Project in FAFF25

Niklas Sundin Johansson, dt08njo
Lars Gustafson, ada10lgu

March 7, 2017

1 Projection Error

1.1 Implemation

1.2 Theroy

Som Newton upptckte s har olika vglngder av ljus olika brytningsindex i samma material, detta kallas kromatisk aberration. Fel kan uppst ven med monokromatisk ljus vid anvndandet av sfriska prismor, detta kallas d sfrisk aberration. I uppgiften skall man med hjlp av, s kallad, ray tracing berkna de fel som genereras. Genom att simulera standardstrlar och berkna dess bana.

Brytningslagen, $n_1 \cdot sin(\alpha_1) = n_2 \cdot sin(\alpha_2)$ anvnds fr att berkna var fokalpunkten hamnar beroende p de tv mediernas brytningsindex samt in och utfallsvinklarna vid vergngen.

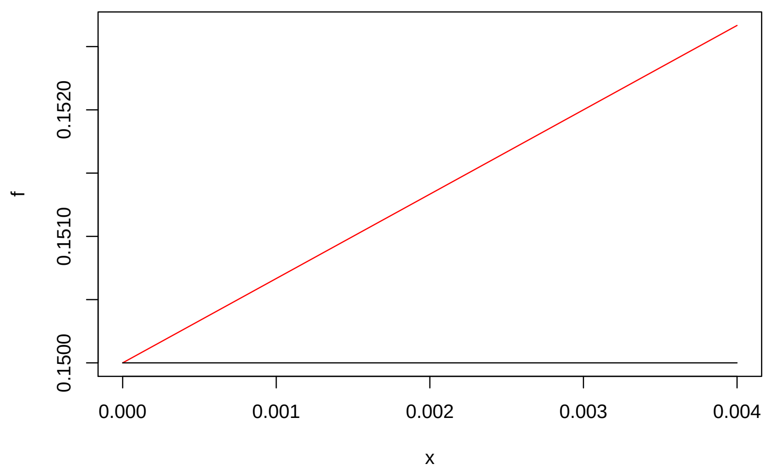
Materialet BK7 anvnds d kromatisk abberation skall berknas. Formeln fr dess brytningsindex r som nedan och gavs av uppgiften.
$n^2 = a_1 + a_2\lambda^2 + a_3\lambda^{-2} + a_4\lambda^{-4} + a_5\lambda^{-6} + a_6\lambda^{-8}$
$a_1 = 2,271176$
$a_2 = -9.700709 \cdot 10^{-3} * \mu m^{-2}$
$a_3 = 0.0110971 \cdot \mu m^2$
$a_4 = 4.622809 \cdot 10^{-5} \cdot \mu m^4$
$a_5 = 1.616105 \cdot 10^{-5} \cdot \mu m^6$
$a_6 = -8.285043 \cdot 10^{-7} \cdot \mu m^8$

1.3 Method

Frst berknas felet, som en sfrisk prism genererar nr man kar hjden,h, frn den optiska axeln. Detta jmnfrs med

1.4 Result

1.4.1 A: Paraxial Approximation



With Paraxial Approximation focus point always becomes 0.15 m, independent of where on the lins a beam parallel with the optic normal axis refract on the lens. Mean while without the approximation it will drift futher away from the lins when the standard beam closes the edge of the lens.

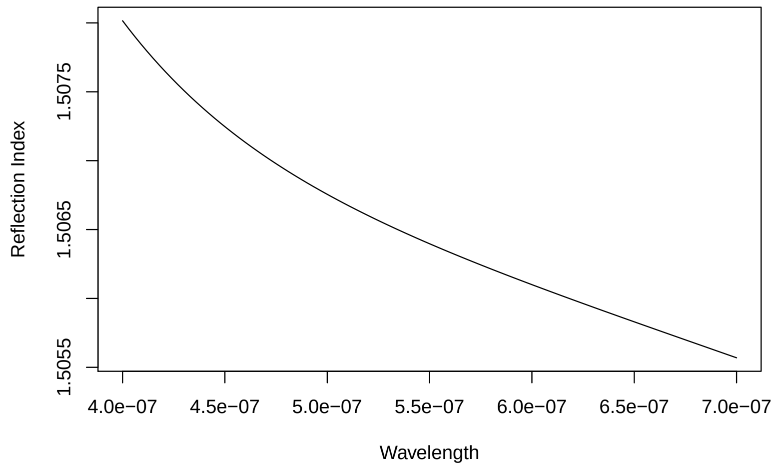
$$write$$

$$down$$

$$difference$$

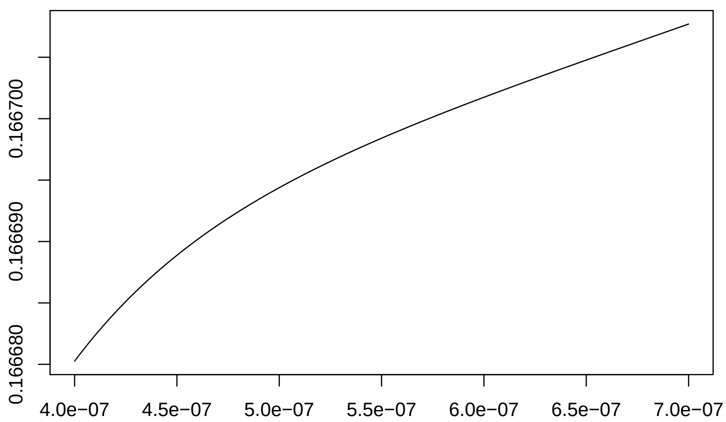
$$[placeholder]$$

1.4.2 B: Material Replacement



When plotting BK7 refraction index compared to wavelength of the incoming light, the graph shows that Δ 0.0025 in index between lowest value $400nm$ and highest $700nm$. The function is close to liner at the high spectrum, which is wort noting.

1.4.3 C: Chromatic Aborations



Applying the BK7 material on the none Paraxial Approximated function, gives us an Δ in focus point of close to 0.00028. Generally close to 0.15

1.5 Conclusion and Commentary

Replacement material $BK7$ has the properties to almost bring focus back to 0.15, where it was when we applied Paraxial Approimation. Conclusion with the new material we can easy apply Paraxial Approximation without to large errors in calculations, depending on specification limitation of cource.

1.6 Conclusion

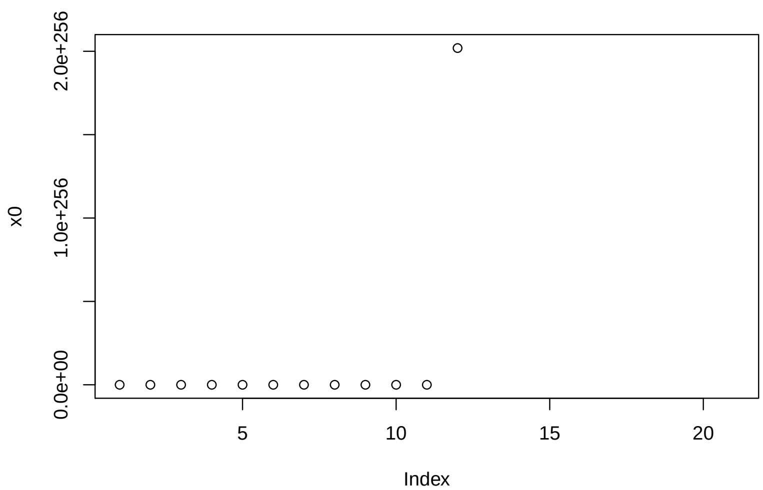
2 Laser Pulse

2.1 Implemation

2.2 Result

2.2.1 A: Nummeric Solution

2.2.2 B: Differential Plot



2.3 Conclusion and Commentary

2.4 Conclusion

A Implementation R Code

A.1 Assignment I:

```r

R <- 0.15                                       #Radious
D <- 0.1                                        #Diameter of lens
n1 <- 1.0                                       #Refraction index
n2 <- 1.5                                       #Refraction index
#Gaussian function (Radious,Hight,incoming refraction index, material refraction index, Use approximation BOOLEAN)
Gaussian <- function(r,h,n1,n2,b){
        a1 = 0                                  #Paraxial Approximation
        if(!b){
                a1 = alph1(h,r)         #None Approximated Angle
        }
        a2 = alph2(r,a1,n1,n2)
        f = r*sin(a2)/cos(a2)+r
        return(f)
}

#Refraction angle to norm of surface
alph2 <- function(r,a1,n1,n2){
        a2 = asin(sin(a1)*(n1/n2))
        return(a2)
}
#Light angle without Paraxial Approxation to norm of surface
alph1 <- function(h,r){
        a1 = asin(h/r)
        return(a1)
}

#Refreaction index calculation of Glass material BK7
BK7n <- function(x){
        a1 = 2.271176
        a2 = -9.700709*(10^-3)*(10^-6)^-2
        a3 = 0.0110971*(10^-6)*(10^-6)^2
        a4 = 4.622809*(10^-5)*(10^-6)^4
        a5 = 1.616105*(10^-5)*(10^-6)^6
        a6 = -8.285043*(10^-7)*(10^-6)^8
        n2 = a1+a2*x^2+a3*x^(-2)+a4*x^(-4)+a5*x^(-6)+a6*x^(-8)
        n = a1;
        n = abs(sqrt(as.complex(n2)));
        return(n);
}
par(mfrow = c(2,3));

#Paraxoide Approximation applied
Gauss_Approx <- function(x) Gaussian(R,x,n1,n2,TRUE);
fa <- Vectorize(Gauss_Approx);
plot.function(fa, from=0, to=D/2, xlab="Hight", ylab="Focus␣Point",
                ylim=c(fa(0)-0.01,0.20));

#No Paraxoide Approximation
Gauss <- function(x) Gaussian(R,x,n1,n2,FALSE);
f <- Vectorize(Gauss);
plot.function(f,from=0,to=D/2, add=TRUE, col="red");

#BK7n Reflection index
n2v <- Vectorize(BK7n);
plot.function(n2v, from=(400/(10^9)), to=(700/(10^9)), ylab="Reflection␣Index", xlab="Wavelength");

#BK7 replace material of lens
h <- 0.025;
f_chrom <- function(x){
        f = Gaussian(R, h, n1, BK7n(x), FALSE);
        return(f)
}
v_chrom <- Vectorize(f_chrom);
plot.function(v_chrom, from=(400/(10^9)), to=(700/(10^9)), xlab="", ylab="");

#Assignment no. 2
L <- 0.2                        #Length
```

A.2 Assignment II:

```
L <- 0.2                        #Length
D <- 0.008                      #Diameter
tb <- 200/10^6                  #pulse duration
tau <- 230/10^6                 #Lifespan
N0 <- 1.4*10^20                 #Number of Ions cm^-3
sigma <<- 2.8/10^23             #
c <- 299792458                  #Speed of Light m/s

V <- L*pi*(D/2)^2;              #cavity Volyme
B <- sigma*c/V;                 #Probability of stimulated emission ion and photon

N_inf <<- 0.01*N0;
P <- N_inf/tau;                 #Pump strength

#Assignment 2:b definitions
R1 <- 1;
R2 <- 0.05;
tb <- 200/10^6;

tau_c <- function(r1,r2) {      #Lifespan in cavity for photons
        tau_r = -2*L/(c*(log(r1)+log(r2)));
        return(tau_r)
}


#Differential eqvations:
N_prim <- function(N, Phi){             #Number of Ions
        y = P-B*N*Phi-N/tau;
        return(y)
}

Phi_prim <- function(Phi, N) {  #
        y = B*V*N*(Phi+1)-Phi/tau_c(R1,R2);
        return(y)
}


Solv <- function(f0, f_prim, g_prim, t){
        h = (t[1]-t[2]);
        f = rep(0, length(t));
        g = rep(0, length(t));
        f[1] <- (f0[1]);
        g[1] <- (f0[2]);
        for(i in as.single(1:length(t))){
                f[i+1] = f[i] - f_prim(f[i],g[i])*h;
                g[i+1] = g[i] - g_prim(g[i],f[i])*h;
        }
        return(f);
}

N <- function(x){
        return(Solv(c(N0,0), N_prim, Phi_prim, x))
}
Phi <- function(x){
        return(Solv(c(0,N0), Phi_prim, N_prim, x))
}
x0 = N(seq(0, 0.0002, length=20));
x1 = Phi(seq(0, 0.0002, length=20));
plot(x0);
plot(x1);
dev.off()

#Write to file #1
pdf("para_approx.pdf", width=7, height=5)
plot.function(f,from=0,to=D/2, col="red");
plot.function(fa, from=0, to=D/2, xlab="Hight",add=TRUE, ylab="Focus Point");
dev.off()

print(2)
#Write to file #2
pdf("BK7_index.pdf", width=7, height=5)
plot.function(n2v, from=(400/(10^9)), to=(700/(10^9)), ylab="Reflection Index", xlab="Wavelength");
dev.off()

```

```
#Write to file #3
pdf("BK7_abo.pdf", width=7, height=5)
plot.function(v_chrom, from=(400/(10^9)), to=(700/(10^9)), xlab="", ylab="");
dev.off()

#Write to file #4
pdf("N.pdf", width=7, height=5)
plot(x0)
dev.off()
```