

Visualisering av ljusets brytning i ett roterande prisma

Aniisa Bihi
Lucas Rung
Erik Nilsson
Molly Middagsfjell
Daniel Olsson

2018-03-12

Sammanfattning

Denna rapport redogör arbetsprocessen för modelleringen och simuleringen av ett fysikalisk fenomen, i detta fall en ljusstråle som passerar igenom ett prisma. Slutresultatet av projektet visar hur dispersionen från ett prisma gjort av glas ser ut när prismet roterar kring sin z-axel. Rapporten går igenom det som krävs för att utföra projektet, som de fysikaliska principerna och implementationsprocessen. Reflektioner och diskussioner av de hinder som uppkom under processen tas upp efter det slutgiltiga resultatet presenterats. Resultatet för projektet blev en prisma som roterar. Rotationen består av att vinkelhastigheten skickas in i en differentialekvation som sedan beräknar prismans nästa position. Det skickas en stråle genom prisma som bryts beroende på infallsvinkeln och utfallsstrålen består av ett färgspektrum.

Innehållsförteckning

1	Inledning	1
1.1	Bakgrund	1
1.2	Syfte	2
1.3	Avgränsningar	2
2	Genomförande	3
2.1	Förundersökning	3
2.1.1	Prisma	3
2.1.2	Snells Lag	5
2.1.3	Eulers stegmetod	5
2.1.4	MATLAB	6
2.2	Implementation	8
3	Resultat och Diskussion	9
3.1	Resultat	9
3.2	Diskussion	10
4	Slutsats	11
	Litteraturförteckning	12

Figurer

2.1	Utvikt prisma bestående av två trianglar och tre rektanglar.	4
2.2	Infallande vitt ljus passerar genom prismet och delas upp i ett spektrum av färger.	4
2.3	Snells lag.	5
2.4	Hur rotationen fungerar med hjälp av Eulers stegmetod	6
2.5	Den grafiska representationen av brytningen i ett prisma från MATLAB.	7
2.6	Vinklar som behövs för att beräkna ljusstrålens linje.	8
3.1	Slutliga grafiska representationen av prismet och ljusstrålen.	9

Kapitel 1

Inledning

För att underlätta förståelsen för fysikaliska system kan visuella redskap användas. I denna rapport kommer arbetsprocessen bakom ett modellprojekt att redovisas. Målet med projektet var att med hjälp av förundersökning, fysik, datorgrafik, modellering och lämpliga API¹, skapa en verklighetstrogen visualisering av ljusets brytning i ett prisma. Projektets syfte var att praktiskt använda sig av de kunskaper projektgruppen erhållit från kursen Modellbygge och Simulering, i anknytning till tidigare kurser.

1.1 Bakgrund

Målet med detta projekt var att utveckla tidigare kunskaper vilket bidrog till en djupare förståelse. Projektvalet grundades i att tidigare kunskaper kunde appliceras på prisma och gav projektgruppen en möjlighet att utvecklas på flera olika moment och ämnen. De ämnen som bearbetats är modellering, simulering, datorgrafik, fysik, trigonometri, linjär algebra och flervariabelanalys.

I detta projekt har ett triangulärt prisma behandlats. Speciellt vid ett triangulärt optiskt glas eller ett transparent prisma, är att det separera infallande vitt ljus i ett spektrum av färger [1]. Detta är användbart inom många delar av IT-industrin. För att kunna skapa reflekterande eller transparenta material inom till exempel datorgrafik krävs det kunskap om hur ljus beter sig vid kollision av denna sorts objekt.

¹API-Application Programming Interface

1.2 Syfte

Syftet med arbetet var att inom en liten projektgrupp åstadkomma en modellering av ett fysikaliskt system med hjälp av en eller flera differentialekvationer. Innebörden av detta var att först bestämma en matematisk modell som blev grunden till systemets simulation och sedan presentera en grafisk representation av systemet.

1.3 Avgränsningar

Avgränsningar som gjordes inför projektet var att ingen reflekterande ljusstråle beräknades eller skapades. Eftersom projektet var tidsbegränsat fick olika moment olika prioriteringsordning för att en färdig produkt skulle kunna presenteras. Bland de moment som hade lägre prioriteringsordning finns 3D-rotation och totalreflektion i prismet.

Kapitel 2

Genomförande

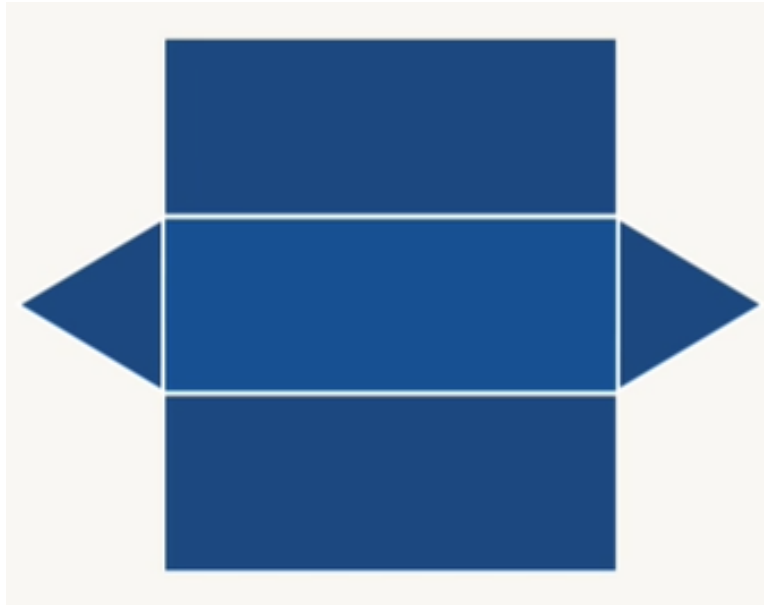
Projektet delades upp i två delar, en förundersökningsfas och en implementationsfas. Under förundersökningen togs differential ekvationer och matematiska modeller fram för att sedan valideras. Under implementationsfasen användes delarna från förundersökningen för att framställa de grafiska komponenterna.

2.1 Förundersökning

Under förundersökningen genomfördes preliminära matematiska beräkningar i MATLAB för att kontrollera att fysiken fungerade korrekt innan implementering. Dessa beräkningar involverade Snells lag för ljusets brytning, 2D-rotation med hjälp av differentialekvationer och Eulers stegmetod.

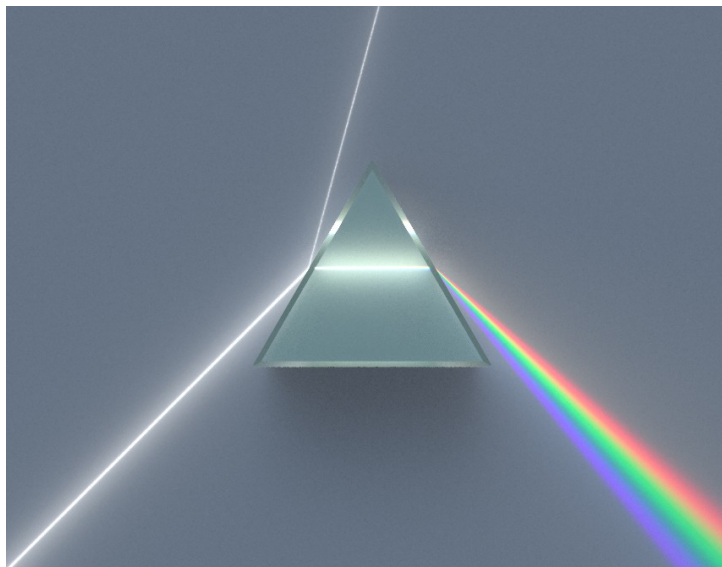
2.1.1 Prisma

Ett prisma är inom optiken ett transparent geometrisk element som användas för att sprida ljuset i till exempel en spektrometer eller för att polarisera ljus. Den vanligaste formen av ett prisma har en triangulär bas som förlängs med tre rektangulära ytor som tillsammans formar en triangel som kan ses i Figur 2.1.



Figur 2.1: Utvikt prisma bestående av två trianglar och tre rektanglar.

Det är denna typ av prisma som skapar ett spektrum av färger när vitt ljus passerar genom den. Detta uppstår på grund av dispersion av ljuset. I detta fall påverkar glasets brytningsindex och ljusets hastighet våglängden beroende på dess frekvens. Det blir då tidsvariationer i perioder mellan de skilda våglängderna i spektrumet vilket liknar en regnbåge. Figur 2.2 påvisar denna brytning, då rött ljus har en mindre brytning och violett ljus bryts mer [2].

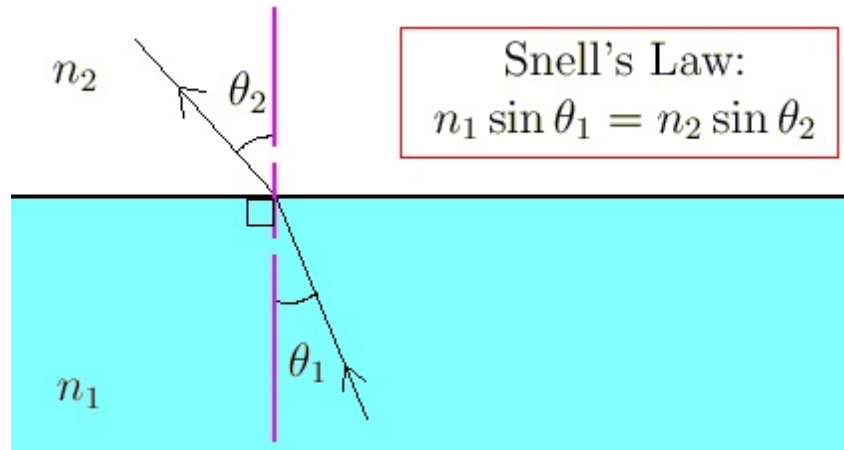


Figur 2.2: Infallande vitt ljus passerar genom prismet och delas upp i ett spektrum av färger.

Ljusets dispersion från prismet beror på vilken vinkel den infallande ljusstrålen har i förhållande till den träffade sidans normal. Vinklarna beräknas med Snells lag. Under detta projekt har brytningsindex för luft och glas använts. Dessa variabler är satta till $n_l = 1.00029$ och $n_g = 1.51$.

2.1.2 Snells Lag

Snells lag[3], även känd som brytningslagen, fastställer en ljusstråles riktningsändring när den passerar genom gränssytan mellan två medier. Refraktionen sker på grund av våglängdens hastighetsförändring då den byter medium [4].



Figur 2.3: Snells lag.

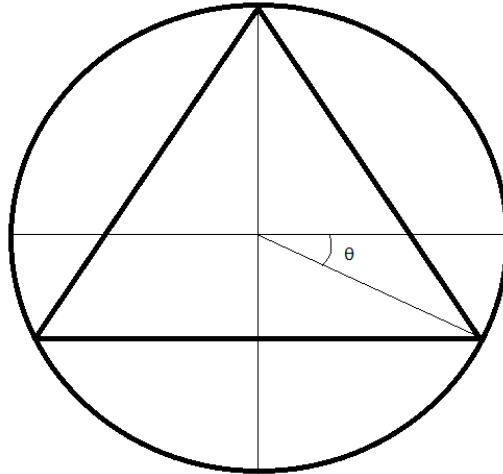
Totalreflektion är fenomenet som uppstår då infallsvinkeln är större än den kritiska vinkeln som definieras enligt ekvation 2.1.

$$\theta_g = \arcsin\left(\frac{n_2}{n_1}\right) \quad (2.1)$$

Den kritiska vinkeln är gränsvinkeln för totalreflektion och ges av Snells lag. Vid totalreflektion kommer ljusstrålen från ett optiskt tätare medium. Om infallsvinkeln är så pass stor att det inte finns tillräcklig med utrymme för den brutna strålen i det tunnare mediet så reflekteras allt ljus tillbaka i det tätare mediet.

2.1.3 Eulers stegmetod

För att kunna använda en differentialekvation till det tänkta projektet approximerades prismats hörnpunkter med hjälp av Eulers stegmetod. Eftersom prismats sex hörnpunkter ligger i en liksidig triangel kan dessa tänkas ligga i en cirkel med samma radie och att de sedan förflyttas efter cirkelns kant, vilket kan ses i Figur 2.4. Punkternas position ses som en funktion av vinkelhastigheten, se ekvation 2.2.



Figur 2.4: Hur rotationen fungerar med hjälp av Eulers stegmetod

Via detta skapades en differentialekvation, se 2.3 som förklarade punktens position med avseende till dess vinkelhastighet. Ur denna differentialekvation skapades sedan 2.4 med hjälp av Eulers stegmetod.

$$\theta_n = \int_{-\infty}^{\infty} w_n(t) dt \quad (2.2)$$

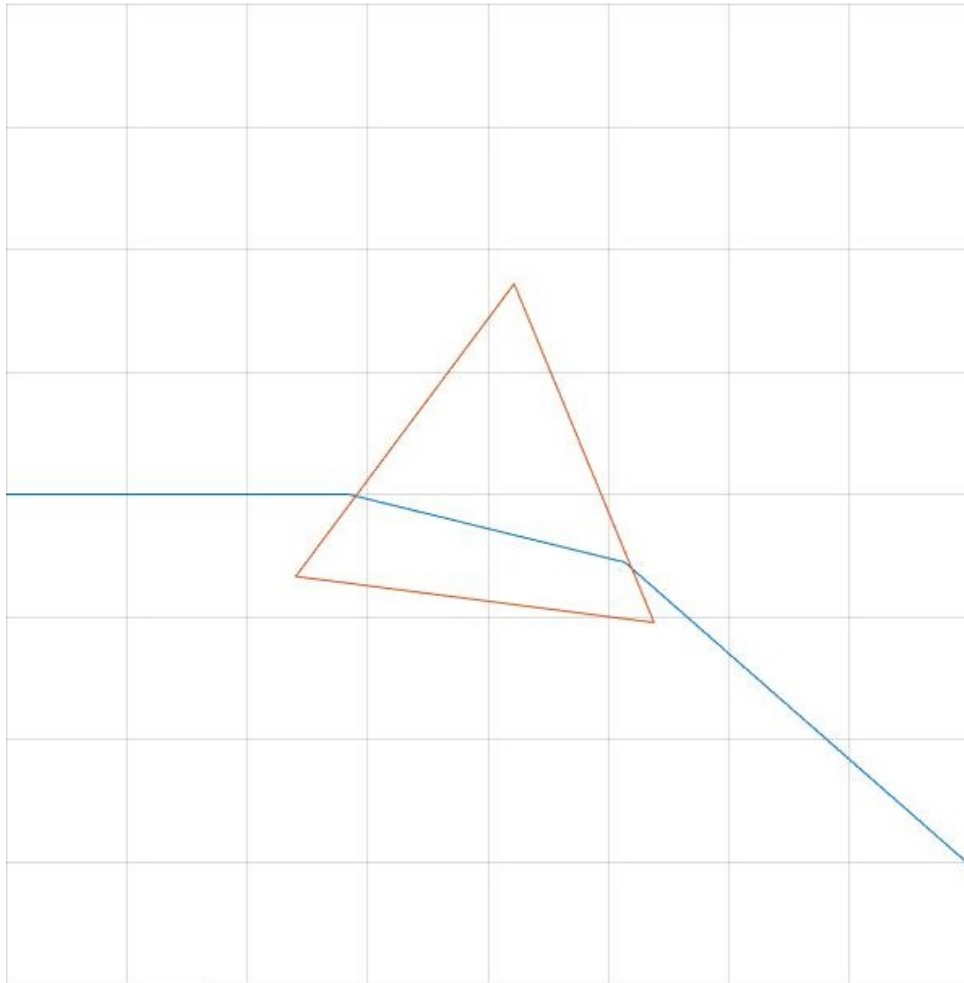
$$\theta_{n+1} = \theta_n + h * \theta' \quad (2.3)$$

$$\theta_{n+1} = \theta_n + h * w \quad (2.4)$$

Där θ_n är ursprungsvinkeln till positiva x-axeln, h är steglängden som stegas varje gång bilden uppdateras. h är i detta fall satt till 0.1 och w är vinkelhastigheten som skickas in är beroende av enheten programmet körs på. Den är satt till datorns uppdateringshastighet multiplicerat med π för att det ska upplevas som att prismet roterar lika snabbt på olika datorer. Vinkeln för varje punkt stegas sedan med Eulers stegmetod och nya punkter beräknas med hjälp av Pythagoras sats och trigonometri så att en rotation av triangeln uppstår.

2.1.4 MATLAB

För att säkerställa att alla formler och ekvationer ger rätt resultat implementerades lösningarna i MATLAB, ett datorprogram för matematiska och tekniska lösningar. Det som implementerades i MATLAB var rotationen av prismet och brytningen av ljuset inuti prismet som kan ses i 2.5.



Figur 2.5: Den grafiska representationen av brytningen i ett prisma från MATLAB.

Rotationen skapades med hjälp av cirkelns ekvationen och triangeln förflyttades med Eulers stegmetod i samband med en differentialekvationen som presenterades i 2.4 med samma parametervärden med undantag för vinkelhastigheten som är satt till $-\pi/20$.

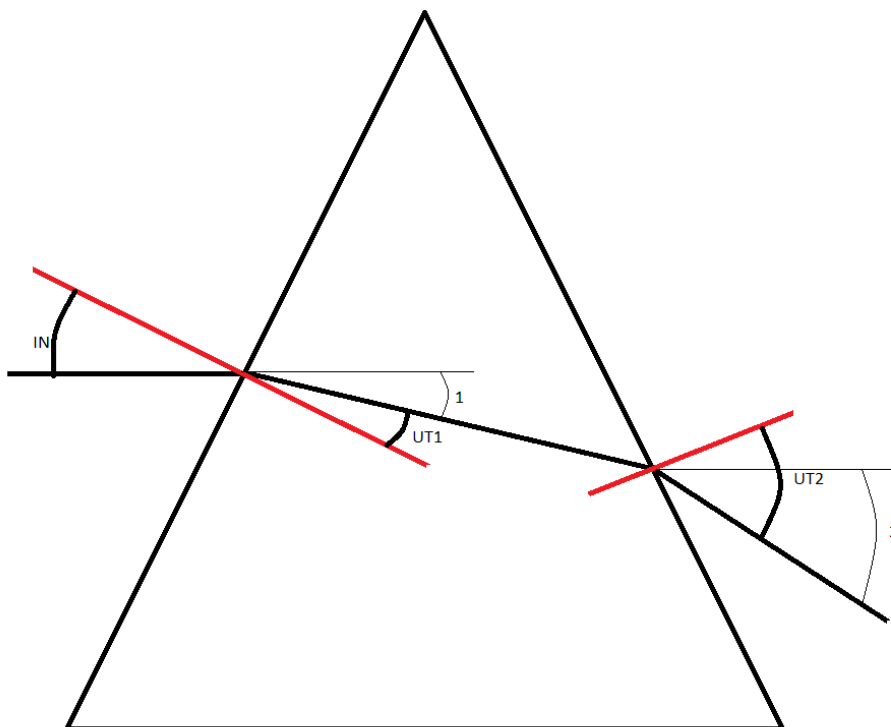
Svårigheterna som stöttes på i MATLAB var hur programmet skulle detektera när ljuset passerade en vägg. Denna position användes som utgångspunkt för nästa ljusstråle. För att hitta denna position betraktades ljusstrålen och de tre väggarna som olika linjens ekvation. Sambanden mellan ljusstrålen och de olika väggarna kan genererar positionen där ljuset bryts.

Ljuset representerades av en linjens ekvation och för att kunna beräkna om den efter en brytning behövs två punkter på linjen. Den första punkten är den detekterade positionen där ljuset bryts men nästa punkten behövs beräknas fram. För att beräkna den andra punkten används trigonometri där en kort stegning i x-led sker för att ge ett nytt y-värde. Då brytningen från Snells lag ger vinkel mellan ytans normal och ljusets riktning så kan ej den vinkeln användas. Den önskade vinkeln är mellan ljusets riktning och x-axeln. Då utstrålen har bryts två gånger behövs det olika formler mellan den för brytningen och den andra. Ekvation 2.5 används för första brytningen och ekvation 2.6 används för den andra brytningen.

$$\alpha_1 = \alpha_{In} - \alpha_{ut1} \quad (2.5)$$

$$\alpha_2 = \alpha_{ut2} + \alpha_{In} - \pi/3 \quad (2.6)$$

Formlerna kommer från definitionen av en rättvinklig triangel och vinklarna som används i ekvation 2.5 och 2.6 kan ses i figur 2.6.



Figur 2.6: Vinklar som behövs för att beräkna ljusstrålens linje.

När vinklarna har beräknats så används tangens för att beräkna den andra punkten. För första brytningen används ekvation 2.7 och andra brytningen använder sig av ekvation 2.8.

$$y_2 = y_1 \pm x_{steg} * \tan(\alpha_1) \quad (2.7)$$

$$y_2 = y_1 \pm (1 - x_{steg}) * \tan(\alpha_2) \quad (2.8)$$

Om minus eller plus används i ekvationerna 2.7 och 2.8 beror på vilken sida av normalen som ljusstrålen bryter prismet.

2.2 Implementation

Den grafiska implementationen av de fysikaliska beräkningarna skedde i C++ med biblioteket OpenGL. De standarder som följdes var efter C++ 11 och OpenGL 3.3 och programmet byggdes för Windows 10. Mjukvaran för texteditor var Visual Studio 2017 och dess inbyggda kompilator användes. Ljusstrålen som skapades är inte ljus i praktiken utan ett plan. Utstrålen är en triangel på som förflyttar sig beroende på brytningsvinkeln.

Kapitel 3

Resultat och Diskussion

I följande kapitel presenteras projektets resultat samt dess implementering och utveckling diskuteras.

3.1 Resultat

Det slutgiltiga resultatet blev en scen som användaren kan röra sig fritt i. Scenen innehåller ett roterande prisma där en ljusstråle bryts för att generera en regnbågsliknande ljusstråle, se Figur 3.1.



Figur 3.1: Slutliga grafiska representationen av prisma och ljusstrålen.

Implementationen saknar den reflekterande vinkel inuti prisma som hade generat ytterligare två fall där en ljusstråle bryts ut ur prisma.

3.2 Diskussion

Eftersom projektet kan utvecklas på många sätt var en gräns tvungen att fastställas. Tidsbrist ledde till att totalreflektion i prisma och 3D-rotation inte kunde implementeras på ett acceptabelt sätt, trots att mycket arbete hade lagts på dessa moment. Problemet med 3D-rotationen var att den inte gick att implementera på samma sätt som 2D-rotationen ty alla hörnpunkter inte ligger lika långt från origo. Detta medförde att radien blev olika för två av hörnpunkterna och sfäriska koordinater kunde inte användas. Då tillkom även ett annat problem vid implementeringen, att ϕ , som är vinkeln till z-axeln i sfäriska koordinater [5], inte definieras för alla tal utan endast mellan 0 till π . Det medförde att implementeringen blev markant mycket svårare jämfört med 2D-rotationen. En möjlig lösning på detta som ännu inte testats är att tänka att tänka att prismats hörnpunkter ligger i olika sfärer med olika radie. Detta gör att radien inte alltid måste vara densamma men det medför också att ekvationerna för de olika hörnpunkterna blir betydligt mer avancerade.

Vid implementeringen av 3D-rotationen hade detekteringen av positionen för ljusbrytningen slutat fungera. Differentialekvationen ger alla punkter på prisma och kan således spänna upp en ekvation för planet. Om ljusstrålen betraktas som en vektor kan positionen där strålen bryter planet hittas via planets ekvation.

Ett fall som inte implementerades var fallet för totalreflektion. Anledningen till detta var att det är en extremt tidskrävande uppgift då många specialfall som måste tas i åtanke. Eftersom det är mycket arbete för en liten visuell utdelning fick detta moment lägre prioritering än andra punkter. Detta medförde att det inte hann utföras innan deadline. Det här innebär också att totalreflektion inne i prisma som skulle ha gett en utstråle inte kunde heller kunde skapas. Resultatet blir därför att endast strålar som bara bryts en gång kommer att visualiseras.

Arbete i ett vidare skede vore att implementera de punkter som tagits upp ovan och dessutom implementera avgränsningen som gjorde: att en reflekterande stråle skapades när ljuset träffade prisma. Förbättring av glas materialet så att det är mer fysikaliskt korrekt skulle också ge en bättre visuell representation.

Kapitel 4

Slutsats

Under projektets gång tillkom problem som ansågs vara triviala teoretiskt men som visade sig vara svårare att implementera grafiskt. Detta medförde att projektets delar tog betydligt längre tid, vilket resulterade i att det slutliga resultatet inte uppfyllde de förväntningar som satts. Huvudsakligen är det 3D-rotationen och ljusbrytningen som är bristfällig på grund av detta. Trots detta blev gruppen nöjda med den resulterande prisman och brytning, med tanke på den korta tidsperioden och svårigheterna vid implementering.

Litteraturförteckning

- [1] Encyclopedia. Prism; 2009. [Hämtad 2018-02-21]. Available from: <http://www.encyclopedia.com/humanities/dictionaries-thesauruses-pictures-and-press-releases/prism-1>.
- [2] Nationalencyklopedin. prisma;. [Hämtad 2018-02-23]. Available from: <http://www.ne.se/uppslagsverk/encyklopedi/lng/prisma>.
- [3] Walker J, Halliday D, Resnick R. Principles of physics. Hoboken, NJ: Wiley; 2014. Pg. 944-945.
- [4] Nationalencyklopedin. brytningslagen;. [Hämtad 2018-02-23]. Available from: <https://www.ne.se/uppslagsverk/encyklopedi/l%C3%A5ng/brytningslagen>.
- [5] Neymark M. Matematisk analys : flera variabler. Upplaga 2 ed. Stockholm: Liber; 2017. Pg. 44.