



INSTITUTIONEN FÖR FYSIK

Driftsättning av femtosekund-pump-prob-uppställning för joner

Kandidatarbete i fysik

Amanda Olsson & Sara Tjäder

Institutionen för fysik

GÖTEBORGS UNIVERSITET

Göteborg, Sverige 2023

KANDIDATARBETE 2023

Driftsättning av femtosekund-pump- prob-uppställning för joner

Amanda Olsson

Sara Tjäder

Handledare: Raimund Feifel & Hélène Coudert-Alteirac

Examinator: Martina Ahlberg



GÖTEBORGS UNIVERSITET

Institutionen för fysik
GÖTEBORGS UNIVERSITET
Göteborg, Sverige 2023

© Amanda Olsson & Sara Tjäder, 2023.

Handledare: Raimund Feifel, Professor, Atom- och molekylfysik,
Institutionen för fysik, Göteborgs universitet
Hélène Coudert-Alteirac, Forskare, Atom- och molekylfysik,
Institutionen för fysik, Göteborgs universitet
Examinator: Martina Ahlberg, Fil. Dr,
Institutionen för fysik, Göteborgs universitet

Kandidatarbete 2023
Institutionen för fysik
Göteborgs universitet

Göteborg, Sverige 2023

Driftsättning av femtosekund-pump-prob-uppställning för joner
Amanda Olsson & Sara Tjäder
Institutionen för fysik
Göteborgs universitet

Abstract

The objective of this project was to commission an UV-IR femtosecond pump-probe-set-up for ions. The aim was to identify at what delay the UV and IR pulses overlap in time, as well as determine the intensity of the pulses in the spot where they overlap in space. The intensity was calculated using known values of the energy and duration of the pulses, together with the measured area of each pulse in the overlap spot. By integration of a mass spectrum of xenon at different delays a curve was obtained, from which the maximum overlap could be determined. A MATLAB code was created, along with a user interface designed in MATLAB App Designer, in order to simplify this and future analyses.

Sammandrag

Detta projekt syftar till att driftsätta en UV-IR femtosekund-pump-prob-uppställning genom att identifiera fördröjningen som ger maximalt överlapp mellan en UV-puls och en IR-puls, samt genom att ta reda på intensiteten i fokuspunkten där de två pulserna möts. Intensiteten beräknades utifrån kända värden på energi och pulslängd för respektive puls samt uppmätt area av pulserna där de möttes. Genom integrering av masspektrum för xenon vid olika fördröjningar erhöles en kurva ur vilken den maximala överlappningen kunde utläsas. Med målet att förenkla denna och framtida analyser skapades en kod i MATLAB samt ett användargränssnitt i MATLAB App Designer.

Nyckelord: Masspektrometri, Joner, Icke-linjär optik, Laser, Användargränssnitt, MATLAB, MATLAB App Designer

Innehåll

1	Introduktion	1
2	Teori	3
2.1	Atomfysik	3
2.2	Icke-Linjär Optik	4
2.3	Laser	6
2.4	Masspektrometri	7
3	Metod	9
3.1	Uppställning	9
3.2	Intensitetsberäkning	11
3.2.1	Noggrannhet	12
3.3	Användargränssnitt	13
3.3.1	Datainsamling	14
3.3.2	Dataanalys	14
4	Resultat och diskussion	17
4.1	Rumslig överlappning	17
4.2	Tidsöverlappning	18
4.3	Användargränssnitt	19
5	Slutsatser	25
	Litteratur	27
A	MATLAB-kod för beräkning av intensitet	I
B	MATLAB-kod för användargränssnitt	V

1

Introduktion

Från det att ett embryo skapas börjar cellerna i kroppen delas genom mitos, vilket fortsätter genom hela människans liv. Vid celledelning kan inte kromosomerna kopieras hela vägen ut i ändarna, vilket leder till att kromosomen förkortas vid varje delning. För att inte gå miste om viktig information finns därför skyddande ändar, så kallade telomerer, i slutet på kromosomerna. I takt med att telomererna förkortas åldras cellen och till slut, när telomererna blivit för korta, kan cellen inte längre dupliceras.

I en del celler kan telomererna dock förlängas igen, vilket kopplas till enzymet telomeras. I de flesta celler i kroppen är enzymet inaktivt, men i de fallen där det är aktivt kan cellen fortsätta dela sig utan att åldras. Exempelvis syns i 85 – 90 procent av cancerceller att telomeras är aktivt och kromosomerna behåller sin fulla längd vid celledelning [1]. Genom att studera den så kallade G-kvadruplexstrukturen (G4) som förekommer i slutet av telomererna kan vi få tillgång till mer information om hur processen för cellernas åldrande ser ut. Stabilisering av G4-strukturen i telomererna skulle kunna utgöra en framtida behandlingsform för cancer.

Vid studier av dynamiska processer i dessa G4-molekyler krävs en tidsupplösning som kan uppnås genom användning av pump-prob-teknik, där en laserpuls (pump) interagerar med molekylen och ger upphov till en förändring, varpå en andra puls (prob) inkommer och analyserar förändringen. Analysen sker genom att molekylen joniseras av prob-pulsen och accelereras av ett elektriskt fält för att sedan registreras av en detektor. Därigenom fås ett masspektrum från vilket data kan avläsas. Detta förutsätter att de två pulserna överlappar i rummet samt att de har tillräckligt hög intensitet för att åstadkomma förändringen och jonisera molekylen.

För undersökning av G4 behövs ett sätt att beräkna pulsernas intensitet och att ta reda på fördröjningen mellan pump-pulsen och prob-pulsen. Syftet med projektet är därför att utveckla en MATLAB-kod för beräkning av pulsintensiteten, samt att skapa ett program som kan kalibrera uppställningen med hjälp av xenon och göra informationen från den insamlade datan lättillgänglig via en app i MATLAB App Designer.

2

Teori

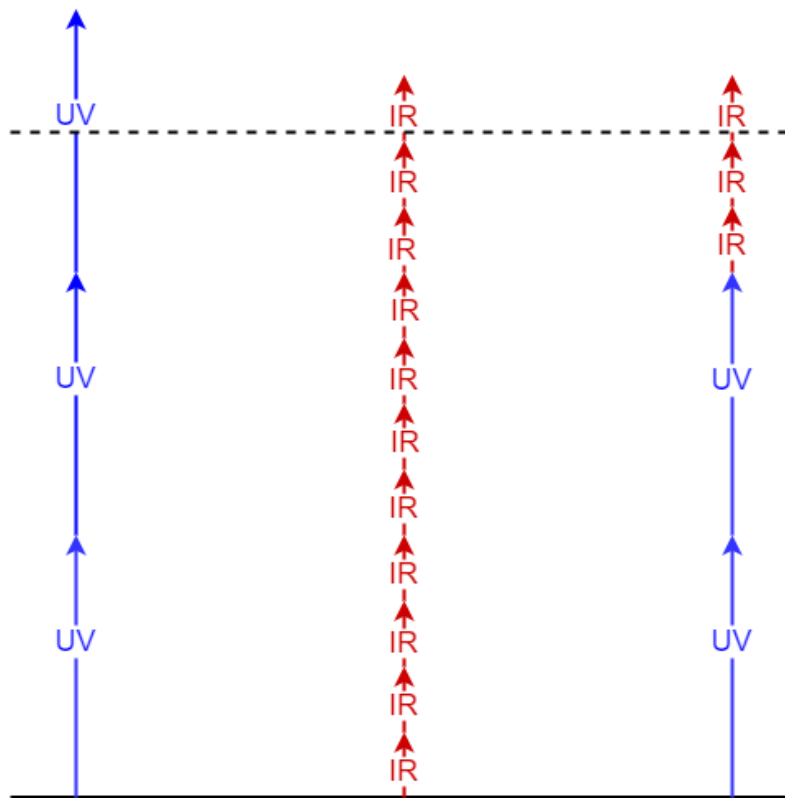
I undersökandet av intensitets- och fördröjningsförhållanden mellan UV- och IR-pulserna krävs kunskap inom bland annat atomfysik, laserfysik och masspektrometri. Vid omvandling av frekvensen för en puls, från IR till UV, spelar icke-linjär optik en central roll.

2.1 Atomfysik

Enligt Bohrmodellen beskrivs atomens struktur som en kärna som består av neutroner och protoner och som omges av elektroner, vilka rör sig i olika orbitaler med kvantiserade avstånd från kärnan. En atom exciteras när den tar emot energi, exempelvis från en foton som absorberas, och en elektron flyttar sig då till en orbital längre från kärnan vilket motsvarar en högre energinivå. Energin blir därmed lagrad i det nya tillståndet men avges igen i form av en foton när elektronen lämnar sitt exciterade tillstånd och återgår till grundtillståndet. Om tillräckligt mycket energi absorberas kan atomen istället joniseras, det vill säga att en elektron slits loss från atomen som då blir positivt laddad. Gränsen för den lägsta energi som krävs för att det ska ske kallas jonisationsenergi.

För att en foton ska kunna jonisera en atom krävs att fotonens energi är lika stor som eller större än jonisationsenergin, alternativt att atomen exciteras av flera fotoner. För att det senare ska vara möjligt krävs att det finns tillräckligt långlivade mellanliggande tillstånd till vilka atomen först kan exciteras för att sedan joniseras av ytterligare en foton. Om ljusets intensitet är tillräckligt hög är det dock även möjligt att absorption av flera fotoner kan ske samtidigt, vilket innebär att atomen kan joniseras av flera fotoner även när det inte finns några mellanliggande tillstånd.

Den multifotonabsorption som beskrivs i föregående stycke kan exempelvis användas för att jonisera xenon, vars jonisationsenergi ligger på 12,1 eV [2]. För att jonisera xenon med UV-fotoner av våglängd 257 nm krävs att tre fotoner absorberas samtidigt, medan det krävs elva IR-fotoner av våglängd 1 030 nm. Genom att låta laserpulser av bägge våglängderna överlappa i tid och rum kan atomen istället joniseras av exempelvis två UV-fotoner och tre IR-fotoner, se figur 2.1.



Figur 2.1: Exempel på hur lasrar med olika våglängd kan excitera och jonisera en atom eller molekyl som saknar mellanliggande tillstånd, genom att fotonerna adderas. För att detta ska ske krävs att ljusintensiteten är tillräckligt hög. Vektorernas längd svarar mot fotonenergin för de olika lasertyperna, den streckade linjen symboliserar jonisationsenergin och den heldragna linjen motsvarar grundtillståndet.

2.2 Icke-Linjär Optik

Icke-linjär optik beskriver hur ljuset beter sig i icke-linjära medium, det vill säga i medium där ljusets elektriska fält E inte förhåller sig linjärt till materialets polarisation P . Dessa medium är oftast kristaller, regelbundna strukturer som upprepar sig i samtliga dimensioner. I icke-linjär optik används kristaller vars brytningsindex beror på ämnet i kristallen och styrkan hos det elektriska fältet. Brytningsindex för de icke-linjära egenskaperna kommer bero på vakuumperrmittiviteten ϵ_0 (även känd som den elektriska konstanten), den elektriska susceptibiliteten χ hos materialet samt dess brytningsindex i linjärt tillstånd.

Susceptibiliteten kan uttryckas som $\chi = \epsilon_r - 1$, där ϵ_r är permittivitetstalet. Detta uttryck beskriver materialets förmåga att förändra sin polarisation vid påverkan av ett elektriskt fält [3]. Om $\chi = 0$ innebär det att materialet inte alls påverkas av ett elektriskt fält. Brytningsindex beskriver hur våglängden förändras inuti ett material och är en kvot mellan ljushastigheten i vakuum och ljushastigheten inuti materialet [4]. Det totala brytningsindexet fås enligt $n = n_0 + n_2 \cdot I$ där n_0 är index i

linjärt tillstånd, $n_2 = 3\chi^{(3)}/(4\epsilon_0cn_0Re(n_0))$ är index i icke-linjärt tillstånd och I är vågens intensitet. När en kristall utsätts för ett elektriskt fält och brytningsindexet förändras proportionerligt mot kvadraten av fältet kallas detta för Kerr-effekten [5].

Kristallers inverkan på vågor som propagerar genom dem beror på vilket ämne kristallen består av samt vilket elektriskt fält den utsätts för. Hur frekvensen av en våg ökar när den passerar kristallen kopplas till den icke-linjära susceptibiliteten χ av olika grad. Exempelvis kan frekvensen dubbleras, vilket kallas andra ordningens övertton (SHG) [6]. Om frekvensen istället tre- eller fyrdubblas kallas det tredje (THG) respektive fjärde (FHG) ordningens övertton. Vid högre ordning genereras så kallad hög ordnings övertton (HHG).

Andra ordningens övertton skapas på grund av den kvadratiska ökningen från det elektriska fältet, där två fotoner med frekvens ω kombineras till en foton med frekvens 2ω , se figur 2.2. Med två kristaller kan frekvensen dubbleras två gånger vilket ger en foton med frekvens 4ω från vad som ursprungligen var fyra fotoner med frekvens ω . På så vis kan en IR-puls omvandlas till en UV-puls via endast kristaller av andra ordningens övertton. Genom denna justering av våglängd kan även flera fotoner tillsammans bilda ett vågpaket, se figur 2.3.

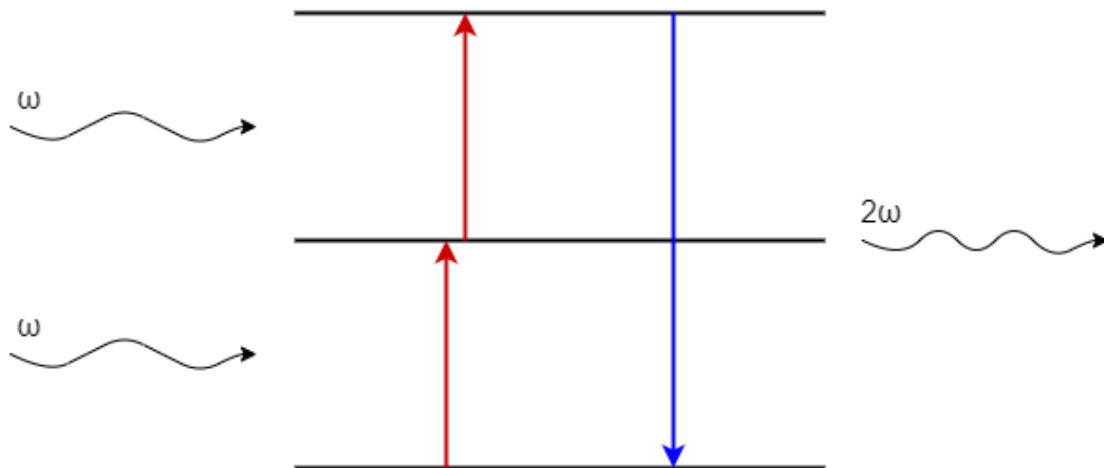
För andra ordningen kan polarisationen uttryckas enligt $P^{(2)} = \epsilon_0\chi^{(2)}E^2$, där ϵ_0 är vakuumperrmittiviteten [F/m], $\chi^{(2)}$ är susceptibiliteten [m/V], E är det elektriska fältet [V/m] och P är polarisationen [C/m²]. Generellt gäller för ordning n att $P^{(n)} = \epsilon_0\chi^{(n)}E^n$ [7].

Då fotoner kan ses som enskilda elektriska fält kan deras rörelse beskrivas likt fältet vid växelström som uttrycks $\tilde{E}(t) = Ee^{-i\omega t} + c.c.$ där *c.c.* står för komplexkonjugat (eng. complex conjugate). Dipolmomentet per volymenhet, eller polarisationen $\tilde{P}(t)$, beror på det elektriska fältet $\tilde{E}(t)$ enligt $\tilde{P} = \epsilon_0[\chi^{(1)}\tilde{E}^1 + \chi^{(2)}\tilde{E}^2 + \dots + \chi^{(n)}\tilde{E}^n] = \tilde{P}^{(1)} + \tilde{P}^{(2)} + \dots + \tilde{P}^{(n)}$. Slutligen fås följande vågekvationen enligt icke-linjär optik i ett medium:

$$\nabla^2\tilde{E} - \frac{n^2}{c^2} \cdot \frac{\partial^2\tilde{E}}{\partial t^2} = \frac{1}{\epsilon_0c^2} \cdot \frac{\partial^2\tilde{P}^{NL}}{\partial t^2}$$

där \tilde{E} är det elektriska fältet, n är brytningsindex, c ljusets hastighet i vakuum, ϵ_0 är permittiviteten i vakuum och \tilde{P}^{NL} är den icke-linjära polarisationen [7].

En kristall som ofta används i icke-linjär optik är beta-bariumborat (BBO), en bariumboratfas i låg temperatur. Dessa kristaller är dubbelbrytande, vilket innebär att de har olika brytningsindex för inkommande ljus beroende på om det är polariserat parallellt med eller vinkelrätt mot den optiska axeln. Brytningsindexet för parallellt polariserat ljus kallas ordinärt index, n_o , medan det för ljus med vinkelrät polarisation kallas extraordinärt index, n_e [7]. Brytningsindex är våglängdsberoende, men för exempelvis $\lambda = 1\,064$ nm är $n_o = 1,6551$ medan $n_e = 1,5425$ [8].



Figur 2.2: Två fotoner med frekvens ω inkommer till kristallen och omvandlas till en foton med frekvens 2ω .

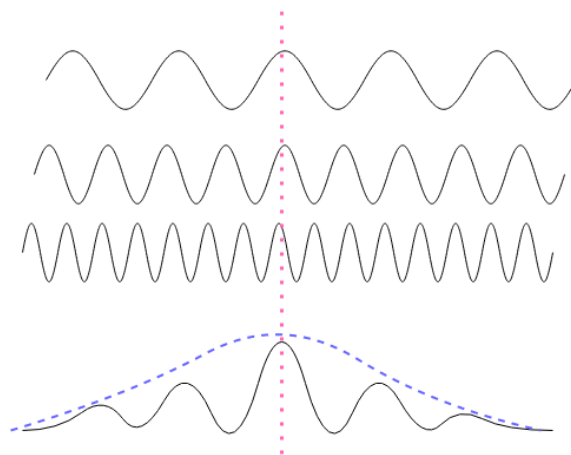
2.3 Laser

Laser är en akronym för Light Amplification by Stimulated Emission of Radiation vilket kan översättas till "ljusförstärkning genom stimulerad emission av strålning". När en molekyl eller atom exciteras övergår den från sitt grundtillstånd till ett mer energirikt tillstånd för att sedan återgå tillbaka till grundtillståndet, varpå den lagrade energin frigörs och en foton sänds ut. Vid spontan emission kan en foton utsändas i godtycklig riktning och fas, medan stimulerad emission innebär att en inkommande foton får atomerna att relaxera utan att fotonen absorberas, samtidigt som ytterligare en foton utsänds i samma riktning och med samma fas som den inkommande [9]. Fotonen som stimulerar emissionen blir alltså duplicerad. Detta ger ett koherent monokromatiskt ljus, alltså ljus som består av vågor som är i fas med varandra och som har samma våglängd. Genom så kallad pumpning tillförs energi så att populationsinversion uppnås, det vill säga att fler atomer befinner sig i ett högre energitillstånd än i ett lägre [10]. Det medför att det finns tillräckligt många atomer som kan svara på stimuleringen.

En metod för analys är pump-prob (eng. pump-probe) där två laserpulser interagerar med ett medium strax efter varandra. Den första pulsen (pump) är till för att skapa en förändring i mediet, exempelvis för att förändra elektronfördelningen, medan den senare pulsen (prob) analyserar tillståndet som den första pulsen skapat [11]. Vanligtvis är den första pulsen starkare än den andra och tidsupplösningen är i regel av storleksordning pico- eller femtosekunder.

Vid summering av vågor med olika frekvenser kan vågpaket bildas genom att konstruktiv interferens sker i centrum av vågpaketet, se figur 2.3. Genom att synkronisera vågor på detta sätt kan nya frekvenser skapas. Inom varje vågpaket finns ett elektriskt fält som oscillerar igenom vågpaketet.

För att fokusera lasrar kan en lins användas för att böja av strålen i en vinkel så att



Figur 2.3: Illustration över hur tre vågor av olika frekvens summeras till ett vågpaket, visat i streckad blå linje, med maximum enligt punktmarkerad rosa linje.



Figur 2.4: Två olika beskrivningar av hur en laser fokuseras genom en lins. **(a)** I geometrisk optik beskrivs strålar som konvergerar mot en fokuspunkt. **(b)** I vågoptik påverkas storleken på fokusområdet av diffraktion och aberration.

den fokuseras i den så kallade brännpunkten, se figur 2.4a. I vågoptik är det dock inte en singular punkt som lasern samlas i, utan ett område där den är som mest fokuserad, den så kallade midjan (eng. waist, w_0) som syns i figur 2.4b. Storleken på midjan beror på våglängden λ , linsens fokallängd f samt laserns diameter D innan den passerar linsen, enligt följande samband $w_0 = 2\lambda f/\pi D$. Experimentellt finns även astigmatism som gör att midjan inte är perfekt cirkelformad. För att ta reda på det verkliga fokuset behöver lasern testas för att se vid vilket avstånd diametern är kortast.

2.4 Masspektrometri

En joniserad atom eller molekyl som befinner sig i ett elektriskt fält kommer accelereras av fältet. Genom att accelerera jonen och låta den kollidera mot en detektor kan dess flygtid (eng. time of flight) mätas. Flygtiden är den tid det tar för jonerna att nå detektorn, vilket är beroende av jonernas massa [12]. Därför kan massan beräknas utifrån den uppmätta flygtiden. Baserat på hur många utslag detektorn får vid olika flygtider skapas ett spektrum med toppar vid de vanligast förekommande flygtiderna. Genom dessa steg skapas ett masspektrum.

Eftersom flygtiden är relativ och varierar beroende på uppställningen behöver varje uppställning kalibreras. För det kan exempelvis xenon eller kväve användas. De

stabila isotopernas relativa förekomster och massor kan jämföras med de relativa intensiteterna och avstånden mellan topparna i masspektrumet. Utifrån det kan flygtiderna kopplas till jonernas respektive massor. Efter kalibreringen kan ett annat ämne placeras i gasbehållaren och de uppmätta flygtiderna kan användas för att avgöra vad som träffat detektorn.

Generellt är det enklare att jonisera större molekyler medan det är svårare för enskilda atomer. Jonisationsenergin för de enskilda atomerna varierar mellan olika grundämnen, men xenon och kväve som nämndes i föregående stycke har bägge relativt höga jonisationsenergier [2]. Om en IR-puls och en UV-puls inkommer samtidigt kan intensiteten i fokuspunkten bli tillräckligt hög för att jonisera atomer med denna höga jonisationsenergi. Om pulserna däremot inte överlappar i tid, det vill säga om den ena pulsen kommer före den andra, joniseras inte tillräckligt många atomer och resultatet blir att inga toppar syns i masspektrumet. Genom att förlänga eller förkorta färdsträckan för IR-pulsen kan en fördröjningen mellan pulserna justeras så att de överlappar.

För att undvika att masspektrumet innehåller toppar från andra ämnen än de som studeras krävs att joniseringen sker i vakuum. Vakuum kan skapas med olika tekniker, varav en metod är med en så kallad rotary pump som för bort molekyler. Det skapas först ett förvakuum på ca 10^{-2} mbar och om bättre vakuum önskas används en turbopump som sänker trycket till 10^{-6} mbar. Högst vakuum behöves exempelvis i områden där lasrar fokuseras eller endast ska interagera med en önskad gas och inte ämnen i luften. I sådana fall skapar en turbopump ett större vakuum. Hur stort vakuum som kan uppnås med turbopumpen beror bland annat på vilken utrustning som används för att foga mellan olika material. Exempelvis kan en gummiring (KF) användas, med vilken det finns möjlighet att nå 10^{-6} mbar. En fördel med en sådan gummiring är att den kan återanvändas. Används istället en kopparring (CF) nås som lägst ett tryck på 10^{-12} mbar, men den kan endast användas en gång då den repas vid montering.

3

Metod

Genom programmering i MATLAB och MATLAB App Designer utvecklades program för kalibrering av uppställningen samt för analys av mätdata.

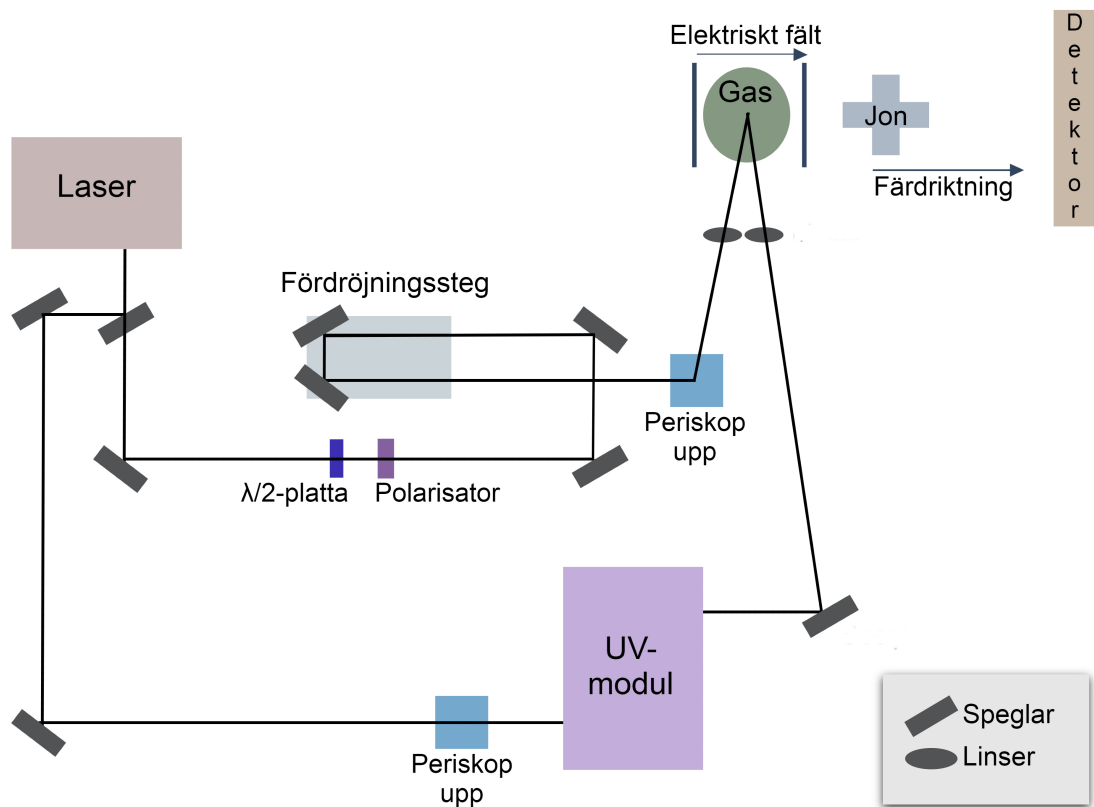
3.1 Uppställning

Uppställningen byggde på att två pulser överlappades och joniserade atomer som sedan accelererades och detekterades, se figur 3.1. Laserpulser med våglängden 1 030 nm utsändes från en IR-laser med pulsfrekvens 3 kHz. Varje puls delades i två med hjälp av en halvreflekterande spegel och de två pulserna färdades olika vägar. För att ändra pulsernas riktning i det horisontella planet användes speglar medan periskop användes för att styra pulserna vertikalt.

En av pulserna passerade genom en UV-modul där frekvensen fyrfaldigades, vilket omvandlade IR-pulsen till en UV-puls med våglängd 257,5 nm. UV-modulen innehöll bland annat beta-bariumboratkristaller av ordning $\chi^{(2)}$ som fördubblade frekvensen i två steg. Den andra pulsens våglängd bevarades och dess intensitet justerades genom att pulsen passerade genom en $\lambda/2$ -platta och polariserades linjärt innan den filtrerades genom en polarisator. $\lambda/2$ -plattan hade olika brytningsindex längs olika axlar vilket gav en snabb och en långsam optisk axel. Det medförde att de två vinkelräta komponenterna av ljusvågen försköts i förhållande till varandra. Denna fasförändring mellan komponenterna ändrade ljusets polarisation. Genom att justera $\lambda/2$ -plattan kunde polarisationen väljas. När ljuset sedan nått den linjära polarisatorn kom endast den andel av ljuset som låg längs polarisatorns axel igenom medan ljuskomponenten som låg vinkelrät mot polarisatorns axel stoppades. På så vis kunde mellan 0 – 100 % av ljuset släppas igenom.

För att synkronisera de två pulserna genomgick IR-pulsen ett fördröjningssteg där pulsen försköts för att bäst överlappa med UV-pulsen. Därefter fokuserades pulserna genom varsin lins innan de förenades inne i gasbehållaren där de joniserade atomerna i gasen.

De laddade jonerna accelererades av ett elektriskt fält över behållaren och färdades sedan fritt fram till en detektor. Utifrån den uppmätta flygtiden för jonerna skapades ett masspektrum.



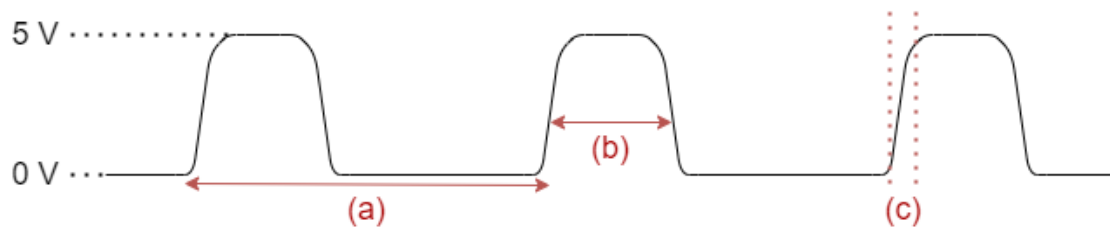
Figur 3.1: Skiss över uppställningen. Lasern skickar ut pulser som delas upp med hjälp av en halvreflekterande spegel. Den ena pulsen passerar ett periskop där den förflyttas vertikalt och sedan fortsätter den via en UV-modul där frekvensen fyrfaldigas. Den andra pulsen går via en $\lambda/2$ -platta och en polarisator till fördröjningssteget där IR-pulsen fördröjs i förhållande till UV-pulsen. Därefter passerar även denna puls ett periskop. Pulserna förenas sedan i gasbehållaren där de joniserar atomerna. Ett elektriskt fält läggs över gasen och accelererar jonerna som sedan flyger fritt mot detektorn.

Med en pulsfrekvens på 3 kHz blir avståndet mellan två pulser

$$c \cdot f^{-1} = 3 \cdot 10^8 \text{ m/s} \cdot \frac{1}{3} \cdot 10^{-3} \text{ s} = 10^5 \text{ m},$$

så trots att lasern avgav 3 000 pulser per sekund kom två på varandra följande pulser fram till gasbehållaren med ett tillräckligt stort avstånd för att inte överlappa.

För att mäta flygtiden krävdes ett sätt att ta reda på vid vilken tidpunkt jonerna påbörjade färden mot detektorn, alltså när de joniserades. För det användes triggerpulser från lasern, se figur 3.2. Dessa utsändes med ett tidsintervall på 3 ms och stigtiden, det vill säga tiden det tog för oscilloskopet att gå från 10 % till 90 % av maxvärdet, var lite mindre än 1 ns. När triggerpulsen nått detektorn började den detektera joner under en period på omkring 200 μs . Därefter slutade den detektera i väntan på nästa triggerpuls. Eftersom ljuset är snabbare än elektroniken användes triggerpulsen från en föregående laserpuls för att beräkna tidpunkten för joniseringen.



Figur 3.2: (a) Tidsintervallet mellan två triggerpulser ligger på 3 ms. (b) Detektorn mäter intensiteten av jonerna under en period på omkring $200 \mu\text{s}$. Därefter slutar den detektera i väntan på nästa puls. (c) Stigtiden, det vill säga tiden det tar för oscilloskopet att gå från 10 % till 90 % av maxvärdet, är lite mindre än 1 ns.

Informationen som samlades in från detektorn var tiden det tog för varje jon att nå detektorn (flygtiden) samt antalet joner som träffade detektorn vid varje tidpunkt (intensiteten).

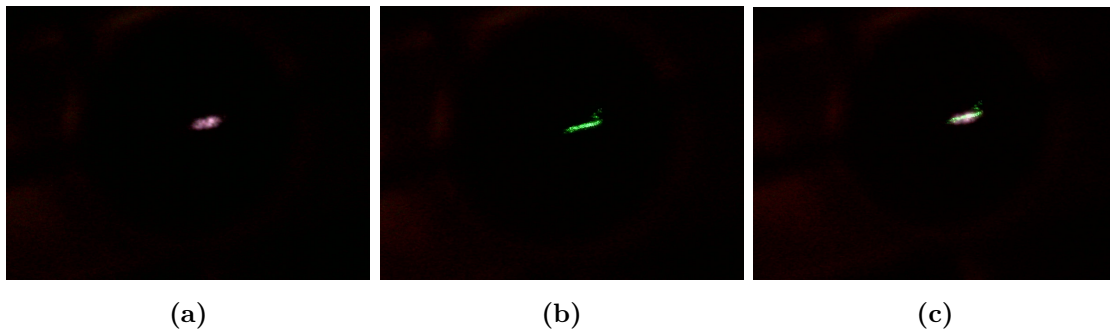
3.2 Intensitetsberäkning

En ca 1 mm bred metallplatta med fosfor placerades i fokuspunkten för IR- och UV-pulserna, vilket gjorde pulserna synliga för mänskliga ögat. Genom att ta fram halvvärdesbredden (eng. full width at half maximum, FWHM) kunde arean för respektive puls beräknas och användas för att beräkna intensiteten i fokuspunkten.

En del av de bilder som analyserades var mättade, vilket innebär att pixlarna nått ett maxvärde. Varje pixel översattes till en cell i en matris med ett värde mellan 0 och 255 beroende på dess intensitet. För de pixlar som nått värdet 255 fanns inget sätt att avgöra om värdet var exakt 255 eller högre. För att få en representativ bild av intensitetsfördelningen av pixlarna kunde de mättade bilderna därmed inte användas för analys, då de verkliga signalerna inte fångats upp. Istället valdes tre omättade bilder ut för att beräkna intensiteten, se figur 3.3.

På grund av att pulserna träffade plattan i 45 graders vinkel blev de utsträckta längs ena axeln. Detta gav en avlång ljusbild istället för en cirkulär, som hade uppstått vid direkt infall. Vid beräkning av area och intensitet kunde detta problem kringgå genom att rotera bilderna så att utsträckningen låg längs en vertikal axel. Den roterade bilden översattes till en matris där varje cell indikerade intensiteten i området och matrisen summerades över samtliga kolumner. Därefter användes den summerade kolumnen till värden i en graf och då kunde bredden på pulsen beräknas längs den horisontella axeln.

Genom att analysera kurvan beräknades halvvärdesbredden (FWHM) genom att dividera värdet av kurvans maximala intensitet med två och notera avståndet mellan de två positionerna där halvvärdet uppnåts. Därefter kunde pulsens area beräknas genom att anta att den har formen av en cirkel vars radie var hälften av halvvärdesbredden.



Figur 3.3: Bilder som analyserades för att ta reda på intensiteten i fokuspunkten. En platta med fosfor placerades i fokuspunkten och avgav ljus av olika färg beroende på om den träffades av en IR- eller UV-puls. **(a)** Fosforplattan avgav ett vitt ljus där den träffades av IR-pulsen. **(b)** Där UV-pulsen träffade fosforplattan lyste den grönt. **(c)** När IR- och UV-pulserna träffade fosforplattan samtidigt avgav den både vitt och grönt ljus.

Med den beräknade arean samt given tid och energi beräknades laserpulsernas intensitet I genom sambandet $I = \frac{E}{t} \cdot \frac{1}{A}$ där E är energin i joule, t är tiden för pulsen i sekunder och A är arean för pulsen i cm^2 .

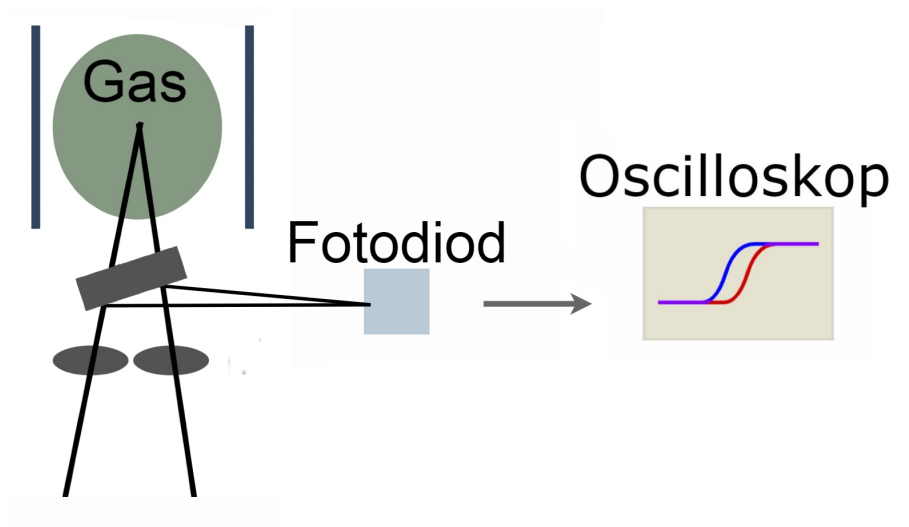
3.2.1 Noggrannhet

I fördröjningssteget justerades fördröjningen för IR-pulsen genom att justera dess färdsträcka. Hur mycket sträckan behövde justeras för att förändra fördröjningen med exempelvis 200 femtosekunder beräknades enligt

$$t \cdot c = 200 \cdot 10^{-15} \text{ s} \cdot 3 \cdot 10^8 \text{ m/s} = 60 \text{ } \mu\text{m}.$$

Ljuset färdas alltså 60 μm per 200 fs (eller 0,3 μm per fs). Noggrannheten vid justering av fördröjningen på femtosekundsskala behövde därmed vara av storleksordning mikrometer.

För en grov uppskattning av vilken fördröjning som gav maximalt överlapp mellan pulserna användes en fotodiod och ett oscilloskop. Eftersom gasen var inne i en vakuumkammare placerades en spegel utanför kammaren, för att rikta pulserna mot fotodioden som skickade information till oscilloskopet, se figur 3.4. Då oscilloskopets tidsupplösning var en knapp nanosekund gav denna metod inte tillräcklig precision, eftersom ljus färdas ungefär 30 cm per nanosekund. Efter den grova uppskattningen finjusterades därför fördröjningen genom att låta pulserna jonisera xenon och studera det masspektrum som genererades vid olika fördröjningar.



Figur 3.4: En spegel placerad utanför vakuumkanmaren riktade pulserna mot en fotodiod, som skickade signaler till ett oscilloskop. Därigenom erhöles en grov uppskattning av vilken fördröjning som gav överlapp mellan pulserna.

3.3 Användargränssnitt

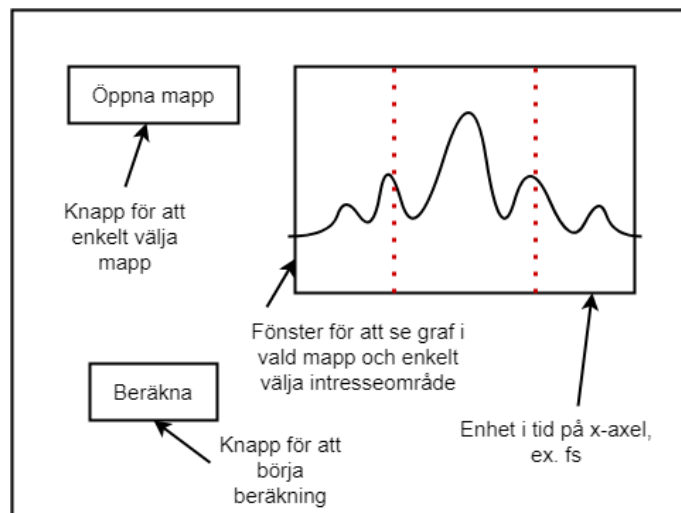
Vid skapandet av det användargränssnitt som krävdes för att underlätta kalibrering och dataanalys övervägdes vilka användningsområden som behövde utvecklas. Av de områden som ansågs relevanta att prioritera valdes användarvänligheten, information om den valda datan samt att på ett enkelt sätt kunna analysera grafer, med några önskade värden tydligt utskrivna.

För att enkelt överföra data utan att ändra koden manuellt vid byte av fil krävdes att koden kunde ta ut informationen från filnamnet oberoende av var filen var placerad eller vilken enhet koden kördes på.

Genom att öka användarvänligheten kunde datan analyseras enklare av fler personer utan direkt förståelse för koden som bearbetade datan. För att leda användaren att bearbeta datan i rätt ordning skapades en separat panel där en direkt uppdatering visade var i processen användaren befann sig och där det framgick vad nästa steg var. Användaren behövde även ett enkelt sätt att avgöra om datan aktivt bearbetades eller om det krävdes ett aktivt val för att nå nästa steg. Denna vägledning behövde vara tydlig och på ett språk som förstås av samtliga användare.

Informationen om den valda datan behövde analyseras i ett rimligt intresseområde (eng. region of interest, ROI) som bestämdes av användaren och beräkningarna utfördes efter att användaren klickade på en knapp. Användaren skulle inte behöva lämna applikationen för att genomföra ett val och därför skulle all information vara tillgänglig och enkelt synlig i applikationen. Valen behövde dessutom vara av en okomplicerad karaktär av typen att klicka på en knapp, välja fil eller skriva in siffervärden.

Graferna med värden av intresse skulle sedan presenteras tydligt, efter att samtliga steg genomförts. Av intresse var bland annat vilket fördröjningssteg som gav högst intensitet, det vill säga bäst överlapp av laserpulserna. Med samtliga krav i fokus planerades en skiss över de funktioner som appen behövde innehålla samt över ett ungefärligt användargränssnitt, se figur 3.5.



Figur 3.5: Skiss av det planerade användargränssnittet med sektioner för att öppna filer, konstruera grafer och göra beräkningar med datan.

3.3.1 Datainsamling

Detektorn samlade in data om jonernas flygtid samt om intensiteten, det vill säga antalet joner som träffade detektorn vid varje tidpunkt. Informationen samlades in av ett LabVIEW-program som överförde datan till en Excel-fil. För varje justering av avståndet i fördröjningssteget skapades en ny Excel-fil innehållande flygtid i första kolumnen och intensitet i andra kolumnen. Varje försök med olika ämnen tilldelades en mapp innehållande lika många Excel-filer som antalet justeringar av fördröjningen.

För varje Excel-fil summerades samtliga värden för intensiteten och en graf ritades med dessa summerade intensiteter som funktion av fördröjningen. Datainsamlingen planerades ge en graf för intensiteten som funktion av flygtiden, samt en graf med den sammanlagda intensiteten som funktion av fördröjningen.

3.3.2 Dataanalys

Först analyserades grafen med intensiteten mot flygtiden. Där avgränsades ett fönster där mest aktivitet skett. För denna kalibrering valdes xenon för att kunna jämföra den insamlade datan mot kända värden för de stabila isotoperna. För dessa förväntades karakteristiska toppar i spektrumet, där isotopernas relativa massor

svarar mot avståndet mellan topparna (det vill säga mot flygtiden) medan den relativa förekomsten av respektive isotop svarar mot toppens intensitet.

För grafen med intensitet som funktion av fördröjning valdes en rimlig gaussisk kurvanpassning för att förenkla analys av bland annat halvvärdesbredd.

4

Resultat och diskussion

Projektet kan delas in i mindre delprojekt. En aspekt av projektet har varit att hitta rumslig överlappning mellan IR- och UV-pulser, medan en annan har handlat om tidsöverlapp mellan pulserna. Dessutom har ett användarvänligt användargränssnitt utgjort en signifikant del av projektet. Nedan presenteras resultat och diskussion för respektive delprojekt.

4.1 Rumslig överlappning

IR-pulsens diameter uppmättes till 0,25 mm och halvvärdesbredden till 9/55 mm ($\approx 0,16$ mm). Med en effekt på 720 $\mu\text{J}/165$ fs ($\approx 4,4 \cdot 10^9$ W) blir intensiteten:

$$I_{IR} = \frac{720 \mu\text{J}}{165 \text{ fs}} \frac{1}{\left(\frac{1}{2} \cdot \frac{9}{550} \text{ cm}\right)^2 \cdot \pi} \approx 2,1 \cdot 10^{13} \frac{\text{W}}{\text{cm}^2}.$$

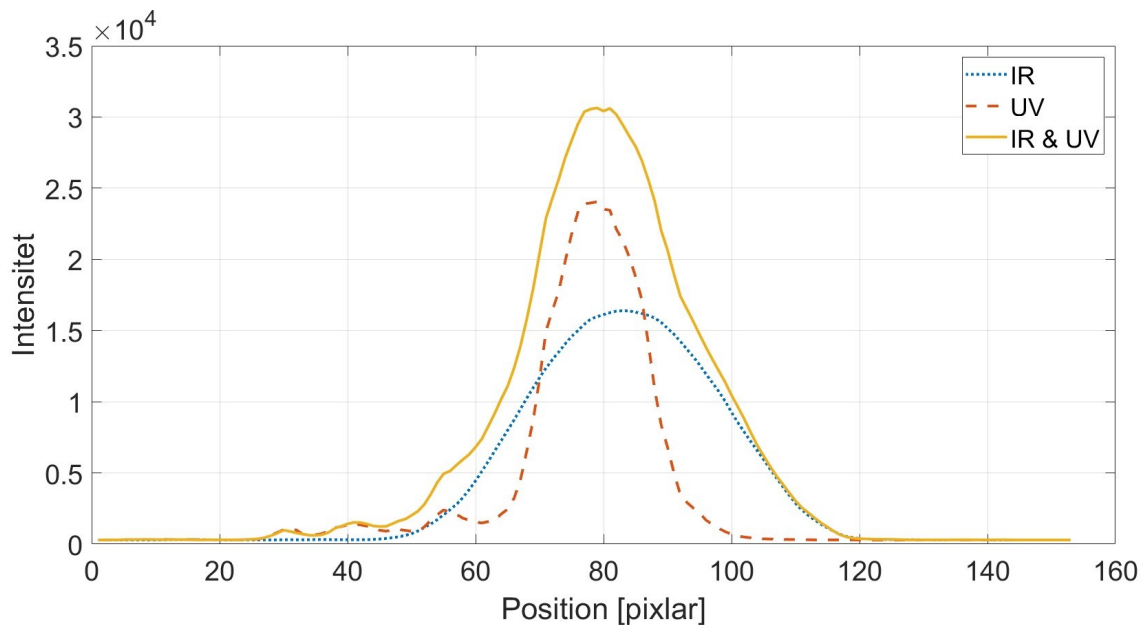
För UV-pulsen uppmättes en diameter på 0,1 mm och halvvärdesbredden till 17/220 mm ($\approx 0,077$ mm). Med effekten 95 $\mu\text{J}/165$ fs ($\approx 0,58 \cdot 10^9$ W) blir intensiteten för UV-pulsen:

$$I_{UV} = \frac{95 \mu\text{J}}{165 \text{ fs}} \frac{1}{\left(\frac{1}{2} \cdot \frac{17}{2200} \text{ cm}\right)^2 \cdot \pi} \approx 1,2 \cdot 10^{13} \frac{\text{W}}{\text{cm}^2}.$$

Att effekten är högre för IR-pulsen än för UV-pulsen beror på att det finns fler IR-fotoner än UV-fotoner i fokuspunkten. Det i sin tur beror dels på att en IR-fotons energi utgör en fjärdedel av den hos en UV-foton, vilket innebär att det krävs fyra IR-fotoner för att skapa en UV-foton. En annan faktor är att verkningsgraden för UV-modulen är lägre än 100 %, vilket innebär att inte alla inkommande IR-fotoner omvandlas till UV-fotoner.

Trots att UV-pulsens effekt är omkring hälften av IR-pulsens har den ett högre intensitetsmaximum än IR-pulsen, se figur 4.1. Detta är en följd av att UV-pulsens diameter är mindre än IR-pulsens.

Det uppmätta värdet på intensiteten var av tillräckligt hög storleksordning för att jonisering skulle ske och kalibreringen kunde därför fortsätta. Att den ungefärliga



Figur 4.1: Intensitet för respektive puls samt summan av dem som funktion av position i intresseområdet (ROI) mätt i pixlar. Figuren visar lägst topp för IR-pulsen, följt av UV-pulsen och högst topp för den kombinerade IR & UV-pulsen.

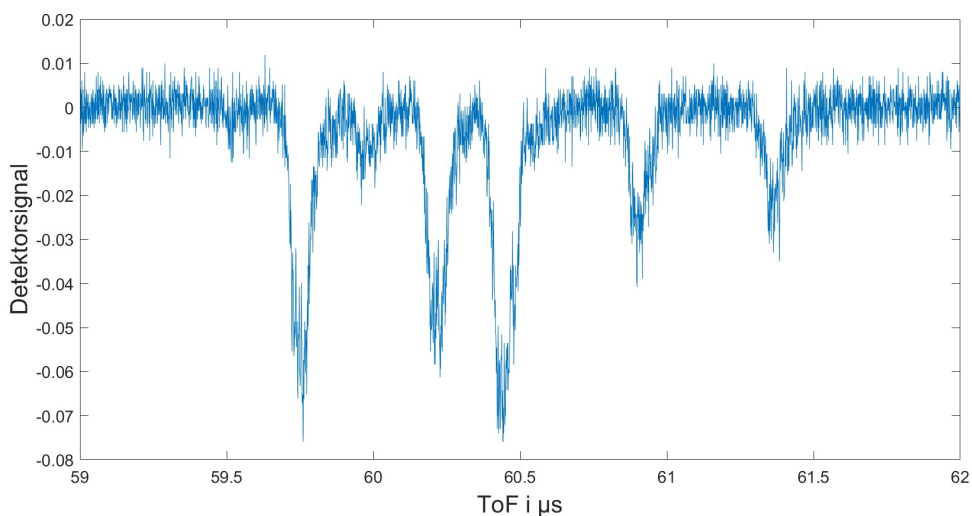
intensiteten kunde beräknas från grafen anses uppfylla syftet, då denna metod nu kan appliceras på andra lasrar.

4.2 Tidsöverlappning

Vid kalibreringen med xenon framträder sex tydliga toppar i masspektrumet, se figur 4.2. Högst intensitet uppmäts i denna mätning vid flygtider på ca 59,8 fs, 60,3 fs och 60,5 fs medan mindre toppar uppmäts vid ca 60 fs, 60,9 fs och 61,4 fs. I samma mätning uppmättes fördröjningen för maximalt överlapp mellan UV- och IR-pulserna vid ca 2 335 fs, med ett värde på halvvärdesbredden på 170 fs, för beräkningar med 500 fördröjningssteg på 10 fs per steg. Vid övriga mätningar noterades andra värden beroende på positionen av fördröjningssteget vid starten.

Det maximala överlappet sker alltså vid ett relativt värde beroende av uppställningen, men så som koden är uppbyggd kan värdet av maximalt överlapp mellan UV- och IR-pulserna tas fram för varje uppställning. Användaren kan då avläsa vilket fördröjningssteg som ger maximalt överlapp och kalibrera uppställningen därefter. Utöver just fördröjningssteget som ger maximalt värde kan användaren även avgöra ett specifikt fördröjningssteg som ger en önskad fördröjning mellan pulserna.

De uppmätta topparnas relativa intensitet stämmer överens med vad som förväntas för de olika isotoperna av xenon. Topporna vid 59,8 fs, 60 fs, 60,3 fs, 60,5 fs, 60,9 fs och 61,4 fs bedöms tillhöra isotoperna ^{129}Xe , ^{130}Xe , ^{131}Xe , ^{132}Xe , ^{134}Xe respektive



Figur 4.2: Masspektrum för xenon, med sex toppar motsvarande de vanligast förekommande isotoperna.

^{136}Xe , medan förekomsten av de tre isotoperna ^{124}Xe , ^{126}Xe och ^{128}Xe är för låg för att de ska synas i masspektret, se tabell 4.1. Då flygtiden bland annat beror på styrkan hos det elektriska fältet som accelererar jonerna kan dessa värden på flygtiderna endast användas för den specifika uppställningen som kalibreringen utförs i. Förhållandet för flygtiderna, alltså de relativa avstånden mellan topparna samt deras relativa intensitet, är dock konstanta för respektive ämne och genom att analysera ämnets karakteristiska toppar kan uppställningen kalibreras.

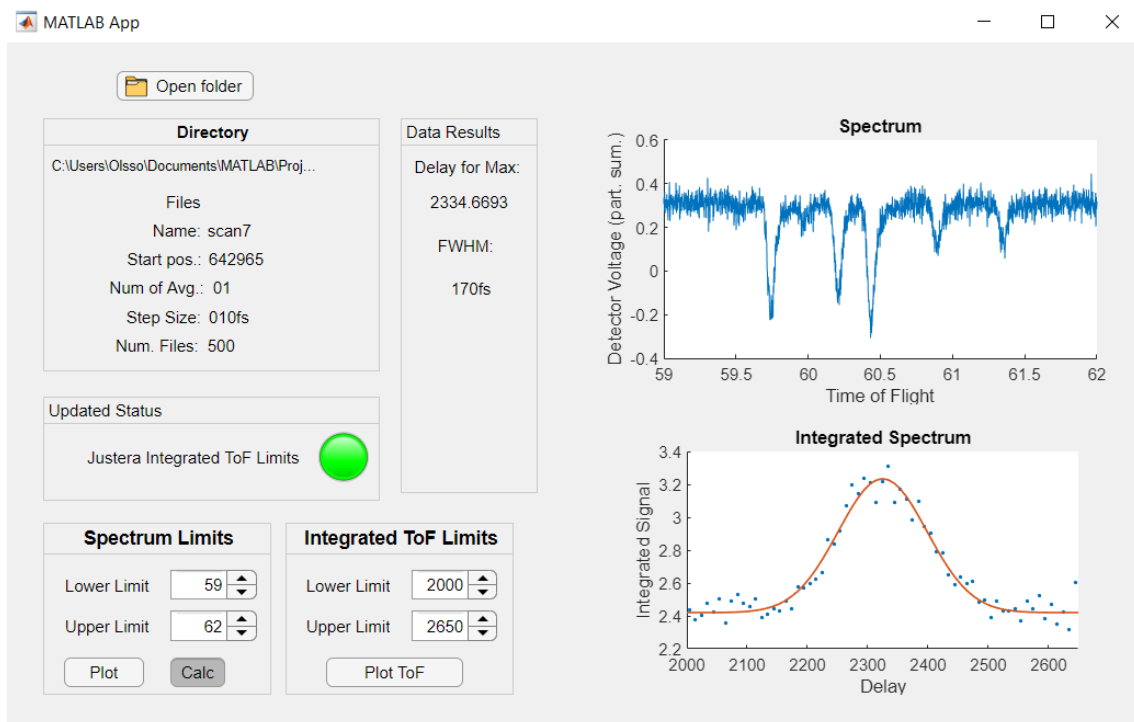
Att endast kalibrera med ett ämne ger ett snävt fönster i masspektret där kalibreringen anses korrekt. För ökad precision utanför de flygtider där xenonisotoperna ger utslag krävs att ytterligare ett ämne används, till exempel kväve. Detta eftersom xenonisotoperna ger utslag inom ett begränsat intervall och kalibreringen blir osäkrare för värden längre bort från intervallet. Att kalibrera med två ämnen ger ökad precision inte bara inom de två intervallen utan även utanför de aktuella intervallen. Koden upplevs åstadkomma målet som förväntades men har endast testats för kalibrering med xenon. För en mer noggrann kalibrering krävs ytterligare ämnen.

4.3 Användargränssnitt

I figur 4.3 ses ett användargränssnitt som underlättar analys av data. I *Directory* presenteras den öppnade filen och information från filnamnet, medan statusfältet ger användaren uppdateringar med beskrivning av nästa steg, för att visa i vilken ordning datan behandlas. Grafernas gränser kan väljas i *Spectrum Limits* respektive *Integrated ToF Limits* och graferna uppdateras genom att klicka på *Plot* respektive *Plot ToF*. Värdena för fördröjningssteget som ger maximalt överlapp samt för halvvärdesbredden presenteras i *Data Results*.

Tabell 4.1: Massa samt förekomst av de nio stabila isotoperna av xenon [2]

Isotop	Massa, u	Förekomst, %
Xe 124	123,905 892	0,095
Xe 126	125,904 298	0,089
Xe 128	127,903 531	1,910
Xe 129	128,904 781	26,401
Xe 130	129,903 509	4,071
Xe 131	130,905 084	21,232
Xe 132	131,904 155	26,909
Xe 134	133,905 394	10,436
Xe 136	135,907 215	8,857

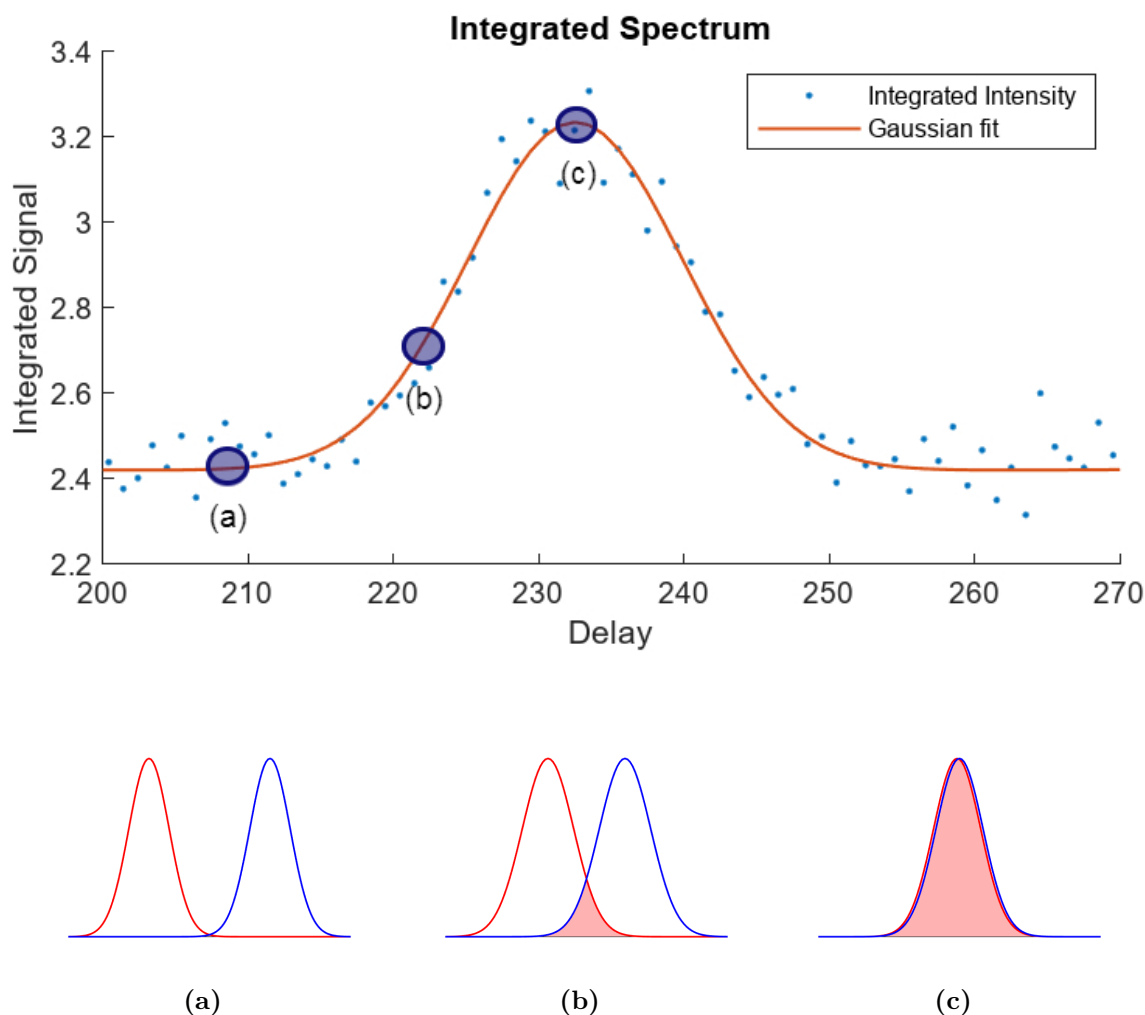
**Figur 4.3:** Användargränssnittet som skapades för att konstruera grafer och ta fram information från data. I bilden syns fönster uppdelade i *Directory*, *Status*, *Limits* och *Spectrum*.

Vid en av mätningarna uppmäts det maximala värdet för den integrerade intensiteten av detekterade joner vid en fördröjning på ca 2 335 fs mellan UV-pulsen och IR-pulsen. Denna mätning gjordes med 500 filer och 10 fs per steg. Vid denna fördröjning överlappar de två pulserna helt och maximalt antal xenonatomer joniseras, se figur 4.4 och figur 4.5.

Den gaussiska anpassningen valdes till MATLABs Gauss3 funktion som använder minst nio filer för analys och ger nio koefficienter till gauss-kurvan. I kurvan är följande variabler koefficienterna som är resultaten av anpassningen;

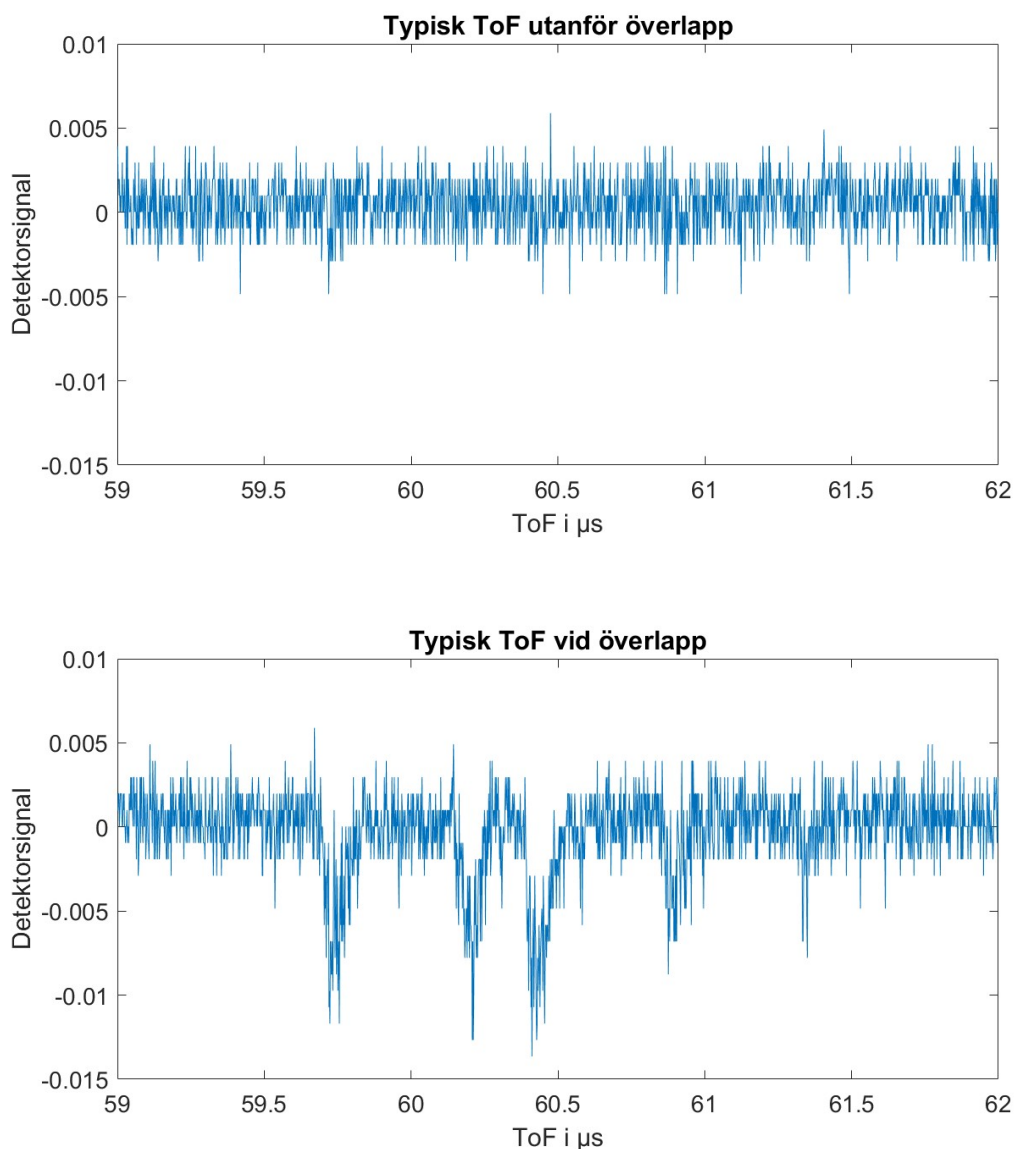
$a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3, c_3$. Gaussanpassningen Y följer funktionen enligt nedan:

$$Y = a_1 \cdot e^{-((x-b_1)/c_1)^2} + a_2 \cdot e^{-((x-b_2)/c_2)^2} + a_3 \cdot e^{-((x-b_3)/c_3)^2}.$$



Figur 4.4: Integrerat spektrum som ritas upp i MATLAB-applikationen. Datapunkterna och den gaussiska kurvanpassningen visar den integrerade intensiteten av detekterade joner som funktion av fördröjningen mellan IR-pulsen och UV-pulsen. **(a)** När pulserna inte alls överlappar ges inget masspektrum då xenatomerna inte joniseras. **(b)** När pulserna delvis överlappar ökar sannolikheten att atomerna joniseras vilket medför att ett masspektrum kan urskiljas. **(c)** Med helt överlappade pulser joniseras maximalt antal atomer och masspektret blir som tydligast.

Målet med appen vara att ge en tydlig överblick, men även att ta fram specifik data som kan vara till hjälp vid kalibrering. Detta mål uppnås och den fördröjning relativt fördröjningsstegets startposition som ger maximal överlapp mellan pulserna presenteras korrekt. Värdena i exemplen är relativa och beroende av uppställningen.



Figur 4.5: Om IR- och UV-pulserna inte överlappar joniseras inte tillräckligt många atomer för att något annat än brus ska kunna urskiljas i spektrumet, se den övre grafen. Detta motsvarar figur 4.4(a). Då pulserna är fullständigt överlappade joniseras maximalt antal xenonatomer och de karakteristiska topparna syns som tydligast i spektrumet, se den nedre grafen. Detta motsvarar figur 4.4(c).

För noggrannare kalibrering kan fler ämnen än xenon användas och kortare fördröjningssteg tas, men precisionen begränsas även av pulslängden.

I användargränssnittet finns utrymme för att presentera mer data från graferna, se figur 4.3, så som topparnas placering. I vissa datamängder fanns dock enstaka värden som var av flera storleksordningar större än de andra. Dessa värden tolkades inte som användbar data. Funktioner som rensar ut värden som inte ska ingå i datan, samt minimerar bruset i datan, är områden som kan förbättras. Dessutom saknar

appen möjlighet att analysera ny mätdata efter att en mätning redan öppnats, vilket behöver åtgärdas.

5

Slutsatser

Intensiteten för IR- och UV-pulserna beräknades till $2,075 \cdot 10^{13}$ W/cm² respektive $1,228 \cdot 10^{13}$ W/cm². Den högre intensiteten hos IR-pulsen beror på ett större antal IR-fotoner än UV-fotoner.

Appen utför de planerade funktionerna och kan användas för kalibrering av fördröjningssteget samt analys av datan i form av grafer och valda värden som presenteras visuellt. I ett exempel där mätningen består av 500 filer med 10 fs fördröjningssteg syns ett tydligt maximalt överlapp mellan UV- och IR-pulserna som uppnås runt 2 300 fs. I det fallet uppmättes värdet på halvvärdesbredden (FWHM) till 170 fs. Eftersom varje puls är 165 fs lång bör den totala bredden på kurvan ligga på 330 fs vilket innebär att 170 fs är ett rimligt FWHM-värde. De uppmätta värdena är relativa och beroende av inställningarna för mätningen i fråga.

Användargränssnittet innehåller de planerade funktionerna. För närvarande saknas möjlighet att genomföra analys av flera mätningar i följd och omstart av applikationen krävs för att kunna öppna en ny mapp med mätdata. Vidareutveckling skulle därför kunna innefatta en uppdatering som möjliggör öppning av en ny mapp direkt efter att en analys har genomförts.

Litteratur

1. Henriksson, S., *Telomerer – stamceller, cancer och livslängd* (<https://cancer.se/telomerer-stamceller-cancer-och-livslangd/>), (hämtad: 09.05.2023).
2. Nordling, C. & Österman, J., *Physics Handbook for Science and Engineering (9 uppl.)* (Studentlitteratur, Lund, 2020), ISBN: 978-91-44-12806-1.
3. Nationalencyklopedin, *permittivitet* (<https://www.ne.se/uppslagsverk/encyklopedi/l%C3%A5ng/permittivitet>), (hämtad 26.05.2023).
4. Young, H. D. & Freedman, R. A., *University Physics with Modern Physics, Global Edition* (Pearson Education Limited, Harlow, utg. 15, 2020), ISBN: 9781292314730.
5. Paschotta, R., *Nonlinear Index* (https://www.rp-photonics.com/nonlinear_index.html), (hämtad 26.05.2023).
6. Paschotta, R., *Frequency Doubling* (https://www.rp-photonics.com/frequency_doubling.html), (hämtad: 26.05.2023).
7. Boyd, R. W., *Nonlinear Optics (3:e Upplagan)* (Academic Press Inc, 2008), ISBN: 9780123694706.
8. EK SMA Optics, *Beta barium borate – BBO crystals* (<https://eksmaoptics.com/nonlinear-and-laser-crystals/nonlinear-crystals/beta-barium-borate-bbo-crystals/>), (hämtad 25.05.2023).
9. Paschotta, R., *Stimulated Emission* (https://www.rp-photonics.com/stimulated_emission.html), (hämtad: 27.05.2023).
10. Paschotta, R., *Population inversion* (https://www.rp-photonics.com/population_inversion.html), (hämtad 02.05.2023).
11. Paschotta, R., *Pump-probe Measurements* (https://www.rp-photonics.com/pump_probe_measurements.html), (hämtad 10.05.2023).
12. Mamyryn, B., *Time-of-flight mass spectrometry (concepts, achievements, and prospects)* (<https://www.sciencedirect.com/science/article/pii/S1387380600003924>), (hämtad 27.05.2023).

A

MATLAB-kod för beräkning av intensitet

```
1 % ----- Options ----- %
2
3 clear all
4 close all
5 % Forklaring: pl = 1 ger plot/bild.
6 % pl = 0 ger ej plot/bild.
7 pl = 1;
8
9 % Forklaring: Lagg alla bilder du vill analysera i en mapp
10 % Importera mapp av bilder
11 myFolder = 'C:\Users\ST\OneDrive\Dokument\MATLAB\Bilder';
12
13
14 %%
15 % ----- Import and readjust ----- %
16
17 % Skapa ett 'directory' med bl. a. namn över bmp filer.
18 filePattern = fullfile(myFolder, '*.bmp');
19 bmpFiles = dir(filePattern);
20
21 % Valjer en bild att skapa inställningar efter
22 a = bmpFiles(3).name;
23 iImage = fullfile(myFolder, a);
24 I = imread(iImage);
25
26 % Gör om 3d matris till 2d.
27 % Resulting image will be grayscale.
28 I=rgb2gray(I);
29
30
31 %%
32 % ----- Rotate ----- %
33
```

```

34 % Valja antalet grader att rotera
35 figure
36 imshow(I)
37 title('Click two points on the line, then enter:')
38 [xi, yi] = getpts;
39
40 x = round(xi(2)-xi(1));
41 y = round(yi(2)-yi(1));
42 angle = 90 + (180/pi)*atan(y/x);
43 J = imrotate(I, angle);
44
45
46 %%
47 % ----- Choose ROI ----- %
48
49 % Valj ROI
50 imshow(J);
51 title('Drag a rectangle, then double click:')
52 h = imrect;
53 position = wait(h);
54
55 croppedImage = imcrop(J, position);
56 close all
57
58
59 %%
60 % ----- Finalize first image ----- %
61
62 % Integrera över alla rader.
63 % Skapa graf över intensiteten i pixlarna
64 Curvy = sum(croppedImage);
65
66 % Plot and show
67 if pl == 1
68     figure;
69     imshow(croppedImage);
70     plot(Curvy)
71 end
72
73
74 %%
75 % ----- Finalize the rest ----- %
76
77 % Skapa loop som använder installningarna till alla
78 % bmp-filer
79 linS = {':', '—', '-'};

```

```

80 for n = 1:size(bmpFiles,1)
81     FilNamn = bmpFiles(n).name;
82     Bild = fullfile(myFolder, FilNamn);
83     K = imread(Bild);
84     K = imrotate(K, angle);
85     K=rgb2gray(K);
86     Croppy = imcrop(K, position);
87     Kurva = sum(Croppy);
88     Curvymax = max(Kurva);
89     Curvymin = min(Kurva);
90
91     %Hitta halva maxvardet
92     halfmax = max(Kurva) / 2;
93
94     %Hitta forsta punkten dar kurvan nar halva max
95     index1 = find(Kurva >= halfmax, 1, 'first');
96
97     %Hitta sista punkten dar kurvan nar halva max
98     index2 = find(Kurva >= halfmax, 1, 'last');
99
100    % Index for FWHM
101    fwhm = index2-index1 + 1;
102    fwhm_list(n) = fwhm;
103
104    if pl == 1
105        %plot(Kurva)
106        txt = [bmpFiles(n).name];
107        plot(Kurva, 'DisplayName', txt, 'linestyle', linS{n})
108        xlabel('Position [pixlar]')
109        ylabel('Intensitet')
110        ax = gca;
111        ax.FontSize = 20;
112        grid on
113        hold on
114    end
115 end
116
117 if pl ==1
118     legend('IR', 'UV', 'IR & UV')
119 end
120
121 legend show
122
123
124 %%
125 % ----- FWHM & intensity ----- %

```

A. MATLAB-kod för beräkning av intensitet

```
126
127 %Berakna FWHM och intensitet , I
128 %Berakna arean med FWHM som diameter
129 %I= (1/area) * energi/pulstid = (1/area) *
130 % (microjoule / 165fs)
131 microjoule = [720*10(-6) 95*10(-6) (720+95)*10(-6)];
132 Intens = [];
133
134 for i = 1:size(fwhm_list,2)
135     % plattans bredd: 220 pixlar = 1 mm =>
136     % diameter: (fwhm_list/220) mm = (fwhm/(220*10)) cm =>
137     % radie: (fwhm/(2*220*10)) cm = (fwhm/(4400)) cm
138     I = (1/(pi*(fwhm_list(i)/(4400))^2)) * (microjoule(i)...
139         ... / (165*10(-15)));
140     Intens(i)=I;
141 end
142
143 Intensity = Intens
```


B

MATLAB-kod för användargränssnitt

```
1 classdef AppXe4AO < matlab.apps.AppBase
2
3     % Properties that correspond to app components
4     properties (Access = public)
5         UIFigure                matlab.ui.Figure
6         UIAxes                  matlab.ui.control.UIAxes
7         DirectoryPanel          matlab.ui.container.Panel
8         StepSizeLabel           matlab.ui.control.Label
9         NumberOfAveragesLabel   matlab.ui.control.Label
10        StartpositionLabel       matlab.ui.control.Label
11        NameLabel                 matlab.ui.control.Label
12        PathLabel                 matlab.ui.control.Label
13        NameLabel_2              matlab.ui.control.Label
14        StartposLabel            matlab.ui.control.Label
15        NumofAvgLabel            matlab.ui.control.Label
16        StepSizeLabel_2         matlab.ui.control.Label
17        FilesLabel               matlab.ui.control.Label
18        NumFilesLabel            matlab.ui.control.Label
19        NumberLabel              matlab.ui.control.Label
20        OpenfolderButton         matlab.ui.control.Button
21        UIAxes_2                 matlab.ui.control.UIAxes
22        SpectrumLimitsPanel      matlab.ui.container.Panel
23        PlotButton               matlab.ui.control.Button
24        CalcButton               matlab.ui.control.StateButton
25        LowerLimitSpinnerLabel   matlab.ui.control.Label
26        LowerLimitSpinner        matlab.ui.control.Spinner
27        UpperLimitSpinnerLabel   matlab.ui.control.Label
28        UpperLimitSpinner        matlab.ui.control.Spinner
29        IntegratedToFLimitsPanel matlab.ui.container.Panel
30        LowerLimitSpinnerLabel_2 matlab.ui.control.Label
31        LowerLimitSpinner_2      matlab.ui.control.Spinner
32        UpperLimitSpinnerLabel_2 matlab.ui.control.Label
33        UpperLimitSpinner_2      matlab.ui.control.Spinner
```

```

34         PlotToFButton          matlab.ui.control.Button
35         UpdatedStatusPanel     matlab.ui.container.Panel
36         pppnamappviaOpenFolderLamp
37         Label matlab.ui.control.Label
38
39         pppnamappviaOpenFolderLamp matlab.ui.control.Lamp
40         DataResultsPanel       matlab.ui.container.Panel
41         DelayforMaxLabel       matlab.ui.control.Label
42         DelayLabel             matlab.ui.control.Label
43         FWHMLabel             matlab.ui.control.Label
44         ValLabel              matlab.ui.control.Label
45         ContextMenu
46         matlab.ui.container.ContextMenu
47
48         Menu                   matlab.ui.container.Menu
49         Menu2                  matlab.ui.container.Menu
50     end
51
52     % Callbacks that handle component events
53     methods (Access = private)
54
55         % Button pushed function: OpenfolderButton
56         function OpenfolderButtonPushed(app, event)
57
58             % TOF tid till detektorn,
59             % spektrum ar intensiteten for varje
60             % tof, centrera grafen for time zero.
61             % Integrated signal i 0 och
62             % offset i delay linspace
63
64             % Bestammer filens plats och strukturerat
65
66             f = figure('Renderer', 'painters', 'Position',
67                 [-100 -100 0 0]);
68             % create a dummy figure so that uigetfile
69             % doesn't minimize our GUI
70             folder = uigetdir();
71             app.PathLabel.Text = folder;
72             files=dir([folder, '\*.csv']);
73             % Struct with csv-files.
74             % Hamtar alla .cvs-filer ur folder
75             delete(f); %delete the dummy figure
76             % Uppdaterar status
77
78             app.pppnamappviaOpenFolderLampLabel.Text =
79                 'Noterat mapp';

```

```
80     app.ppnamappviaOpenFolderLamp.Color = 'blue';
81     drawnow;
82
83     % Delar folder och tar ut det efter sista '\'
84
85     x = strsplit(folder, '\');
86     scan = x{end};
87
88     % Hittar dar vi delar upp filnamnet med '_'
89
90     o = strfind(scan, '_');
91
92     % Tar ut information fran filnamnet
93
94     scan_name = scan(1:o(1)-1);
95     start_position = scan(o(1)+1:o(2)-1);
96     number_of_average = scan(o(2)+1:o(3)-1);
97     step_size = scan(o(3)+1:length(scan));
98
99     % Bestammer storleken pa filen
100
101     big=size(files);
102     stlk = str2double(big(1,1));
103
104     % Uppdaterar namnen i panelen och status
105
106     app.NameLabel.Text = scan_name;
107     app.StartpositionLabel.Text = start_position;
108     app.NumberofAveragesLabel.Text =
109     number_of_average;
110     app.StepSizeLabel.Text = step_size;
111     app.NumberLabel.Text = num2str(length(files));
112     app.ppnamappviaOpenFolderLampLabel.Text =
113     'Hamtar filer ...';
114     app.ppnamappviaOpenFolderLamp.Color = 'blue';
115     drawnow;
116
117     % Gor om namnen fran strangar till double
118
119     number_of_average =
120     cat(1, str2double(number_of_average));
121
122     ind = find(step_size(:) ~= 0);
123     step_size = eraseBetween(step_size, 1, ind(1));
124     Remove_These = isletter(step_size);
125     Units = step_size(Remove_These);
```

```
126         step_size(Remove_These) = [];
127         step_size = cat(1, str2double(step_size));
128
129
130         % Hamtar filderna från Directory och
131         % uppdaterar status
132
133         for u=1:size(files,1)
134             % Fran 1 till storleken pa "files"
135
136                 filename{u}= [folder, '\', files(u).name];
137                 % Hamtar filnamn pa "u:te" filen i folder
138                 data{u} = importdata(filename{u});
139         end
140
141         app.ppnamappviaOpenFolderLampLabel.Text =
142             'Valj ROI via Spectrum limits och Plot';
143         app.ppnamappviaOpenFolderLamp.Color = 'green';
144
145         %-----%
146         % Tar ut data om spektrumet och time of flight
147
148         spectrum = 0;
149         spectrum2 = 0;
150         for u=1:size(files,1)
151             data_point=data{u};
152             time_flight=data_point(:,1);
153             background=-1.2e-3;
154             if max(abs(data_point(:,2))) >
155                 max(abs(spectrum2))/1.2
156                 % Div vardet kan andras
157                 spectrum= spectrum +
158                     data_point(:,2)-background;
159                 spectrum2 = data_point(:,2)-background;
160             end
161         end
162
163         % Plottar in i de tva graferna
164
165         plot(app.UIAxes, time_flight, spectrum);
166         app.ppnamappviaOpenFolderLampLabel.Text =
167             'Valj ROI via Spectrum limits';
168         app.ppnamappviaOpenFolderLamp.Color = 'green';
169         drawnow;
170
171         % Startar berakning efter klickat i calc.
```

```

172     % uppdaterar status
173
174     while (app.CalcButton.Value == 0)
175         pause(1);
176     end
177
178     app.CalcButton.Value = 1;
179     app.ppnamappviaOpenFolderLampLabel.Text =
180     'Beraknar ...';
181     app.ppnamappviaOpenFolderLamp.Color = 'blue';
182     drawnow;
183     ned = int32(app.LowerLimitSpinner.Value);
184     upp = int32(app.UpperLimitSpinner.Value);
185
186     app.UIAxes.XLim = [ned, upp];
187
188
189     ned2=(ned-time_flight(1))*length(data_point)/
190     (time_flight(length(data_point))-
191     time_flight(1));
192     upp2=length(data_point)-
193     (time_flight(length(data_point))-upp)*
194     length(data_point)/
195     (time_flight(length(data_point))-
196     time_flight(1));
197
198     if upp2 == 0
199         upp2 = length(data_point)
200     end
201
202     for u=1:size(files,1)
203
204         data_point=data{u};
205         time_flight=data_point(:,1);
206         background=-1.2e-3;
207         spectrum=data_point(:,2)-background;
208         %plot(app.UIAxes,spectrum);
209         cut_time_flight=time_flight(ned2+1:upp2);
210         cut_spectrum=spectrum(ned2+1:upp2);
211
212         integration(u)=sum(abs(cut_spectrum));
213         %peak_value(u)=max(abs(cut_spectrum));
214     end
215     integration = transpose(integration);
216     si = size(integration);
217     leg = si(1);

```

```
218     ned2 = double(ned2);
219     upp2 = double(upp2);
220     x = linspace(0,leg,leg);
221     x = transpose(x);
222     x = step_size.*x;
223
224     % Tar datan efter Gauss
225
226     fitline = fit(x,integration,'gauss3');
227     koeff = coeffvalues(fitline);
228     % 9st koefficients for gauss3
229
230
231     koeff = coeffvalues(fitline);
232     a1 = double(koeff(1,1));
233     b1 = double(koeff(1,2));
234     c1 = double(koeff(1,3));
235     a2 = double(koeff(1,4));
236     b2 = double(koeff(1,5));
237     c2 = double(koeff(1,6));
238     a3 = double(koeff(1,7));
239     b3 = double(koeff(1,8));
240     c3 = double(koeff(1,9));
241     Y = a1*exp(-((x-b1)/c1).^2)+
242     a2*exp(-((x-b2)/c2).^2) +
243     a3*exp(-((x-b3)/c3).^2);
244
245     plot(app.UIAxes_2, x, integration,
246     'Marker','.', 'LineStyle','none');
247     hold(app.UIAxes_2, 'on');
248     plot(app.UIAxes_2, x, Y, 'LineWidth', 1);
249     %legend(app.UIAxes_2, 'Integrated Intensity',
250     'Gaussian fit');
251
252     % Find the half max value.
253     halfMax = (min(Y) + max(Y)) / 2;
254     % Find where the data first drops
255     % below half the max.
256     index1 = find(Y >= halfMax, 1, 'first');
257     % Find where the data last rises
258     % above half the max.
259     index2 = find(Y >= halfMax, 1, 'last');
260     fwhm = index2-index1 + 1; % FWHM in indexes.
261     % OR, if you have an x vector
262     fwhmx = x(index2) - x(index1);
263     % Times stepsize
```

```
264
265
266     app.ValLabel.Text =
267     string(fwhm*step_size) + string(Units);
268
269     % Tar data till bast delay
270
271     [maxval_int, idx_int] = max(integration) ;
272     maxvalx = x(idx_int) ;
273     app.DelayLabel.Text = string(maxvalx);
274
275
276     % Plottar integration
277
278     app.UIAxes_2.XLim = [1, max(x)];
279     app.UIAxes_2.Title.String =
280     'Integrated Spectrum';
281
282     app.ppnamappviaOpenFolderLampLabel.Text =
283     'Justera Integrated ToF Limits';
284     app.ppnamappviaOpenFolderLamp.Color = 'green';
285
286     drawnow;
287
288
289 end
290
291 % Callback function
292 function PlotButtonValueChanged(app, event)
293
294 end
295
296 % Button pushed function: PlotButton
297 function PlotButtonPushed(app, event)
298
299     ned = app.LowerLimitSpinner.Value;
300     upp = app.UpperLimitSpinner.Value;
301
302     app.UIAxes.XLim = [ned, upp];
303
304 end
305
306 % Button pushed function: PlotToFButton
307 function PlotToFButtonPushed(app, event)
308     ned = app.LowerLimitSpinner_2.Value;
309     upp = app.UpperLimitSpinner_2.Value;
```

```
310         app.UIAxes_2.XLim = [ned, upp];
311
312     end
313 end
314
315 % Component initialization
316 methods (Access = private)
317
318     % Create UIFigure and components
319     function createComponents(app)
320
321         % Create UIFigure and hide until
322         % all components are created
323         app.UIFigure = uifigure('Visible', 'off');
324         app.UIFigure.Position = [100 100 770 509];
325         app.UIFigure.Name = 'MATLAB App';
326
327         % Create UIAxes
328         app.UIAxes = uiaxes(app.UIFigure);
329         title(app.UIAxes, 'Spectrum')
330         xlabel(app.UIAxes, 'Time of Flight')
331         ylabel(app.UIAxes,
332              'Detector Voltage (part. sum.)')
333         app.UIAxes.PlotBoxAspectRatio =
334         [1.98101265822785 1 1];
335         app.UIAxes.Position = [421 243 327 214];
336
337         % Create DirectoryPanel
338         app.DirectoryPanel = uipanel(app.UIFigure);
339         app.DirectoryPanel.TitlePosition = 'centertop';
340         app.DirectoryPanel.Title = 'Directory';
341         app.DirectoryPanel.FontWeight = 'bold';
342         app.DirectoryPanel.Position = [28 268 246 185];
343
344         % Create StepSizeLabel
345         app.StepSizeLabel = uilabel(app.DirectoryPanel);
346         app.StepSizeLabel.Position = [121 29 118 22];
347         app.StepSizeLabel.Text = {'Step Size'; ''};
348
349         % Create NumberofAveragesLabel
350         app.NumberofAveragesLabel =
351         uilabel(app.DirectoryPanel);
352         app.NumberofAveragesLabel.Position =
353         [124 50 115 22];
354         app.NumberofAveragesLabel.Text =
355         'Number of Averages';
```



```
356
357 % Create StartpositionLabel
358 app.StartpositionLabel =
359     uilabel(app.DirectoryPanel);
360 app.StartpositionLabel.Position =
361     [121 71 118 22];
362 app.StartpositionLabel.Text =
363     'Start position';
364
365 % Create NameLabel
366 app.NameLabel = uilabel(app.DirectoryPanel);
367 app.NameLabel.Position = [120 92 119 22];
368 app.NameLabel.Text = 'Name';
369
370 % Create PathLabel
371 app.PathLabel = uilabel(app.DirectoryPanel);
372 app.PathLabel.HorizontalAlignment = 'center';
373 app.PathLabel.FontSize = 10;
374 app.PathLabel.Position = [4 144 197 12];
375 app.PathLabel.Text = 'Path';
376
377 % Create NameLabel_2
378 app.NameLabel_2 = uilabel(app.DirectoryPanel);
379 app.NameLabel_2.HorizontalAlignment = 'right';
380 app.NameLabel_2.Position = [45 92 71 22];
381 app.NameLabel_2.Text = 'Name: ';
382
383 % Create StartposLabel
384 app.StartposLabel = uilabel(app.DirectoryPanel);
385 app.StartposLabel.HorizontalAlignment = 'right';
386 app.StartposLabel.Position = [24 71 92 22];
387 app.StartposLabel.Text = 'Start pos.:';
388
389 % Create NumofAvgLabel
390 app.NumofAvgLabel = uilabel(app.DirectoryPanel);
391 app.NumofAvgLabel.HorizontalAlignment = 'right';
392 app.NumofAvgLabel.Position = [44 50 74 22];
393 app.NumofAvgLabel.Text = 'Num of Avg.:';
394
395 % Create StepSizeLabel_2
396 app.StepSizeLabel_2 =
397     uilabel(app.DirectoryPanel);
398 app.StepSizeLabel_2.HorizontalAlignment =
399     'right';
400 app.StepSizeLabel_2.Position = [25 29 91 22];
401 app.StepSizeLabel_2.Text = {'Step Size: '; ''};
```

```
402
403     % Create FilesLabel
404     app.FilesLabel = uilabel(app.DirectoryPanel);
405     app.FilesLabel.HorizontalAlignment = 'center';
406     app.FilesLabel.Position = [88 113 29 22];
407     app.FilesLabel.Text = 'Files';
408
409     % Create NumFilesLabel
410     app.NumFilesLabel = uilabel(app.DirectoryPanel);
411     app.NumFilesLabel.HorizontalAlignment = 'right';
412     app.NumFilesLabel.Position = [48 8 66 22];
413     app.NumFilesLabel.Text = 'Num. Files';
414
415     % Create NumberLabel
416     app.NumberLabel = uilabel(app.DirectoryPanel);
417     app.NumberLabel.Position = [120 8 87 22];
418     app.NumberLabel.Text = 'Number';
419
420     % Create OpenfolderButton
421     app.OpenfolderButton =
422     uibutton(app.UIFigure, 'push');
423     app.OpenfolderButton.ButtonPushedFcn =
424     createCallbackFcn(app,
425     @OpenfolderButtonPushed, true);
426     app.OpenfolderButton.Icon = '3767084.png';
427     app.OpenfolderButton.Position = [82 466 100 22];
428     app.OpenfolderButton.Text = 'Open folder';
429
430     % Create UIAxes_2
431     app.UIAxes_2 = uiaxes(app.UIFigure);
432     title(app.UIAxes_2, 'Waiting for input...')
433     xlabel(app.UIAxes_2, 'Delay')
434     ylabel(app.UIAxes_2, 'Integrated Signal')
435     app.UIAxes_2.PlotBoxAspectRatio =
436     [1.98295454545455 1 1];
437     app.UIAxes_2.YTick = [0 0.5 1];
438     app.UIAxes_2.Position = [421 31 327 196];
439
440     % Create SpectrumLimitsPanel
441     app.SpectrumLimitsPanel = uipanel(app.UIFigure);
442     app.SpectrumLimitsPanel.TitlePosition =
443     'centertop';
444     app.SpectrumLimitsPanel.Title =
445     'Spectrum Limits';
446     app.SpectrumLimitsPanel.FontWeight = 'bold';
447     app.SpectrumLimitsPanel.FontSize = 14;
```

```
448     app.SpectrumLimitsPanel.Position =
449     [28 31 159 125];
450
451     % Create PlotButton
452     app.PlotButton =
453     uibutton(app.SpectrumLimitsPanel, 'push');
454     app.PlotButton.ButtonPushedFcn =
455     createCallbackFcn(app, @PlotButtonPushed, true);
456     app.PlotButton.Position = [15 5 59 22];
457     app.PlotButton.Text = 'Plot';
458
459     % Create CalcButton
460     app.CalcButton =
461     uibutton(app.SpectrumLimitsPanel, 'state');
462     app.CalcButton.Text = 'Calc';
463     app.CalcButton.Position = [93 5 40 22];
464
465     % Create LowerLimitSpinnerLabel
466     app.LowerLimitSpinnerLabel =
467     uilabel(app.SpectrumLimitsPanel);
468     app.LowerLimitSpinnerLabel.
469     HorizontalAlignment = 'right';
470     app.LowerLimitSpinnerLabel.Position =
471     [11 69 67 22];
472     app.LowerLimitSpinnerLabel.Text =
473     'Lower Limit';
474
475     % Create LowerLimitSpinner
476     app.LowerLimitSpinner =
477     uispinner(app.SpectrumLimitsPanel);
478     app.LowerLimitSpinner.Position = [93 69 57 22];
479
480     % Create UpperLimitSpinnerLabel
481     app.UpperLimitSpinnerLabel =
482     uilabel(app.SpectrumLimitsPanel);
483     app.UpperLimitSpinnerLabel.
484     HorizontalAlignment = 'right';
485     app.UpperLimitSpinnerLabel.Position =
486     [11 39 67 22];
487     app.UpperLimitSpinnerLabel.Text = 'Upper Limit';
488
489     % Create UpperLimitSpinner
490     app.UpperLimitSpinner =
491     uispinner(app.SpectrumLimitsPanel);
492     app.UpperLimitSpinner.Position = [93 39 57 22];
493
```

```
494 % Create IntegratedToFLimitsPanel
495 app.IntegratedToFLimitsPanel =
496 uipanel(app.UIFigure);
497 app.IntegratedToFLimitsPanel.TitlePosition =
498 'centertop';
499 app.IntegratedToFLimitsPanel.Title =
500 'Integrated ToF Limits';
501 app.IntegratedToFLimitsPanel.FontWeight =
502 'bold';
503 app.IntegratedToFLimitsPanel.FontSize = 14;
504 app.IntegratedToFLimitsPanel.Position =
505 [197 31 159 125];
506
507 % Create LowerLimitSpinnerLabel_2
508 app.LowerLimitSpinnerLabel_2 =
509 uilabel(app.IntegratedToFLimitsPanel);
510 app.LowerLimitSpinnerLabel_2.
511 HorizontalAlignment = 'right';
512 app.LowerLimitSpinnerLabel_2.Position =
513 [10 69 67 22];
514 app.LowerLimitSpinnerLabel_2.Text =
515 'Lower Limit';
516
517 % Create LowerLimitSpinner_2
518 app.LowerLimitSpinner_2 =
519 uispinner(app.IntegratedToFLimitsPanel);
520 app.LowerLimitSpinner_2.Position =
521 [92 69 57 22];
522
523 % Create UpperLimitSpinnerLabel_2
524 app.UpperLimitSpinnerLabel_2 =
525 uilabel(app.IntegratedToFLimitsPanel);
526 app.UpperLimitSpinnerLabel_2.
527 HorizontalAlignment = 'right';
528 app.UpperLimitSpinnerLabel_2.Position =
529 [10 39 67 22];
530 app.UpperLimitSpinnerLabel_2.Text =
531 'Upper Limit';
532
533 % Create UpperLimitSpinner_2
534 app.UpperLimitSpinner_2 =
535 uispinner(app.IntegratedToFLimitsPanel);
536 app.UpperLimitSpinner_2.Position =
537 [92 39 57 22];
538
539 % Create PlotToFButton
```

```
540     app.PlotToFButton =
541     uibutton(app.IntegratedToFLimitsPanel, 'push');
542     app.PlotToFButton.ButtonPushedFcn =
543     createCallbackFcn(app,
544     @PlotToFButtonPushed, true);
545     app.PlotToFButton.Position = [30 5 100 22];
546     app.PlotToFButton.Text = 'Plot ToF';
547
548     % Create UpdatedStatusPanel
549     app.UpdatedStatusPanel = uipanel(app.UIFigure);
550     app.UpdatedStatusPanel.Title = 'Updated Status';
551     app.UpdatedStatusPanel.Position =
552     [28 172 246 76];
553
554     % Create ppnamappviaOpenFolderLampLabel
555     app.ppnamappviaOpenFolderLampLabel =
556     uilabel(app.UpdatedStatusPanel);
557     app.ppnamappviaOpenFolderLampLabel.
558     HorizontalAlignment = 'right';
559     app.ppnamappviaOpenFolderLampLabel.Position =
560     [11 21 176 22];
561     app.ppnamappviaOpenFolderLampLabel.Text =
562     'oppna mapp via Open Folder';
563
564     % Create ppnamappviaOpenFolderLamp
565     app.ppnamappviaOpenFolderLamp =
566     uilamp(app.UpdatedStatusPanel);
567     app.ppnamappviaOpenFolderLamp.Position =
568     [201 14 37 37];
569     app.ppnamappviaOpenFolderLamp.Color = [1 1 1];
570
571     % Create DataResultsPanel
572     app.DataResultsPanel = uipanel(app.UIFigure);
573     app.DataResultsPanel.Title = 'Data Results';
574     app.DataResultsPanel.Position =
575     [289 178 100 275];
576
577     % Create DelayforMaxLabel
578     app.DelayforMaxLabel =
579     uilabel(app.DataResultsPanel);
580     app.DelayforMaxLabel.Position = [10 229 83 22];
581     app.DelayforMaxLabel.Text = 'Delay for Max: ';
582
583     % Create DelayLabel
584     app.DelayLabel = uilabel(app.DataResultsPanel);
585     app.DelayLabel.HorizontalAlignment = 'center';
```

```
586     app.DelayLabel.Position = [0 203 100 22];
587     app.DelayLabel.Text = 'Delay';
588
589     % Create FWHMLabel
590     app.FWHMLabel = uilabel(app.DataResultsPanel);
591     app.FWHMLabel.Position = [27 171 46 22];
592     app.FWHMLabel.Text = 'FWHM: ';
593
594     % Create ValLabel
595     app.ValLabel = uilabel(app.DataResultsPanel);
596     app.ValLabel.HorizontalAlignment = 'center';
597     app.ValLabel.Position = [10 140 83 22];
598     app.ValLabel.Text = 'Val';
599
600     % Create ContextMenu
601     app.ContextMenu =
602     uicontextmenu(app.UIFigure);
603
604     % Assign app.ContextMenu
605     app.ppnamappviaOpenFolderLamp.ContextMenu =
606     app.ContextMenu;
607
608     % Create Menu
609     app.Menu = uimenu(app.ContextMenu);
610     app.Menu.Text = 'Menu';
611
612     % Create Menu2
613     app.Menu2 = uimenu(app.ContextMenu);
614     app.Menu2.Text = 'Menu2';
615
616     % Show the figure after all components
617     % are created
618     app.UIFigure.Visible = 'on';
619 end
620 end
621
622 % App creation and deletion
623 methods (Access = public)
624
625     % Construct app
626     function app = AppXe4AO
627
628     % Create UIFigure and components
629     createComponents(app)
630
631     % Register the app with App Designer
```

```
632         registerApp(app, app.UIFigure)
633
634         if nargin == 0
635             clear app
636         end
637     end
638
639     % Code that executes before app deletion
640     function delete(app)
641
642         % Delete UIFigure when app is deleted
643         delete(app.UIFigure)
644     end
645 end
646 end
```



GÖTEBORGS
UNIVERSITET