

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

Tompa Júlia Boglárka

Alkalmazott matematikus MSc

**OPTIKAI RENDSZEREK MODELLEZÉSE ÉS
OPTIMALIZÁCIÓJA**

Szakdolgozat

Témavezető:

Jüttner Alpár

Operációkutatás Tanszék



Budapest, 2024

Köszönetnyilvánítás

Ezúton szeretnék köszönetet mondani témavezetőmnek, Jüttner Alpárnak, amiért éveken át támogatott, és segített elmélyedni a C++, az optika és a globális optimalizálás világában. Köszönöm neki a világos magyarázatokat, amiknek segítségével gyakran olyan fogalmakat értettem meg, amik korábban mindig összezavartak, valamint az érdekességeket, amik elérték, hogy őszintén lelkesedjek a téma iránt, amivel foglalkoztunk.

Köszönöm a családomnak és a barátaimnak a folyamatos támogatást. Külön köszönet illeti páromat, Attilát, amiért mindig mellettem állt és számíthattam rá a tanulmányaim teljes ideje alatt.

Tartalomjegyzék

Bevezetés	1
1. Lencserendszerek modellezésének fizikai alapjai	2
1.1. A Snellius–Descartes-törvény	3
2. Gauss-féle optika	5
2.1. Az ideális képalkotás tulajdonságai	5
2.2. A mátrix elemeinek meghatározása	8
2.2.1. Fókuszpontok és fősíkok alapján	9
2.2.2. Görbület és törésmutatók alapján	10
3. Sugárkövetéses modellezés	14
3.1. Sugármenetek vizualizációja	14
3.2. Optikai aberrációk	15
3.2.1. Kromatikus aberráció	15
3.2.2. Szférikus aberráció	17
3.2.3. Kóma	20
3.2.4. Asztigmatizmus	21
3.2.5. Képmezőelhajlás	22
3.2.6. Torzítás	22
4. Globális optimalizálás	23
4.1. A globális optimalizálási feladat	23
4.2. A globális optimalizálási feladat megoldhatósága	24

4.3. Problémák csoportosítása	24
4.4. Megoldási módszerek csoportosítása	25
5. Akromát lencse tervezése	27
5.1. Egy akromatikus lencse paramétereinek optimalizálása	27
5.2. Az Arany metszés-módszer	29
5.3. Eredmény	31
6. Egy objektív paramétereinek optimalizálása	32
6.1. Keresés a koordinátatengelyek mentén	34
7. A Controlled Random Search algoritmus	36
7.1. A CRS algoritmus módosított verziója és a CRS2	38
7.2. A Shuffled Complex Evolution módszer	39
7.3. A CRS algoritmussal elért eredmények	42
7.3.1. Az 5.5 fényerejű Tessar	42
7.3.2. A 2.8 fényerejű Tessar	44
8. Implementálás	47
8.1. Ideális képalkotás	47
8.2. Sugárkövetéses modell	48
Kitekintés	50

Bevezetés

Szakedolgozatom középpontjában egy gyakorlati jellegű probléma áll, méghozzá az optika témaköréből. Lencserendszerek, objektívek tervezésekor, az egyes paraméterek megválasztásánál felmerülnek bizonyos képalkotási hibák, aberrációk, amelyeket szeretnénk minél jobban lecsökkenteni. Ezek között vannak olyanok, amelyeket analitikus úton ki tudunk számolni a paraméterek alapján, azonban egyes hibák esetében ez nem ilyen egyszerű. Ezek meghatározásához szükségünk van egy kellően pontos modellre, valamint valamilyen optimalizálási eljárásra, amellyel a hibákat a lehető legnagyobb mértékben ki tudjuk küszöbölni.

Egyszerűsége miatt kézenfekvőnek tűnhet az ún. Gauss-féle modellt alkalmazni, azonban ez nem alkalmas az optikai aberrációk analizálására és megjelenítésére. Ezzel szemben a sugárkövetéses modellezés már használható erre is, ezért mi a modellező program megírására ezt választottuk. Az optikai aberrációk csökkentése egy globális optimalizálási feladatra vezetett, aminek során egy többváltozós, analitikusan nem megadott ("black-box") függvénynek kerestük egy megfelelően kicsi függvényértékű lokális minimumát bizonyos korlátozások mellett, amelyek az optimalizálandó objektívra előírt tulajdonságokból adódtak. A feladat megoldására a globális optimalizálási módszerek áttekintése és néhány másik módszer kipróbálása után a Controlled Random Search-algoritmust választottuk, amelynek segítségével sikerült a célnak megfelelő eredményeket elérnünk egy Tessar típusú objektív optimalizálása során.

Dolgozatom elején röviden kitérek a lencserendszerek modellezéséhez szükséges fizikai alapismertetekre, majd modellezési módszerekkel kezdek foglalkozni: először a Gauss-féle (ideális) képalkotási modellt mutatom be, majd a fent említett sugárkövetéses modellt, amely jelen szakedolgozat céljai szempontjából a legcélravezetőbbnek bizonyult. Ezután előkerül a globális optimalizálás témaköre, a globális optimalizálási problémák tulajdonságai és csoportosítása, valamint megoldási módszereik. Az első konkrét optimalizálási feladat, amellyel foglalkoztunk, egy akromát megtervezése volt. Ehhez az egyszerű feladathoz ugyan nem lenne feltétlenül szükség a sugárkövetéses modellre és a globális optimalizálásra – a színhiba csökkentését analitikus úton is megoldhatnánk –, azonban azt szeretnénk elérni, hogy az általunk létrehozott program más aberrációk figyelembevételére és bonyolultabb lencserendszerek optimalizálására is alkalmas legyen. Ezt egy Tessar objektív optimalizálásán keresztül mutatjuk be. Az ehhez implementált Controlled Random Search-algoritmus eredeti és újabb változatait, valamint néhány rokon módszerét mutatjuk be a 7. fejezetben, a 8. fejezet pedig a C++-ban megírt modellező és optimalizáló program szerkezetét ismerteti.

1. fejezet

Lencserendszerek modellezésének fizikai alapjai

Az optika tudománya a fizikának a fény terjedésével és tulajdonságaival foglalkozó ága. A fény elektromágneses sugárzás, és míg hétköznapi értelemben véve általában a látható fényt értjük alatta – vagyis azt a sugárzást, melynek hullámhossza 400-tól 700 nm-ig terjed – optikai értelemben ide számítjuk az ennél a spektrumnál nagyobb (infravörös) és kisebb (ultraviola) hullámhosszal rendelkező sugárzást is. Az optika fontos részterületei az alábbiak:

A *geometriai optika* olyan egyszerűsített modellt alkalmaz a fény jelenségeinek leírására, melyben a fényt mint sugarak összességét tekintjük. A fénysugarak útját egyes közegeken belül egyenesnek vesszük, különböző közegek határán pedig a sugarak megtörnek vagy visszaverődnek. A geometriai optika nem alkalmas az olyan jelenségek megmagyarázására és modellezésére, mint például a diffrakció, illetve az interferencia, melyek a fény hullámtermészetéből következnek. Ennek ellenére számos esetben nagyon praktikus modellje a fény terjedésének, kiváló közelítésnek számít ugyanis minden olyan esetben, amikor a fény hullámhossza kicsi azoknak az objektumoknak a méretéhez képest, amelyekkel útja során interakcióba lép. Különösen hasznos tehát optikai rendszerek képalkotásának geometriai leírásához, illetve optikai aberrációk vizsgálatához. Ezen okból kifolyólag jelen szakdolgozatban az optikának a geometriai ága kerül elő, és bár a jelenségek egy része, melyekkel foglalkozunk, precízen a hullámoptika eszközeivel lenne megmagyarázható, mi ebben a témakörben nem merültünk el, az elsődleges cél ugyanis a modellezés és optimalizálás volt.

A *hullámoptika* a fényt hullámként modellezi, így alkalmas arra is, hogy a diffrakció, az interferencia, illetve a polarizáció jelenségét vizsgáljuk vele. A történelem során az optikának ez az ága alakult ki később, mint a geometriai optika – bár a diffrakció jelenségét már a 17. században megfigyelték, és Robert Hooke (1635-1703) már ekkor hullámként tekintette a fényt, részben Newton elutasításának okán végül csak a 19. század elején vált elfogadottá ez a modell [3].

Bizonyos jelenségek a fény hullám-részecske kettősségén alapulnak. Ez a kettősség a kvantummechanika egyik központi fogalma. A fény részecske-tulajdonságainak modellezéséhez tekinthet-

jük a fényt részecskékből ("fotonokból") álló anyagnak. Az optika ezzel foglalkozó ágának neve *kvantumoptika*.

1.1. A Snellius–Descartes-törvény

A fény vákuumbeli sebessége a fizika egyik fontos állandója, értéke $c_0 = 299\,792\,458$ m/s. Anyagi közegekben a fény lassabban terjed, mint vákuumban. Ennek mértéke az anyag *törésmutatója* vagy *abszolút törésmutatója* (n), mely az alábbi összefüggés szerint adható meg:

$$n = \frac{c_0}{c},$$

ahol c a fény közegbeli terjedési sebessége.

Egy anyag *relatív törésmutatóján* egy másik anyaghoz viszonyítva az alábbi értéket értjük:

$$n_{2;1} = \frac{c_1}{c_2} = \frac{n_2}{n_1},$$

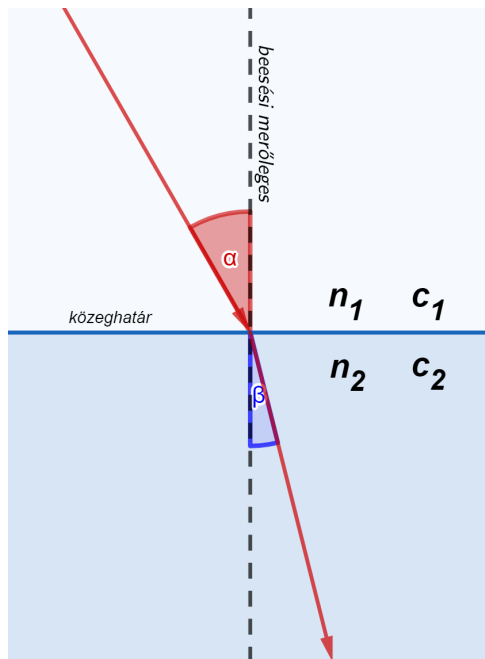
ahol c_1 és c_2 a fény terjedési sebességét adják meg az egyes közegekben, míg n_1 -gyel és n_2 -vel az anyagok abszolút törésmutatóit jelöltük.

Egy anyagot leggyakrabban a levegőhöz mért relatív törésmutatójával jellemzünk. Ez praktikus, mivel a levegő abszolút törésmutatója egyenlő 1.00029-del, ami elég jó közelítéssel 1.

A *fénytörés törvénye* a törésmutatók segítségével írja le, hogy két közeg határán hogyan tör meg a fény. A magyar nyelvű tudomány Snellius–Descartes-törvényként hivatkozik erre az eredményre, Willebrord van Roijen Snellius (1591–1626) és René Descartes (1596–1650) nevéhez kötve azt. Az angol nyelven többnyire Snell törvényeként emlegetett formula felfedezésében valóban volt része mindkét tudósnak, bár mérés útján – hibásan és kevésbé hibásan – már az ókor óta akadtak kutatók, akik újra és újra felfedezték azt. Helyesen először Ibn Sahn írta le 984-ben [10]. Christiaan Huygens (1629-1695) 1678-as, *Traité de la Lumière* című művében leírja, hogy hogyan vezethető le az összefüggés a fény hullámtermészetéből, arra a jelenségre támaszkodva, melyet ma Huygens–Fresnel-elvként ismerünk [3].

1.1.1. Tétel (A Snellius–Descartes-törvény). (a) *A beeső fénysugár, a beesési merőleges és a megtört fénysugár egy síkban van.* (b) *A merőlegesen beeső fénysugár nem tör meg.* (c) *A beesési szög (α) szinuszának és a törési szög (β) szinuszának aránya a közegekben mért terjedési sebességek arányával egyenlő, ami megegyezik a két közeg relatív törésmutatójával, azaz*

$$\frac{\sin(\alpha)}{\sin(\beta)} = \frac{c_1}{c_2} = \frac{n_2}{n_1} = n_{2;1}.$$



1.1. ábra. A Snellius–Descartes-törvény

2. fejezet

Gauss-féle optika

2.1. Az ideális képalkotás tulajdonságai

Az alábbiakban a fényt geometriai optikai megközelítésből vizsgáljuk, azaz sugarak összességének tekintjük. Az általunk vizsgált lencserendszerek egy optikai tengely mentén, arra merőlegesen elhelyezkedő sík- és gömbfelületű lencsékéből állnak.

A formulák egyszerűsége és használhatósága érdekében közelítsük a vizsgált szögek szinusztát: $\sin(x) \approx x$. Ez a közelítés kicsi szögekre lesz pontos, így csak az optikai tengely közvetlen közelében tekinthető jó modellnek. A képalkotás forgásszimmetriája miatt elegendő a lencserendszernek egy hosszanti metszetét tekinteni, és az euklideszi tér helyett síkban dolgozni, az optikai tengelyt a derékszögű koordináta-rendszer x tengelyének feleltetve meg.

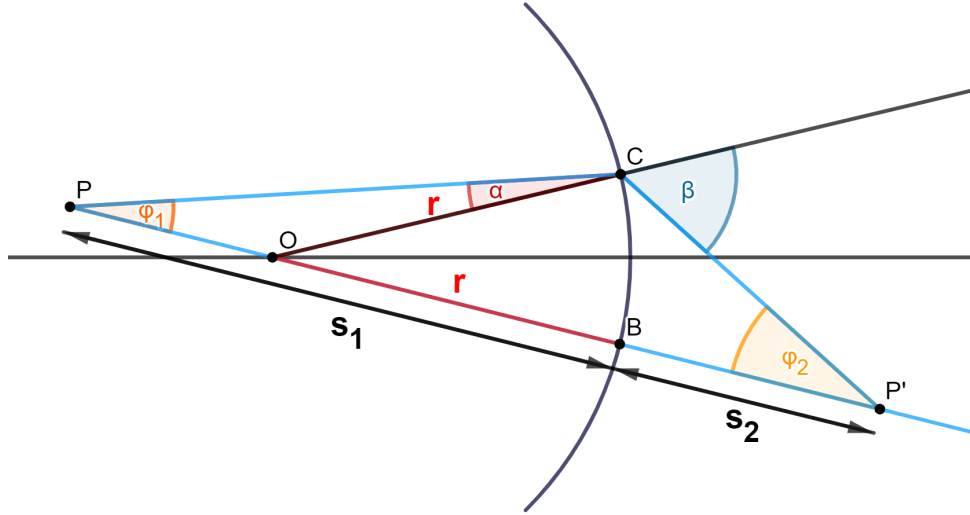
Vizsgáljuk most meg, hogy $\sin(x)$ -et elsőrendben közelítve milyen megállapításokat vonhatunk le egy rendszer leképezésére vonatkozóan. Vegyünk először egy gömbfelületet, melynek gömbi sugara r , gömbi középpontja pedig O , és egy n_1 , valamint egy n_2 törésmutatójú anyagot választ el. Legyen P egy tárgy pont. Vizsgáljuk meg a P -ből kiinduló, a gömbfelületen áthaladó fénysugarakat (2.1. ábra). Az a sugár, amelyik áthalad a gömbi középponton (PO), a Snellius–Descartes-törvény értelmében irányváltoztatás nélkül halad át a felületen. Legyen PC egy ezzel ϕ_1 szöget bezáró sugár, legyen az ehhez tartozó beesési szög α , a törési szög β . Jelöljük a PO egyenes és a C pontban megtört másik fénysugár metszéspontját P' -vel, a két sugár által bezárt szöget P' -ben ϕ_2 -vel.

Ekkor a szinusztétel alapján a POC háromszögben az alábbi összefüggést kapjuk:

$$\frac{r}{\sin(\phi_1)} = \frac{s_1 - r}{\sin(\alpha)}. \quad (2.1)$$

Írjuk fel a szinusztételt a $P'OC$ háromszögre is:

$$\frac{r}{\sin(\phi_2)} = \frac{s_2 + r}{\sin(\pi - \beta)}. \quad (2.2)$$



2.1. ábra. Egy tárgypontból induló sugarak útja egy gömbfelületen keresztül

A két egyenletet átrendezve, és figyelembe véve a $\sin(\pi - \beta) = \sin(\beta)$ azonosságot, az alábbi összefüggésekhez jutunk:

$$r \cdot \sin(\alpha) = (s_1 - r) \cdot \sin(\phi_1), \quad (2.3)$$

valamint

$$r \cdot \sin(\beta) = (s_2 + r) \cdot \sin(\phi_2). \quad (2.4)$$

Most osszuk el egymással (2.3)-at és (2.4)-et:

$$\frac{\sin(\alpha)}{\sin(\beta)} = \frac{(s_1 - r) \cdot \sin(\phi_1)}{(s_2 + r) \cdot \sin(\phi_2)}. \quad (2.5)$$

A Snellius–Descartes-törvény alapján

$$\frac{\sin(\alpha)}{\sin(\beta)} = \frac{n_2}{n_1}. \quad (2.6)$$

Most tekintsük a PBC háromszöget. A BC ív a szögek kicsisége miatt vehető egyenes szakasznak, így a PBC háromszöget vehetjük a B csúcsnál derékszögűnek. Ekkor

$$\frac{BC}{PB} = \tan(\phi_1) \approx \phi_1.$$

A $P'BC$ háromszögben hasonló logikát követve

$$\frac{BC}{P'B} = \tan(\phi_2) \approx \phi_2.$$

Vagyis

$$\frac{\sin(\phi_1)}{\sin(\phi_2)} \approx \frac{\phi_1}{\phi_2} \approx \frac{BC}{PB} \cdot \frac{P'B}{BC} = \frac{P'B}{PB} = \frac{s_2}{s_1}. \quad (2.7)$$

(2.5), (2.6) és (2.7) összevetésével az alábbi egyenlethez jutunk:

$$\frac{n_2}{n_1} = \frac{(s_1 - r)}{(s_2 + r)} \cdot \frac{s_2}{s_1}. \quad (2.8)$$

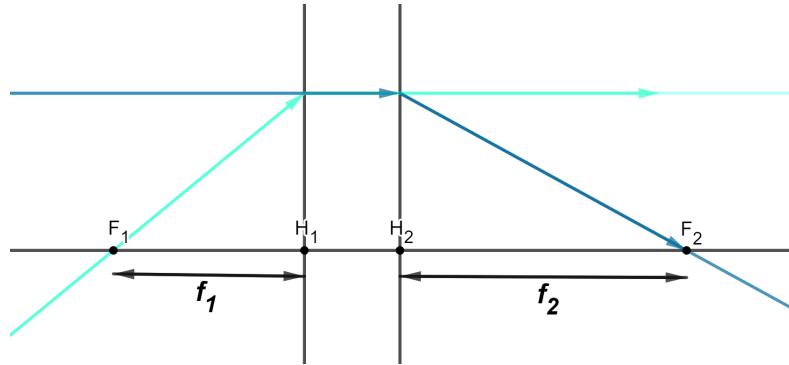
Adott P tárgypontra esetén ismerjük az n_1, n_2, s_1 és r értékeket, a (2.8) egyenlet szerint így s_2 -t is kiszámolhatjuk. Ez azt jelenti, hogy s_2 független ϕ_1 -től, tehát minden P -ből induló sugár, amely a felületen áthalad, egy pontban (P' -ben) fogja metszeni egymást. Ezek szerint minden tárgypontra képe egy képpont. Hasonló módon belátható, hogy ez fordítva is igaz, azaz minden képpont ősképe egy tárgypontra lesz. Mindezt belátható abban az esetben is, ha a tárgypontra máshogy helyezkedik el a gömbi középponthoz képest, mint a 2.1. ábrán, illetve ha a gömbi középpont a felület másik oldalán van, és ha a felület által elválasztott anyagok törésmutatói máshogy aránylanak egymáshoz. Ha egy tárgyponthoz tartozó képpont a tárgypontra azonos oldalán helyezkedik el a felületnek – azaz a tárgypontra kiinduló sugarak valójában széttartanak a felületen áthaladva, csupán a képoldali megfigyelő számára tűnnek úgy, mintha egy pontból indulnának ki –, akkor ezt *virtuális képpont* nevezünk. Ellenkező esetben *valódi képpont* beszélünk.

Vegyünk észre, hogy ebből egy másik tulajdonsága is következik a leképezésnek: a kollinearitás. Vegyünk ugyanis néhány tárgypontra, amelyek egy egyenes mentén helyezkednek el. Vegyünk azt a fénysugarat, amely mindegyikén áthalad. Ez a fénysugár a közeghatárok túloldalán keresztül kell, hogy menjen mindegyik pont képén, vagyis a pontok képeinek egy egyenesre kell esniük.

Később belátjuk, hogy a $\sin(x) = x$ közelítés mellett az optikai tengellyel párhuzamosan érkező sugarak a felületen áthaladva az optikai tengely egy pontjában metszik egymást. Ez a képoldali fókuszpont (F_2). Hasonlóan a rendszert az optikai tengellyel párhuzamosan elhagyó fénysugarak ősképei egy pontban metszik egymást az optikai tengelyen, ez a tárgyoldali fókuszpont (F_1). Az optikai tengellyel párhuzamosan érkező sugarak és képeik metszéspontjai egy síkot határoznak meg, ez a második fősík (H_2). A rendszerből az optikai tengellyel párhuzamosan kilépő sugaraknak és ősképeknek metszéspontjai szintén egy síkot határoznak meg, ez az első fősík (H_1).

A leképezés kollinearitásából következik, hogy ha a tekintett hosszanti metszet síkját beágyazzuk a térbe $z = 1$ magasságban, akkor projektív koordinátákat tekintve a leképezések lineárisak lesznek, tehát leírhatók 3×3 -as mátrixok segítségével. Ekkor egy több elemből álló optikai rendszert jellemezhetünk az egyes elemekhez tartozó mátrixok szorzatával. Így minden optikai rendszerhez kiszámolható egy 3×3 -as mátrix, amely leírja az ő képpalkotását. Azt a rendszert, amelyre mindezek a jellemzők igazak (minden tárgypontra konjugáltja egy képpont és fordítva, valamint tartozik hozzá két fősík és két fókuszpont a fent ismertetett tulajdonságokkal) *ideális rendszernek* nevezünk, a leképezését pedig *ideális képpalkotásnak*. Beláttuk tehát, hogy ha $\sin(x)$ -et elsőrendben közelítjük, akkor a kapott rendszer ideális rendszer lesz. Ezt a modellt *Gauss-féle*, vagy *paraxiális* közelítésnek is nevezünk, utóbbi elnevezés utal arra, hogy a modell csak az optikai tengely közelében lesz pontos. Az, hogy a képpalkotás ideális, azt is jelenti, hogy a valóságban előforduló képpalkotási hibák itt nem jelennek meg. Így tehát ez a modell optikai aberrációk vizsgálatára nem használható.

Az alábbiakban optikai rendszereket jellemző mátrixok elemeinek meghatározására keresünk



2.2. ábra. Az ideális képalkotás jellemzői: F_1 és F_2 a fókuszpontok/gyújtópontok, H_1 és H_2 a fősíkok, f_1 és f_2 a fókusz távolságok/gyújtótávolságok.

formulákat a rendszer paramétereinek ismeretében.

2.2. A mátrix elemeinek meghatározása

Legyen az x tengely továbbra is a vizsgált rendszer optikai tengelye. Ahogy említettük, a forgásszimmetria miatt dolgozhatunk síkban, egy hosszanti metszetet tekintve. Így egy pont jellemezhető egy x és egy y koordinátával, projektív koordinátákat tekintve pedig az (x, y) pontnak az $(x, y, 1)$ vektor és ennek skalárszorosai felelnek meg. A továbbiakban dolgozzunk sorvektorok helyett oszlopvektorokkal. Vegyük észre, hogy ha egy $(x, y, 1)^T$ pont képe egy $(x', y', 1)^T$ pont, akkor, mivel a leképezés az optikai tengelyre szimmetrikusan történik, az $(x, -y, 1)^T$ pont képe az $(x', -y', 1)^T$ pont kell legyen. Legyen a leképezést leíró mátrix a következő:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix}.$$

Tetszőleges $(x, y, 1)^T$ pont képe az A mátrix szerint

$$\begin{pmatrix} a_{1,1}x + a_{1,2}y + a_{1,3} \\ a_{2,1}x + a_{2,2}y + a_{2,3} \\ a_{3,1}x + a_{3,2}y + a_{3,3} \end{pmatrix},$$

míg az $(x, -y, 1)^T$ pont képe

$$\begin{pmatrix} a_{1,1}x - a_{1,2}y + a_{1,3} \\ a_{2,1}x - a_{2,2}y + a_{2,3} \\ a_{3,1}x - a_{3,2}y + a_{3,3} \end{pmatrix}$$

lesz. Így a leképezés szimmetriájára vonatkozó megfigyelésből a következő három összefüggéshez jutunk:

$$\begin{aligned} a_{1,1}x + a_{1,2}y + a_{1,3} &= a_{1,1}x - a_{1,2}y + a_{1,3}, \\ a_{2,1}x + a_{2,2}y + a_{2,3} &= -(a_{2,1}x - a_{2,2}y + a_{2,3}), \\ a_{3,1}x + a_{3,2}y + a_{3,3} &= a_{3,1}x - a_{3,2}y + a_{3,3}. \end{aligned}$$

Ez a három összefüggés tetszőleges x, y értékekre fennáll, amiből az következik, hogy $a_{1,2} = a_{2,1} = a_{3,2} = a_{3,3} = 0$, azaz a leképezés mátrixa a következőképpen néz ki:

$$\begin{pmatrix} a_{1,1} & 0 & a_{1,3} \\ 0 & a_{2,2} & 0 \\ a_{3,1} & 0 & a_{3,3} \end{pmatrix}.$$

Továbbá mivel projektív koordinátákat nézünk, ezért egy mátrix és a skalárszorosa ugyanazt a leképezést határozzák meg. Emiatt a mátrix tetszőleges elemét 1-nek választhatjuk. Az $a_{2,2} = 1$ jó választás, mivel ekkor két ilyen alakú mátrixot összeszorozva a szorzatmátrix középső eleme szintén 1 lesz. Ezek szerint egy lencserendszer matematikai leírásához négy értéket kell meghatározunk:

$$\begin{pmatrix} a_1 & 0 & c_1 \\ 0 & 1 & 0 \\ a & 0 & c \end{pmatrix}.$$

2.2.1. Fókuszpontok és fősíkok alapján

Tegyük fel, hogy ismerjük egy rendszer két fókuszpontját: F_1, F_2 , valamint a fősíkok helyét: H_1, H_2 . Ekkor az alábbi megfigyelések alapján számolhatjuk ki a keresett négy mátrixelemet.

1. Az $(F_1, 0, 1)^T$ pont képe az ideális pont, vagyis az optikai tengellyel párhuzamos egyenesek "végpontja": $(1, 0, 0)^T$.

$$\begin{pmatrix} a_1 & 0 & c_1 \\ 0 & 1 & 0 \\ a & 0 & c \end{pmatrix} \begin{pmatrix} F_1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} F_1 \cdot a_1 + c_1 \\ 0 \\ F_1 \cdot a + c \end{pmatrix}.$$

A szorzat pontosan akkor fog megegyezni $(1, 0, 0)^T$ skalárszorosával, ha

$$c = -F_1 \cdot a. \tag{2.9}$$

2. Az $(F_2, 0, 1)^T$ pont ősképe az ideális pont.

$$\begin{pmatrix} a_1 & 0 & c_1 \\ 0 & 1 & 0 \\ a & 0 & -F_1 \cdot a \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} a_1 \\ 0 \\ a \end{pmatrix}.$$

Az eredményvektor normált alakja pontosan akkor fog megegyezni $(F_2, 0, 1)^T$ -vel, ha

$$a_1 = F_2 \cdot a \quad (2.10)$$

teljesül.

3. A $(H_1, y, 1)^T$ pontot a leképezés a $(H_2, y, 1)^T$ pontba viszi tetszőleges y esetén. Tekintsük az $y = 1$ esetet.

$$\begin{pmatrix} F_2 \cdot a & 0 & c_1 \\ 0 & 1 & 0 \\ a & 0 & -F_1 \cdot a \end{pmatrix} \begin{pmatrix} H_1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} H_1 \cdot F_2 \cdot a + c_1 \\ 1 \\ H_1 \cdot a - F_1 \cdot a \end{pmatrix}.$$

Az eredményvektornak egyenlőnek kell tehát lennie $(H_2, 1, 1)^T$ -tal, amiből az alábbi következtetéseket vonhatjuk le:

$$a = \frac{1}{H_1 - F_1}, \quad (2.11)$$

valamint

$$c_1 = H_2 - H_1 \cdot F_2 \cdot a = H_2 - \frac{H_1 \cdot F_2}{H_1 - F_2}. \quad (2.12)$$

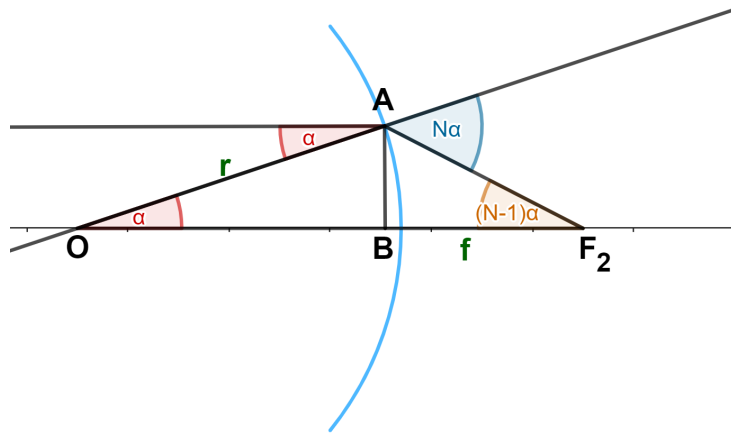
A (2.9)-(2.12) képleteket összesítve az alábbi formulákat kapjuk a mátrix elemeire:

$$\begin{pmatrix} \frac{F_2}{H_1 - F_1} & 0 & H_2 - \frac{H_1 \cdot F_2}{H_1 - F_1} \\ 0 & 1 & 0 \\ \frac{1}{H_1 - F_1} & 0 & -\frac{F_1}{H_1 - F_1} \end{pmatrix}. \quad (2.13)$$

2.2.2. Görbület és törésmutatók alapján

Most legyen adott egy felület előjeles görbülete (ezt akkor tekintjük negatívnak, ha a görbületi középpont a felülettől a tárgyter irányába helyezkedik el): g , valamint a felület két oldalán található anyagok törésmutatója: n_1, n_2 , továbbá a felület helye az x tengelyen: H_1 . (Mivel itt csak egy felületről van szó, az első és a második fősík helye is a felület helyével fog megegyezni. Több felületből álló rendszer mátrixát az egyes felületekhez tartozó mátrixok összeszorzásával kaphatjuk meg.) Ezen adatok segítségével ismét a felület képalkotását jellemző mátrix elemeit szeretnénk megállapítani. Elsőként úgy határozzuk meg a mátrixelemeket, hogy H_1 -et nullának vesszük, majd egy eljárás segítségével eltoljuk a rendszert H_1 -gyel.

Tekintsünk először tehát egy felületet, amely az origóban helyezkedik el. Először az F_2 fókuszpontot határozzuk meg a felület előjeles görbülete, valamint az általa elválasztott két anyag törésmutatója alapján. Legyen N a két törésmutató aránya, azaz $N = \frac{n_1}{n_2}$.



2.3. ábra. F_2 meghatározása

A 2.3. ábrán egy negatív görbületű lencsébe belépő, az optikai tengellyel párhuzamos fénysugarat láthatunk. A fénysugár α beesési szöggel lépi át a felületet. A Snellius–Descartes-törvény alapján tehát a törési szög (β) kiszámítható az alábbi képlettel:

$$\frac{\sin(\alpha)}{\sin(\beta)} = \frac{n_2}{n_1} = \frac{1}{N}.$$

Mivel $\sin(x)$ -et x -szel közelítettük, így a fenti képletből

$$\beta = N \cdot \alpha$$

következik. Csúcsszögek és váltószögek észrevételével azt is megkapjuk, hogy a 2.3. ábra jelöléseivel $AOB\angle = \alpha$ és $AF_2B\angle = (N - 1) \cdot \alpha$. Az ABO háromszögben

$$\alpha \approx \sin(\alpha) = \frac{AB}{r},$$

amiből

$$AB = r \cdot \alpha.$$

Továbbá

$$(N - 1)\alpha \approx \tan((N - 1)\alpha) = \frac{AB}{f} = \frac{r \cdot \alpha}{f},$$

ahonnan

$$f = \frac{r}{(N - 1)}.$$

Hozzávéve mindehhez, hogy az előjeles görbület ebben az esetben $g = -\frac{1}{r}$ -rel egyezik meg, továbbá a felület az origóban van, az alábbi képlethez jutunk:

$$F_2 = -\frac{1}{g \cdot (N - 1)}. \quad (2.14)$$

Hasonló logika mentén megvizsgálhatjuk azokat az eseteket is, amikor az előjeles görbület pozitív, illetve ha $N < 1$. A tárgyoldali fókuszpont (F_1) meghatározásához nem kell mást tennünk, mint az előbbi esetet "jobbról balra" végigvenni, ami annyi változtatást jelent az eredményben, hogy N helyére $\frac{1}{N}$ -et kell írunk. Azaz

$$F_1 = -\frac{1}{g \cdot (\frac{1}{N} - 1)} = -\frac{N}{g \cdot (1 - N)} = \frac{N}{g \cdot (N - 1)}. \quad (2.15)$$

Vegyük észre, hogy a (2.14) és a (2.15) képletek egyben be is bizonyítják, hogy az optikai tengellyel párhuzamosan belépő sugarak egy pontban metszik az optikai tengelyt (F_2), valamint hogy az F_1 ponton áthaladó sugarak a lencserendszeren áthaladva az optikai tengellyel párhuzamosan haladnak tovább.

H_1 és H_2 helyére 0-t írva így a (2.13), valamint a (2.14)-(2.15) összefüggésekből megkapjuk a négy mátrixelemet:

$$\begin{pmatrix} \frac{1}{N} & 0 & 0 \\ 0 & 1 & 0 \\ \frac{g \cdot (1 - N)}{N} & 0 & 1 \end{pmatrix}. \quad (2.16)$$

Ekkor tehát kiszámoltuk egy felület mátrixát az előjeles görbület és az általa elválasztott anyagok törésmutatója alapján, abban az esetben, ha a felület éppen az origóban helyezkedik el, és így $H_1 = H_2 = 0$. Ahhoz, hogy tetszőleges H_1 -re meg tudjuk határozni a mátrixelemeket, ki kell dolgoznunk egy eljárást az x tengely mentén eltoló rendszer mátrixának meghatározására.

Az eltolásmátrix meghatározását két különböző módszerrel is elvégezhetjük. Az egyik módszer hasonló ahhoz, ahogy a rendszer fókuszpontjai és főtengelyei alapján kiszámoltuk a mátrix elemeit: tudjuk, hogy ha az $(x, y, 1)^T$ pont képe az eredeti rendszer szerint az $(x', y', 1)^T$ pont, akkor a d -vel eltoló rendszer az $(x, y + d, 1)^T$ pontot az $(x', y' + d, 1)^T$ pontba kell, hogy vigye. Ez a megfigyelés meghatároz három egyenletet, amelyek segítségével meghatározható az eltoló rendszer mátrixa.

A másik módszer ennél is egyszerűbb. Legyen $D(v_x, v_y)$ a (v_x, v_y) -nal való eltolás mátrixa [12]. Ekkor

$$D(v_x, v_y) = \begin{pmatrix} 1 & 0 & v_x \\ 0 & 1 & v_y \\ a & 0 & 1 \end{pmatrix}.$$

Most ha egy $(x, y, 1)^T$ pont d -vel eltoló rendszer szerinti képét szeretnénk megkapni, és az eredeti (origóban lévő) rendszer mátrixát ismerjük:

$$A = \begin{pmatrix} a_1 & 0 & c_1 \\ 0 & 1 & 0 \\ a & 0 & c \end{pmatrix},$$

akkor az $(x, y, 1)^T$ pontot $-d$ -vel eltolva az x tengely mentén, a kapott pontra az A transzformációt alkalmazva, majd d -vel visszatolva megkapjuk a pont eltoló rendszer szerinti képét. Eszerint

az eltolt rendszer mátrixát a következőképpen számolhatjuk ki:

$$A' = D(d, 0) \cdot A \cdot D(-d, 0),$$

vagyis

$$A' = \begin{pmatrix} a_1 + ad & 0 & c_1 + cd - a_1d - ad^2 \\ 0 & 1 & 0 \\ a & 0 & c - ad \end{pmatrix}. \quad (2.17)$$

(2.16) és (2.17) alapján megkaphatjuk a mátrixelemeket tetszőleges H_1 pontban elhelyezkedő felület esetén.

3. fejezet

Sugárkövetéses modellezés

A fejezet ábráit a 8. fejezetben bemutatott kód segítségével generáltuk.

A $\sin(x)$ függvény elsőrendbeli közelítése olyan képalkotási modellhez vezet, melyben a leképezés hibái nem jelennek meg. Ahhoz, hogy ezeket a hibákat vizsgálni tudjuk, ennél pontosabb modellre van szükségünk. A sugárkövetéses modellezés megfelel erre a célra. Ez a módszer a fényt továbbra is mint egyenes úton haladó sugarakat összességét tekinti, így egy fénysugarat tudunk a kiindulópontjával és egy irányvektorral jellemezni. Egy sugárnak a lencsék alkotta sík-, illetve gömbfelületekkel való metszéspontjait kiszámolhatjuk a koordinátageometria eszközeivel, a metszéspontból továbbhaladó sugár irányvektorát pedig a Snellius–Descartes-törvény formulájával tudjuk meghatározni.

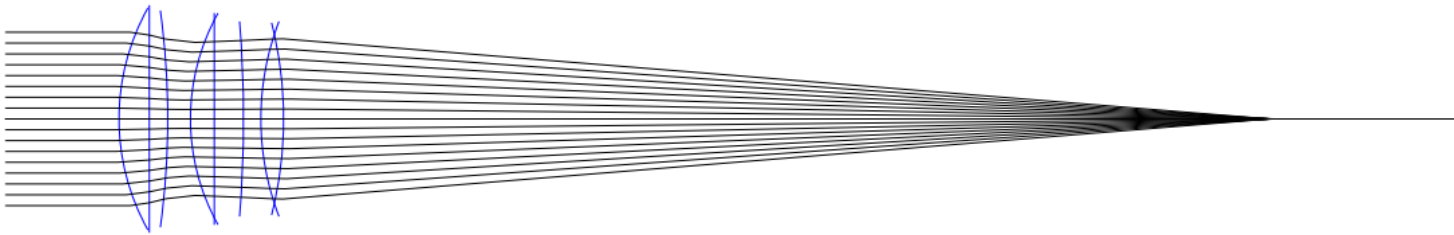
Sugárkövetéses modellezés segítségével végigkövethetjük tehát a fénysugarak útját egy lencseszerkezeten keresztül, és ez már alkalmas a képalkotás hibáinak, az ún. optikai aberrációknak a vizualizációjára és analizálására is.

3.1. Sugármenetek vizualizációja

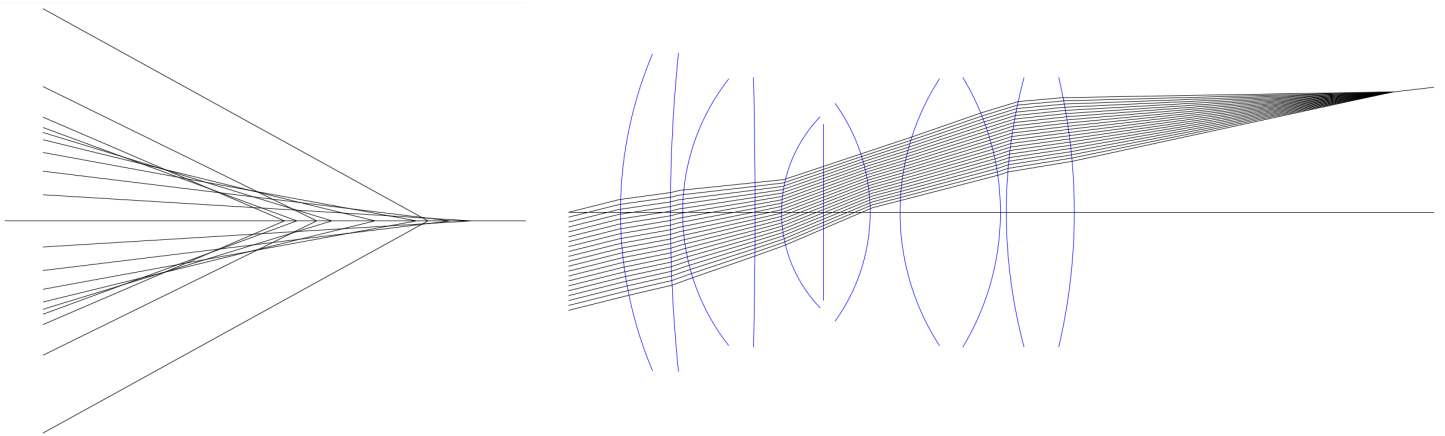
A sugárkövetéses modellt implementáló program segítségével 3 objektívet vizsgáltunk: egy Tessar, egy Biotar és egy Biogon típusút. Először az x tengellyel párhuzamosan érkező sugarak meneteit rajzoltuk ki (3.1. ábra).

Ha kinagyítjuk a Tessar objektíven áthaladó sugarakról készült 3.1. ábra jobb szélét (3.2. ábra), akkor az is látszik, hogy az ideális képalkotással szemben itt nem egy pontban metszik egymást és az x tengelyt a sugarak. Ezt a képalkotási hibát hívják szférikus aberrációnak [1] (lásd 3.2.2. fejezet).

Egymással párhuzamos, az x tengellyel kis szöget bezáró sugár-sereg útját is kirajzoltuk a programmal, a 3.3. ábrán a Biotar típusú objektíven áthaladó sugarak menete látható.



3.1. ábra. Tessar - sugármenetek



3.2. ábra. Tessar - szférikus aberráció

3.3. ábra. Biotar - ferde sugármenetek

3.2. Optikai aberrációk

A képalkotás aberrációit két csoportba sorolhatjuk: *monokromatikus aberrációknak* nevezzük azokat a képalkotási hibákat, melyek a rendszer lencséinek geometriai tulajdonságaiból következnek, és így monokromatikus fény használata esetén is fellépnek. Ezzel szemben *kromatikus aberrációkkal* csak akkor kell számolnunk, ha különböző hullámhosszú fénysugarak haladnak át a rendszeren.

3.2.1. Kromatikus aberráció

Kromatikus aberrációnak nevezzük a képalkotás azon hibáit, amelyeket az okoz, hogy optikai közegek határán a különböző hullámhosszú fénysugarak különböző mértékben törnek meg, mivel a törésmutató függ a hullámhossztól. Így a különböző színű sugarak nem egy pontban fókuszálódnak, és egy tárgy pont képe elmosódik, nem egy képpont lesz [3]. Monokromatikus fény esetén kromatikus aberrációk nem lépnek fel.

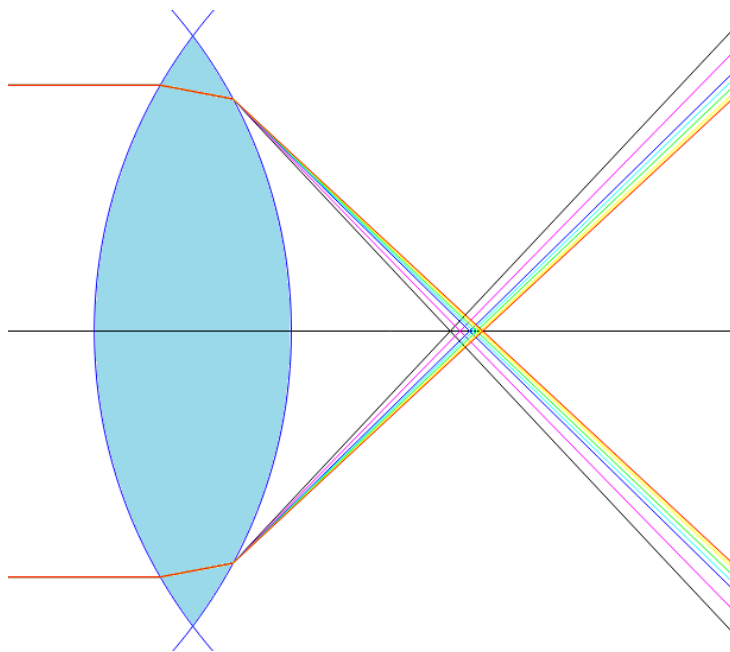
A kromatikus aberrációknak két típusát különböztetjük meg:

- *Axiális kromatikus aberráció*nak nevezzük azt a jelenséget, amikor a különböző hullámhosszú sugarakhoz tartozó fókuszpontok különböznek, azaz a különböző hullámhosszú sugarak a lencsefelülettől különböző távolságokban fókuszálódnak.
- *Transzverzális kromatikus aberráció* alatt azt értjük, amikor különböző hullámhosszú sugarak a fókusz sík különböző pontjain fókuszálódnak, vagyis például a lencserendszer nagyítása kék színű fény esetén kisebb mértékű, mint vörös fény esetén.

A kromatikus aberrációk modellezéséhez szükségünk van a lencsék anyagának törésmutatójára, ami tehát nem állandó, hanem a fénysugár hullámhosszának függvényében változik. A hullámhossz és a törésmutató közötti összefüggést adja meg például a Sellmeier-egyenlet [13], [18]:

$$n^2 - 1 = C_1 + \sum_{i=1}^8 \frac{C_{2i}\lambda^2}{\lambda^2 - C_{2i+1}}, \quad (3.1)$$

ahol a C_i ($i = 1, \dots, 17$) együtthatók az adott anyagra jellemző állandók, λ az áthaladó sugár hullámhossza, n pedig a levegőhöz mért relatív törésmutató.

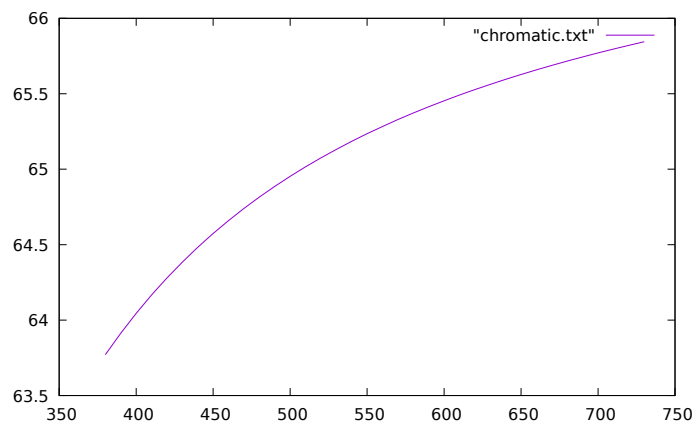


3.4. ábra. Axiális kromatikus aberráció egy bikonvex lencsén szemléltetve

Az axiális kromatikus aberráció jelensége megfigyelhető a 3.4. ábrán, ahol különböző hullámhosszú sugarakat szimuláltunk, amint áthaladnak egy $K7$ (koronaüveg) anyagú lencsén. A lencsébe való belépéskor még együtt haladnak a sugarak, két nyalábban, majd mindkét nyaláb különböző színű sugarakra oszlik. Az egyes hullámhosszokhoz tartozó fókuszpontok különböző távolságokra vannak a lencsefelülettől.

A kromatikus aberráció mértékét szemléltethetjük grafikonon is: ábrázoljuk az x tengelyen a hullámhosszt (nm-ben), az y tengelyen pedig az adott hullámhosszhoz tartozó fókuszpontot

(mm-ben). A 3.5. ábrán a fenti lencséhez tartozó grafikon látható.



3.5. ábra. Fókuszpont a hullámhossz függvényében, egy lencse esetén

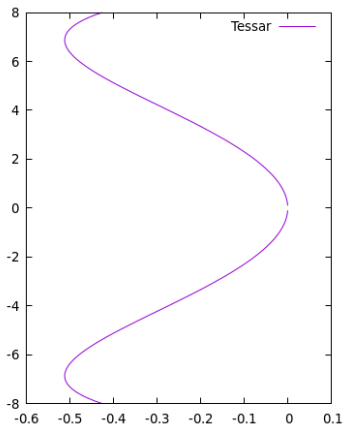
3.2.2. Szférikus aberráció

Szférikus aberráció akkor lép fel, ha az optikai tengellyel párhuzamosan, de különböző magasságokban belépő nyalábok eltérő távolságokban fókuszálódnak, azaz eltérő távolságokban metszik az optikai tengelyt [1]. Ez a jelenség a lencsék geometriai tulajdonságainak következménye, és monokromatikus fény használata esetén is megfigyelhető, ezért soroljuk a monokromatikus aberrációk közé. Az alábbiakban több módot is bemutatunk a szférikus aberráció mértékének szemléltetésére sugárkövetéses modellezés segítségével.

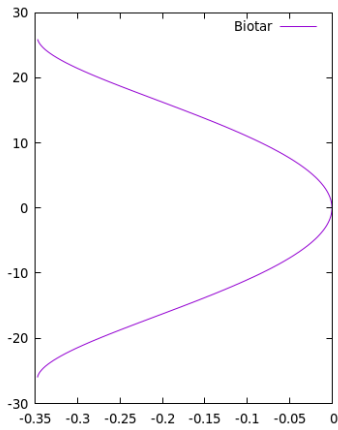
Az első ilyen módszer során a következőképpen járunk el: szimuláljuk az optikai tengellyel párhuzamosan induló sugarakat, amint áthaladnak a lencserendszeren, majd ábrázoljuk egy grafikonon az y tengelyen a sugarak kiinduló magasságát, az x tengelyen a sugár és az optikai tengely metszéspontjának paraxiális fókuszponttól vett távolságát. (Paraxiális fókuszpont alatt az ideális képalkotás szerinti fókuszpontot értjük.) A grafikonokon (3.6-3.8. ábrák) megfigyelhető, hogy ha az optikai tengelyhez közel indítunk azzal párhuzamos nyalábokat, akkor azok a várakozásunknak megfelelően közel esnek a paraxiális fókuszponthoz (hiszen a paraxiális közelítés lényege éppen az, hogy az optikai tengely közelében pontos), míg az optikai tengelytől távolodva egy ideig egyre nő a különbség.

Összehasonlításként érdemes lehet egy, az adott objektívvel megegyező fókusz távolságú szimpla síkdomború lencse szférikus aberrációját is szemléltetni (3.9-3.11. ábrák). Mindhárom vizsgált objektív esetében jól látható, hogy e hiba szempontjából jelentősen jobban teljesítenek egy egyszerű lencsénél. Az összehasonlításhoz használt lencsék anyagát ún. rövid flintüvegnek választottuk, melynek levegőhöz mért relatív törésmutatója (587.6 nm hullámhosszú fénysugarak esetén) 1.5211.

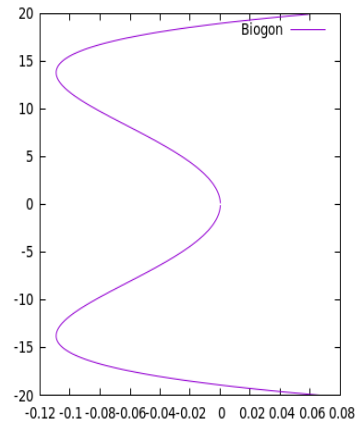
Más módon is szemléltethetjük a szférikus aberrációt: helyezzünk az optikai tengelyre merőleges síkokat a paraxiális fókusz közelébe, és vizsgáljuk meg, hogy az objektíven áthaladó sugarak hol metszik el ezeket. Ezen szemléltetés bemutatására a Biogon típusú objektívet használtunk. A



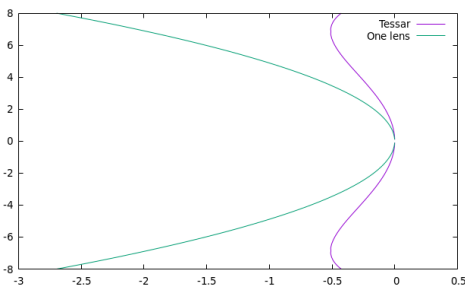
3.6. ábra. Tessar



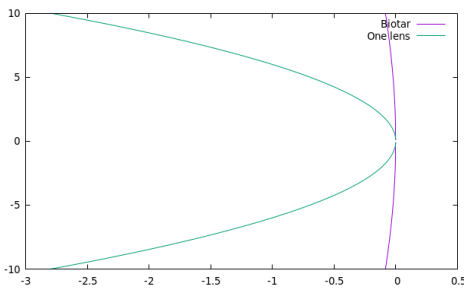
3.7. ábra. Biotar



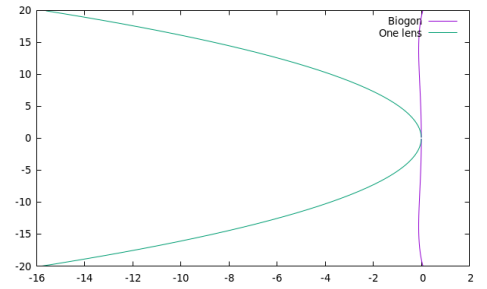
3.8. ábra. Biogon



3.9. ábra. Tessar

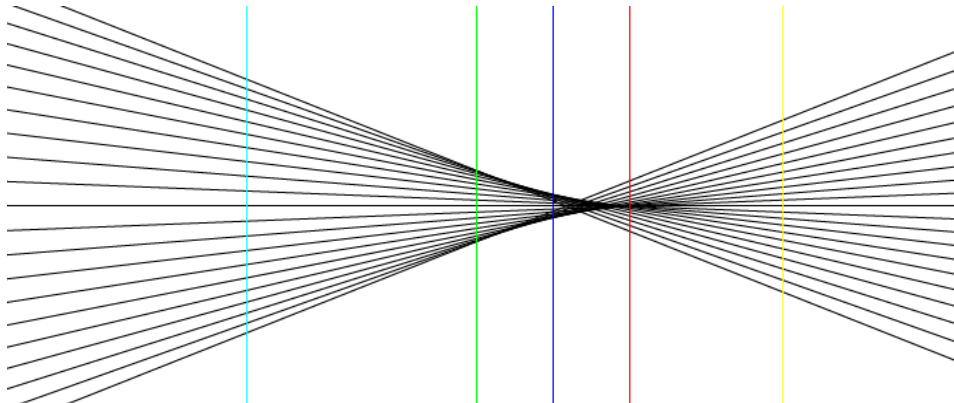


3.10. ábra. Biotar



3.11. ábra. Biogon

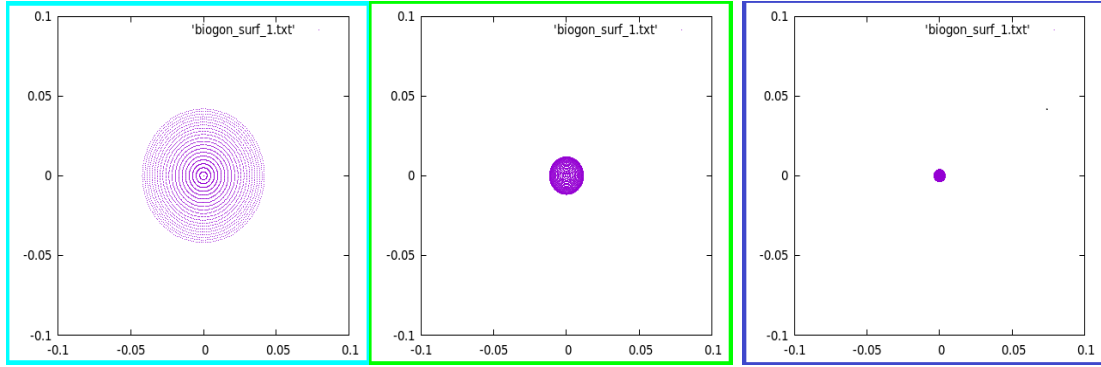
3.12. ábrán az egyes síkok elhelyezkedése látható, a 3.13-3.17. ábrák mutatják, hogy az optikai tengellyel párhuzamosan, körkörösén indított sugárnyalábok hol metszik el az adott síkot.



3.12. ábra. A síkok elhelyezkedése. Cián: 0.5 mm-rel, zöld: 0.2 mm-rel, kék: 0.1 mm-rel a paraxiális fókusz előtt, piros: a paraxiális fókusz sík, citromsárga: 0.2 mm-rel mögötte.

A 3.18-3.22. grafikonokon az y tengelyen a sugarak kiindulási magasságát, az x tengelyen az optikai tengelytől való távolságát ábrázoltuk, amikor a sugár az egyes síkokat átfúrja.

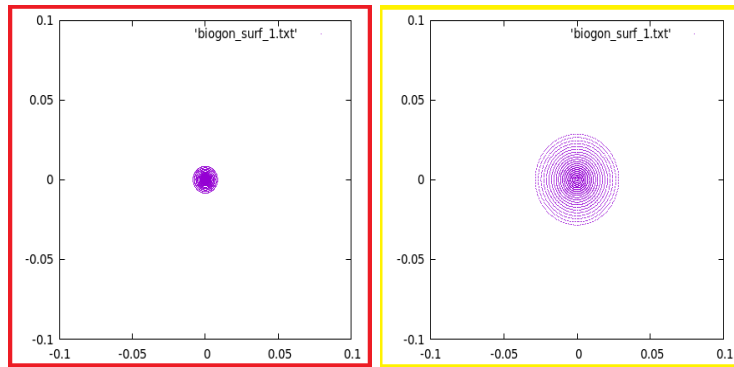
A szemléltetési módok mindegyike jól láttatja, hogy az optikai tengellyel párhuzamos sugárnyalábok nem a paraxiális fókuszpontban, hanem valamivel előtte fókuszálódnak a legjobban.



3.13. ábra. 0.5 mm

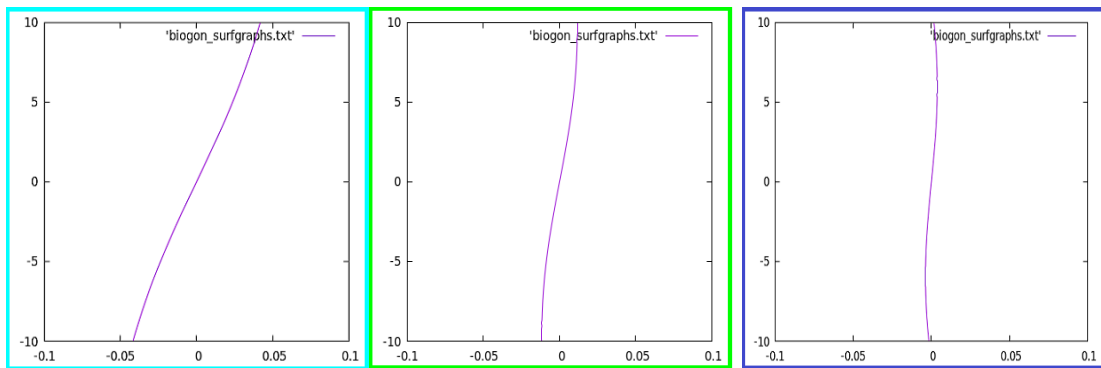
3.14. ábra. 0.2 mm

3.15. ábra. 0.1 mm



3.16. ábra. 0 mm

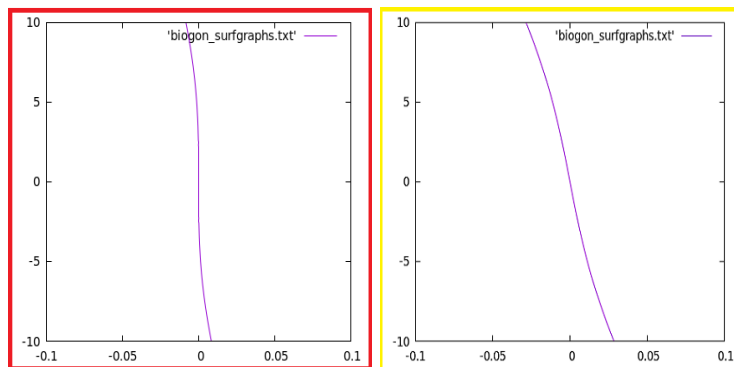
3.17. ábra. 0.2 mm



3.18. ábra. 0.5 mm

3.19. ábra. 0.2 mm

3.20. ábra. 0.1 mm

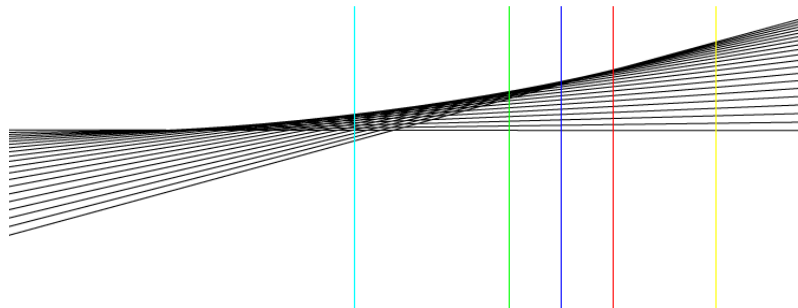


3.21. ábra. 0 mm

3.22. ábra. 0.2 mm

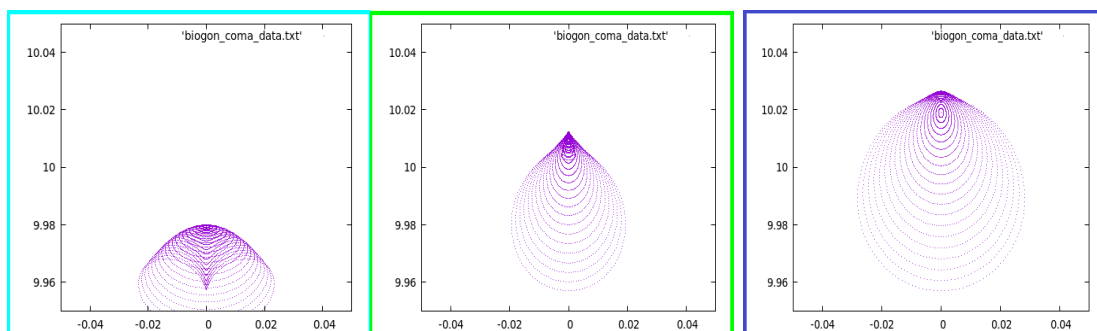
3.2.3. Kóma

Kóma akkor léphet fel, amikor az optikai tengellyel szöget bezáró sugarak lépnek be az apertúrán. Az objektív szélein belépő ún. tengelyen kívüli vagy off-axis sugarak más magasságban metszik a fókuszfelületet, mint az apertúra közepén belépők. Ennek eredményeként egy pontforrás képének intenzitáseloszlása nem lesz szimmetrikus, hanem elnyúlt, üstökösszerű alakot ölt [5]. A kóma vizsgálatának szemléltetéséhez szintén a Biogon objektívet használtuk - a fentiekhez hasonlóan vettünk 5 síkot, melyek a paraxiális fókusz közelében, az optikai tengelyre merőlegesen helyezkednek el, és megvizsgáltuk, hogy az optikai tengelyhez képest kis szögben induló sugarak hol dőfik ezeket a síkokat.



3.23. ábra. A síkok elhelyezkedése. Cián: 0.5 mm-rel, zöld: 0.2 mm-rel, kék: 0.1 mm-rel a paraxiális fókusz előtt, piros: a paraxiális fókusz sík, citromsárga: 0.2 mm-rel mögötte.

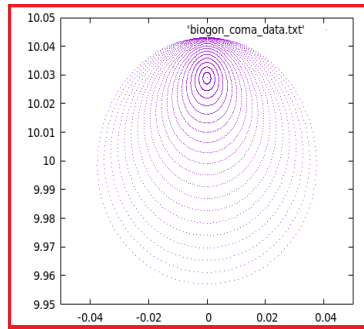
A 3.24-3.28. ábrákon jól látszik az üstökösszerű alak, amit az egyes síkokat elmetező sugarak rajzolnak ki. A sugarakat koncentrikus körök mentén, 10 mm-rel az objektív első lencsefelvénye előtről, nagyjából 5.71° -os emelkedéssel indítottuk - ezek az adatok véletlenszerűek, a kóma illusztrálásához választottuk őket.



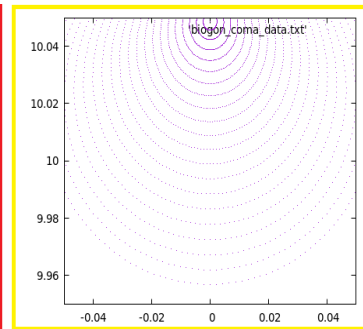
3.24. ábra. 0.5 mm

3.25. ábra. 0.2 mm

3.26. ábra. 0.1 mm

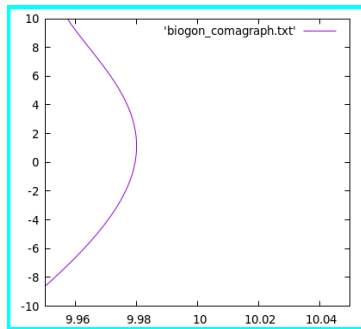


3.27. ábra. 0 mm

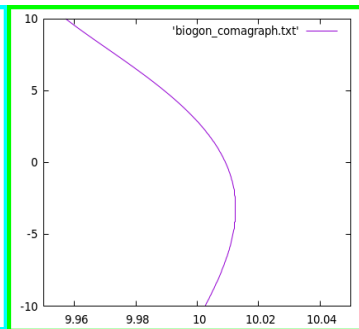


3.28. ábra. 0.2 mm

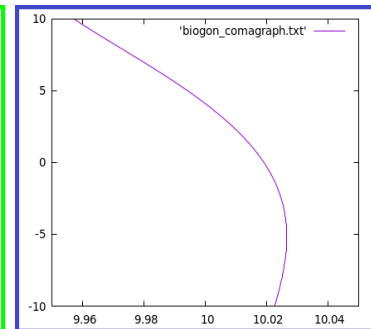
A 3.29-3.33. grafikonok a fentiekhez hasonlóan szemléltetik a kóma mértékét: az y koordináta mutatja a sugárnyalábok magasságát, az x koordináta pedig az optikai tengelytől való távolságot.



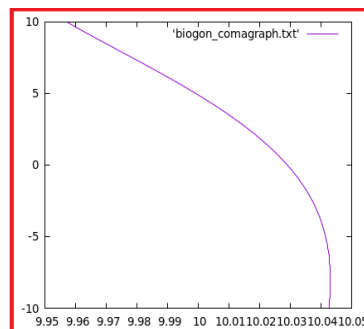
3.29. ábra. 0.5 mm



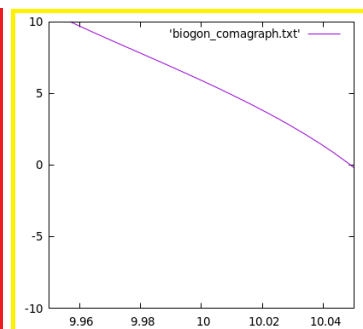
3.30. ábra. 0.2 mm



3.31. ábra. 0.1 mm



3.32. ábra. 0 mm



3.33. ábra. 0.2 mm

3.2.4. Asztigmatizmus

Asztigmatizmusnak azt az optikai aberrációt nevezzük, amikor két, egymásra merőleges síkban haladó sugarak fókuszsíkja nem esik egybe. Az optikai tengelyre szimmetrikus rendszerek esetén ez olyan tárgyponthok esetén fordulhat elő, amelyek nem az optikai tengelyen helyezkednek el. Ekkor megtörténhet, hogy a tárgyponthból kiinduló, az optikai tengely síkjában ("tangential plane") haladó fénysugarak más síkban fókuszálódnak, mint az optikai tengely síkjára merőleges síkban

("sagittal plane") haladó sugarak [1], [7]. Azokat az objektíveket, melyekben az asztigmatizmus korrigálva van, anasztigmátnak nevezzük.

3.2.5. Képzőelhajlás

Képzőelhajlásnak vagy fókuszfelület-görbülésnek azt hívjuk, amikor a fókuszfelület nem sík, hanem görbült, vagyis az éles képet nem sík, hanem görbült felületre lehet kivetíteni [5]. Ezt az optikai aberrációt szimuláció nélkül, analitikusan is vizsgálhatjuk az ún. Petzvál-összeg segítségével [9], amely a magyar Petzvál József nevéhez fűződik. Ez az összeg egy optikai rendszer felületeinek görbületi sugara, valamint a levegőhöz viszonyított törésmutatójuk alapján megadja a képfelület előjeles görbületét.

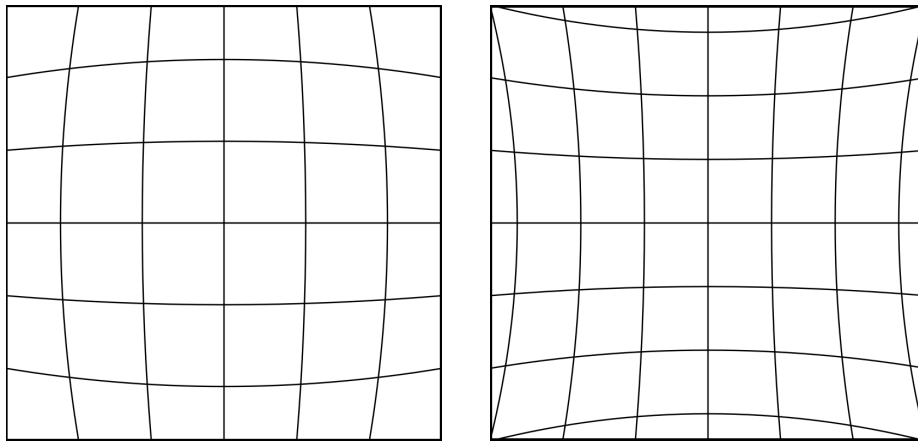
A Petzvál-összeg a következőképpen számítható [9]:

$$\sum_i \frac{n_{i+1} - n_i}{r_i n_{i+1} n_i}, \quad (3.2)$$

ahol r_i az i -edik felület görbületi sugara, az n -ek pedig a felület által elválasztott anyagok törésmutatói.

3.2.6. Torzítás

A torzítás annyiban különbözik az eddig tárgyalt leképezési hibáktól, hogy nem a kép élességére, hanem annak alakjára van hatással. Azt az aberrációt értjük alatta, amikor a tárgy egyenes vonaljai a képen nem maradnak egyenesek [7]. A torzításnak sok formája létezik, de leggyakrabban az alábbiakkal találkozhatunk: *hordó torzítás* és *párnahiba*.



3.34. ábra. A hordó torzítás és a párnahiba

4. fejezet

Globális optimalizálás

4.1. A globális optimalizálási feladat

Számos gyakorlati mérnöki feladat megfogalmazható globális optimalizálási problémaként, azaz olyan optimalizálási problémaként, ahol a célfüggvény nem konvex, és lehet multimodális: tartalmazhat sok lokális minimumot az értelmezési tartomány releváns részén [19]. A releváns rész alatt azt az A tartományt értjük, amelyet az határoz meg, hogy a változóknak mely kombinációi adnak értelmes megoldást az adott alkalmazás szempontjából. A globális optimalizálás célja egy olyan megoldás megtalálása A -ban, melyre a célfüggvényérték a lehető legkisebb, vagyis ami egy *globális minimumot* határoz meg. Lokális optimalizálási feladatoknál léteznek kritériumok, amelyek segítségével ellenőrizhetjük, hogy találtunk-e lokális minimumot (például a gradienssel kapcsolatos feltételek) – ezzel szemben globális optimalizálás esetén nem áll rendelkezésünkre ilyen kritérium.

A legtöbb globális optimalizálási feladat esetében kevés *a priori* információval rendelkezünk a célfüggvényről. Általában feltehetjük, hogy a célfüggvény majdnem mindenhol folytonos, ezenfelül minden esetben ismerjük a tartományt, amelyben az optimumot keressük, valamint adott pontban ki tudjuk számolni a függvényértéket. Gyakran azonban ez minden, amit tudunk. Ilyenkor az egyetlen mód arra, hogy bármilyen információt szerezzünk a függvényről, az, ha kiértékeljük valahol.

A megoldási stratégia egy *globális szakasz*ból és *lokális szakaszok*ból áll. *A priori* információk híján A minden részét egyformán alaposan át kell vizsgálnunk (globális szakasz), különben fennáll a lehetőség, hogy elkerüljük a globális minimumot. Miután gyűjtöttünk bizonyos mennyiségű információt a függvényről, a keresési tartomány egyes részei ígéretesebbnek bizonyulhatnak, ezekben pontosabb megoldásokat szeretnénk találni (lokális szakasz).

4.2. A globális optimalizálási feladat megoldhatósága

A globális optimalizálási problémán belül több feladatot különböztetünk meg aszerint, hogy mi az optimalizálás célja. Legyen $P(f^*)$ az a probléma, amikor csupán a globális minimum értékére – f^* -ra – vagyunk kíváncsiak, $P(X^*)$ pedig az, amikor az összes globális minimumhelyet (legyen ezek halmaza X^*) keressük. Bizonyos esetekben elég, ha X^* egy pontját meg tudjuk határozni, ezt a problémát jelöljük $P(x^*)$ -gal. Numerikus módszerek használatával természetesen az összes fenti problémára csupán közelítő megoldást remélhetünk.

$P(f^)$ megoldhatósága [19]:* Az f^* érték közelítésének megoldhatóságát vizsgálandó tegyük fel, hogy f folytonos A -n és A kompakt. Ekkor f^* folytonos függvénye f -nek és így a $P(f^*)$ probléma jóldefiniált. Azonban nem garantált, hogy véges sok függvényérték-számolással tudunk találni pontot az M_ε színhalmazban, ahol

$$M_\varepsilon = \{x \in A \mid f(x) \leq f^* + \varepsilon\}.$$

Tegyük fel ugyanis, hogy véges sok helyen kiértékeljük a függvényt. Ekkor a függvény globális minimuma tetszőleges mértékben eltérhet a legkisebb talált értéktől.

A fentiekből az is következik, hogy az f^* approximálására vonatkozó feladat nem megoldható, azaz nem létezik olyan algoritmus, amely minden ilyen típusú problémát véges sok lépésben megold. Ez a megállapítás olyan f függvények esetén is áll, amelyek tetszőlegesen sokszor folytonosan deriválhatóak A -n.

$P(x^)$ megoldhatósága [19]:* Az x^* approximálására vonatkozó feladatot tovább bonyolítja az a tény, hogy $P(x^*)$ nem jóldefiniált. Ez azt jelenti, hogy léteznek folytonos függvények A -n, melyekre a függvényértékek maximális abszolút különbsége tetszőlegesen kicsi, de a globális optimumhelyek távol esnek egymástól.

Tekintsük például az f_δ függvényosztályt, ahol

$$f_\delta(x) = \cos(\phi x) - \delta x, \quad x \in [2, -2].$$

A $\delta > 0$ esetben $x^* \simeq 1$, míg $\delta \leq 0$ esetén $x^* \simeq -1$.

Bár a megoldhatóságra vonatkozó megállapítások nem hangzanak biztatóan, mégis fontos foglalkoznunk a globális optimalizálás témakörével, mert a gyakorlatban számos olyan probléma felmerül, amely ilyen feladatra vezet. Szerencsére léteznek olyan heurisztikák, amelyek bár nem garantálnak minden esetben jó megoldást, a gyakorlatban mégis nagyon hasznosnak bizonyultak.

4.3. Problémák csoportosítása

A globális optimalizálási problémákat – amellet, hogy mi az optimalizálás célja – számos tulajdonság mentén vizsgálhatjuk, illetve sorolhatjuk csoportokba. Elsősorban aszerint szoktuk cso-

portosítani a globális optimalizálási feladatokat, hogy tartalmazzanak-e *korlátozó feltételeket*. Másik szempont lehet a *megoldhatóság*: léteznek olyan globális optimalizálási problémák, amelyekre van hatékony algoritmus, de mint láttuk, nem mindig ez a helyzet. Nagy különbségek lehetnek az egyes problémák között abban, hogy mekkora a *célfüggvény kiértékelésének erőforrásigénye*. Számít a *lokális minimumhelyek száma* is: azok a problémák, melyekben a célfüggvénynek viszonylag kevés lokális minimuma van, könnyebben megoldhatók, mert ilyenkor a globális minimum "vonzástartománya" várhatóan nagyobb lesz. A lokális minimumhelyek elhelyezkedése is lényeges: ha egy csoportban helyezkednek el A -n belül, akkor az egyik megtalálása könnyen vezethet több másik felfedezéséhez is. Hasonlóan fontos a *dimenzió*: minél magasabb dimenziójú problémáról van szó, annál nehezebb megoldani a feladatot. Például a teljes keresési tartománynak már a nagy vonalakban történő átvizsgálása is rendkívül nehéz magasabb dimenziókban, ahogy az erőforrásigény és a szükséges tárhely is nő, ha növeljük n -et. Ezenfelül könnyítheti vagy nehezítheti a probléma megoldását az, hogy állnak-e rendelkezésünkre információk a célfüggvény bizonyos tulajdonságairól, például az analitikus deriváltokról, a függvény növekedésére vonatkozó korlátokról, a függvényértékekre vonatkozó korlátokról, a lokális minimumhelyek számáról A -ban, vagy azok eloszlásáról.

4.4. Megoldási módszerek csoportosítása

A különböző globális optimalizálási problémák megoldására alkotott módszereket sokféleképpen csoportosíthatjuk [19]. Indokolt lehet elsősorban két diszjunkt osztályra bontani őket aszerint, hogy milyen pontossággal oldják meg a problémákat: garantálják-e a pontos megoldás megtalálását, vagy sem. Garantált pontos megoldás csak A kimerítő átkutatása esetén érhető el, ez pedig csak olyan problémák esetében lehetséges, amelyekről rendelkezésünkre állnak bizonyos *a priori* információk az alapvetőeken felül. A két megoldási típus alapvetően különbözik abban, hogy milyen célt szolgálnak, így nem érdemes a két osztályba tartozó módszerek teljesítményét egymással összevetni. A garantáltan pontos megoldást adó módszereket nevezzük *fedési módszereknek*, a többi módszert pedig osszuk két csoportra: a *direkt módszerek* csak lokális információk (függvénykiértékelések) felhasználásával működnek, míg az *indirekt módszerek* a lokális információkat arra használják, hogy globális modellt hozzanak létre a szinthalmazokról vagy a célfüggvényről.

Fedési módszerek: Fedési módszereknek tehát azokat a megoldási módszereket nevezzük, amelyek garantálják a pontos megoldás megtalálását. A legegyszerűbb fedési módszerek olyan tartományok felfedezésén és az optimalizálásból való kizárásán alapulnak, amelyek biztosan nem tartalmazzák a globális minimumot (ilyen például az Aranymetszés-módszer, melynek lényegi gondolatáról és implementálásáról szó esik a 5.2. fejezetben). *Lokalizáló* vagy *kettéosztó* folyamatként is megközelíthetjük a feladatot, melynek eredményeképpen az előírt pontosságnak megfelelően rövid, a globális minimumokat tartalmazó intervallumokhoz jutunk. Egy harmadik megközelítés alapján egyre jobb felső és alsó (szakaszonként lineáris) közelítéseket keresünk a célfüggvényre, és ezek minimumhelyeit tartjuk számon.

Random keresési módszerek: A random keresések a direkt módszerek közé tartoznak. A

három legegyszerűbb random keresési eljárás a *tiszta random keresés*, a *singlestart* és a *multistart*. A tiszta random keresés módszere egyáltalán nem alkalmaz lokális finomítást. A singlestart-módszernél a random minta legjobb eleméből indítunk lokális keresést, míg a multistart-eljárásnál minden random pont egy lokális kereső algoritmus kezdőpontja lesz.

A Controlled Random Search-algoritmus, amelynek vizsgálatáról, implementálásáról és rokon módszereiről szó esik a 7. fejezetben, ezek közül egyik kategóriába sem illik bele tökéletesen, de lényegét tekintve ez is random keresési módszer.

Klaszterező módszerek: Ez a csoport szintén a direkt módszerek közé sorolható. A globális optimalizálási probléma ezen megoldási módszereinél valamilyen klaszterező eljárást alkalmazunk annak elkerülésére, hogy újra és újra megtaláljunk korábban már felfedezett lokális minimumokat. A cél lényegében az, hogy minden lokális minimum meghatározására, melynek "vonzáskörzetét" megtaláljuk, egyetlen lokális keresést futtassunk.

A pontok lokális minimumok köré csoportosítására két általános módszer létezik. Az egyik azon alapul, hogy kis függvényértékű pontok megtartásával ezek nagy valószínűséggel csoportokat fognak alkotni a lokális minimumok körül (ezt a gondolatot a Controlled Random Search-algoritmus is felhasználja). A másik stratégia pedig az, hogy valamilyen lokális keresési algoritmusból néhány lépést lefuttatva közelebb tolhatjuk a pontokat a lokális optimumokhoz.

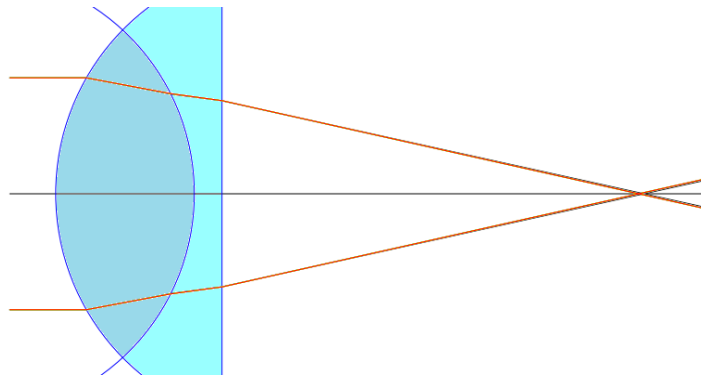
"Generalized descent"-módszerek: Ezek a metódusok szintén direkt módszernek számítanak. Lényegi gondolatuk, hogy a felesleges lokális kereséseket elkerülhetjük úgy, hogy folytatjuk a random keresést A -ban egészen addig, amíg találunk egy pontot, melyre kisebb a függvényérték az összes eddigi megtalált pontnál. Ebből a pontból lokális keresést indítva garantáltan az eddiginél jobb minimumot találunk. Ez a technika tekinthető a lokális keresési módszerek globális optimalizálásra való általánosításának.

Szinthalmazokat vagy célfüggvényt approximáló módszerek: Ezeket nevezzük *indirekt módszereknek*, mivel a fentiekkel ellentétben nem közvetlenül használják fel a célfüggvényről szerzett lokális információkat, hanem azokat valamilyen globális modell felállítására használják. A módszerek matematikailag és implementálás szempontjából egyaránt komplikáltabbak az eddigiéknél, és bár emléletben biztatónak tűnnek, gyakran túl bonyolultak az algoritmikus megvalósításhoz.

5. fejezet

Akromát lencse tervezése

Egy akromatikus lencse, vagy más néven akromát tervezésekor kiválasztunk két hullámhosszt, általában úgy, hogy az egyik a vörös, a másik a kék szín egy bizonyos árnyalatának felel meg (ez a két hullámhossz általában a következő: 486.1 nm és 656.3 nm), majd megnézzük, hogy hova fókuszálnának az egyik, illetve a másik hullámhosszal rendelkező sugarak. A cél a két fókuszpont közötti távolság minimalizálása, ezáltal az axiális kromatikus aberráció csökkentése [17]. (Léteznek ún. apokromatikus lencsék/objektívek is: ezeknél már három kiválasztott hullámhossz fókuszpontját igyekszünk egymáshoz közelíteni.) Általában egy akromát két elemből áll: egy koronaüvegből készült bikonvex, és egy flintüvegből készült konkáv lencséből [17].



5.1. ábra. Kromatikus aberráció egy akromátban

A 5.1. ábrán megfigyelhető, hogy amikor egy akromátba lépnek be különböző hullámhosszú fénysugarak, akkor ezek jóval kevésbé szóródnak, alig látszik, hogy különválnának egymástól a fókuszpontjaik – szemben a 3.4. ábrával, ahol a bikonvex lencsébe lépő sugárnyalábok jól láthatóan különböző színű komponensekre bomlanak.

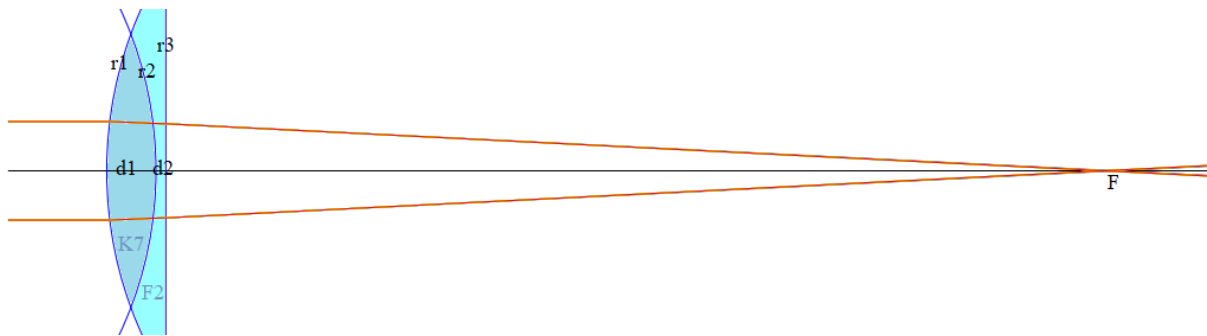
5.1. Egy akromatikus lencse paramétereinek optimalizálása

Bonyolultabb optikai rendszerek, objektívek paramétereinek optimalizálása előtt először egy egyszerűbb dolgot, egy akromátot terveztünk meg. Az volt tehát a célunk, hogy a 486.1 nm és a

656.3 nm hullámhosszú sugarak fókuszpontjai között minimalizáljuk a távolságot az alábbi feltételek mellett:

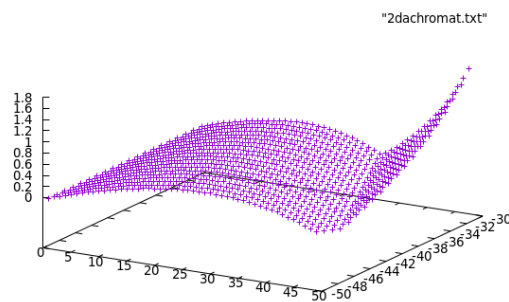
- álljon a lencse két elemből: egy *K7* anyagú bikonvex részből, és egy *F2*-ből (flintüvegből) készült lencséből, melynek a képoldali felülete sík;
- az első lencse vastagságát 5 mm-re, a másodikét 1 mm-re állítottuk be;
- továbbá előírtuk, hogy a lencse fókusz távolsága 100 mm legyen.

Mint ahogy a fókuszpontok és fókusz távolságok paraxiális közelítés mellett is kiszámolhatók, így ez az optimalizálási feladat valójában analitikus úton, az ideális képalkotás formuláit használva is megoldható lett volna, nem lett volna feltétlenül szükség szimulációra. Mi azért választottuk mégis ezt az opciót, mert azt szerettük volna, ha a megoldási módunk bonyolultabb rendszerek esetén, más optikai aberrációkat figyelembe véve is működne.



5.2. ábra. Akromát terve

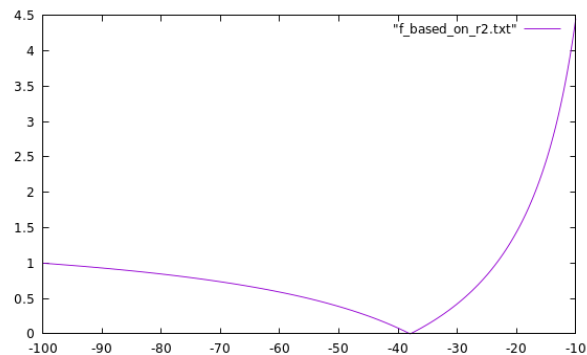
Az volt tehát a feladatunk, hogy a 5.2. ábrán látható r_1 és r_2 paraméterek meghatározásával minimalizáljuk a két adott hullámhossz fókuszpontjainak eltérését, miközben adott, hogy $r_3 = \infty$, $d_1 = 5$ és $d_2 = 1$, valamint a fókusz távolság 100 mm. (Ha egy sugár ∞ , az alatt síkfelületet értünk, míg a pozitív sugár olyan gömbfelületet takar, melynek gömbi középpontja a felülettől a képtér felé esik, negatív sugár pedig a tárgytér felé eső gömbi középpontot jelent.) Ez azt jelenti, hogy a 5.3. ábrán látható kétváltozós függvényt kellett minimalizálnunk bizonyos feltételek teljesülése mellett.



5.3. ábra. A két hullámhossz fókuszpontjának különbsége r_1 és r_2 függvényeként

Fontos megfigyelés, hogy ha a két sugár közül az egyiket (például r_2 -t) rögzítjük, akkor a fókusztávolságra vonatkozó előírás miatt a másik sugár értéke is adott lesz. Ez alapján először írtunk egy függvényt, amely rögzített r_2 -höz megtalálja azt az r_1 -et, melyre a fókusztávolság 656.3 nm hullámhosszú sugarak esetén a legkevésbé tér el 100 mm-től. Ez a függvény felező keresés segítségével minimalizálja azt, hogy a fókusztávolság mennyire tér el az előírttól. Vegyük észre, hogy ebben az esetben a megfelelő r_1 megtalálásához nem lett volna szükség szimulációra és felező keresésre (vagy bármilyen más optimalizálási algoritmusra), mivel a fókusztávolság kiszámolása a Gauss-féle közelítés használatával is megoldható feladat – ami azt jelenti, hogy r_1 -et a fókusztávolság ismeretében kiszámolhatnánk analitikusan. Mivel azonban a felező keresés implementálása nagyon egyszerű feladat, és a futtatása sem vesz igénybe annyi időt, hogy az analitikus megoldási mód jelentős időnyereséget jelentene, így gyakorlatilag mindegy, melyik módszert választjuk.

Ezután r_2 -re kell egy olyan értéket találnunk, melyre a hozzá kiszámolható r_1 -gyel a lencse optimális akromát lesz, tehát a két hullámhossz fókusza között minimális az eltérés. Ez az eltérés ezúttal r_2 -nek lesz egy kvázikonvex függvénye, ahogy a 5.4. ábrán megfigyelhető. Itt r_2 minden értékéhez -100 mm és -10 mm között meghatároztuk r_1 értékét a fent említett bináris kereséssel, majd kiszámoltuk a 486.1 nm és 656.3 nm hullámhosszú sugarak fókuszpontjainak különbségét. Ezen kvázikonvex függvény minimalizálásához az ún. Aranymetszés-módszert (Golden Section Method) használtuk [2].



5.4. ábra. A két fókuszpont távolsága r_2 függvényeként

5.2. Az Aranymetszés-módszer

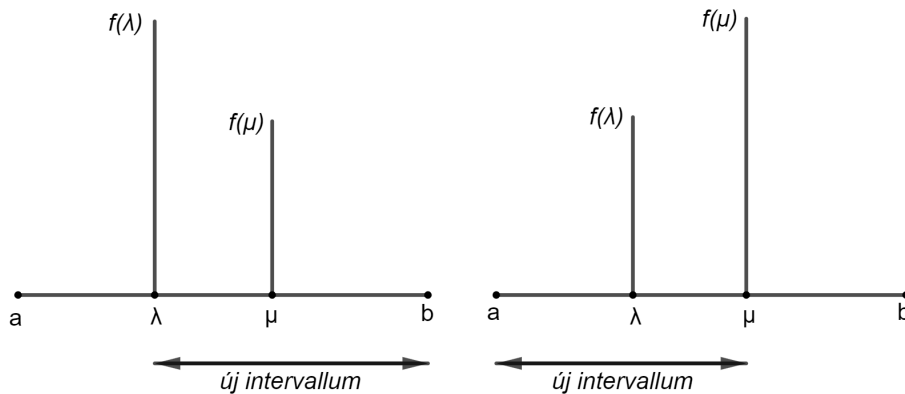
A folytonos optimalizálás egyik kulcseszköze a deriválás. Jelen esetben azonban el kell tekintenünk ezen eszköz használatától, ugyanis a függvény, melyet minimalizálni szeretnénk, nem adott analitikusan, képlettel, hanem csupán szimuláció útján számolhatjuk ki az értékeit. Mivel azonban feltehetjük, hogy célfüggvényünk szigorúan kvázikonvex, így megoldhatjuk a feladatot például az Aranymetszés-módszerrel.

A módszer megértéséhez szükségünk van az alábbi definícióra.

5.2.1. Definíció. Az $[a, b]$ intervallumot *bizonytalansági intervallumnak* nevezzük, ha a célfügg-

vény egy minimumpontja az $[a, b]$ intervallumban van, de pontos helyzetét nem tudjuk.

A módszer lényege, hogy egy szigorúan kvázikonvex függvényt a bizonytalansági intervallumon belül két pontban kiértékelve kizárhatjuk a bizonytalansági intervallum bizonyos részeit, amelyek biztosan nem tartalmazzák a minimumot, és így folyamatosan csökkenthetjük a bizonytalansági intervallum hosszát, amíg az valamilyen küszöbérték alá nem megy (5.5. ábra).

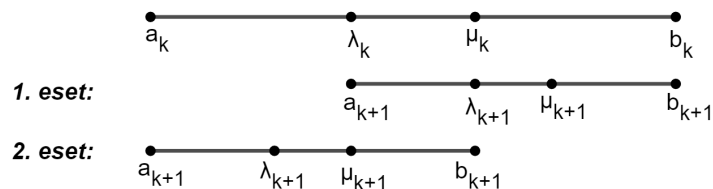


5.5. ábra. A bizonytalansági intervallum csökkentése

Tegyük fel például, hogy egy $f(x)$ függvényt szeretnénk minimalizálni úgy, hogy $a \leq x \leq b$, tehát a kiinduló bizonytalansági intervallum $[a, b]$. Legyen $\lambda, \mu \in [a, b]$, $\lambda < \mu$. Ha $f(\lambda) > f(\mu)$, akkor a minimum a $[\lambda, b]$ intervallumba esik, ha pedig $f(\lambda) < f(\mu)$, akkor az $[a, \mu]$ intervallumba. Így tehát csökkenteni tudjuk a bizonytalansági intervallum hosszát azáltal, hogy a függvényt két pontban kiértékeljük. A két pont kiválasztására számos módszer létezik, ezek közül az egyik leghatékonyabb az Aranymetszés-módszer. Ezen módszer alkalmazásakor a bizonytalansági intervallum csökkenésének aránya, azaz

$$\frac{\text{a bizonytalansági intervallum hossza } v \text{ függvény-kiértékelés után}}{\text{kiinduló bizonytalansági intervallum hossza}}$$

egyenlő $(0.618)^{v-1}$ -nel.



5.6. ábra. A következő pont kiválasztásának módja az Aranymetszés-módszer szerint

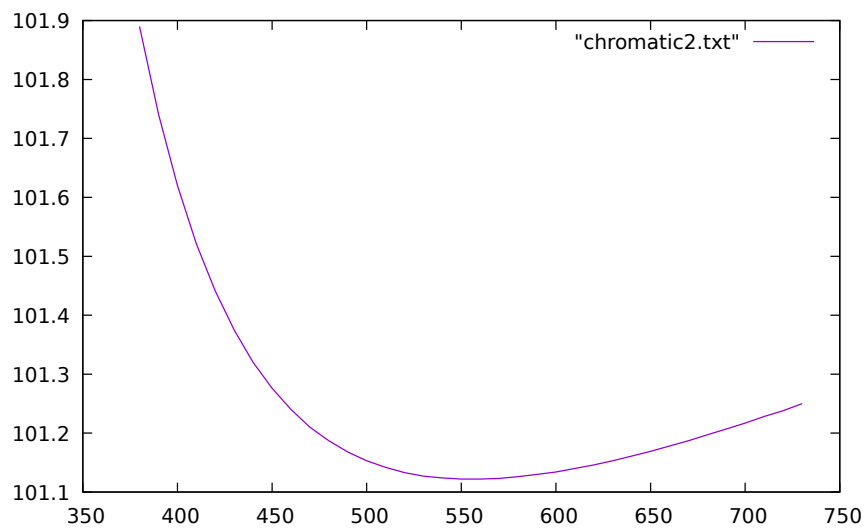
5.3. Eredmény

Az Aranymetszés-módszer implementálásával az alábbi értékeket kaptuk a két paraméterre:

- $r1 = 40.0526$ mm,
- $r2 = -37.968$ mm.

A kész akromátban a két fókuszpont különbsége $1.55 \cdot 10^{-7}$ mm volt, ami jelentős javulás a kiinduló értékhez képest (0.667 mm).

A 5.7. ábrán az optimalizált akromatikus lencse fókuszpontjának optikai tengelyen való elhelyezkedését ábrázoltuk az áthaladó sugár hullámhosszának függvényében. Ezt a görbét érdemes összevetni a 3.5. ábrával, ahol egy bikonvex lencse esetében szemléltettük ugyanezt. Látszik, hogy ott a fókuszpont annál távolabbra kerül a lencsefelülettől, minél nagyobb a hullámhossz, míg az akromát esetében a távolság először csökken, majd nő, így a két kijelölt hullámhossz – 486.1 nm és 656.3 nm – esetén ugyanannyi lesz. Az is látszik, hogy az értékek itt jóval kisebb intervallumon belül változnak, mint a lencse esetében.

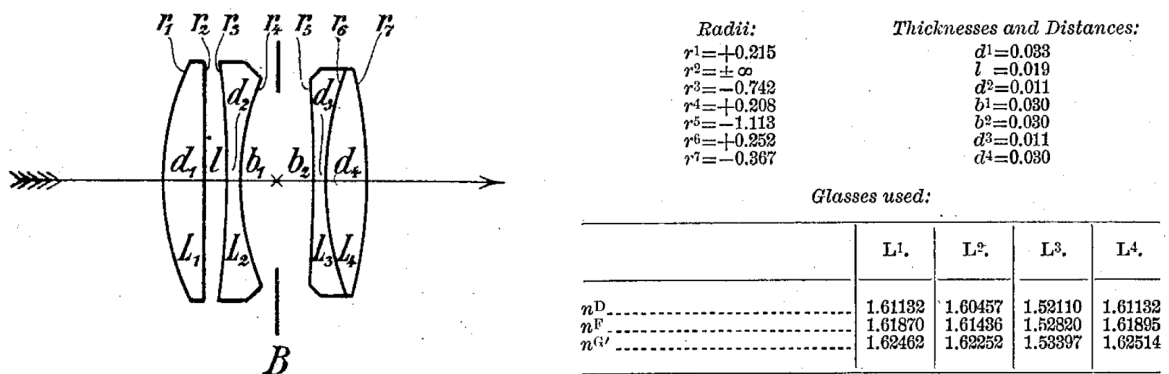


5.7. ábra. Fókuszpont elhelyezkedése (mm) a hullámhossz (nm) függvényében

6. fejezet

Egy objektív paramétereinek optimalizálása

Az akromatikus lencse tervezése után áttértünk a komplexebb objektívek optimalizálásának témakörére. Azt az 1903-ban szabadalmaztatott Tessar típusú objektívet vettük elő ismét, mely a 3. fejezetben már előkerült. Erről az objektívről az alábbi adatok álltak rendelkezésünkre: a lencsék által meghatározott 7 felület sugara, a lencsék vastagsága és távolsága, valamint az anyaguk törésmutatója (két hullámhosszon) és Abbe-száma. Ezen adatok alapján tudtunk az eredeti objektívben használt anyagokhoz hasonló tulajdonságúakat találni (mivel az eredetiekét ma már nem gyártják).



6.1. ábra. A Tessar objektív adatai [16]

Célunk az objektív paramétereinek módosítása volt úgy, hogy bizonyos leképezési hibák a lehető legkisebbek legyenek. A fókusztávolságot ismét 100 mm-re szeretnénk volna állítani. A feladat tehát hasonlít az akromát optimalizálásához, csak itt kettő helyett összesen 13 paramétert kell optimalizálni: 7 felület görbületét és a közöttük lévő 6 távolságértéket.

Mivel a fókusztávolság ismét adott, így ha a 13 paraméter közül 12-t rögzítünk, akkor a 13. is kiszámolható. Ehhez létrehoztunk egy függvényt, ami az objektív utolsó felületének görbületét

számolja ki bináris kereséssel a másik 12 paraméter függvényében.

Az objektívben az alábbi aberrációkat szeretttük volna minimalizálni:

- axiális kromatikus aberráció,
- szférikus aberráció és
- képmezőelhajlás.

Az axiális kromatikus aberráció mértékének csökkentését az akromáthoz hasonló módon szeretttük volna elérni, azaz úgy, hogy a 486.1 nm és a 656.3 nm hullámhosszokhoz tartozó fókuszpontok távolságát minimalizáljuk.

A szférikus aberráció mibenlétéről és vizsgálatáról sok szó esett a 3. fejezetben: ez az a képalkotási hiba, amikor az optikai tengellyel párhuzamosan érkező sugarak nem egy pontban fókuszálódnak. A 3. fejezetben láttunk példákat arra, hogy a szférikus aberráció mértékét hogyan lehet szemléltetni, most azonban ez nem elég: számosítani is kell, hogy lehessen minimalizálni. Ehhez az optikai tengellyel párhuzamos fénysugarakat bocsátottunk az objektívbe szimuláció útján, a teljes apertúra szélességében, és megnéztük, hogy melyik metszi az optikai tengelyt a paraxiális fókuszról legtávolabb. Ezt a távolságot, azaz a szférikus aberráció maximumát szeretttük volna minimalizálni.

A képmezőelhajlás mértékét kiszámolhatjuk a Petzvál-összeg (3.2) segítségével.

Annak érdekében, hogy a kromatikus aberrációt, a szférikus aberrációt és a Petzvál-összeget egyszerre tudjuk minimalizálni, az alábbi célfüggvényt definiáltuk:

$$\begin{aligned} f(\mathbf{x}) = & c_1 \cdot (\text{focus}(\mathbf{x}, 656.3) - \text{focus}(\mathbf{x}, 486.1))^2 \\ & + c_2 \cdot s^2(\mathbf{x}) \\ & + c_3 \cdot k^2(\mathbf{x}), \end{aligned}$$

ahol a $\text{focus}(\mathbf{x}, y)$ függvény az \mathbf{x} paramétervektorral rendelkező objektív fókuszpontját adja meg y hullámhosszú sugarak esetén, $s(\mathbf{x})$ a szférikus aberráció értéke \mathbf{x} helyen, $k(\mathbf{x})$ -szel pedig a képmezőelhajlást jelöltük. A c_1, c_2 és c_3 konstansokat úgy választottuk meg, hogy mindhárom aberráció értéke a hozzá tartozó konstanssal szorozva nagyjából egy nagyságrendbe essen, majd az optimalizálási folyamat során változtatgattunk rajtuk az alapján, hogy melyik hiba kijavítására szeretnénk nagyobb hangsúlyt fektetni. Szintén folyamatosan finomhangoltuk azokat az intervallumokat, amelyekből a 12 paramétert (azaz a 6 görbületet és 6 távolságot) vesszük.

Összefoglalva tehát egy 12 változós, képlettel nem megadott ("black-box"), nem lineáris függvényt szeretttünk volna minimalizálni bizonyos korlátozások mellett. Ezek a korlátozások a következők:

- a megfelelő paraméterek essenek a megfelelő intervallumokba;

- az objektív legyen "értelmes", azaz lehessen úgy megválasztani a 13. paramétert (vagyis az utolsó felület görbületét), hogy a fókusz-távolság 100 mm legyen (ez a feltétel persze már eleve szükséges ahhoz, hogy az aberrációkat ki tudjuk értékelni);
- később az objektív fényerejét is figyelembe vettük, azaz előírtuk, hogy a gyújtótávolság és a legnagyobb hasznos átmérő hányadosa legfeljebb 2.8 legyen.

Annak érdekében, hogy az optimalizáló algoritmusunkat tesztelni tudjuk, kicseréltük az objektívben használt anyagokat. Így a kiinduló értékek kellően rosszak lettek ahhoz, hogy az algoritmus teljesítményét szemléltetni tudjuk rajtuk.

A feladat valójában nem a szó szoros értelmében vett globális optimalizálási probléma, ugyanis nem feltétlenül van szükségünk a globális optimum megtalálására, és még ha meg is találnánk azt, további nehéz feladat lenne bebizonyítani, hogy valóban globális optimumot találtunk. A célunk inkább az volt, hogy a kicserélt anyagokkal legalább annyira jó értékeket érjünk el, mint amilyenek az eredeti anyagokat használó, hasonló fényerejű Tessar objektívet jellemezték. Ennek érdekében célszerű legalábbis egy lokális optimumot megtalálni, és azok közül is minél jobbat.

A probléma megoldására több módszert is kipróbáltunk, de az egyetlen, amivel igazán jó eredményeket sikerült elérnünk, a Controlled Random Search algoritmus [14] volt. Erről a 7. fejezetben lesz bővebben szó.

6.1. Keresés a koordinátatengelyek mentén

Elsőnek az ún. Cyclic Coordinate Method-ot implementáltuk [2]. Ezen módszer során a koordinátatengelyek irányában végzünk vonalmenti kereséseket, valamilyen kezdőpontból kiindulva. Egyszerre mindig egy paramétert változtatunk, a többit rögzítjük, és így keressük meg a minimumot. Ezután áttérünk a következő paraméterre, és így tovább. Az algoritmus tehát az alábbi lépésekből áll:

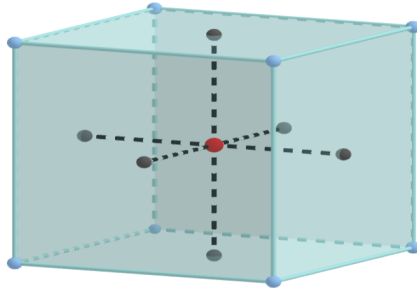
Inicializáló lépés: Válasszunk $\varepsilon > 0$ skalárt a leállási kritériumhoz, és legyenek $\mathbf{d}_1, \dots, \mathbf{d}_n$ a koordináta-irányok. Válasszunk egy kiindulópontot: \mathbf{x}_1 , legyen $\mathbf{y}_1 = \mathbf{x}_1$, legyen $k = j = 1$, és menjünk tovább a fő lépésre.

Fő lépés:

1. Legyen λ_j a $\min f(\mathbf{y}_j + \lambda \mathbf{d}_j)$, $\lambda \in E_1$ feladat egy optimális megoldása, és legyen $\mathbf{y}_{j+1} = \mathbf{y}_j + \lambda_j \mathbf{d}_j$. Ha $j < n$, akkor cseréljük j -t $j + 1$ -re, és ismételjük az 1. lépést. Különben, ha $j = n$, folytassuk a 2. lépéssel.
2. Legyen $\mathbf{x}_{k+1} = \mathbf{y}_{k+1}$. Ha $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| < \varepsilon$, akkor álljunk meg. Különben legyen $\mathbf{y}_1 = \mathbf{x}_{k+1}$, legyen $j = 1$, cseréljük k -t $k + 1$ -re és ismételjük az 1. lépést.

Kezdőpontnak az eredeti objektív adatait választva futtattuk az algoritmust, különböző intervallumokat választva korlátozó feltételnek. A kapott eredmények olyannyira nem lettek jók, hogy nem is érdemes közölni őket.

Ezután egy, az előzőhöz nagyon hasonló keresési módszert is kipróbáltunk. Ez a módszer is a koordinátatengelyek mentén végez vonalmenti kereséseket, de nem lép mindig át a következő pontba, hanem a kiindulópontból vizsgál meg minden koordinátaírányt, és miután az összeset végigvette, átlép az addig talált legjobb pontba. Tulajdonképpen egy 12 dimenziós téglá közepéből indulva vizsgálja át az összes koordinátaírányt a téglá falai között. (Három dimenzió esetén valahogy úgy nézne ki ez a folyamat, ahogy a 6.2. ábrán látszik.) Ezt a folyamatot ismételjük úgy, hogy az újabb téglá középpontjának az így megtalált legjobb pontot választjuk, és a téglák fokozatosan egyre kisebbek, a keresés pedig egyre pontosabb lesz. A vonalmenti keresések elvégzésére ebben és az előző esetben is az ún. Uniform Search módszert használtuk [2], itt egyszerűen apró részekre osztjuk az intervallumot, amelyben keresünk, és amelyik osztópontban a legkisebb a függvényérték, azt választjuk. Ezt a keresést egyszerűen tehetjük fokozatosan egyre pontosabbá: mindössze a felosztást kell finomítani hozzá.



6.2. ábra. Keresés a koordinátatengelyek mentén a téglán belül

Az algoritmusnak ezzel a verziójával az előbbinél valamivel jobb, de még mindig nem megfelelő eredményeket kaptunk, így ez a próbálkozás is kudarccal végződött.

7. fejezet

A Controlled Random Search algoritmus

A Controlled Random Search (röviden: CRS) algoritmust W. L. Price alkotta meg 1977-ben globális optimalizálási problémák megoldására [14]. Az algoritmus a *direkt módszerek* kategóriáján belül a *random keresések* közé sorolható. Nem tartozik tehát a garantáltan pontos megoldást ígérő módszerek közé, hanem heurisztika, de ennek köszönhetően problémák széles skálájához használható – nem szükséges hozzá a deriváltak ismerete, de még az sem, hogy a célfüggvény minden pontban folytonos legyen, továbbá korlátozó feltételek jelenléte sem akadály. Az algoritmus *globális szakaszában* egyenletes eloszlás szerint generálunk random pontokat a keresési tartományból, utána pedig egyfajta evolúciós módszerrel cserélgetjük a minta pontjait.

Tegyük fel, hogy egy n -változós f függvényt szeretnénk minimalizálni valamilyen V tartományon. Az algoritmus a következőképpen működik: válasszunk N random pontot V -ből egyenletes eloszlás szerint, ahol N előre adott szám, úgy, hogy ezek a pontok a korlátoknak megfeleljenek. A választott pontok mindegyikében értékeljük ki a függvényt, és a kapott értékeket tároljuk el egy A tömbben. Ezután minden iterációban választunk egy új P pontot a lehetséges választandó pontok halmazából – ez a halmaz az eddigi N db pont elhelyezkedése alapján kerül meghatározásra, később kitérek rá, hogy hogyan. Amennyiben ez a P pont is megfelel a korlátoknak, azaz V -be esik, itt is kiértékeljük a függvényt: a kapott függvényérték legyen f_p . Legyen M az a pont az eddigi tesztpontjaink közül, amelyhez a legnagyobb (tehát legrosszabb) függvényérték tartozik, ez az érték pedig legyen f_m . Ha $f_p < f_m$, akkor M -et kidobjuk az A tömbből, helyette pedig berakjuk P -t. Amennyiben viszont P nem felel meg a korlátoknak vagy $f_p > f_m$, akkor P -t elvetjük, és helyette másik pontot választunk a lehetséges új pontok halmazából.

Az algoritmus futása során a tárolt pontok általában klaszterbe rendeződnek olyan lokális minimumpontok körül, amelyek értéke kisebb az aktuális f_m -nél. Annak a valószínűsége, hogy a pontok végül a globális minimumhoz (vagy minimumokhoz) konvergálnak, függ N választásától, a függvény komplexitásától, a korlátozó feltételek milyenségétől és attól, hogy hogyan választjuk meg a lehetséges új pontok halmazát.

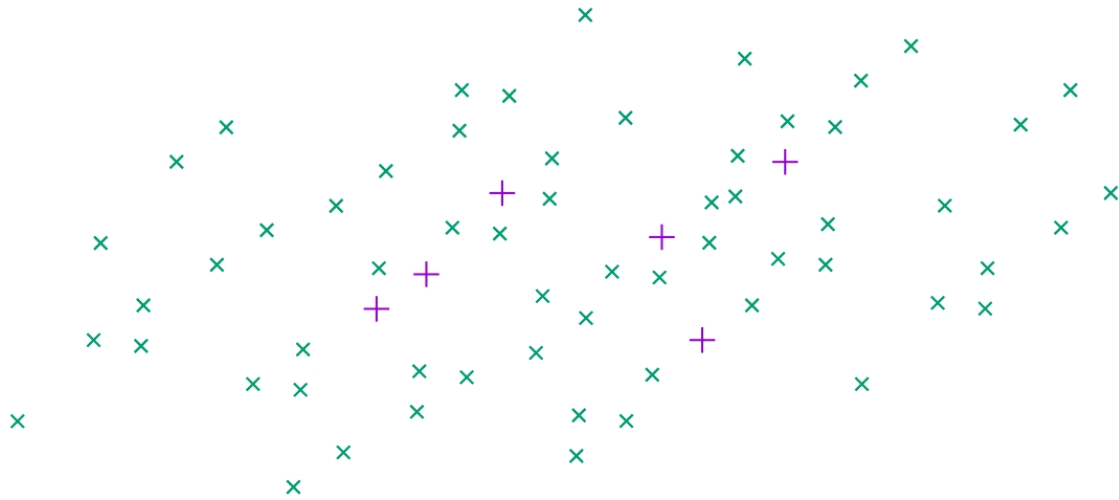
A lehetséges új pontok halmaza a Price által megalkotott verzióban az N jelenlegi pont konfigurációjától függ. Minden iterációban választunk $n + 1$ db pontot az N tesztpont közül véletlenszerűen: R_1, R_2, \dots, R_{n+1} . (Itt n tehát a változók száma, és N -et tanácsos úgy megválasztani, hogy $n \ll N$ teljesüljön.) Ez az $n + 1$ pont egy szimplexet alkot az n dimenziós térben. Válasszuk ki R_{n+1} -et a szimplex csúcsai közül, és tükrözzük a maradék n pont centroidjára, \bar{G} -re. Az így kapott P lesz az új tesztpontunk. Képlettel leírva tehát az alábbi módon kapjuk P -t:

$$\bar{P} = 2 \times \bar{G} - \bar{R}_{n+1}, \quad (7.1)$$

ahol

$$\bar{G} = \frac{\sum_{i=1}^n \bar{R}_i}{n}. \quad (7.2)$$

Az $n + 1$ db random pontot N pont közül $\binom{N}{n+1}$ -féleképpen választhatjuk ki, és mivel R_{n+1} választása is random (mivel az $n + 1$ pontot véletlen sorrendben választjuk), így a következőnek választható pontok halmaza $(n + 1) \cdot \binom{N}{n+1}$ méretű, és innen minden pontot egyforma valószínűséggel választunk. Az 7.1. ábrán a következőként választható pontok halmaza látható egy olyan esetben, amikor az algoritmust két dimenzióban futtatjuk, és 6 tesztpontunk van (ezek lila $+$ -jellel, a lehetséges választhatók pedig zöld x -szel vannak jelölve az ábrán).



7.1. ábra. A következő tesztpont lehetséges helyzetei

Az algoritmusban nincs meghatározva leállási kritérium – az író arra biztatja a módszert alkalmazókat, hogy az aktuális feladat alapján ők határozzák meg, meddig fusson az algoritmus. Lehet a kritérium például, hogy bizonyos számú iterációt futtassunk le, vagy hogy a célfüggvény-értékek az N tesztpontban kellően közel legyenek egymáshoz, vagy az N tesztpont helyezkedjen el egymáshoz kellően közel az n dimenziós térben.

7.1. A CRS algoritmus módosított verziója és a CRS2

Hat évvel a CRS algoritmus megalkotása után W. L. Price újabb cikket publikált, amelyben egyrészt továbbfejlesztette a CRS algoritmust, másrészt egy "második verzióval" is előállt, melyet CRS2-nek nevezett [15]. Mindkét változat az eredeti algoritmus azon gyengeségén hivatott javítani, miszerint az eredeti verzió erősen a keresésre van kihegyezve a konvergencia gyorsaságával szemben, és ez bizonyos alkalmazások esetén túl lassú konvergenciát eredményez. Az újabb verziók lényege, hogy megtalálják az egyensúlyt a hatékony keresés és a megfelelő gyorsaságú konvergencia között.

Először vessünk egy pillantást a továbbfejlesztett CRS algoritmusra. Ennek a lényege, hogy elsődleges tesztpontok mellett másodlagos tesztpontokat is definiál. Az eredeti algoritmus során az N tesztpontból random kiválasztjuk az R_1, \dots, R_{n+1} pontokat, majd az általuk alkotott szimplex egy random csúcsát tükrözzük a maradék n pont centroidjára:

$$\bar{P} = 2 \times \bar{G} - \bar{R}_{n+1}.$$

Ezt a P -t nevezzük *elsődleges tesztpontnak*. Az új verzióban emellé egy *másodlagos tesztpontot* is definiálunk az iteráció során:

$$\bar{Q} = (\bar{G} + \bar{R}_{n+1})/2.$$

Míg az elsődleges tesztpontok funkciója a keresés (P a szimplexen kívülre esik), addig a másodlagos tesztpontok a konvergenciát hivatottak gyorsítani (Q a szimplex belsejében van). Az algoritmus futtatása során számon tartjuk az addigi sikeres cserék arányát, azaz azt, hogy az összes addigi iterációnak hányadrészében sikerült olyan tesztpontot találnunk, amelyre $f_p < f_m$ teljesült, tehát amit meg tudtunk tartani. Ha ez az arány 50% alá megy, akkor a következő iterációban, amennyiben az nem sikeres, a másodlagos tesztpontot is megvizsgáljuk. Így a sikeres cserék aránya végig 50% körül lesz, fenntartva a keresés és a konvergencia közti egyensúlyt.

A CRS2 algoritmus ezzel szemben csak elsődleges tesztpontokat használ. Ez a változat mindössze annyiban különbözik az eredetitől, hogy R_1 választása nem random: R_1 -nek mindig azt a tesztpontot választjuk, amelyre a legkisebb a függvényérték az N tárolt pont közül. Ez a választás úgy befolyásolja a keresést, hogy az a legkisebb értékű pont környezete felé tendáljon, ezzel gyorsítva a konvergenciát. Felmerül ugyanakkor a veszély, amelyet minden globális keresési módszer magában hordoz: hogy a keresés benne ragad egy lokális minimumban. Ez a veszély N megfelelő megválasztásával csökkenthető: minél nagyobb N -et választunk, annál inkább átkerül a fókusz a gyors konvergenciáról a hatékony keresésre. A megfelelő N tehát a problémától függ, tapasztalataink függvényében határozhatjuk meg. A cikk írója ugyanakkor előáll egy javaslattal mindkét algoritmushoz: $N = 25n$ -et javasol a CRS, míg $N = 10(n + 1)$ -et a CRS2 esetében.

Price abban látja a CRS2 algoritmus fő előnyét a CRS-sel szemben, hogy kisebb a helyigénye, valamint egyszerűbb is az előbbinél, míg teljesítményben hasonló eredményeket produkál.

7.2. A Shuffled Complex Evolution módszer

Többek között a CRS-ből és a CRS2-ből kiindulva született meg 1993-ban a Shuffled Complex Evolution algoritmus, röviden SCE [6]. Ez a módszer négy olyan fogalmon alapul, amelyek hasznosnak bizonyultak globális optimalizálási problémák megoldásakor, ezek pedig:

- (a) random és determinisztikus megközelítések kombinálása;
- (b) a klaszterezés (kategorizálás) fogalma;
- (c) a teret feszítő pontok komplexusának szisztematikus evolúciója globálisan javító irányba;
- (d) kompetitív evolúció.

Az SCE módszer megalkotói úgy látták, hogy a CRS algoritmus, melyet kiindulópontként használtak, sok olyan tulajdonsággal rendelkezik, amelyek hasznosak globális optimalizálási problémák megoldásakor. Megállapították ugyanakkor azt is, hogy emellett a CRS-nek vannak gyenge pontjai. Az eredeti verzió egyformaként kezeli a tér minden pontját, a keresés nincs befolyásolva egyik vagy másik irányba valamilyen paraméter alapján. Ez, ahogy Price maga is megállapította, lassú konvergenciához vezethet. A CRS2 ugyanakkor nagy hangsúlyt helyez a pillanatnyi legjobb tesztpontra, ami magában hordozza azt a veszélyt, hogy az algoritmus megakad egy lokális minimumban. Ezenfelül az, hogy minden (sikeres) iterációban lecseréljük az addigi legrosszabb pontot, ahhoz vezet, hogy a pontok nagyon hamar egy kis régióra fognak korlátozódni, és így ismét könnyen beleragadhatunk egy lokális minimumba. Ezt pedig csak a tesztpontok nagyon nagy számával tudjuk elkerülni. Az SCE módszer megalkotói igyekeztek úgy átvenni a CRS algoritmus jó tulajdonságait, hogy közben a fenti gyengeségeket is orvosolják.

Így született az alábbi algoritmus [6]:

Inicializáló lépés: Válasszunk p -t és m -et úgy, hogy $p \geq 1$ és $m \geq n + 1$ teljesüljön: p lesz a komplexusok száma, m pedig a pontok száma az egyes komplexusokban. Számítsuk ki a minta elemszámát: $s = p \cdot m$.

Algoritmus:

1. Válasszunk s db random pontot V -ből: x_1, \dots, x_s . Minden x_i -re számoljuk ki a hozzá tartozó f_i függvényértéket.
2. Rendezzük a kiválasztott pontokat a függvényértékek szerint növekvő sorrendbe, és tároljuk el őket egy D tömbben: $D = \{x_i, f_i, i = 1, \dots, s\}$. Most tehát x_1 az a pont, amelyhez a legkisebb függvényérték tartozik.
3. Particionáljuk D -t p db, egyenként m pontot tartalmazó komplexusba: A^1, \dots, A^p az alábbi módon:

$$A^k = \{x_j^k, f_j^k \mid x_j^k = x_{k+p(j-1)}, f_j^k = f_{k+p(j-1)}, j = 1, \dots, m\}.$$

4. Minden $A^k, k = 1, \dots, p$ komplexusra alkalmazzuk a Competitive Complex Evolution (röviden CCE) algoritmust, melyet később külön ismertetünk.

5. Tegyük vissza az A^1, \dots, A^k komplexusok evolvált változatát a D tömbbe, majd rendezzük D elemeit a függvényértékek szerint növekvő sorrendbe.
6. Ellenőrizzük a leállási kritériumot.
 - Ha teljesül: STOP,
 - ha nem: ugorjunk a 3. lépésre.

A 4. lépésben minden komplexusra futtatjuk az ún. CCE algoritmust:

Inicializáló lépés: Válasszunk q -t, α -t és β -t úgy, hogy $2 \leq q \leq m$, $\alpha \geq 1$ és $\beta \geq 1$ teljesüljön.

Algoritmus:

1. Rendeljünk valószínűségértékeket A^k pontjaihoz az alábbi eloszlás szerint:

$$\rho_i = \frac{2(m+1-i)}{m(m+1)}, \quad i = 1, \dots, m.$$

Így a legnagyobb valószínűség az x_1^k ponthoz tartozik: $\rho_1 = 2/(m+1)$. A legkisebb valószínűséget pedig x_m^k -hoz rendeltük: $\rho_m = 2/m(m+1)$.

2. Válasszunk q random pontot A^k -ből a fent definiált eloszlás szerint (ezek lesznek a "szülő"): u_1, \dots, u_q . (Ez a q pont egy szubkomplexust definiál.) Tároljuk őket egy B tömbben: $B = \{u_i, v_i, i = 1, \dots, q\}$, ahol v_i az u_i ponthoz tartozó függvényérték. B pontjainak A^k -beli pozícióit tároljuk el egy L tömbben.
3. Hozzuk létre a leszármazottakat:

- (a) Rendezzük B -t és L -et úgy, hogy a q db pont függvényérték szerint növekvő sorrendben legyen. Határozzuk meg a centroidot:

$$g = \frac{1}{q-1} \sum_{j=1}^{q-1} u_j.$$

- (b) Határozzuk meg az új tesztpontot: $r = 2g - u_q$ (tükröző lépés).
- (c) Ha $r \in V$: számoljuk ki az f_r függvényértéket, és ugorjunk a (d) lépésre. Ha nem: határozzuk meg az A^k -t tartalmazó legkisebb $H \subset \mathbb{R}^n$ hiperkockát, generáljunk egy random $z \in H$ pontot, számítsuk ki az f_z függvényértéket, majd legyen $r = z$ és $f_r = f_z$ (mutációs lépés).
- (d) Ha $f_r < f_q$, akkor cseréljük u_q -t r -re, és ugorjunk az (f) lépésre. Ha nem, számoljuk ki $c = (g + u_q)/2$ -t és f_c -t (összehúzó lépés).
- (e) Ha $f_c < f_q$, cseréljük u_q -t c -re, és ugorjunk az (f) lépésre. Ha nem, generáljunk egy random $z \in H$ pontot, számítsuk ki az f_z függvényértéket (mutációs lépés). Cseréljük u_q -t z -re.
- (f) Ismételjük az (a)-(e) lépéseket α -szor, ahol $\alpha \geq 1$ a felhasználó által meghatározott paraméter.

4. Cseréljük a szülőket a gyerekeikre, azaz tegyük be B elemeit A^k -ba az L tömbben eltárolt pozíciókba. Rendezzük A^k elemeit a függvényértékek szerint növekvő sorrendbe.
5. Ismételjük az 1-4. lépéseket β -szor, ahol $\beta \geq 1$ a felhasználó által megadott paraméter. β azt határozza meg, hogy hány gyereket generáljunk, azaz milyen mértékben evolváljunk a komplexust.

Az SCE módszer úgy kezeli a globális optimalizálást, mint egy természetes evolúciós folyamatot: az eredeti s db tesztpont felel meg a populációnak, amelyet közösségekre osztunk fel (ezek a komplexusok). Minden egyes közösség egymástól függetlenül fejlődik (ez felel meg a tér különböző irányában történő keresésnek). Bizonyos generációnyi fejlődés után a közösségek összekeverednek, és új közösségekre osztjuk a populációt. Ez a folyamat növeli a populáció életképességét az információk megosztása által, melyet az egyes közösségek egymástól függetlenül szereztek.

A közösségek (komplexusok) minden tagja potenciális szülő, azaz megvan a lehetősége arra, hogy részt vegyen a reprodukciós folyamatban. Az egyes komplexusokból kiválasztott szubkomplexusok olyanok, mint egy szülőpár, azzal a különbséggel, hogy egy szubkomplexus állhat 2-nél több pontból. Annak érdekében, hogy az evolúciós folyamat kompetitív legyen, a jobb tulajdonságokkal rendelkező szülők nagyobb valószínűséggel vesznek részt a gyerekek létrehozásában, mint a rosszabbak – ezt biztosítja a pontokhoz rendelt eloszlás. Az egyes szubkomplexusokban a leszármazottak generálása a Nelder és Mead által megalkotott, a CRS-hez hasonló módszer szerint történik [11], de ezenfelül bizonyos esetekben random leszármazottak is generálódnak, amely jelenség valamilyen szinten hasonlít a természetes biológiai folyamatok során történő mutációhoz. A mutációs lépések növelik a populáció által tárolt információ mértékét. Az újonnan létrejött leszármazottak a szubkomplexus legrosszabb függvényértékű pontjának helyét foglalják el, nem pedig a teljes populáció legrosszabb pontját, ezzel biztosítva, hogy minden szülőnek lesz esélye a szaporodásra, mielőtt kidobnánk vagy lecserélnénk. Így a mintában tárolt információ semkora részét nem hagyjuk figyelmen kívül.

A módszer megalkotói két verzióját ismertették az SCE módszernek: Az SCE1, valamint az SCE2 algoritmust.

Az SCE1 algoritmus az a verziója az SCE módszernek, ahol a teljes, s méretű minta egy komplexust alkot, tehát $m = s$, és $p = 1$. Az egy iterációban generált leszármazottak száma megegyezik a komplexus méretével: $\beta = m$. Ez az algoritmus csupán a komplexus evolúciójának módjában különbözik a CRS2-től, a leszármazottak generálására kiválasztott szubkomplexus mérete is megegyezik a a Nelder és Mead által meghatározott értékkel, amelyet a CRS is használ, vagyis $n + 1$ -gyel [11]. Az α paraméter értékét is 1-re állítjuk (tehát minden szimplex evolúciója egy lépésből áll), így ebben is hasonlít a módszer a CRS-re. Az SCE módszer ezen verziójában a legfontosabb beállítandó paraméter a minta mérete (s).

Az SCE2 algoritmus során a komplexusok méretét $m = 2n + 1$ -nek választjuk, ahol n a dimenziót jelöli. A leszármazottak száma (β), melyek két keveredés között generálódnak az egyes komplexusokban egymástól függetlenül, legyen egyenlő a komplexusok méretével, azaz legyen szintén $2n + 1$. A gyerekek generálásához kiválasztott szubkomplexusok mérete legyen ismét

$n + 1$, és α -t ismét állítsuk 1-re. Az SCE2 algoritmusban a legfontosabb meghatározandó paraméter a komplexusok száma (p).

7.3. A CRS algoritmussal elért eredmények

A Controlled Random Search algoritmus implementálása során sokféle kombinációját kipróbáltuk a paramétereknek, és több különböző leállási kritériumot is kipróbáltunk.

Az első leállási kritériumunk az volt, hogy a tesztpontok a tér kellően kis részébe tömörüljenek: azaz a legnagyobb eltérés két tesztpont között a görbületi sugaraknak megfelelő koordinátákban 0.001-nél, a felületek közötti távolságoknak megfelelő koordinátákban 0.1-nél kisebb legyen. Ezen kritérium teljesülése kellő bizonyosságot nyújt arról, hogy valóban elértünk egy lokális minimum-pontba. A gyakorlatban ez nem igazán vált be, ugyanis ahhoz, hogy teljesüljön, hosszú órákig kellett futtatni a programot, miközben a célfüggvény gyakorlatilag már párezer lépés után nem változott.

Ezért vezettünk be később egy új leállási kritériumot: akkor álljon le az algoritmus futása, ha a tesztpontok célfüggvényértékei között a legnagyobb különbség kisebb, mint 0.01. Ez a kritérium, bár az előbbinél kevésbé meggyőző bizonyítéka egy lokális minimum elérésének, a gyakorlatban sokkal jobban bevált.

Legtöbbször $N = 50$ tesztponttal futtattuk az algoritmust, de sok más értéket is kipróbáltunk, köztük a Price által konyhaszabályként javasolt értékeket is (a CRS eredeti verziójához ez $N = 25n$, míg a CRS2-höz $N = 10(n + 1)$). Az eredmények nem javultak több tesztpont használata esetén, azonban a futási idő jelentősen megnőtt, így az $N = 50$ érték bizonyult a legjobbnak a feladatunkhoz.

Az eredeti CRS algoritmus mellett a CRS módosított verzióját is implementáltuk, ezzel azonban az eredetihez képest gyengébb eredményeket értünk el.

7.3.1. Az 5.5 fényerejű Tessar

Egy objektív *fényerejét* úgy tudjuk kiszámolni, hogy a fókusztávolságot elosztjuk az objektív legnagyobb hasznos átmérőjével [7]. Minél kisebb ez az érték, annál több fényt képes befogadni az objektív. Átlagos objektívek fényereje általában 2.8 vagy 3.5, de profi kamerák gyakran 1.4-es, 1.8-as fényerővel is rendelkezhetnek.

Az eredeti, 1903-ban szabadalmaztatott Tessar objektív fényereje 5.5 volt, így először ezt a fényerőt írtuk elő az optimalizálás során. Célunk az volt, hogy az eredeti objektív hibaértékeihez hasonlókat érjünk el úgy, hogy a lencsék anyagát kicseréljük, ezzel elrontva az objektív teljesítményét.

Az eredeti objektívben az aberrációk értékei az alábbiak voltak:

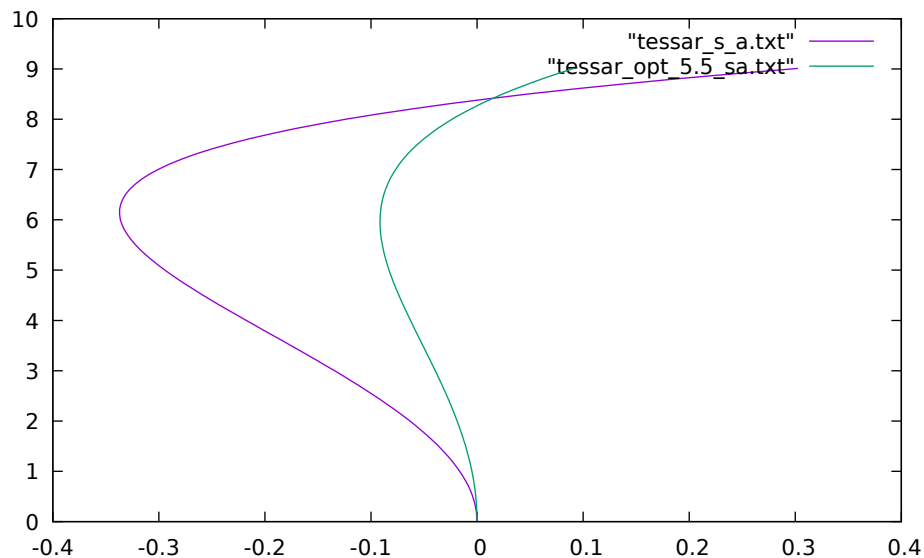
- kromatikus aberráció: 0.0676,
- szférikus aberráció: 0.3371,
- képmezőelhajlás: 0.0030.

(Itt a kromatikus aberráció és a szférikus aberráció mm-ben értendő, míg a képmezőelhajlás görbületet mér, tehát a mértékegysége $\frac{1}{\text{mm}}$.)

Az általunk optimalizált, 5.5-ös fényerejű, a kicserélt anyagokat használó objektívben a hibadatok az alábbi módon alakultak:

- kromatikus aberráció: 0.0043,
- szférikus aberráció: 0.0915,
- képmezőelhajlás: $7.8 \cdot 10^{-6}$.

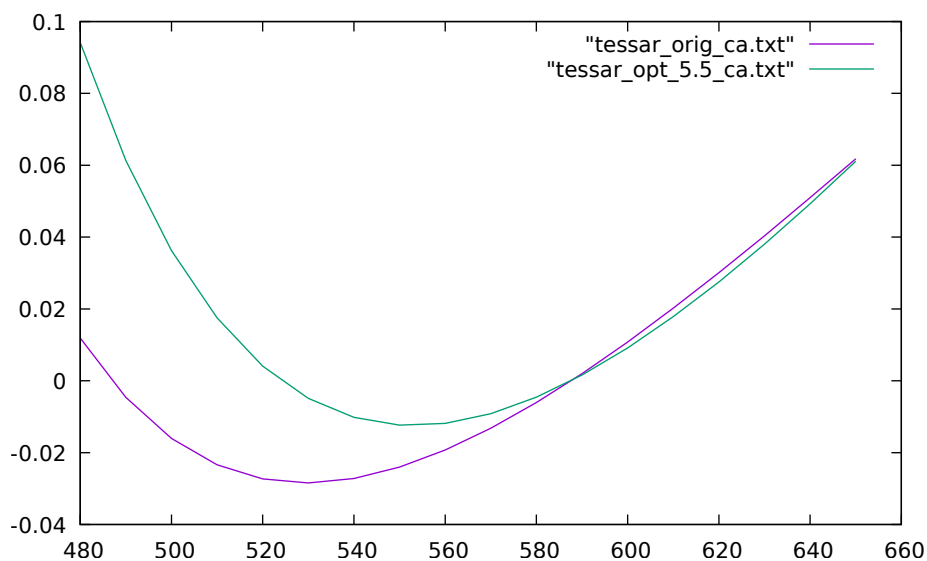
Az eredeti és az optimalizált objektívek optikai aberrációit összehasonlító ábrákon is szemléltettük. A 7.2. ábrán lilával az eredeti, zölddel az általunk optimalizált objektív szférikus aberrációját ábrázoltuk a 3. fejezetben bemutatott módon.



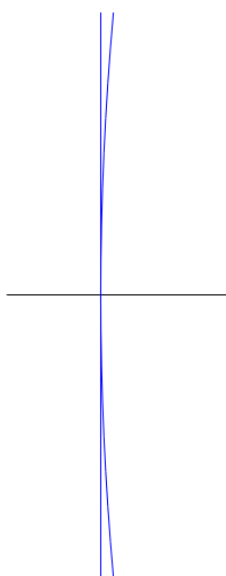
7.2. ábra. Az eredeti objektív és a kicserélt anyagokkal optimalizált objektív szférikus aberrációjának összehasonlítása

A 7.3. ábrán a különböző hullámhosszakhoz tartozó fókuszpontok távolságát szemléltettük a 587.6 nm-es hullámhosszú sugarak fókuszától. Zölddel ismét az általunk optimalizált objektívhez tartozó görbét színeztük, ezen látszik, hogy az objektív akromatikus.

A 7.4. ábrán a bal oldali görbe mutatja az általunk optimalizált objektív képmezőelhajlását, azaz azt, hogy mekkora a fókuszfelület görbülete (annyira kicsi a görbület, hogy a felület gyakorlatilag sík). A jobb oldali görbe tartozik az eredeti Tessarhoz.



7.3. ábra. Az eredeti objektív és a kicserélt anyagokkal optimalizált objektív kromatikus aberrációjának összehasonlítása



7.4. ábra. Az eredeti objektív és a kicserélt anyagokkal optimalizált objektív képmezőelhajlásának összehasonlítása

7.3.2. A 2.8 fényerejű Tessar

Később az optimalizálás során azt írtuk elő, hogy az optimalizált objektív fényereje 2.8 legyen. Az anyagokat ismét kicseréltük az objektívben, de ezúttal a fényerő különbsége miatt nem lett volna értelme az elért eredményeket az eredeti Tessarral összehasonlítani. Így a célunk ezúttal az volt, hogy egy 2.8-as fényerejű objektív hibaértékeihez hasonlókat érjünk el. Ehhez az összehasonlításhoz egy Olympus Tessar típusú objektívet választottunk [8].

Az Olympus Tessarban a vizsgált aberrációk mértéke az alábbiakban látható:

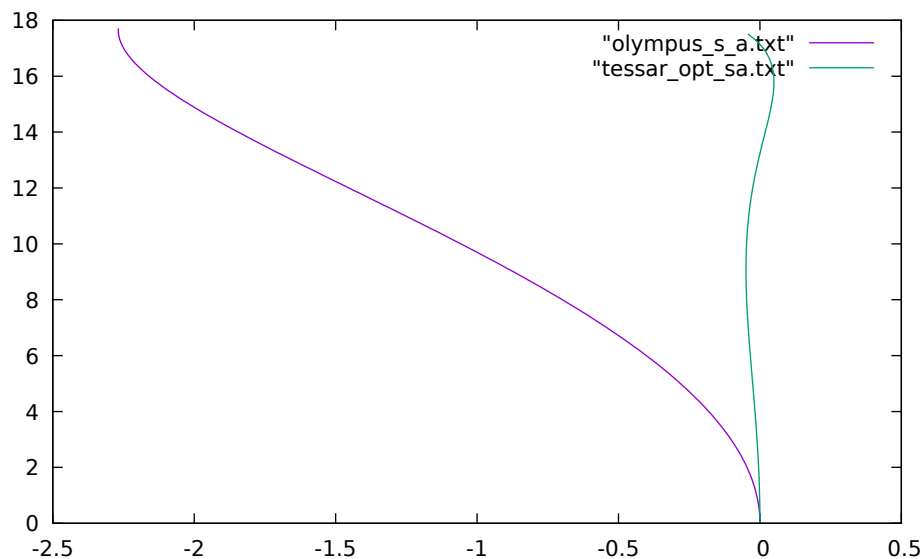
- kromatikus aberráció: 0.4706,
- szférikus aberráció: 2.2683,
- képmezőelhajlás: 0.0027.

Az optimalizálás során elért hibaértékek pedig az alábbiak szerint alakultak:

- kromatikus aberráció: 0.0019,
- szférikus aberráció: 0.0495,
- képmezőelhajlás: 0.0039.

Mivel az algoritmus random, így minden egyes futtatásnál más és más eredményeket kaptunk, ezek azonban nem tértek el egymástól jelentősen a paraméterek ideális beállítása után. Az itt bemutatott objektív, amelyet az optimalizálás eredményeképpen kaptunk, egy általunk kiválasztott verzió a sok közül.

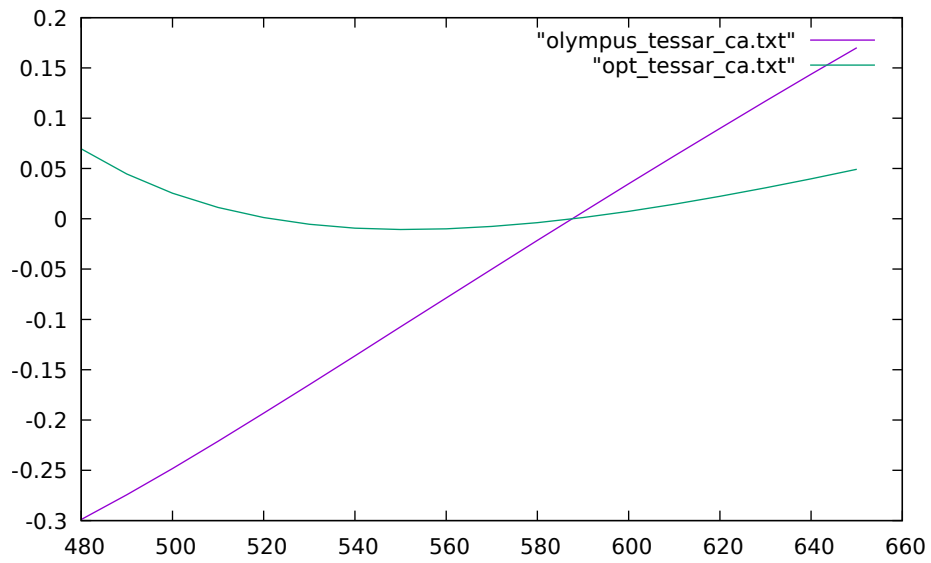
A 7.5. és a 7.6. ábrákon a fentiekhez hasonlóan szemléltettük a két objektív szférikus aberrációját, illetve kromatikus aberrációját, lilával az Olympus Tessarhoz, zölddel az optimalizált objektívhez tartozó görbét színezve.



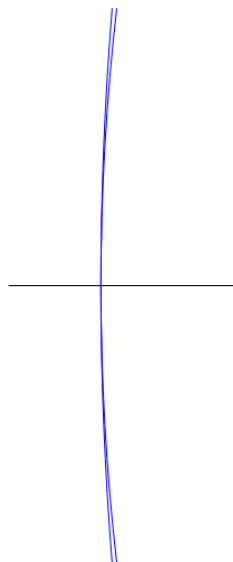
7.5. ábra. Az Olympus Tessar és a kicserélt anyagokkal optimalizált objektív szférikus aberrációjának összehasonlítása

A 7.7. ábrán a jobb oldali görbe szemlélteti az általunk optimalizált objektív képmezőelhajlását. Látszik, hogy bár az Olympus Tessarénál nagyobb a fókuszfelület görbülete, a kapott felület így is szinte sík.

Összefoglalva tehát mindkét fényerő esetében sikerült elérnünk azt a célt, hogy az anyagok kicserélése által elrontott objektíveket annyira feljavítsuk, hogy a hibaértékeik legalább olyan jók legyenek, mint egy hasonló fényerejű Tessar objektívben. Sőt, majdnem minden hibaérték jelentősen kisebb is lett, mint az eredeti verziókban.



7.6. ábra. Az Olympus Tessar és a kicserélt anyagokkal optimalizált objektív kromatikus aberrációjának összehasonlítása



7.7. ábra. Az Olympus Tessar és a kicserélt anyagokkal optimalizált objektív képmezőelhajlásának összehasonlítása

8. fejezet

Implementálás

Az alábbiakban a dolgozat első felében tárgyalt modellek C++-beli implementálásáról lesz szó. Összesen két modellező programot készítettünk, egyet az ideális képalkotás, egyet pedig a sugárkövetéses modellezés alapján. Ez utóbbit használtuk az eddigiekben bemutatott ábrák létrehozására, valamint az optimalizáló algoritmusok implementálására is, ezzel értük el a fent részletesen ismertetett eredményeket.

8.1. Ideális képalkotás

Kiindulásként egy C++ osztályt hoztunk létre, amely az ideális képalkotást modellezi. Az osztályt használó program alkalmas arra, hogy

- egy optikai rendszert jellemző mátrix elemeinek megadásával modellezze a rendszer képalkotását, azaz megállapítsa adott pont képét a rendszer szerint, illetve kiszámolja annak fókuszpontjait, főtávolságait és fókusz távolságait;
- egy rendszer fókuszpontjainak és főtávolságainak ismeretében kiszámolja a rendszert jellemző mátrix elemeit;
- egy felület előjeles görbületének, elhelyezkedésének és az általa elválasztott anyagok törésmutatójának ismeretében megadja a felületet jellemző mátrixot;
- több felület adatainak ismeretében modellezze az eredő rendszert, amely a felületek egymás után helyeződésével adódik;
- modellezze adott rendszer optikai tengely mentén való eltoltját.

A gyakorlatban ez a modell csak az optikai tengely közvetlen közelében pontos, nem veszi figyelembe a képalkotás hibáit, így optimalizálásra sem alkalmas.

8.2. Sugárkövetéses modell

A második C++ program már jóval több mindenre használható. Modellezhetünk vele sík- és gömbfelületű lencséből álló optikai rendszereket, szimulálhatjuk a rajtuk áthaladó sugarakat, alkalmas arra, hogy a lencserendszereket optikai aberrációk szempontjából analizáljuk, hogy a hibák mértékét szemléltessük, valamint a Cairo könyvtár [4] segítségével sugarak és az objektívek vizualizációjára is használható. Fő funkciója a lencserendszerek optimalizációja. Alkalmazására számos példát láthattunk az eddigi fejezetekben, most bemutatjuk a kód felépítését.

A program legfőbb elemei a sugarakat modellező *Ray* rekord, a felületeket modellező *Surface* osztály, az objektívek leírására alkalmas *Obj* osztály és az anyagokra vonatkozó *Material* rekord. Mindez a három dimenziós euklideszi térben értendő, ahol a koordinátarendszer egységei 1 mm-nek felelnek meg.

- A *Ray* struktúra három paraméterrel adható meg, egy kiindulóponttal, egy irányvektorral és a mikrométerben mért hullámhosszával. Legfontosabb hozzá tartozó függvény:
 - *intersection*, amely két sugár metszéspontját adja meg.
- A *Surface* objektumnak két származtatott osztálya van: a *FlatSurface* és a *SphericalSurface* osztályok. Előbbi sík-, utóbbi gömbfelületek modellezésére alkalmas. Egy síkfelületet a következő paraméterekkel lehet megadni: a felület normálvektora, egy pontja, a mérete, valamint az általa elválasztott anyagok levegőhöz mért relatív törésmutatója. Egy gömbfelület megadásához a szükséges paraméterek: gömbi középpont, sugár hossza, a gömbi középpontból a felület középpontja felé mutató vektor ("irány"), nyílásszög, valamint a síkfelülethez hasonlóan a törésmutatók. Mindkét osztályhoz az alábbi függvények tartoznak:
 - az *intersection* függvény a felület és egy adott sugár metszéspontját adja vissza,
 - a *goThrough* függvény egy sugarat kap paraméterként, visszatérési értéke pedig az a sugár, amit akkor kapunk, ha az előbbi áthalad a felületen,
 - *draw*: a cairo program segítségével lerajzolja a felület metszetét az $y = 0$ síkkal.
- A *Material* rekord modellezi az anyagokat, melyekből a lencsék készülnek. Ezen rekord segítségével azonfelül, hogy egy lencse anyagát egyszerűen a nevével megadhatjuk, figyelembe tudjuk venni a különböző hullámhosszú sugarak eltérő viselkedését, vagyis azt, hogy az anyagok törésmutatója a hullámhossz függvénye. A törésmutató számítását a program a Sellmeier-egyenlettel végzi, amihez azonban hozzáférésre van szüksége az egyes anyagokhoz tartozó paraméterekhez minden használt anyagnál. Ezen adatokat egy online adatbázisból [13] nyertük ki, melyet a *bs4* modulból származó *BeautifulSoup* osztály felhasználásával írt Python program segítségével elemeztünk ki. Az adatokat egy JSON fájlban gyűjtöttük össze, amelyet a program egyszer olvas be, majd a *Material* osztály egy statikus adattagjaként tárol.
A *Material* osztály egy példánya egy paraméterrel hozható létre: az üveg nevével (pl. "N-SF14"). Az osztály legfontosabb tagfüggvénye:

- A n_i függvény meghatározza az anyag törésmutatóját a fénysugár hullámhossza alapján (alapértelmezés szerint 587.6 nm).
- Az *Obj* osztály lencserendszereket ír le. Tulajdonképpen nem más, mint egymás után pakolt, az x tengelyre merőlegesen és szimmetrikusan elhelyezkedő felületek összessége. Legfontosabb hozzá tartozó függvények:
 - *readObjective*: Számos konkrét objektív adatai álltak rendelkezésünkre, ezen eljárás használatával tudtuk beolvasni egy lencserendszer adatait egy bizonyos formátumú szövegfájlból, és létrehozni a neki megfelelő modellt.
 - *drawPaths*: Egy vektorban tárolt sugár-sereg adott lencserendszeren átmenő útjának kirajzolását végezhetjük ezzel az eljárással egy pdf fájlba.
 - *focus*: adott hullámhossz alapján megadja az optikai rendszer második fókuszpontjának helyét.
 - *focalLength*: ez a függvény a második, azaz képoldali fókusztávolságot adja meg adott hullámhossz esetén.

Kitekintés

Dolgozatomban bemutattam egy megközelítést arra vonatkozóan, hogy hogyan lehet lencserendszereket számítógéppel modellezni, és a képalkotásuk hibáit elemezni, valamint a paramétereiket optimalizálni. Foglalkoztunk az ideális képalkotással, azzal az egyszerű modellel, amely lehetővé teszi, hogy háromdimenziós mátrixok segítségével jellemezzük optikai rendszerek leképezését. Mivel azonban ez a közelítés nem volt alkalmas az optikai aberrációk vizsgálatára, ezért áttértünk a sugárkövetéses modellezésre. A modellező program elkészítése után megvizsgáltuk a globális optimalizálási feladat tulajdonságait, amelyhez jutottunk, és a lehetséges megoldási módszerek közül kiválasztottunk egy random keresési algoritmust, amelynek segítségével elértük a projekt célját: sikerült olyan eredményeket kapnunk az optimalizálási feladatra, amilyeneket szeretttünk volna.

Úgy fogalmaztam, hogy ezzel elértük a célunkat – ez azonban nem jelenti azt, hogy ne lehetne innen még továbbhaladni. Érdekes lenne a vizsgált aberrációkon felül más képalkotási hibákat is bevenni a modellezésbe és az optimalizálásba egyaránt, ezzel hatékonyabbá téve az optimalizáló programot. Nagy fejlődés lenne egy jól használható user interface létrehozása is, ahol a felhasználó tetszőleges paraméterek – úgymint lencsék görbülete, vastagsága, elhelyezkedése és anyaga – megadásával vizualizációkat, ábrákat, grafikonokat generálhatna a lencserendszer tulajdonságairól, valamint egy gombnyomással optimalizálhatná is azt a kívánt paraméterekre. Lehetne ezenfelül kísérletezni más optimalizáló algoritmusok implementálásával is a feladatra, például azzal a Shuffled Complex Evolution-módszerrel, amelyet a Controlled Random Search rokonaként ismertettünk a szakdolgozatban. A lehetőségek tárháza végtelen egy ilyen sokfelé ágazó téma esetében.

Irodalom

- [1] *Amateur Telescope Optics*. URL: <https://www.telescope-optics.net/>.
- [2] M. S. Bazaraa, H. D. Sherali és C. M. Shetty. *Nonlinear Programming – Theory and Algorithms*. John Wiley & Sons, Inc., 1993. ISBN: 0-471-55793-5.
- [3] Max Born és Emil Wolf. *Principles of Optics*. Cambridge University Press, 1959. ISBN: 978-1-108-47743-7.
- [4] *Cairo Graphics*. URL: <https://www.cairographics.org/>.
- [5] *Csillagászati műszertechnika online jegyzet*. URL: https://titan.physx.u-szeged.hu/~pierre/muszertech_1/I-02/.
- [6] Q.Y. Duan, V.K. Gupta és S. Sorooshian. „Shuffled complex evolution approach for effective and efficient global minimization”. *Journal of Optimization Theory and Applications* 76 (1993), 501–521. old. DOI: <https://doi.org/10.1007/BF00939380>.
- [7] John E. Greivenkamp. *Field guide to geometrical optics*. SPIE Bellingham, WA, 2004. ISBN: 0-8194-5294-7.
- [8] Toshihiro Imai. *Behind-stop tessar type photographic lens system*. US Patent No. 3,895,857. 1975. júl. URL: <https://patentimages.storage.googleapis.com/43/f8/3d/ff1d13421f338e/US3895857.pdf>.
- [9] Rudolf Kingslake. *A history of the photographic lens*. Elsevier, 1989. ISBN: 0-12-408640-3.
- [10] Alistair Kwan, John Dudley és Eric Lantz. „Who really discovered Snell’s law?”: *Physics World* 15.4 (2002), 64. old. DOI: [10.1088/2058-7058/15/4/44](https://doi.org/10.1088/2058-7058/15/4/44).
- [11] John A. Nelder és Roger Mead. „A simplex method for function minimization”. *The computer journal* 7.4 (1965), 308–313. old.
- [12] Richard P. Paul. *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators*. The MIT Press, 1981. ISBN: 0-262-16082-X.
- [13] Mikhail Polyanskiy. *refractiveindex.info - online database*. URL: <https://github.com/polyanskiy/refractiveindex.info-database>.
- [14] W. L. Price. „A controlled random search procedure for global optimisation”. *The Computer Journal* 20.4 (1977), 367–370. old. DOI: <https://doi.org/10.1093/comjnl/20.4.367>.
- [15] W. L. Price. „Global Optimization by Controlled Random Search”. *Journal of Optimization Theory and Applications* 40 (1983), 333–348. old. DOI: <https://doi.org/10.1007/BF00933504>.

- [16] Paul Rudolph. *Photographic Objective*. US Patent No. 721,240. 1903. febr. URL: <https://patentimages.storage.googleapis.com/5b/da/59/4075b2ac2d340e/US721240.pdf>.
- [17] José Sasián. *Introduction to aberrations in optical imaging systems*. Cambridge University Press, 2013. ISBN: 978-1-107-00633-1.
- [18] Wilhelm Sellmeier. „Ueber die durch die Aetherschwingungen erregten Mitschwingungen der Körpertheilchen und deren Rückwirkung auf die ersteren, besonders zur Erklärung der Dispersion und ihrer Anomalien”. *Annalen der Physik* 223.12 (1872). DOI: <https://doi.org/10.1002/andp.18722231203>.
- [19] Aimo Törn és Antanas Žilinskas. *Global optimization*. 350. köt. Springer, 1989. ISBN: 978-3-540-50871-7. DOI: <https://doi.org/10.1007/3-540-50871-6>.