

**SIMULASI DAN PERHITUNGAN ALIGNMEN MOLEKUL
SERTA PEMBANGKITAN HARMONIK TINGGI PADA
MOLEKUL N_2 DAN O_2**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar
Sarjana Sains dalam bidang fisika

Oleh:

AWAN PASCA RAMADIKA

115090300111-017



**JURUSAN FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2018**

LEMBAR PENGESAHAN SKRIPSI

SIMULASI DAN PERHITUNGAN ALIGNMEN MOLEKUL SERTA PEMBANGKITAN HARMONIK TINGGI PADA MOLEKUL N_2 DAN O_2

Oleh:

AWAN PASCA RAMADIKA

115090300111017

Setelah dipertahankan di depan Majelis Penguji pada tanggal

.....

**dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Sains dalam bidang fisika**

Dosen Pembimbing I

Dosen Pembimbing II

Dr.rer.nat Abdurrouf, S.Si.,M.Si
NIP. 197209031994121001

Mauludi Ariesto Pamungkas,
S.Si.,M.Si.,Ph.D
NIP. 197304122000031013

Mengetahui,
Ketua Jurusan Fisika
Fakultas MIPA Universitas Brawijaya

Prof. Dr. rer. nat. M. Nurhuda
NIP. 196407021989031001

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : AWAN PASCA RAMADIKA

NIM : 115090300111017

Jurusan : FISIKA

Penulis Skripsi berjudul :

**SIMULASI DAN PERHITUNGAN ALIGNMEN MOLEKUL
SERTA PEMBANGKITAN HARMONIK TINGGI PADA
MOLEKUL N₂ DAN O₂**

Dengan ini menyatakan bahwa:

1. Isi dari Skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam Skripsi ini.
2. Apabila dikemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, Desember 2017

Yang Menyatakan,

**Awan Pasca Ramadika
NIM. 115090300111-017**

SIMULASI DAN PERHITUNGAN ALIGNMEN MOLEKUL SERTA PEMBANGKITAN HARMONIK TINGGI PADA MOLEKUL N₂ DAN O₂

ABSTRAK

Penelitian ini bertujuan untuk mensimulasikan alignmen molekul serta pembangkitan harmonik tinggi pada molekul N₂ dan O₂ dengan menggunakan program MATLAB. Simulasi pada penelitian ini adalah hasil modifikasi simulasi MATLAB yang dikembangkan oleh Sebastien Weber. Weber mengembangkan simulasi tentang alignmen molekul pada N₂ dan O₂. Dengan menggunakan ungkapan teoritis pembangkitan harmonik orde tinggi, didapatkan simulasi pembangkitan harmonik tinggi (*high harmonic generation*, HHG) pada molekul nitrogen dan oksigen. Teori *fast fourier transform* yang dapat mengubah sinyal HHG pada domain waktu ke domain frekuensi juga telah diterapkan dalam modifikasi simulasi ini. Simulasi HHG ini dapat digunakan untuk memprediksi bentuk sinyal maupun kapan terjadinya amplitudo HHG tertinggi dan terendah pada suatu molekul. Didapatkan juga kesimpulan bahwa temperatur sangat mempengaruhi besaran sinyal yang didapatkan. Semakin besar suhu awal akan memperkecil besaran amplitudo sinyal HHG, sebaliknya semakin kecil suhu awal besaran amplitudo sinyal HHG akan semakin besar.

Kata kunci: *alignmen molekul, simulasi MATLAB, pembangkitan harmonik tinggi, fast fourier transform, oksigen, nitrogen, temperatur.*

SIMULATION AND CALCULATION ON Molecular Alignment AND HIGH HARMONIC Generation OF MOLEKUL N₂ AND O₂

ABSTRACT

This study has the objective to simulate the molecular alignment and the high harmonics generations of molecules N₂ and O₂ using MATLAB program. The simulation of this research is the result of MATLAB modification simulation developed by Sebastien Weber. Weber developed a simulation of the molecular alignment at N₂ dan O₂. Using theoretical expression of high harmonic generation (HHG) developed a simulations HHG of nitrogen and oxygen molecules. Fast fourier transform theory which can convert HHG signals in time domain to frequency domain has been applied in the modification of this simulation. This simulation can be used to predict the shape of the signal and when the occurrence of the highest and lowest HHG amplitudes of a molecule. It was also found that the temperature greatly influenced the magnitude of the signal obtained. The larger initial temperature will decrease the amplitude of the HHG signal, otherwise the smaller of the initial temperature will increase the amplitude of the HHG signal.

Keywords: *molecular alignment, MATLAB simulation, high harmonic generation, fast fourier transform, oxygen, nitrogen, temperature.*

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah memberikan curahan ilmu, rahmat dan hidaya-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul **“SIMULASI DAN PERHITUNGAN ALIGNMEN MOLEKUL SERTA PEMBANGKITAN HARMONIK TINGGI PADA MOLEKUL N_2 DAN O_2 ”**. Selesaiannya laporan ini berkat adanya bantuan dan bimbingan dari berbagai pihak. Karena itu, penulis sampaikan penghargaan dan ucapan terimakasih kepada:

1. Kedua orang tua penulis, Bapak Eko Santoso dan Ibu Prinasih yang selalu mencurahkan segala cinta, kasih sayang, doa yang tidak pernah putus serta selalu sabar dalam membesarkan, mendidik, menasehati demi kelancaran dalam menuntut ilmu.
2. Prof. DR. rer. nat. M. Nurhuda. Selaku Ketua Jurusan Fisika Fakultas MIPA Universitas Brawijaya.
3. DR. rer. nat. Abdurrouf, S.Si., M.Si dan Mauludi Ariesto Pamungkas, S.Si.,M.Si.,Ph.D yang telah bersedia meluangkan waktunya dan selalu memberikan perhatian besar dalam membimbing penulis.
4. Seluruh dosen Jurusan Fisika Fakultas MIPA yang telah memberikan ilmu kepada penulis selama perkuliahan.
5. Kakak tercinta yang selalu memberi dukungan dan semangat serta doa.
6. Semua pihak yang telah membantu yang tidak dapat disebutkan satu persatu.

Semoga Allah SWT memberikan balasan yang berlimpah atas semua kebaikan yang telah diberikan kepada penulis. Kesempurnaan hanya milik Allah semata,dengan segala keterbatasan semoga laporan ini dapat bermanfaat.

Malang, 16 Desember 2017

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN SKRIPSI	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR.....	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB II TINJAUAN PUSTAKA.....	7
2.1 Alignmen Molekul.....	7
2.2 Pembangkitan Harmonik Tinggi.....	9
BAB III METODE PENELITIAN.....	17
3.1 Waktu dan Tempat Penelitian.....	17
3.2 Alat dan Bahan	17
3.3 Tahap Pengerjaan.....	17
BAB IV HASIL DAN PEMBAHASAN.....	21
4.1 Sinyal HHG pada Domain Waktu	21
4.2 Pengaruh Temperatur.....	25
4.3 Sinyal HHG pada Domain Frekuensi	28
4.4 Perbandingan Simulasi dan Eksperimen.....	31

BAB V PENUTUP.....	35
5.1 Kesimpulan.....	35
5.1 Saran.....	35
DAFTAR PUSTAKA	37
LAMPIRAN	.41

DAFTAR GAMBAR

Gambar 2.1	Alignmen dan Orientation.....	7
Gambar 2.2	Alignmen dan Orientation.....	8
Gambar 2.3	Proses pembangkitan harmonik tinggi.....	9
Gambar 2.4	HHG spectrum pada CO ₂	10
Gambar 2.5	Model tiga langkah.....	11
Gambar 2.6	Fungsi gelombang asimtotik N ₂ dan O ₂	14
Gambar 3.1	Tampilan progam awal.....	18
Gambar 3.2	Tampilan progam setelah modifikasi.....	19
Gambar 3.3	Diagram alir penelitian.....	20
Gambar 4.1	Grafik sinyal HHG N ₂ pada domain waktu.....	22
Gambar 4.2	Grafik sinyal HHG O ₂ pada domain waktu.....	23
Gambar 4.3	Grafik perbandingan sinyal HHG molekul N ₂ dan O ₂	24
Gambar 4.4	Grafik perbandingan pengaruh temperatur pada molekul N ₂	26
Gambar 4.5	Grafik perbandingan pengaruh temperatur pada molekul N ₂	26
Gambar 4.6	Grafik perbandingan pengaruh temperatur pada molekul O ₂	27
Gambar 4.7	Grafik perbandingan pengaruh temperatur pada molekul O ₂	28
Gambar 4.8	Grafik sinyal Fast Fourier Transform HHG	

	pada molekul N_2	29
Gambar 4.9	Grafik sinyal Fast Fourier Transform HHG	
	pada molekul O_2	30
Gambar 4.10	Grafik perbandingan simulasi dan	
	eksperimen pada molekul N_2	31
Gambar 4.11	Grafik perbandingan simulasi dan	
	eksperimen pada molekul O_2	32

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penelitian tentang *Molekular alignment* dengan menggunakan pulsa laser non-resonan berkembang pesat setelah diawali oleh Friedrich, Herschbach dan Seideman pada pertengahan tahun 90-an. Friedrich dan Herschbach mengembangkan model sederhana dari interaksi antara molekul dan bidang laser yang non-resonan. Model yang digunakan adalah polarisabilitas anisotropik dari (non-polar) molekul, untuk beberapa medan listrik dari bidang laser intens yang memberikan energi potensial. Energi potensial bergantung pada sudut serta dapat menciptakan torsi, yang membuat molekul berputar selaras dengan medan listrik laser. *Laser induced alignment* memiliki keuntungan yaitu dapat diterapkan untuk molekul apapun, karena metode ini merupakan suatu metode umum yang tidak memerlukan dipol permanen. Namun, metode ini hanya bisa menyelaraskan (*align*) tetapi tidak dapat mengarahkan (*orient*) molekul (Abdurrouf, 2008).

Alignmen molekul atau *Molecular Alignment* ditemukan pada tahun 1975, *molecular alignment* disebabkan oleh medan laser kuat yang non-resonan, bidang ini pertama kali dipelajari secara teoritis oleh Zon dan Katsnelson. Pada tahun yang sama (1975) alignmen dengan pembentukan rotasi *wavepacket* dilakukan pada molekul relatif berat (CS_2), dengan menggunakan pulsa 5 ps yang cukup singkat untuk membuat rotasi *wavepacket* sejajar dan teratur. *Aligned* and *oriented* molekul dapat juga diproduksi oleh tumbukan atom, menggunakan *brute force* orientasi dengan medan listrik yang kuat, atau dengan menggunakan radiasi optik inframerah. Merupakan hal penting untuk membedakan antara alignmen (*alignment*) dan orientasi (*orientation*), disebut alignmen apabila sumbu molekul istimewa selaras di beberapa sudut relatif terhadap sumbu ruang-tetap (misalnya garis medan listrik), sedangkan disebut orientasi apabila molekul menunjuk sepanjang sumbu keselarasan yang dikontrol. Derajat orientasi ini diukur dengan $\cos \theta$ (*orientation angle*), sementara derajat alignmen diukur dengan $\cos^2 \theta$ (*alignment degree*).

Sejak metode alignmen ditemukan, telah ada realisasi penelitian pada non-adiabatik alignment dari molekul ringan sampai molekul berat, seperti I_2 , N_2 , O_2 , CO_2 , dan D_2 . Pengamatan juga dilakukan untuk adiabatik *alignment naftalena* dan *turn-off alignment* CO_2 . Selain itu ada usulan teoritis dan realisasi eksperimental peningkatan keberpihakan dengan kombinasi kombinasi pulsa pendek dan panjang (Becker, 2005)

Berdasarkan pada durasi penggunaan pulsa laser, ada tiga jenis dari alignmen, yang dikenal sebagai adiabatik alignmen / keselarasan adiabatik, dinamis alignmen, dan *switched-off* alignment. Pada jenis pertama, bidang laser perlahan dihidupkan dan dimatikan sehingga molekul adiabatik dapat menyesuaikan diri dengan perubahan energi potensi dan berotasi. Kriteria perlahan dapat dipenuhi dengan menggunakan pulsa lebih lama dari periode rotasi, dimana keselarasan ini juga dikenal sebagai keselarasan pulsa panjang. Dengan pulsa panjang, masing-masing eigenstatis dari Hamiltonian bidang bebas dipastikan berubah menjadi adiabatik pada waktu interaksi, sehingga metode ini mendapatkan nama keselarasan adiabatik, keselarasan adiabatik dapat mencapai tingkat keselarasan yang tinggi tetapi hilang setelah pulsa dimatikan. Dinamis alignmen adalah keselarasan dengan menggunakan pulsa singkat yang dengan cepat diaktifkan dan dimatikan. Pada jenis keselarasan ini keselarasan akan tetap terjaga dengan kata lain tidak hilang bahkan setelah pulsa dimatikan. Sayangnya, metode ini tidak dapat mencapai gelar keselarasan setinggi jenis keselarasan adiabatik. Keselarasan ini juga dikenal sebagai keselarasan pulsa pendek(mengacu pada jenis pulsa yang diterapkan), serta disebut juga keselarasan non-adiabatik (mengacu pada interaksi non-adiabatik), serta *kick alignment* (karena adanya interaksi *impuls* oleh pulsa laser). Jenis ketiga adalah campuran antara metode pulsa panjang dan metode pulse singkat, yaitu pulsa secara perlahan dihidupkan namun cepat dimatikan. Hal ini mirip dengan pulsa panjang yang tiba-tiba dimatikan pada puncaknya. Metode terakhir mengambil semua manfaat dari dua kasus sebelumnya dan memberikan keselarasan dinamis dengan tingkat keselarasan yang tinggi, serta terjaganya keselarasan meski sinyal dimatikan, jenis ini

pertama kali ditemukan pada eksperimen molekul CO_2 dengan mengamati Optik Keer efek (OKE) sinyal (Levesque, 2006).

Pembangkitan harmonik tinggi (*High Harmonic Generation*, HHG) Pertama kali diamati pada akhir 80-an, HHG adalah proses dimana atom gas mulia didorong oleh medan laser *intens* oleh frekuensi ω sehingga menghasilkan radiasi dari frekuensi yang lebih tinggi $\Omega = n\omega$, dimana n adalah angka ganjil, dikenal sebagai urutan harmonik. Spektrum harmonik tinggi memiliki sejumlah fitur khas. Yaitu, sinyal harmonik secara eksponensial menurun untuk harmonik orde rendah (disebut wilayah *perturbative*), diikuti oleh dataran tinggi intensitas harmonis, di mana sinyal harmonik tetap konstan dengan meningkatnya orde harmonik, sebelum mencapai *cut-off*, ketika penurunan cepat eksponensial sinyal harmonik diamati. Menurut teori *perturbation*, orang akan beranggapan penurunan yang cepat dari sinyal harmonik dengan meningkatnya pesanan harmonik. Namun, pada intensitas tinggi daerah dataran tinggi yang diamati dan *cut-off* dari harmonik sulit untuk dipahami.

Proses penting dalam *Digital Signal Processing* (DSP) adalah menganalisis suatu sinyal input maupun output untuk mengetahui karakteristik sistem fisis tertentu. Proses analisis dan sintesis dalam domain waktu memerlukan analisis cukup panjang dengan melibatkan turunan dari fungsi, yang dapat menimbulkan ketidaktelitian hasil analisis. Analisis dan sintesis sinyal akan lebih mudah dilakukan pada domain frekuensi, karena besaran yang paling menentukan suatu sinyal adalah frekuensi. Oleh karena itu, untuk dapat bekerja pada domain frekuensi dibutuhkan suatu formulasi yang tepat sehingga proses manipulasi sinyal sesuai dengan kenyataan. Salah satu formulasi yang ampuh untuk proses pengolahan sinyal adalah menggunakan *Discrete Fourier Transform* (DFT). Prinsip DFT adalah mentransformasikan (alih bentuk) sinyal yang semula analog menjadi diskret dalam domain waktu, dan kemudian diubah ke dalam domain frekuensi. Hal ini dilakukan dengan mengalikan sinyal diskret dengan suatu fungsi kernel. Algoritma lain yang lebih cepat adalah *Fast Fourier Transform* (FFT). Prinsip kerja FFT adalah membagi sinyal hasil penyamplingan menjadi beberapa bagian yang kemudian masing-

masing bagian diselesaikan dengan algoritma yang sama dan hasilnya dikumpulkan kembali. Ada tiga kelas FFT yang umum digunakan di dalam suatu software DSP yaitu *Decimation in Time* (DIT), *Decimation in Frequency* (DIF) dan *Split Radix*. Ide ketiga jenis FFT tersebut adalah proses iterasi *sequence* data dilakukan secara berbeda dan memanfaatkan fungsi kernel yang memiliki sifat yang simetris pada suatu nilai tertentu dalam satu periode suatu sinyal. Jenis lain FFT yang sudah digunakan adalah paralel FFT dimana *sequence* data dikerjakan dengan menggunakan *parallel computing* sehingga proses transformasi akan lebih cepat.

Penelitian tentang pembangkitan harmonik tinggi pada molekul CO₂ pernah dilakukan oleh Meiyang Qin, digunakan 800 nm dan 100 fs pulsa alignmen dengan intensitas $1 \times 10^{13} \text{ W / cm}^2$. Suhu awal molekul 40 K. Panjang gelombang, durasi pulsa, dan intensitas pulsa *probe* yang digunakan untuk menghasilkan harmonik tingkat tinggi masing-masing 1450 nm, 18 fs, dan $1.7 \times 10^{13} \text{ W / cm}^2$. Dalam percobaan, pulsa IR ini dapat diproduksi oleh teknologi amplifikasi parametrik optik (*optical parametric amplification*, OPA) berdasarkan konversi panjang gelombang dari sumber laser Ti: sapphire. Didapatkan hasil percobaan yaitu *spectral minima* dengan jelas diamati dalam spektrum HHG pada setiap penundaan waktu. Selain itu, *spectral minima* yang bergantung pada keselarasan memiliki bentuk yang mirip dengan evolusi $\cos^2\theta$. Secara rinci, ketika nilai $\cos^2\theta$ meningkat (atau menurun), posisi minimum spektral secara bersamaan bergeser ke energi foton yang lebih rendah (atau lebih tinggi). Titik tertinggi dan titik terendah dari *spectral minima* muncul pada penundaan ketika $\cos^2\theta$ mencapai titik maksimum dan minimum. (Meiyang Qin, 2014).

Ling-ling Du pada tahun 2015 juga pernah mengamati HHG pada molekul H₂, digunakan 1450 nm dan 100 fs pulsa alignmen dengan intensitas $1 \times 10^{14} \text{ W / cm}^2$. Suhu awal molekul 40 K. Panjang gelombang, durasi pulsa, dan intensitas pulsa *probe* yang digunakan untuk menghasilkan harmonik tingkat tinggi masing-masing 1600 nm, 18 fs, dan $2,14 \times 10^{14} \text{ W / cm}^2$. Didapatkan hasil percobaan bahwa lebar dataran tinggi pertama bisa digunakan untuk

memperkirakan internuclear jarak H_2 . Bentuk gelombang yang didapatkan di percobaan ini menyerupai bentuk gelombang $\cos$, yang semakin naik puncak amplitudo nya pada $\frac{1}{2}$ revival pertama, dan akan mengalami penurunan amplitudo pada interval terakhir. (Lingling Du, 2015).

Pengukuran struktur elektronik dan dinamika untuk molekul membutuhkan pra-alignmen molekul target. Ini dapat dilakukan secara adiabatik atau non-adiabatik dengan menggunakan intensitas sedang pulsa laser . Umumnya, alignmen non-adiabatik lebih disukai karena dapat menghasilkan ensembel makroskopik molekul yang sangat selaras di bawah kondisi *field-free*. Dewasa ini didemonstrasikan bahwa struktur revival dari penyelarasan non-adiabatik molekul dapat diamati dengan lebih akurat melalui pembangkitan harmonik orde tinggi (HHG), sehingga dalam penelitian ini digunakan ungkapan teoritis HHG untuk molekul N_2 dan O_2 untuk mensimulasikan pembangkitan harmonik tinggi pada molekul N_2 dan O_2 , serta memanfaatkan *Fast Fourier Transform* untuk mengubah sinyal HHG pada domain waktu menjadi sinyal HHG pada domain frekuensi.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka diperoleh rumusan masalah adalah bagaimana sifat-sifat alignmen molekul dan pembangkitan harmonik tinggi pada molekul N_2 dan O_2 dengan menggunakan simulasi alignmen molekul.

1.3 Tujuan

Tujuan dari penelitian ini adalah untuk menganalisis ketergantungan HHG pada derajat alignmen dan kemungkinan pengembangan pada perhitungan sinyal HHG pada molekul N_2 dan O_2 .

1.4 Batasan Masalah

Dalam penyusunan penelitian tugas akhir ini, penulis membatasi pokok-pokok pembahasan antara lain : Simulasi yang dilakukan menggunakan program MATLAB, dan pokok bahasan

adalah molekul alignmen pada N_2 dan O_2 serta pembangkitan harmonik tingginya.

1.5 Manfaat

Penelitian ini diharapkan dapat memberikan informasi kepada pembaca tentang dinamika molekul N_2 dan O_2 ketika pembangkitan alignmen molekul terjadi, serta dapat melakukan perhitungan sinyal pembangkitan harmonik tinggi.

BAB II

TINJAUAN PUSTAKA

2.1 Alignmen Molekul

Alignmen molekul (*molecular alignment*) disebabkan oleh medan laser kuat yang non-resonan, pertama kali dipelajari secara teoritis oleh Zon dan Katsnelson. Merupakan hal penting untuk membedakan antara alignmen dan orientasi, disebut Alignmen apabila sumbu molekul istimewa selaras di beberapa sudut relatif terhadap sumbu ruang-tetap (misalnya garis medan listrik), sedangkan disebut orientasi apabila molekul menunjuk sepanjang sumbu keselarasan yang dikontrol. Derajat orientasi ini diukur dengan $\cos \theta$ (*orientation angle*), sementara derajat alignmen diukur dengan $\cos^2 \theta$ (*alignment degree*).



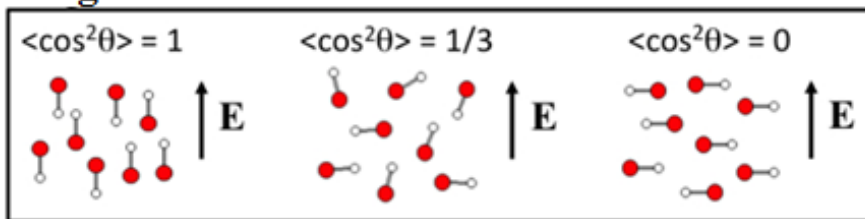
Gambar 2.1. Alignmen dan Orientasi

2.1.1 Pengertian alignmen molekul

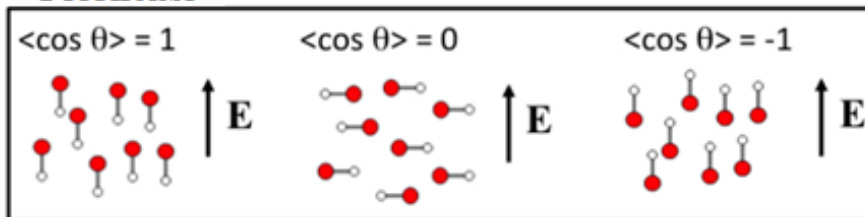
Penelitian tentang *molekular alignment* dengan menggunakan pulsa laser non resonan berkembang pesat setelah diawali oleh Friedrich, Herschbach dan Seideman pada pertengahan tahun 90-an. Freidrich dan Herschbach mengembangkan model sederhana dari interaksi antara molekul dan bidang laser yang non-resonan. Model yang digunakan adalah polarisabilitas anisotropik dari (non polar) molekul, untuk beberapa medan listrik dari bidang laser intensif yang memberikan energi potensial. Energi potensial bergantung pada sudut serta dapat menciptakan torsi, yang membuat molekul berputar selaras dengan medan listrik laser. *Laser induced*

alignment memiliki keuntungan yaitu dapat diterapkan untuk molekul apapun, karena metode ini merupakan suatu metode umum yang tidak memerlukan dipol permanen. Namun, metode ini hanya bisa menyelaraskan (*align*) tetapi tidak dapat mengarahkan (*orient*) molekul.

Alignmen



Orientasi



Gambar 2.2 Alignmen dan Orientasi

2.1.2 Jenis alignmen

Berdasarkan pada durasi penggunaan pulsa laser, ada tiga jenis dari alignmen, yang dikenal sebagai adiabatik alignmen, dinamis alignmen, dan *switched-off* alignmen.

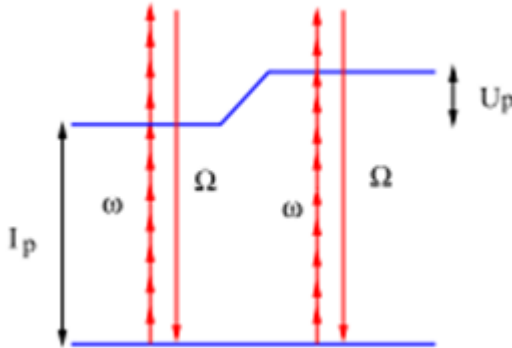
Pada jenis pertama, bidang laser perlahan dihidupkan dan dimatikan sehingga molekul adiabatik dapat menyesuaikan diri dengan perubahan energi potensi dan berotasi. Kriteria perlahan dapat dipenuhi dengan menggunakan pulsa lebih lama dari periode rotasi, dimana keselarasan ini juga dikenal sebagai keselarasan pulsa panjang. Dengan pulsa panjang, masing-masing eigenstates dari Hamiltonian bidang bebas dijamin berubah menjadi adiabatik pada waktu interaksi, sehingga metode ini mendapatkan nama keselarasan adiabatik, keselarasan adiabatik dapat mencapai tingkat keselarasan yang tinggi tetapi hilang setelah pulsa dimatikan. Dinamis alignmen adalah keselarasan dengan menggunakan pulsa singkat yang dengan cepat diaktifkan dan dimatikan. Pada jenis keselarasan ini

keselarasan akan tetap terjaga dengan kata lain tidak hilang bahkan setelah pulsa dimatikan. Sayangnya, metode ini tidak dapat mencapai gelar keselarasan setinggi jenis keselarasan adiabatik. Keselarasan ini juga dikenal sebagai keselarasan pulsa pendek (mengacu pada jenis pulsa yang diterapkan), serta disebut juga keselarasan non-adiabatik (mengacu pada interaksi non-adiabatik), serta *kick alignment* (karena adanya interaksi impuls oleh pulsa laser). Jenis ketiga adalah campuran antara metode pulsa panjang dan metode pulse singkat, yaitu pulsa secara perlahan dihidupkan namun cepat dimatikan. Hal ini mirip dengan pulsa panjang yang tiba-tiba dimatikan pada puncaknya. Kasus terakhir mengambil semua manfaat dari dua kasus sebelumnya dan memberikan keselarasan dinamis dengan tingkat keselarasan yang tinggi, serta terjaganya keselarasan meski sinyal dimatikan, jenis ini pertama kali ditemukan pada eksperimen molekul CO_2 dengan mengamati Optik Keer efek (OKE) sinyal (Becker, 2005).

2.2 Pembangkitan Harmonik Tinggi

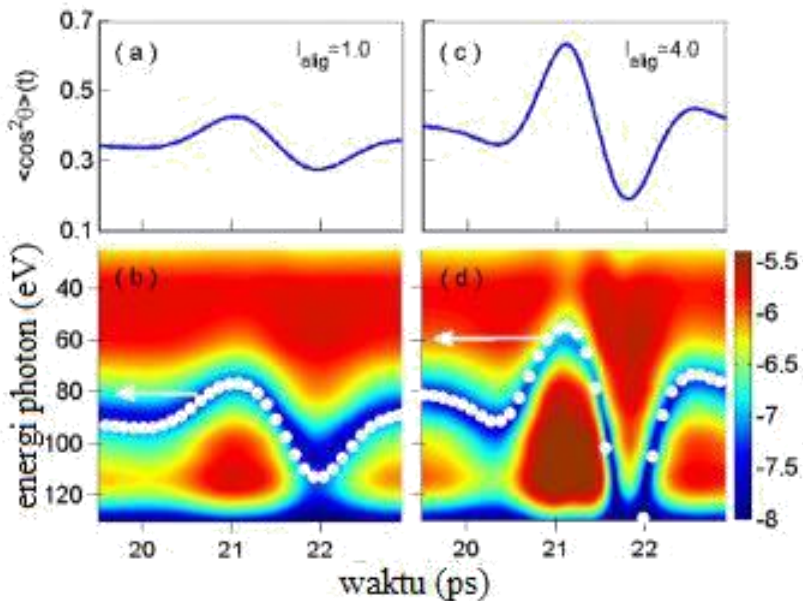
2.2.1 Pengertian pembangkitan harmonik tinggi

Pembangkitan Harmonik Tinggi (HHG) Pertama kali diamati pada akhir 80-an, HHG merupakan proses di mana atom gas mulia didorong oleh medan laser yang memiliki frekuensi yang *intens* sehingga menghasilkan radiasi dengan frekuensi yang lebih tinggi $\Omega = n\omega$ di mana n adalah angka ganjil, dan n merupakan urutan harmonik atau orde. Spektrum harmonik yang tinggi (hasil harmonik sebagai fungsi dari orde) memiliki sejumlah fitur karakteristik. Artinya, sinyal harmonik secara eksponensial menurun untuk orde harmonik rendah (yang disebut daerah *perturbatif*), diikuti oleh dataran tinggi harmonik, ketika sinyal harmonik tetap konstan meski ada peningkatan urutan harmonik, sebelum mencapai *cut-off*, dimana penurunan eksponensial yang cepat dari sinyal harmonik ini diamati. Menurut teori perturbasi, peneliti akan berhipotesa bahwa penurunan pesat pada sinyal harmonik dengan meningkatkan orde harmonik. Namun, pada intensitas tinggi, daerah dataran tinggi yang diamati dan *cut-off* harmonik tidaklah mudah dipahami.



Gambar 2.3 Proses HHG, n foton dengan frekuensi ω menghasilkan radiasi dengan frekuensi $\Omega = n\omega$ (Miyazaki, 2005).

Spektrum HHG yang bergantung pada alignmen yang dihasilkan dari molekul CO_2 telah disimulasikan.



Gambar 2.4 Spektrum HHG pada molekul CO_2

Dalam Gambar 2.4 (a) evolusi dari parameter alignmen $\cos^2\theta$ disekitar pembangkitan setengah sinyal pertama (panel a) dan

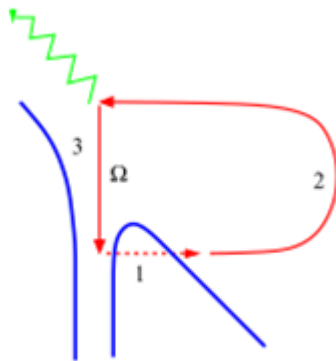
pemetaan yang sesuai dari spektrum HHG sebagai fungsi dari domain waktu dan energi foton (2.4 b). Untuk pemetaan spektrum HHG (panel b), sumbu horizontal mewakili penundaan waktu (ps) sehubungan dengan pulsa alignmen, dan sumbu vertikal mewakili energi foton dari harmonik orde tinggi. Digunakan 800 nm dan 100 fs pulsa alignmen dengan intensitas $1 \times 10^{13} \text{ W / cm}^2$. Suhu awal molekul diatur menjadi 40 K. Panjang gelombang, durasi pulsa, dan intensitas pulsa *probe* yang digunakan untuk menghasilkan harmonik tingkat tinggi masing-masing 1450 nm, 18 fs, dan $1.7 \times 10^{13} \text{ W / cm}^2$. Dalam percobaan, pulsa IR ini dapat diproduksi oleh teknologi amplifikasi parametrik optik (*optical parametric amplification*, OPA) berdasarkan konversi panjang gelombang dari sumber laser Ti:sapphire. Seperti ditunjukkan pada Gambar. 2.4 (b), *spectral minima* dengan jelas diamati dalam spektrum HHG pada setiap penundaan waktu. Selain itu, *spectral minima* yang bergantung pada keselarasan memiliki bentuk yang mirip dengan evolusi $\cos^2\theta$. Secara rinci, ketika nilai $\cos^2\theta$ meningkat (atau menurun), posisi minimum spektral secara bersamaan bergeser ke energi foton yang lebih rendah (atau lebih tinggi). Titik tertinggi dan titik terendah dari *spectral minima* muncul pada penundaan ketika $\cos^2\theta$ mencapai titik maksimum dan minimum. (Meiyan Qin, 2014).

2.2.2 Ketergantungan HHG terhadap derajat alignmen

Pemecahan teoritis pertama dalam pemahaman tentang dataran tinggi HHG muncul pada tahun 1992 ketika Krause dkk menunjukkannya, dengan memecahkan *time-dependent Schrodinger equation* (TDSE) secara numerik, bahwa posisi *cut-off* dalam spektrum harmonik diberikan oleh hukum universal, yaitu $n_{cut-off} = (I_p + 3U_p) / \hbar\omega$, di mana I_p adalah potensi ionisasi dan $U_p = \frac{e^2 E^2}{4m\omega^2}$ (di mana e adalah muatan listrik dan m adalah massa elektron) atau $U_p \text{ (eV)} = 9.33 \times 10^{-14} I (\text{W / cm}^2) \lambda^2 \text{ (}\mu\text{m)}$ merupakan apa yang disebut energi *ponderomotive*, yang merupakan energi kinetik siklus-rata dari elektron di medan listrik laser dengan medan listrik puncak E . Ini membuktikan energi interaksi elektron-laser juga meningkatkan potensi ionisasi I_p oleh U_p . Pada tahun 1993 Corkum serta Kulander dkk, menerbitkan teori *classic/quasiclassical*, yang direproduksi oleh daerah dataran tinggi yang diamati dalam percobaan. Menurut penelitian mereka, HHG

dapat dipecah menjadi proses tiga langkah, yaitu : ionisasi, propagasi, dan rekombinasi. Oleh karena itu, model tersebut dinamai model tiga langkah atau juga biasa disebut “*model simple-man*” karena kesederhanaan klasiknya yang mencolok. Selain itu, mereka menemukan classically juga mengalami cut-off dalam spektrum harmonik yang terjadi pada $n_{cut-off} = (I_p + 3U_p) / \hbar\omega$. Tak lama setelah model tiga langkah diperkenalkan, model kuantum diusulkan oleh Lewenstein dkk yang menegaskan kebenaran pendekatan *classic*.

Model tiga langkah adalah *quasiclassical*, yang berarti bahwa aspek kuantum, seperti tunneling dan ionisasi dimasukkan ke dalam model tetapi gerakan elektron di medan laser penggerak diperlakukan secara klasik.



Gambar 2.5. Deskripsi proses HHG dalam kerangka model tiga langkah, yaitu (1) ionisasi, (2) propagasi, dan (3) rekombinasi.

Model tiga langkah ditunjukkan pada Gambar 2.5 Elektron lahir ke dalam kontinum melalui ionisasi terowongan dengan nol energi kinetik. (2) Elektron diperlakukan sebagai partikel klasik bebas, yang dipercepat oleh medan laser penggerak sehingga mengumpulkan energi kinetik. (3) Emisi harmonik tinggi terjadi karena elektron tertarik kembali untuk bergabung kembali dengan inti dan memancarkan foton tunggal dengan frekuensi $\Omega = n\omega$. Proses tiga langkah ini mengulang setiap setengah siklus dari medan laser penggerak. Periodisitas ini dalam domain waktu menghasilkan

domain frekuensi sebagai pemisahan oleh frekuensi dua kali dari medan laser penggerak, 2ω . Frekuensi fundamental muncul dalam spektrum sebagai emisi orde pertama ($n = 1$), dan pemisahan oleh 2ω menyiratkan bahwa sinyal HHG muncul hanya untuk harmonik orde ganjil.

Menurut prosedur *semi-classic*, amplitudo untuk emisi harmonik dari keadaan murni sebanding dengan modulus kuadrat Transformasi Fourier momen dipol. Sinyal dinamis untuk emisi HHG untuk frekuensi yang diberikan $\Omega = n\omega$ dan waktu t_d , oleh karena itu diberikan dari kuadrat mutlak transformasi Fourier (FT) dari nilai harapan dipol total $D(t; t_d)$. Dalam kasus atom, Untuk campuran termal ensembel molekul, sinyal dinamis untuk frekuensi yang diberikan $\Omega = n\omega$ dan jeda waktu t_d dapat diperoleh dari rata-rata sinyal, melalui distribusi dari eigenstatis perputaran murni yang tidak terganggu sebelumnya $|J_0 M_0\rangle$ pada suhu awal T :

$$S^{(n)}(t_d) = \sum_{J_0 M_0} \rho(J_0) |[F.T.(D(t, t_d))](n\omega)|^2 \quad (2.1)$$

dengan $\rho(J_0) = \frac{1}{Z_p} e^{-E_{J_0}/kT}$ di mana k adalah konstanta Boltzmann, T adalah suhu, E_{J_0} adalah energi rotasi dari eigenstatis $|J_0 M_0\rangle$ dan $Z_p = \sum_{J_0} (2J_0 + 1) e^{-E_{J_0}/kT}$ adalah fungsi partisi rotasi. $D(t, t_d)$ terdiri dari fungsi gelombang rotasi nuklir $\phi_{J_0 M_0}(t_d)$ dan bagian elektronik $D_e(t; t_d)$. Karena kebebasan fungsi gelombang rotasi nuklir $\phi_{J_0 M_0}(t_d)$ pada t , F.T. dari $D(t, t_d)$ sekarang dapat dinyatakan sebagai

$$[F.T.(D(t, t_d))](n) = [\phi_{J_0 M_0}(t_d, \theta) |[F.T.(D_e(t; t_d))](n\omega) | \phi_{J_0 M_0}(t_d, \theta)] + c. \quad (2.2)$$

Dapat diturunkan secara eksplisit ketergantungan fungsi gelombang nuklir sesaat $\phi_{J_0 M_0}(t_d, \theta)$ pada alignmen sudut θ . Sehingga dapat diperoleh operator baru

$$T_e^{(n)}(\theta) = [F.T.(D_e(t; t_d))](n\omega) \quad (2.3)$$

$T_e^{(n)}(\theta)$ dapat dilihat sebagai operator transisi HHG yang elemen matriksnya berhubungan dengan paket gelombang rotasi

mempengaruhi sinyal. Di persamaan di atas, $T_e^{(n)}(\theta)$ bergantung pada

θ melalui *ground states* elektronik ϕ_e^0 muncul di $D_e(t; t_d)$

Akhirnya, sinyal dinamis pada waktu t_d dapat diperoleh dengan mengganti Persamaan. (b) dalam Persamaan. (a)

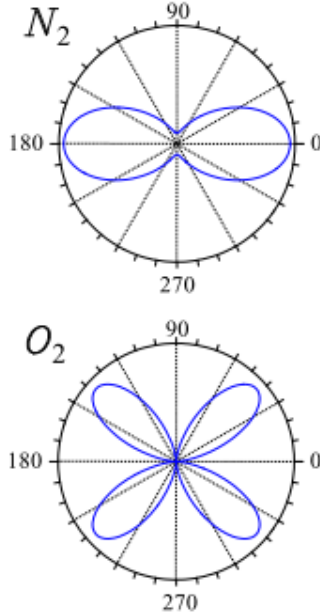
$$S^{(n)}(t_d) = \sum_{J_0 M_0} \rho(J_0) |[\phi_{J_0 M_0}(t_d, \theta) | T_e^{(n)}(\theta) | \phi_{J_0 M_0}(t_d, \theta)]|^2 \quad (2.4)$$

2.2.3 Ungkapan teoritis HHG untuk N_2 dan O_2

Operator HHG $T_e^{(n)}(\theta)$ untuk N_2 dapat diturunkan dari jumlah pangkat dari $\cos^2 \theta$ saja

$$S^{(n)}(t_d) = \sum_{j=0}^3 \sum_{j' \geq j} c_{jj'}^{(n)} \langle \cos^{2j} \theta \rangle \langle \cos^{2j'} \theta \rangle (t_d) \quad (2.5)$$

Fungsi gelombang asimtotik terhitung untuk N_2 dan O_2 dapat dilihat pada gambar berikut.



Gambar 2.6 Gelombang asimtotik N_2 dan O_2

Dan operator O_2 dapat diturunkan sebagai berikut

$$S^{(n)}(t_d) = \sum_{j=1}^3 \sum_{j' \geq j}^3 c_{j,j'}^{(n)} \langle \langle \sin^2 \theta \cos^{2j} \theta \rangle \langle \sin^2 \theta \cos^{2j'} \theta \rangle \rangle (t_d) \quad (2.6)$$

Mengekspresikan operator HHG dari O_2 (dan molekul-molekul lain) dalam $\cos \theta$ kuadrat memberikan representasi yang seragam untuk operator HHG dari molekul yang selaras. Namun, mengekspresikan operator HHG O_2 dalam kekuatan $\sin \theta \cos \theta$ memberikan wawasan fisik yang lebih banyak. (Abdurrouf, 2008).

[Halaman ini sengaja dikosongkan]

BAB III METODOLOGI

3.1 Waktu dan Tempat Pelaksanaan

Penelitian ini dilaksanakan mulai tanggal 30 Juli 2016 sampai 23 Desember 2016 di Laboratorium Simulasi dan Pemodelan Jurusan Fisika Fakultas MIPA Universitas Brawijaya

3.2 Alat dan Bahan Penelitian

Alat yang digunakan adalah seperangkat komputer dengan spesifikasi sebagai berikut :

Processor	: Intel(R) Core (TM) i7CPU @ 2.6 GHz
Memori RAM	: 4 GB
OS	: Windows 7 Professional 32 Bit

Sedangkan Software yang digunakan adalah Ms. Office dan Matlab r2012b.

3.3 Tahap Pengerjaan

3.3.1 Pendahuluan

Ada beberapa tahap identifikasi dalam penelitian ini, yaitu :

- a. Latar belakang yang mendasari adanya permasalahan dalam penelitian ini yang terkait molekular alignmen dan pembangkitan harmonik tinggi
- b. Perumusan masalah yang sesuai dengan latar belakang
- c. Tujuan berdasarkan perumusan masalah yang ada
- d. Batasan masalah dalam penelitian untuk memfokuskan pembahasan
- e. Manfaat penelitian untuk umum dan instansi khususnya.

3.3.1 Pengumpulan data

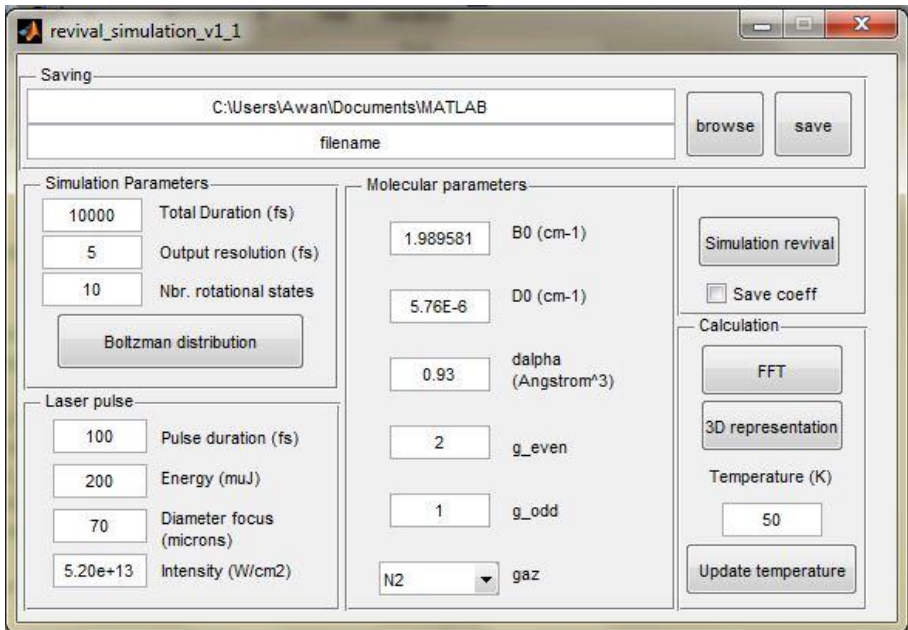
Data yang digunakan dalam penelitian adalah hasil simulasi program Matlab.

3.3.2 Studi literatur

Literatur yang terkait dalam penelitian ini yaitu alignmen molekul, pembangkitan harmonik tinggi, karakteristik molekul Oksigen dan Nitrogen, serta modul program MATLAB

3.3.3 Program MATLAB

Program yang digunakan adalah program matlab, dengan simulasi yang dikembangkan oleh Sebastien Weber kemudian dimodifikasi agar bisa melakukan perhitungan HHG.

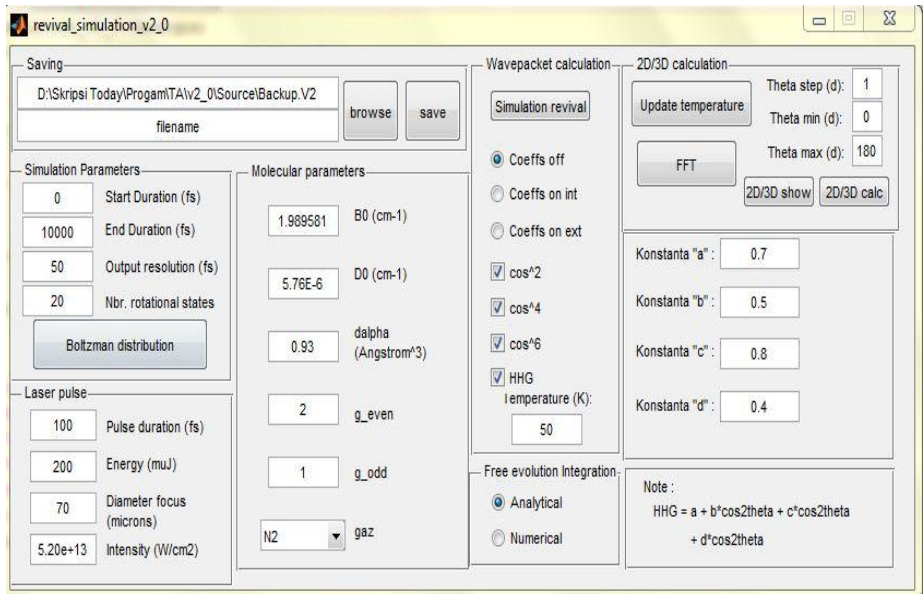


Gambar 3.1 Tampilan program awal

Program awal ini adalah program yang dikembangkan oleh Dr Sébastien Weber pada tahun 2011, beliau membuat simulasi tentang gas N_2 dan O_2 tentang molecular alignment pada $\cos^2\theta$.

Dengan melakukan modifikasi terhadap program tersebut dengan menurunkan $\cos^2\theta$ ke $\cos^4\theta$ dan $\cos^6\theta$ serta menggunakan persamaan tentang *High Harmonic Generation* penulis berharap agar bisa mendapatkan simulasi tentang Pembangkitan Harmonic Tinggi

sehingga dapat membandingkan perhitungan simulasi tersebut kedalam perhitungan eksperimen.



Gambar 3.2 Tampilan program awal

3.3.4 Pengolahan data

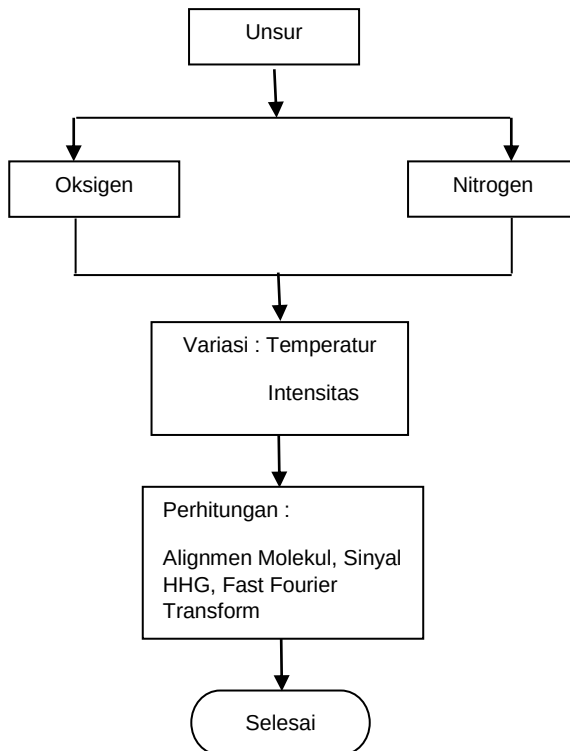
- a. Tahapan analisis data Molekular Alignment
 1. Melakukan running program molekular alignment pada N₂
 2. Melakukan running program molekular alignment pada O₂
- b. Tahapan analisis data High Harmonic Generation
 1. Melakukan running program HHG pada N₂
 2. Melakukan running program HHG pada O₂
 3. Mengubah temperatur
 4. Mencatat semua data yang diperoleh.

3.3.5 Membandingkan hasil simulasi dengan eksperimen

Hasil simulasi dari program dibandingkan dengan hasil eksperimen, dengan tujuan untuk mengetahui keakuratan metode simulasi pada MATLAB.

3.3.6 Analisa hasil

Diagram alir penelitian ditunjukkan pada Gambar 3.1



Gambar 3.3 Diagram alir penelitian

BAB IV

ANALISA DAN PEMBAHASAN

Penelitian ini dilaksanakan di Laboratorium Simulasi dan Pemodelan Jurusan Fisika Fakultas MIPA Universitas Brawijaya. Tahap awal pada penelitian ini adalah dilakukan modifikasi terhadap program simulasi matlab dari Sebastien Weber, yang mengembangkan simulasi sinyal $\cos^2 \theta$ pada alignmen molekul nitrogen dan oksigen, kemudian dengan ungkapan teoritis yang ada, program dikembangkan untuk mendapatkan $\cos^4 \theta$ dan $\cos^6 \theta$ agar bisa diperoleh pembangkitan harmonik tinggi, yang merupakan penelitian yang tergolong masih baru.

Hasil perhitungan pada kedua molekul (nitrogen dan oksigen) dibahas dalam dua bagian. Bagian pertama mendiskusikan mengenai karakteristik sinyal HHG pada domain waktu, termasuk didalamnya membahas pengaruh temperatur pada kekuatan sinyal HHG. Bagian kedua mendiskusikan mengenai karakteristik sinyal HHG pada domain frekuensi. Pada setiap bagian, perbandingan dengan data eksperimen yang tersedia diberikan untuk melihat kesesuaian.

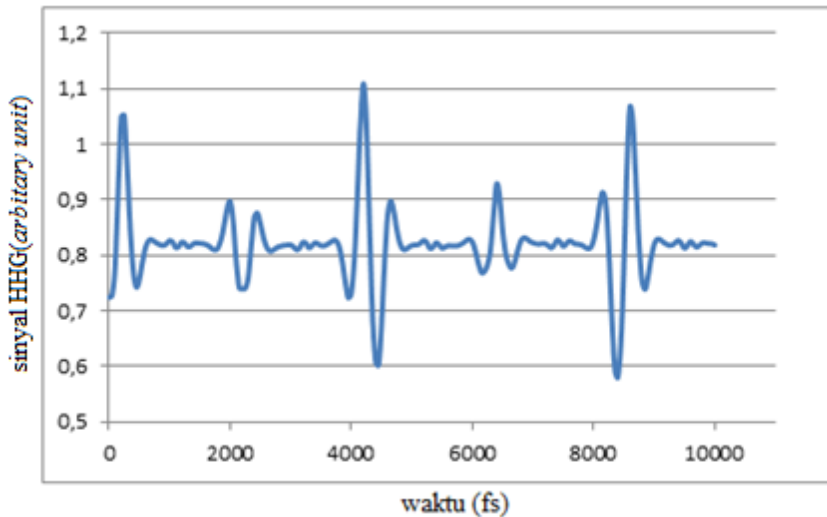
4.1 Sinyal HHG N₂ dan O₂ pada Domain Waktu

Pembangkitan harmonik tinggi atau yang sering disebut HHG, merupakan proses di mana atom gas mulia didorong oleh medan laser yang memiliki frekuensi yang intens sehingga menghasilkan radiasi dengan frekuensi yang lebih tinggi. Sinyal HHG dapat diamati dengan dua cara yaitu mengamati menurut domain waktu dan menurut domain frekuensi.

Sinyal HHG pada Nitrogen dan Oksigen menurut domain waktu memiliki perbedaan yang cukup signifikan, keduanya memiliki amplitudo yang relatif sama, namun sinyal tertinggi antara molekul Nitrogen dan Oksigen terjadi pada waktu yang berbeda.

4.1.1 Sinyal HHG pada N₂

Pada Molekul N₂ memiliki karakteristik B0 1,989581 cm⁻¹, D0 5,76 x 10⁻⁶ cm⁻¹, alpha 0,93 Å³, g_{even} = 2 dan g_{odd} = 1.

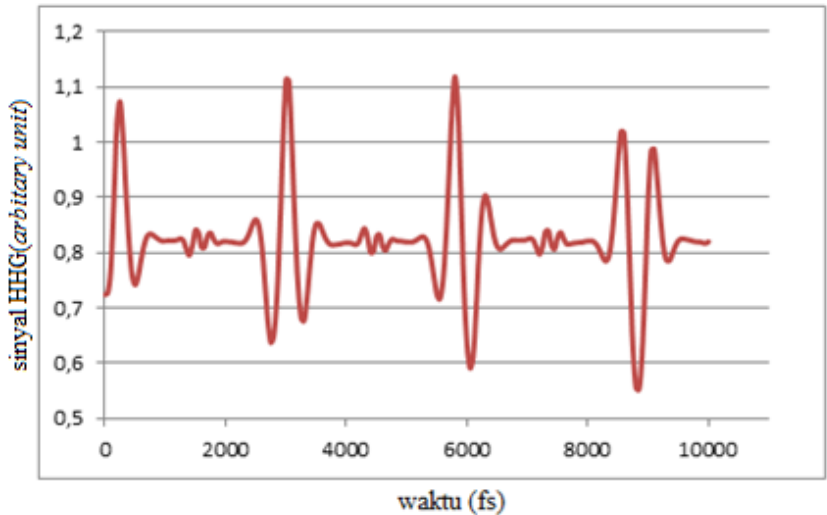


Gambar 4.1 Grafik sinyal HHG N₂ (orde n=19) pada domain waktu

Hasil simulasi sinyal HHG N₂ pada domain waktu diperlihatkan pada Gambar 4.1. digunakan waktu selama 10000 femtosecond (fs) dan temperatur 50 Kelvin, grafik HHG N₂ mendapatkan kekuatan sinyal tertinggi pada waktu 4200 fs yaitu didapatkan sinyal HHG dengan amplitudo sebesar 1,1088, sedangkan sinyal terendah didapatkan pada waktu 8400 fs yaitu 0,5798. Dari hasil simulasi tersebut juga didapatkan kesimpulan bahwa sinyal HHG yang didapatkan dari 10000 fs, dengan step sebesar 50 fs memiliki rata-rata nilai sebesar 0,820723. Sinyal tinggi didapatkan 3 kali yaitu pada waktu 250 fs, 4200 fs, serta 8600 fs. Sedangkan sinyal rendah didapatkan 2 kali yaitu pada waktu 4400 fs dan 8400 fs, dapat disimpulkan bahwa sinyal rendah pertama didapatkan yaitu 200 fs setelah sinyal tinggi yang kedua, dan sinyal rendah kedua didapatkan 200 fs sebelum sinyal tinggi yang ketiga.

4.1.2 Sinyal HHG pada O₂

Pada molekul O₂ memiliki karakteristik B0 1,43768 cm⁻¹ , D0 4,839 x 10⁻⁶ cm⁻¹, dalphi 1,14 Å³, g_{even} = 0 dan g_{odd} = 1.

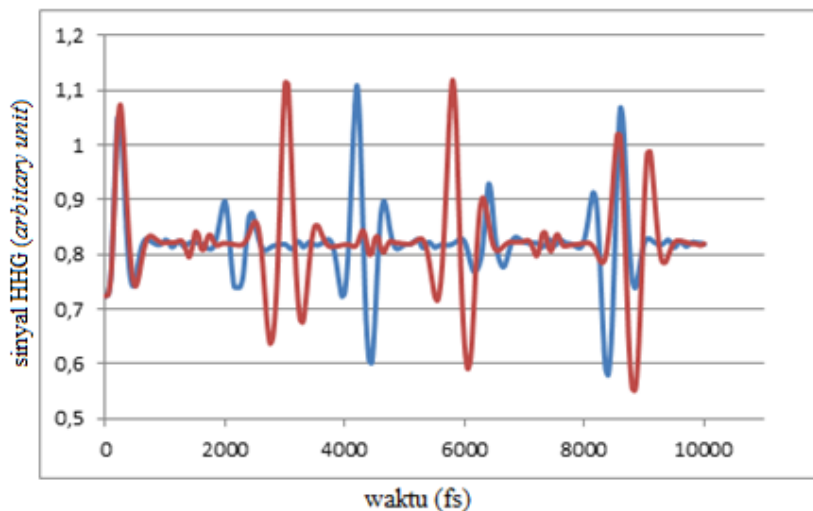


Gambar 4.2 Grafik sinyal HHG O_2 (orde $n=19$) pada domain waktu

Hasil simulasi sinyal HHG O_2 pada domain waktu diperlihatkan pada Gambar 4.2. digunakan waktu selama 10000 femtosecond (fs) dan temperatur 50 Kelvin, grafik HHG O_2 mendapatkan kekuatan sinyal tertinggi pada waktu 5800 fs yaitu didapatkan amplitudo sebesar 1,1186, sedangkan sinyal terendah didapatkan pada waktu 8850 fs yaitu 0,5525. Dari hasil simulasi tersebut juga didapatkan kesimpulan bahwa sinyal HHG yang didapatkan dari 10000 fs, dengan step sebesar 50 fs memiliki rata-rata nilai sebesar 0,822174. Sinyal tinggi didapatkan 4 kali yaitu pada waktu 250 fs, 3000 fs, 5800 fs serta 8600 fs. Sedangkan sinyal rendah didapatkan 2 kali yaitu pada waktu 6050 fs dan 8850 fs, dapat disimpulkan bahwa sinyal rendah pertama didapatkan yaitu 250 fs setelah sinyal tinggi yang kedua, dan sinyal rendah kedua didapatkan 250 fs sebelum sinyal tinggi yang keempat. Sebelum dan sesudah sinyal tinggi kedua juga didapatkan sinyal rendah namun masih diatas nilai 0,6.

4.1.3 Analisis perbedaan sinyal HHG pada N_2 dan O_2

Hasil Simulasi HHG pada N_2 dan O_2 didapatkan hasil perbedaan yang cukup signifikan, perbedaan yang mendasar adalah waktu dimana sinyal HHG tertinggi dan waktu dimana sinyal HHG terendah didapatkan, namun untuk besaran nilai sinyal pada kedua molekul tersebut didapatkan nilai tertinggi dan terendah di titik yang relatif sama.



Gambar 4.3 Grafik sinyal HHG, molekul N_2 (garis biru) dan O_2 (garis merah) pada suhu 50 K dan orde $n=19$

Hasil simulasi sinyal HHG N_2 dan O_2 diperlihatkan pada Gambar 4.3. dapat disimpulkan bahwa perbedaan mencolok antara sinyal HHG N_2 dan O_2 adalah waktu ketika sinyal HHG mencapai titik puncak dan titik terendah, namun titik puncak yang didapatkan dari kedua molekul relatif sama yaitu berada diatas 1,1, demikian halnya dengan titik terendah, namun demikian HHG pada O_2 memiliki titik puncak yang lebih besar, dan memiliki titik rendah yang lebih kecil dibandingkan HHG N_2 . Molekul O_2 juga memiliki rata-rata sinyal HHG yang relatif lebih tinggi dibandingkan N_2 , yaitu sebesar 0,822174 untuk O_2 dan 0,820723 untuk N_2 , dapat

disimpulkan bahwa nilai rata-rata dari N_2 dan O_2 hanya memiliki perbedaan yang sangat kecil.

4.2 Pengaruh Temperatur pada Sinyal HHG

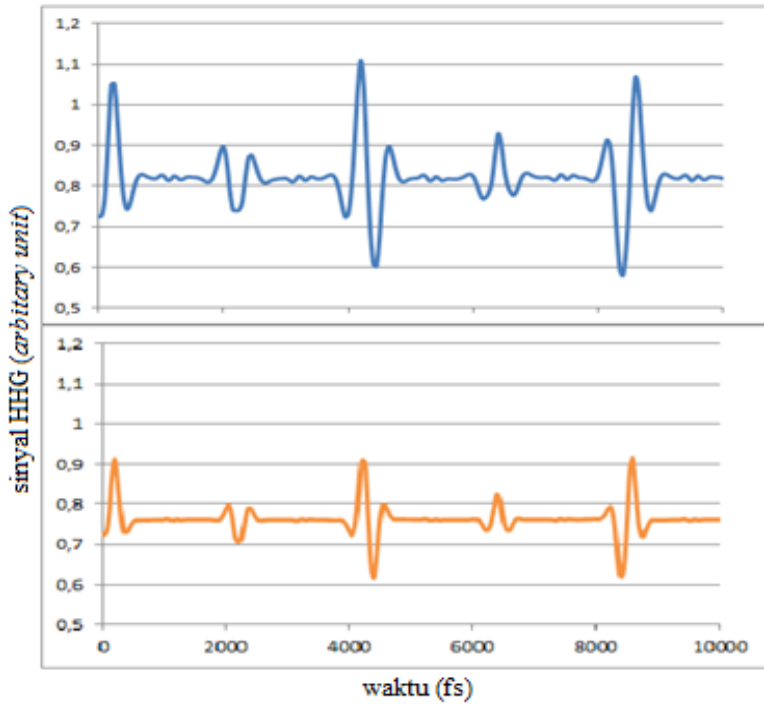
Nilai amplitudo sinyal pembangkitan harmonik tinggi pada molekul Nitrogen dan Oksigen pada suhu 50 Kelvin memiliki titik puncak yang hampir sama yaitu berada diatas 1,1, sinyal HHG memiliki ketergantungan pada suhu awal molekul, seperti yang ditunjukkan pada Gambar. 4.4 dan 4.5. Hal ini dapat dilihat dari angka-angka yang didapatkan bahwa suhu awal molekul yang lebih rendah menghasilkan sinyal HHG dengan amplitudo yang lebih besar.

4.2.1 Pengaruh temperatur pada sinyal HHG molekul N_2

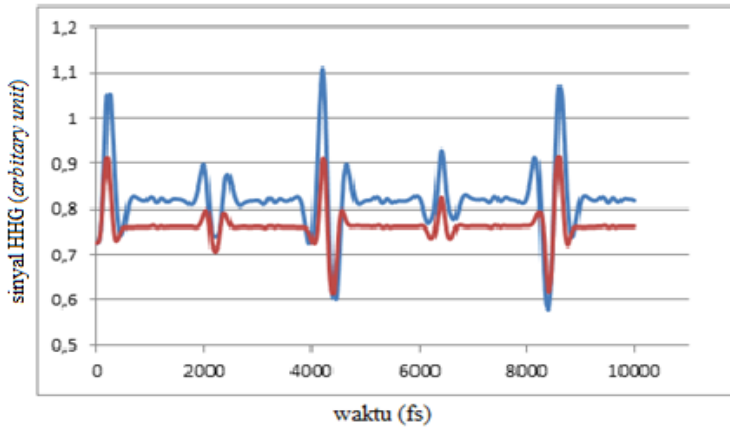
Temperatur memiliki pengaruh penting didalam pembangkitan harmonik tinggi, semakin tinggi suhu awal molekul sinyal harmonik yang dihasilkan akan memiliki amplitudo yang lebih kecil.

Hasil simulasi sinyal HHG molekul N_2 pada suhu 50 K dan 200 K , didapatkan kesimpulan bahwa nilai amplitudo mengalami penurunan ketika suhu dinaikan hal ini terlihat pada Gambar 4.5, untuk garis merah (suhu 200 K) hanya memiliki amplitudo terbesar pada nilai 0,9 θ , sedangkan untuk suhu 50 K berada pada nilai 1,1 θ , demikian halnya untuk nilai terendah juga mengalami penyusutan amplitudo yaitu nilai terendah pada suhu 200 K hanya berkisar pada nilai 0,62 sedangkan untuk suhu 50 K memiliki nilai amplitudo lebih kecil yaitu 5,8.

Dari perhitungan matematis didapatkan nilai rata-rata amplitudo yang memiliki perbedaan cukup besar antara suhu awal 50 Kelvin dengan 200 Kelvin, untuk 50 K didapatkan nilai rata-rata amplitudo 0,820723, sedangkan untuk suhu awal 200 K didapatkan nilai rata-rata amplitudo 0,762223, dapat disimpulkan bahwa semakin tinggi suhu awal molekul N_2 , akan mengakibatkan nilai amplitudo sinyal HHG semakin kecil.



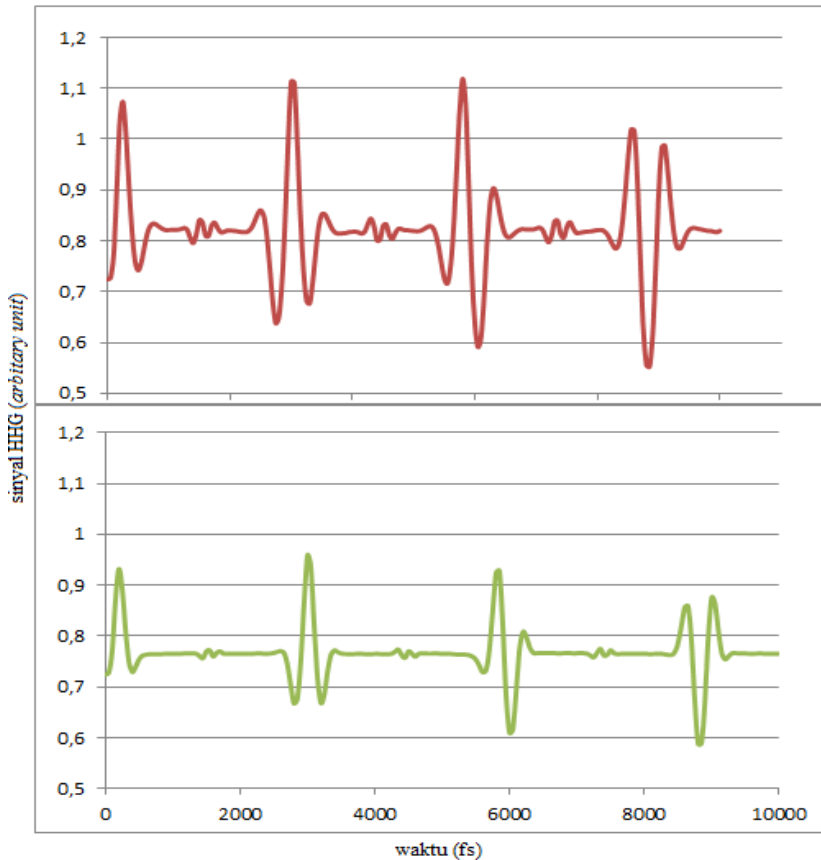
Gambar 4.4 Grafik sinyal HHG, molekul N_2 (orde $n=19$) pada suhu 50 K (garis biru) dan molekul N_2 pada suhu 200 K (garis oranye)



Gambar 4.5 Grafik sinyal HHG, molekul N_2 pada suhu 50 K (garis biru) dan molekul N_2 pada suhu 200 K (garis merah).

4.2.1 Pengaruh temperatur pada sinyal HHG molekul O_2

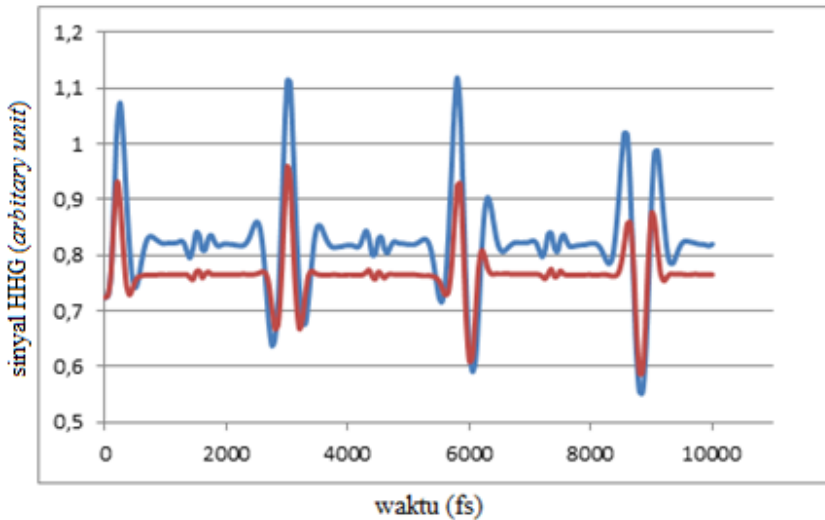
Pengaruh temperatur molekul oksigen memiliki karakteristik yang sama dengan molekul Nitrogen, yaitu besar temperatur memiliki pengaruh berlawanan dengan besar amplitudo, semakin besar temperatur, amplitudo yang dihasilkan akan semakin kecil.



Gambar 4.6 Grafik sinyal HHG, molekul O_2 (orde $n=19$) pada suhu 50 K (garis merah) dan molekul O_2 pada suhu 200 K (garis hijau).

Hasil simulasi sinyal HHG pada molekul O_2 pada suhu 50 K dan 200 K didapatkan kesimpulan yang hampir sama dengan molekul N_2 yaitu mengalami penurunan nilai amplitudo yang cukup

besar ketika suhu awal molekul dinaikkan. Dari perhitungan matematis didapatkan nilai rata-rata amplitudo yang memiliki perbedaan cukup besar antara suhu awal 50 Kelvin dengan 200 Kelvin, untuk 50 K didapatkan nilai rata-rata sinyal HHG sebesar 0,822174, sedangkan untuk suhu awal 200 K didapatkan nilai rata-rata sinyal HHG sebesar 0,766494, dapat disimpulkan bahwa semakin tinggi suhu awal molekul O_2 , akan mengakibatkan nilai amplitudo sinyal HHG semakin kecil, seperti halnya pada molekul N_2 .



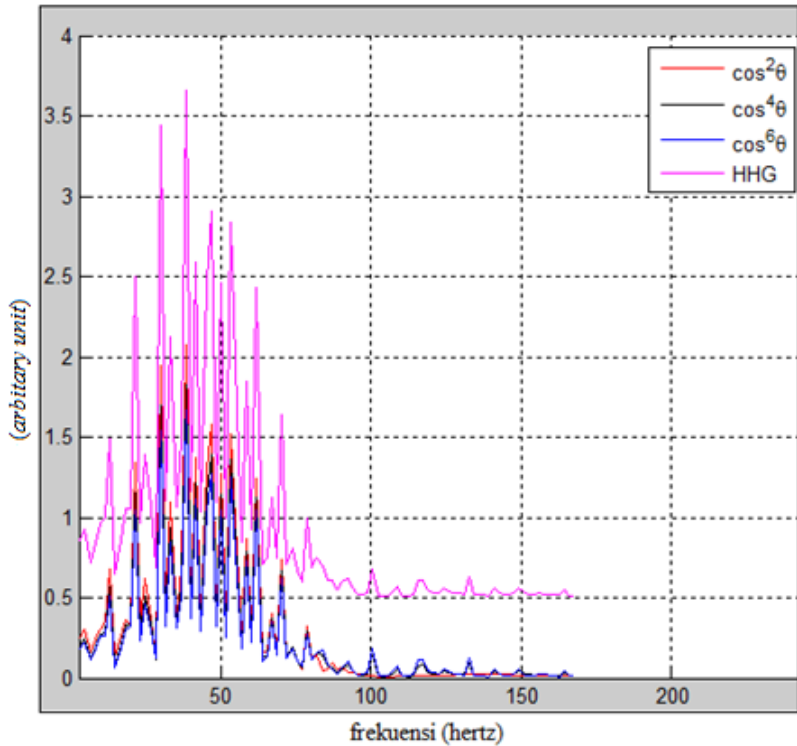
Gambar 4.7 Grafik sinyal HHG, molekul O_2 (orde $n=19$) pada suhu 50 K (garis biru) dan molekul O_2 pada suhu 200 K (garis merah).

4.3 Sinyal HHG pada Domain Frekuensi

Hasil simulasi sinyal HHG pada domain waktu dapat diubah menjadi sinyal HHG pada domain frekuensi dengan menggunakan *Fast Fourier Transform* (FFT) atau dikenal juga sebagai Transformasi Fourier Cepat.

FFT diterapkan dalam beragam bidang dari pengolahan sinyal digital dan memecahkan persamaan diferensial parsial menjadi algoritma-algoritma untuk penggandaan bilangan integer dalam

jumlah banyak. Ada pun kelas dasar dari algoritma FFT yaitu *decimation in time* (DIT) dan *decimation in frequency* (DIF). Garis besar dari kata *Fast* diartikan karena formulasi FFT jauh lebih cepat dibandingkan dengan metode perhitungan algoritma Fourier Transform sebelumnya.

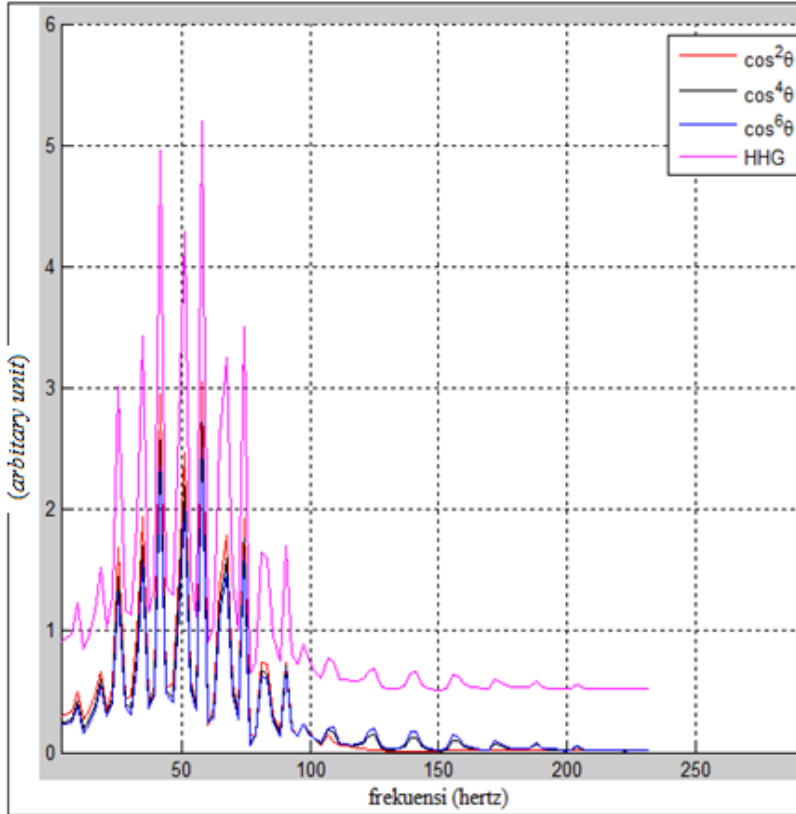


Gambar 4.8 Grafik sinyal *Fast Fourier Transform* HHG pada molekul N_2 (orde $n=19$) suhu 200 K.

Metode FFT memerlukan sekitar 10000 operasi algoritma matematika untuk data dengan 1000 observasi, 100 kali lebih cepat dibandingkan dengan metode sebelumnya. Penemuan FFT dan perkembangan personal komputer, teknik FFT dalam proses analisis data menjadi populer, dan merupakan salah satu metode baku dalam analisis data. Satu bentuk transformasi yang umum digunakan untuk

merubah sinyal dari domain waktu ke domain frekuensi adalah Transformasi Fourier: Persamaan dari bentuk sinyal $x(t)$.

$$(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (4.1)$$

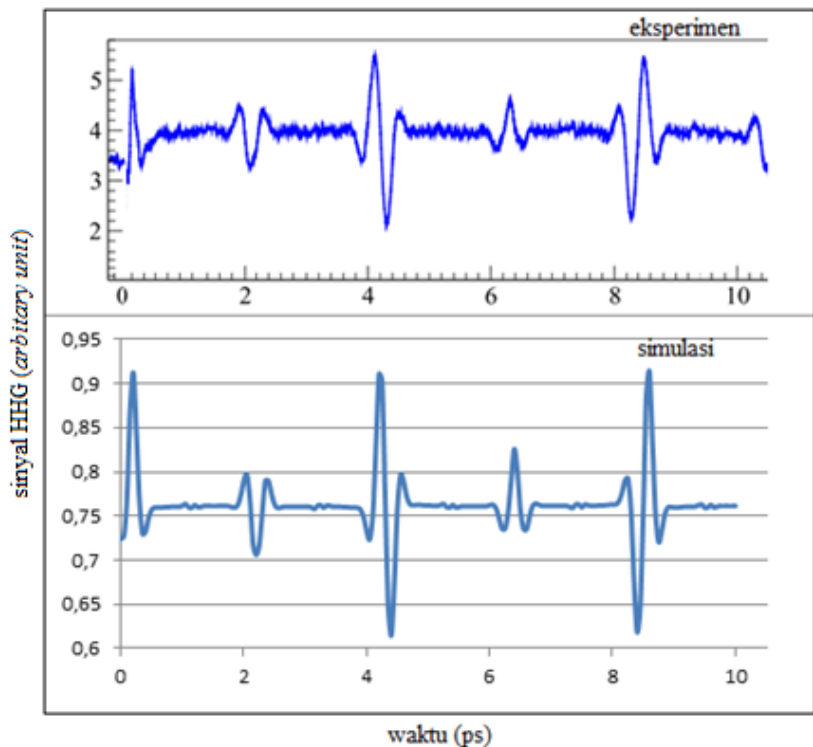


Gambar 4.9 Grafik sinyal *Fast Fourier Transform* HHG pada molekul O_2 (orde $n=19$) suhu 200 K.

Frekuensi didefinisikan secara sederhana sebagai kebalikan dari waktu. Sehingga waktu yang satuannya adalah detik (*second*) akan menjadi Hertz. Gambar 4.9 merupakan tampilan FFT sinyal HHG N_2 pada $\cos^2\theta$, $\cos^4\theta$, $\cos^6\theta$ dan HHG.

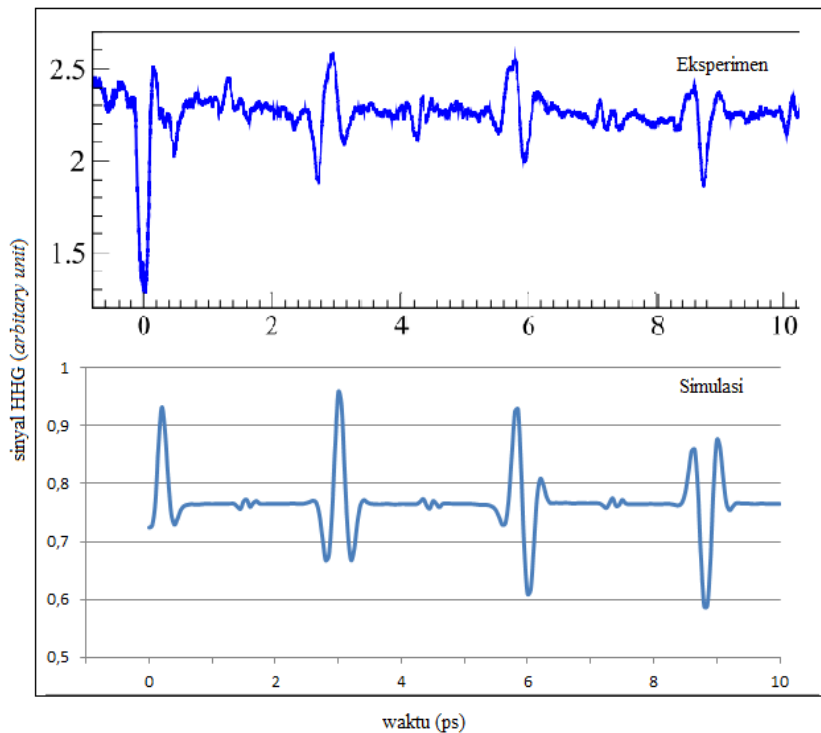
4.4 Perbandingan Simulasi dan Eksperimen

Secara keseluruhan, hasil perhitungan sinyal HHG yang ditampilkan dalam bentuk grafik, menunjukkan bahwa simulasi ini memiliki bentuk sinyal yang relatif sama dengan eksperimen, hal ini menunjukkan bahwa dinamika molekul dari simulasi telah menunjukkan hasil seperti pada eksperimen, namun untuk kekuatan sinyal HHG dari eksperimen dan simulasi terdapat perbedaan yang sangat jauh, hal ini disebabkan karena variabel penguatan laser I_{pump} dan I_{probe} yang tidak bisa diatur pada simulasi.



Gambar 4.10. Perbandingan eksperimental (grafik atas) dan simulasi (grafik bawah) sinyal harmonik untuk N_2 ; intensitas pompa $I_{\text{pump}} = 0,8 \times 10^{14} \text{ W / cm}^2$, intensitas probe $I_{\text{probe}} = 1,7 \times 10^{14} \text{ W / cm}^2$; durasi 40 fs, panjang gelombang 800 nm, suhu 200 K.

namun perbedaan kekuatan sinyal tidak menjadi masalah dalam penelitian ini karena secara teori tidak mempengaruhi karakteristik dinamika molekulnya, Hasil eksperimen ini merupakan hasil eksperimen dari Miyazaki dkk pada tahun 2007. Bentuk sinyal yang didapatkan dari simulasi dalam setiap *picosecond* (ps) memiliki karakteristik yang sama dengan bentuk sinyal dari hasil eksperimen. Dari Gambar 4.10 didapatkan hasil yang relatif sama untuk waktu didapatkannya sinyal tertinggi dan terendah untuk sinyal pembangkitan harmonik tinggi pada molekul Nitrogen yaitu pada 0,2 ps ; 4,2 ps serta 8,2 ps (1 *picosecond* = 1000 *femtosecond*).



Gambar 4. 11 Perbandingan eksperimental (grafik atas) dan simulasi (grafik bawah) HHG untuk O_2 ; intensitas pompa $I_{\text{pump}} = 0,5 \times 10^{14} \text{ W / cm}^2$, intensitas probe $I_{\text{probe}} = 1,2 \times 10^{14} \text{ W / cm}^2$; durasi 40 fs, panjang gelombang 800 nm, suhu 200 K.

Dari Gambar 4.11 didapatkan kesimpulan yang sama dengan perbandingan eksperimen dan simulasi pada molekul N_2 , yaitu didapatkan hasil yang relatif sama untuk waktu sinyal tertinggi dan terendah untuk sinyal pembangkitan harmonik tinggi pada molekul Oksigen yaitu pada 0,2 ps ; 4 ps serta 5,8 ps, namun sama halnya dengan simulasi N_2 , kekuatan sinyal yang didapatkan memiliki perbedaan yang cukup besar. Perlu diperhatikan juga untuk sinyal sebelum waktu 0 ps pada gambar 4.11 (eksperimen) bukanlah merupakan sinyal HHG karena grafik sebelum 0 ps merupakan grafik ketika laser dipaparkan sebelum dilakukan metode *switch-off*.

Secara keseluruhan, hasil simulasi dapat memberikan gambaran yang cukup real untuk bentuk sinyal HHG yang dihasilkan, keselarasan waktu terbentuknya amplitudo tinggi dan amplitudo rendah dari molekul nitrogen maupun molekul oksigen antara simulasi dengan eksperimen secara keseluruhan juga mendapatkan hasil yang relatif sama, hal ini dibuktikan waktu amplitudo tinggi dan rendah simulasi yang sama dengan waktu di eksperimen.

[Halaman ini sengaja dikosongkan]

BAB V

PENUTUP

5.1 Kesimpulan

Simulasi Matlab menggunakan Matlab r2012b dan teori pembangkitan harmonik tinggi dapat digunakan untuk memprediksi bentuk sinyal maupun kapan terjadinya amplitudo HHG tertinggi dan terendah pada suatu molekul, yaitu molekul Nitrogen dan Oksigen. Didapatkan juga kesimpulan bahwa temperatur sangat mempengaruhi besaran sinyal yang didapatkan, yaitu semakin besar suhu awal akan memperkecil besaran amplitudo sinyal HHG.

Fast Fourier Transform atau Transformasi Fourier Cepat, juga dimasukkan didalam simulasi ini, sehingga bisa dianalisis juga mengenai sinyal HHG pada domain frekuensi. Dengan analisis grafik hasil simulasi Matlab dari domain waktu dan domain frekuensi, didapatkan hasil kesimpulan adanya perbedaan spesifik antara molekul nitrogen dan oksigen. Pada molekul oksigen amplitudo tinggi kedua didapatkan lebih cepat yaitu pada 3000 fs (*femtosecond*), sedangkan amplitudo tinggi kedua pada molekul nitrogen didapatkan pada 4200 fs. Jumlah amplitudo yang didapatkan juga memiliki perbedaan,. Sehingga dapat ditarik kesimpulan bahwa oksigen memiliki sinyal HHG dengan periode lebih pendek dibandingkan nitrogen.

5.1 Saran

Intensitas pompa I_{pump} , intensitas probe I_{probe} , serta durasi pemaparan laser seharusnya dimasukkan dalam variabel karena juga mempengaruhi besaran amplitudo.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- Abdurrouf, A. 2007. Intense-field theory of high harmonic generation from freely rotating molecules, Bielefeld University. Germany.
- Cai, L., J. Marango, dan B. Friedrich. 2001. Time-dependent alignment and orientation of molecules in combined electrostatic and pulsed nonresonant laser fields. *Phys. Rev. Lett.* 86(5):775 – 778.
- Corkum, P. B. 1993. Plasma perspective on strong-field multiphoton ionization. *Phys. Rev. Lett.* 71(13):1994 – 1997.
- Faisal, F. H. M. 1987. Theory of Multiphoton Processes. Plenum Press. New York.
- Faisal, F.H.M., A. Abdurrouf, K. Miyazaki, dan G. Miyaji. 2007. Origin of anomalous spectra of dynamic alignments observed in N₂ and O₂. *Phys. Rev. Lett.* 98(14):143001.
- Ferray, M., A L'Huillier, X F Li, L A Lompre, G Mainfray, dan C Manus. 1988. Multiple- harmonic conversion of 1064 nm radiation in rare gases. *J. Phys. B: At. Mol. Opt. Phys.* 21(3):L31.
- Friedrich, B. dan D. 1995. Herschbach. Alignment and trapping of molecules in intense laser fields. *Phys. Rev. Lett.* 74(23):4623 – 4626.
- Ganeev, R. A., C. Hutchison, T. Witting, F. Frank, W. Okell, A. Zaïr, dan S. Weber. 2002. High-order harmonic generation in graphite plasma plumes using ultrashort laser pulses: a systematic analysis of harmonic radiation and plasma conditions. *J. Phys. B: At. Mol. Opt. Phys.* 45 165402.

- Kaku, M., K. Masuda, dan K. Miyazaki. 2004 Observation of revival structure in femtosecond laser induced alignment of N_2 with high-order harmonic generation. *Japan. J. Appl. Phys.* 43(4B):L591 – 593.
- Kanai, T., S. Minemoto, dan H. Sakai. 2005. Quantum interference during high-order harmonic generation from aligned molecule. *Nature*, 435(7041):470 – 474.
- Krause. J. L., K.J. Schafer, dan K.C. Kulander. 1992. High-order harmonic generation from atoms and ions in the high intensity regime. *Phys. Rev. Lett.* 68(24):3535 – 3538.
- Kulander, K. C., K.J. Schafer, dan J.L. Krause. 1993. Dynamics of short-pulse excitation and ionization and harmonic generation. Plenum Press. New York.
- Lein, M., N. Hay, R. Velotta, J.P. Marangos, dan P. L. Knight. 2002. Interference effects in high-order harmonic generation with molecules. *Phys. Rev. A*, 66(2):023805.
- Lewenstein, M., Ph. Balcou, M. Yu. Ivanov, A. L’Huillier, dan P.B. Corkum. 1994. Theory of high-harmonic generation by low-frequency laser fields. *Phys. Rev. A* 49(3):2117 – 2132.
- Li, X. F., M. Ferray, L.A Lompre, dan G. Mainfray. 1989. Multiple-harmonic generation in rare gases at high laser intensities. *Phys. Rev. A*. 39(11):5751–5761.
- Liang, Y., S. Augst, S.L. Chin, Y. Beaudoin, dan M. Chaker. 1994. High harmonic generation in atomic and diatomic molecular gases using intense picosecond laser pulses - a comparison. *J. Phys. B: At. Mol. Opt Phys.* 27(28):5119 – 5130.
- Litvinyuk, I. V., K.F. Lee, P.W. Dooley, D.M. Rayner, D.M. Villeneuve, dan P.B. Corkum. 2003. Alignment-dependent strong field ionization of molecules. *Phys. Rev. Lett.* 90(23):233003.

- Loesch, H. dan A. Remscheid. 1990. Bruto force in molecular reaction dynamics: A novel technique for measuring steric effects. *J. Chem. Phys.* 93(4):4779 – 4790.
- Miyazaki, K., M. Kaku, G. Miyaji, A. Abdurrouf, dan F.H.M. Faisal. 2005. Field-free alignment of molecules observed with high-order harmonic generation. *Phys. Rev. Lett.* 95(24):243903.
- Poulsen, M. D., T. Ejdrup, dan H. Stapelfeldt. 2006. Alignment enhancement by the combination of a short and long laser pulse. *Phys. Rev. A.* 73(3):033405.
- Sakai, H. dan K. Miyazaki. 1995. High-order harmonic generation in nitrogen molecule with subpicosecond visible dye-laser pulse. *Appl. Phys. B: Lasers Opt.* 61(5):493 – 498.
- Seideman, T. 1999. Revival structure of aligned rotational wave packets. *Phys. Rev. Lett.* 83(24):4971 – 4974.
- Seideman, T. dan E. Hamilton. 2006. Nonadiabtic alignment by intense pulse. Concepts, theory, and directions. *Adv. in Atomic, Mol, and Opt. Phys.* 52:289 – 329
- Shan, B., X.M. Tong, Z. X. Zhao, Z. Chang, dan C.D. Lin. 2002. High-order harmonic cut-off extension of the O₂ molecule due to ionization suppression. *Phys. Rev. A.* 66(6):061401(R).
- Stapelfeldt, H. 2004. Laser aligned molecules: Application in physics and chemistry. *Phys. Scr.* T110:132 – 136.
- Stapelfeldt, H. dan T. Seideman. 2003. Colloquium: Aligning molecules with strong laser pulses. *Rev. Mod. Phys.* 75(2):543 – 557.
- Torres, R., R. de Nalda, dan J.P. Marangos. 2005. Dynamics of laser-induced molecular alignment in the impulsive and adiabatic regime: A direct comparison. *Phys. Rev. A.* 72(2):023420.

- Vellota, R., N. Hay, M.B. Mason, M. Castillejo, dan J.P. Marangos. 2001. High-order harmonic generation in aligned molecules. *Phys. Rev. Lett.* 87(18):183901.
- Zhou, X. X., X.M. Tong, Z.X. Zhao, dan C.D. Lin. 2005. Alignment dependence of high-order harmonic generation from N_2 and O_2 molecules in intense laser fields. *Phys. Rev. A*, 72(3):033412.
- Zon, B. dan B. Katsnelson. 1976. Nonresonant scattering of intense light by a molecule. *Zh. Exp. Teor. Fiz.* 69:1166 – 1178.

LAMPIRAN

Lampiran 1. Data Hasil Simulasi

L.1.1 $\cos^2 \theta$ pada molekul nitrogen.

Waktu (fs)	Amplitudo	
	50 K	100 K
0	0,333	0,333
50	0,338	0,338
100	0,389	0,388
150	0,517	0,504
200	0,633	0,591
250	0,648	0,570
300	0,574	0,480
350	0,466	0,391
400	0,384	0,350
450	0,353	0,353
500	0,365	0,374
550	0,395	0,390
600	0,420	0,396
650	0,432	0,397
700	0,434	0,395
750	0,431	0,395
800	0,427	0,395
850	0,425	0,395
900	0,424	0,395
950	0,424	0,395
1000	0,424	0,395
1050	0,424	0,395
1100	0,424	0,395
1150	0,424	0,395
1200	0,424	0,395
1250	0,424	0,395
1300	0,424	0,395

Waktu (fs)	Amplitudo	
	50 K	100 K
1350	0,424	0,395
1400	0,424	0,395
1450	0,424	0,395
1500	0,425	0,395
1550	0,424	0,395
1600	0,423	0,396
1650	0,419	0,395
1700	0,415	0,394
1750	0,413	0,392
1800	0,418	0,391
1850	0,434	0,395
1900	0,460	0,410
1950	0,483	0,431
2000	0,489	0,446
2050	0,466	0,435
2100	0,416	0,395
2150	0,361	0,344
2200	0,329	0,314
2250	0,336	0,325
2300	0,376	0,368
2350	0,427	0,413
2400	0,463	0,434
2450	0,471	0,429
2500	0,458	0,412
2550	0,437	0,397
2600	0,420	0,391
2650	0,413	0,392

Waktu (fs)	Amplitudo	
	50 K	100 K
2700	0,414	0,394
2750	0,418	0,395
2800	0,422	0,395
2850	0,423	0,395
2900	0,424	0,395
2950	0,423	0,395
3000	0,423	0,395
3050	0,423	0,395
3100	0,423	0,395
3150	0,422	0,395
3200	0,422	0,395
3250	0,422	0,394
3300	0,422	0,394
3350	0,422	0,394
3400	0,421	0,394
3450	0,421	0,394
3500	0,420	0,394
3550	0,421	0,393
3600	0,422	0,393
3650	0,425	0,392
3700	0,428	0,393
3750	0,426	0,393
3800	0,415	0,393
3850	0,388	0,387
3900	0,351	0,370
3950	0,324	0,344
4000	0,335	0,329

Waktu (fs)	Amplitudo	
	50 K	100 K
4050	0,407	0,357
4100	0,528	0,441
4150	0,649	0,554
4200	0,701	0,621
4250	0,641	0,579
4300	0,488	0,435
4350	0,314	0,275
4400	0,205	0,194
4450	0,202	0,226
4500	0,285	0,323
4550	0,395	0,409
4600	0,473	0,445
4650	0,497	0,437
4700	0,480	0,415
4750	0,448	0,399
4800	0,424	0,394
4850	0,413	0,394
4900	0,413	0,395
4950	0,417	0,396
5000	0,420	0,396
5050	0,422	0,396
5100	0,423	0,396
5150	0,423	0,396
5200	0,423	0,396
5250	0,423	0,395
5300	0,423	0,395
5350	0,423	0,395

Waktu (fs)	Amplitudo	
	50 K	100 K
5400	0,423	0,395
5450	0,423	0,395
5500	0,423	0,395
5550	0,423	0,395
5600	0,423	0,395
5650	0,423	0,395
5700	0,423	0,395
5750	0,423	0,395
5800	0,425	0,395
5850	0,428	0,396
5900	0,432	0,397
5950	0,434	0,399
6000	0,427	0,399
6050	0,410	0,394
6100	0,384	0,378
6150	0,362	0,358
6200	0,360	0,346
6250	0,387	0,360
6300	0,439	0,402
6350	0,492	0,452
6400	0,520	0,478
6450	0,508	0,462
6500	0,464	0,417
6550	0,414	0,373
6600	0,381	0,355
6650	0,376	0,362
6700	0,391	0,381

Waktu (fs)	Amplitudo	
	50 K	100 K
6750	0,413	0,395
6800	0,429	0,400
6850	0,434	0,399
6900	0,433	0,397
6950	0,429	0,396
7000	0,425	0,395
7050	0,424	0,396
7100	0,423	0,396
7150	0,424	0,396
7200	0,424	0,396
7250	0,424	0,396
7300	0,425	0,396
7350	0,425	0,396
7400	0,425	0,396
7450	0,425	0,396
7500	0,425	0,396
7550	0,426	0,397
7600	0,426	0,397
7650	0,427	0,397
7700	0,427	0,397
7750	0,426	0,398
7800	0,425	0,398
7850	0,422	0,398
7900	0,420	0,398
7950	0,422	0,398
8000	0,436	0,399
8050	0,464	0,406

Waktu (fs)	Amplitudo	
	50 K	100 K
8100	0,500	0,424
8150	0,522	0,448
8200	0,503	0,458
8250	0,425	0,424
8300	0,302	0,336
8350	0,188	0,227
8400	0,149	0,171
8450	0,222	0,225
8500	0,382	0,375
8550	0,553	0,531
8600	0,651	0,601
8650	0,641	0,558
8700	0,549	0,456
8750	0,439	0,372
8800	0,366	0,342
8850	0,349	0,355
8900	0,370	0,379
8950	0,403	0,394
9000	0,426	0,398
9050	0,435	0,397
9100	0,434	0,395
9150	0,430	0,395
9200	0,426	0,395
9250	0,425	0,395
9300	0,424	0,395
9350	0,424	0,395
9400	0,424	0,395

Waktu (fs)	Amplitudo	
	50 K	100 K
9450	0,424	0,395
9500	0,424	0,395
9550	0,424	0,395
9600	0,424	0,395
9650	0,424	0,395
9700	0,424	0,395
9750	0,424	0,395
9800	0,424	0,395
9850	0,424	0,395
9900	0,424	0,395
9950	0,424	0,395
10000	0,422	0,395

L.1.2 $\cos^4\theta$ pada molekul nitrogen.

Waktu (fs)	Amplitudo	
	50 K	100 K
0	0,200	0,200
50	0,204	0,204
100	0,251	0,249
150	0,382	0,366
200	0,512	0,454
250	0,514	0,420
300	0,421	0,328
350	0,317	0,250
400	0,244	0,215
450	0,218	0,220
500	0,231	0,240
550	0,261	0,256
600	0,286	0,262
650	0,298	0,262
700	0,300	0,261
750	0,297	0,261
800	0,293	0,261
850	0,291	0,261
900	0,290	0,261
950	0,294	0,262
1000	0,300	0,266
1050	0,297	0,267
1100	0,286	0,258
1150	0,287	0,259
1200	0,297	0,266
1250	0,295	0,262
1300	0,288	0,259

Waktu (fs)	Amplitudo	
	50 K	100 K
1350	0,290	0,262
1400	0,294	0,263
1450	0,295	0,263
1500	0,295	0,263
1550	0,294	0,263
1600	0,293	0,263
1650	0,289	0,263
1700	0,285	0,262
1750	0,283	0,260
1800	0,286	0,258
1850	0,301	0,262
1900	0,325	0,275
1950	0,354	0,295
2000	0,367	0,313
2050	0,331	0,303
2100	0,257	0,247
2150	0,215	0,200
2200	0,219	0,199
2250	0,220	0,203
2300	0,235	0,229
2350	0,292	0,282
2400	0,342	0,306
2450	0,347	0,296
2500	0,327	0,277
2550	0,304	0,263
2600	0,288	0,258
2650	0,281	0,258

Waktu (fs)	Amplitudo	
	50 K	100 K
2700	0,281	0,260
2750	0,285	0,261
2800	0,288	0,261
2850	0,290	0,261
2900	0,290	0,261
2950	0,291	0,261
3000	0,291	0,261
3050	0,287	0,261
3100	0,281	0,257
3150	0,286	0,257
3200	0,297	0,265
3250	0,295	0,264
3300	0,286	0,258
3350	0,290	0,262
3400	0,296	0,265
3450	0,294	0,262
3500	0,290	0,261
3550	0,290	0,261
3600	0,293	0,261
3650	0,296	0,261
3700	0,300	0,261
3750	0,300	0,262
3800	0,289	0,263
3850	0,265	0,258
3900	0,229	0,242
3950	0,202	0,219
4000	0,208	0,204

Waktu (fs)	Amplitudo	
	50 K	100 K
4050	0,265	0,224
4100	0,369	0,294
4150	0,495	0,395
4200	0,568	0,473
4250	0,500	0,433
4300	0,323	0,278
4350	0,171	0,143
4400	0,096	0,087
4450	0,087	0,108
4500	0,154	0,195
4550	0,263	0,279
4600	0,344	0,312
4650	0,367	0,303
4700	0,350	0,281
4750	0,319	0,266
4800	0,295	0,262
4850	0,285	0,262
4900	0,285	0,264
4950	0,288	0,264
5000	0,291	0,264
5050	0,292	0,263
5100	0,292	0,263
5150	0,296	0,263
5200	0,301	0,267
5250	0,295	0,266
5300	0,284	0,258
5350	0,287	0,259

Waktu (fs)	Amplitudo	
	50 K	100 K
5400	0,296	0,265
5450	0,291	0,260
5500	0,284	0,258
5550	0,287	0,261
5600	0,290	0,262
5650	0,290	0,261
5700	0,289	0,261
5750	0,289	0,261
5800	0,291	0,261
5850	0,295	0,261
5900	0,299	0,263
5950	0,301	0,265
6000	0,296	0,266
6050	0,281	0,262
6100	0,260	0,249
6150	0,247	0,233
6200	0,250	0,227
6250	0,258	0,236
6300	0,279	0,254
6350	0,342	0,302
6400	0,398	0,348
6450	0,373	0,321
6500	0,310	0,268
6550	0,275	0,242
6600	0,260	0,230
6650	0,253	0,234
6700	0,263	0,250

Waktu (fs)	Amplitudo	
	50 K	100 K
6750	0,283	0,264
6800	0,299	0,268
6850	0,304	0,267
6900	0,302	0,264
6950	0,298	0,263
7000	0,295	0,263
7050	0,293	0,263
7100	0,293	0,263
7150	0,294	0,263
7200	0,294	0,264
7250	0,290	0,262
7300	0,285	0,259
7350	0,291	0,260
7400	0,301	0,268
7450	0,297	0,265
7500	0,288	0,260
7550	0,293	0,264
7600	0,300	0,266
7650	0,296	0,263
7700	0,293	0,263
7750	0,292	0,263
7800	0,291	0,263
7850	0,288	0,263
7900	0,285	0,263
7950	0,286	0,262
8000	0,298	0,262
8050	0,323	0,268

Waktu (fs)	Amplitudo	
	50 K	100 K
8100	0,357	0,284
8150	0,380	0,306
8200	0,363	0,316
8250	0,286	0,285
8300	0,169	0,203
8350	0,084	0,112
8400	0,074	0,082
8450	0,128	0,128
8500	0,248	0,241
8550	0,421	0,394
8600	0,530	0,462
8650	0,500	0,404
8700	0,394	0,306
8750	0,292	0,233
8800	0,229	0,208
8850	0,215	0,222
8900	0,237	0,245
8950	0,269	0,260
9000	0,292	0,263
9050	0,300	0,262
9100	0,300	0,261
9150	0,296	0,261
9200	0,292	0,261
9250	0,290	0,261
9300	0,291	0,261
9350	0,296	0,263
9400	0,301	0,267

Waktu (fs)	Amplitudo	
	50 K	100 K
9450	0,294	0,265
9500	0,285	0,257
9550	0,290	0,260
9600	0,298	0,266
9650	0,293	0,261
9700	0,287	0,259
9750	0,291	0,263
9800	0,295	0,263
9850	0,295	0,263
9900	0,294	0,263
9950	0,294	0,263
10000	0,292	0,263

L.1.3 $\cos^6\theta$ pada molekul nitrogen.

Waktu (fs)	Amplitudo	
	50k	100k
0	0,143	0,1429
50	0,146	0,1462
100	0,188	0,187
150	0,318	0,3005
200	0,453	0,3876
250	0,445	0,346
300	0,345	0,2557
350	0,244	0,1841
400	0,178	0,1528
450	0,155	0,1583
500	0,168	0,178
550	0,197	0,1931
600	0,221	0,1988
650	0,234	0,1992
700	0,236	0,1989
750	0,231	0,1977
800	0,228	0,1977
850	0,226	0,1981
900	0,225	0,1982
950	0,231	0,199
1000	0,239	0,2045
1050	0,235	0,2054
1100	0,219	0,1943
1150	0,222	0,1953
1200	0,236	0,2047
1250	0,231	0,1991
1300	0,221	0,195

Waktu (fs)	Amplitudo	
	50 K	100 K
1350	0,222	0,1983
1400	0,232	0,2009
1450	0,235	0,2028
1500	0,231	0,2
1550	0,231	0,2003
1600	0,228	0,2003
1650	0,228	0,2013
1700	0,223	0,1999
1750	0,220	0,1978
1800	0,223	0,1963
1850	0,236	0,1997
1900	0,259	0,2113
1950	0,290	0,2306
2000	0,306	0,2496
2050	0,266	0,2394
2100	0,182	0,1786
2150	0,149	0,1364
2200	0,168	0,1465
2250	0,164	0,1457
2300	0,166	0,1643
2350	0,226	0,2198
2400	0,282	0,2439
2450	0,286	0,2323
2500	0,262	0,2138
2550	0,239	0,2004
2600	0,224	0,195
2650	0,217	0,1953

Waktu (fs)	Amplitudo	
	50k	100k
2700	0,218	0,1971
2750	0,223	0,1989
2800	0,222	0,198
2850	0,223	0,196
2900	0,227	0,1987
2950	0,228	0,1987
3000	0,230	0,1994
3050	0,221	0,1974
3100	0,213	0,1925
3150	0,221	0,1928
3200	0,236	0,2042
3250	0,233	0,2027
3300	0,220	0,1939
3350	0,226	0,2003
3400	0,236	0,204
3450	0,233	0,2001
3500	0,228	0,1993
3550	0,226	0,1986
3600	0,229	0,1986
3650	0,232	0,1987
3700	0,237	0,1996
3750	0,237	0,2006
3800	0,228	0,2009
3850	0,205	0,1966
3900	0,172	0,1826
3950	0,145	0,1605
4000	0,150	0,1462

Waktu (fs)	Amplitudo	
	50 K	100 K
4050	0,199	0,1639
4100	0,292	0,2248
4150	0,416	0,3184
4200	0,500	0,3996
4250	0,428	0,3614
4300	0,242	0,2033
4350	0,105	0,0845
4400	0,045	0,0393
4450	0,036	0,0557
4500	0,095	0,1362
4550	0,200	0,2171
4600	0,280	0,2476
4650	0,302	0,2385
4700	0,285	0,2178
4750	0,255	0,2038
4800	0,232	0,1996
4850	0,222	0,2001
4900	0,221	0,2008
4950	0,227	0,2021
5000	0,229	0,2018
5050	0,229	0,2012
5100	0,228	0,1999
5150	0,233	0,201
5200	0,241	0,2062
5250	0,233	0,205
5300	0,217	0,1936
5350	0,222	0,1959

Waktu (fs)	Amplitudo	
	50k	100k
5400	0,234	0,2042
5450	0,228	0,1972
5500	0,219	0,1941
5550	0,223	0,199
5600	0,224	0,1986
5650	0,224	0,1966
5700	0,226	0,1991
5750	0,226	0,1988
5800	0,229	0,1987
5850	0,229	0,198
5900	0,234	0,1997
5950	0,237	0,2019
6000	0,232	0,2032
6050	0,218	0,1996
6100	0,199	0,188
6150	0,190	0,1737
6200	0,196	0,1708
6250	0,197	0,1781
6300	0,207	0,1873
6350	0,271	0,2319
6400	0,337	0,2846
6450	0,305	0,2531
6500	0,234	0,1985
6550	0,208	0,181
6600	0,202	0,1714
6650	0,194	0,1737
6700	0,202	0,1893

Waktu (fs)	Amplitudo	
	50 K	100 K
6750	0,221	0,2018
6800	0,236	0,2056
6850	0,241	0,2042
6900	0,237	0,2017
6950	0,232	0,1998
7000	0,234	0,2011
7050	0,232	0,2025
7100	0,228	0,1999
7150	0,230	0,2003
7200	0,229	0,2008
7250	0,226	0,2003
7300	0,219	0,1946
7350	0,226	0,1962
7400	0,241	0,2069
7450	0,234	0,2034
7500	0,222	0,1953
7550	0,229	0,2026
7600	0,239	0,2052
7650	0,233	0,2005
7700	0,227	0,1991
7750	0,228	0,2007
7800	0,227	0,2007
7850	0,224	0,2005
7900	0,219	0,1995
7950	0,220	0,1987
8000	0,231	0,1988
8050	0,255	0,204

Waktu (fs)	Amplitudo	
	50k	100k
8100	0,288	0,2182
8150	0,310	0,2385
8200	0,295	0,2481
8250	0,219	0,22
8300	0,109	0,1428
8350	0,036	0,0611
8400	0,034	0,0406
8450	0,079	0,0807
8500	0,185	0,1793
8550	0,357	0,3274
8600	0,470	0,3939
8650	0,427	0,329
8700	0,317	0,235
8750	0,221	0,1689
8800	0,164	0,1467
8850	0,151	0,1604
8900	0,173	0,1831
8950	0,205	0,1968
9000	0,227	0,2002
9050	0,236	0,1994
9100	0,236	0,1986
9150	0,230	0,1973
9200	0,228	0,1979
9250	0,225	0,1981
9300	0,227	0,1985
9350	0,233	0,2002
9400	0,239	0,2055

Waktu (fs)	Amplitudo	
	50 K	100 K
9450	0,231	0,2033
9500	0,217	0,1929
9550	0,225	0,1974
9600	0,237	0,205
9650	0,228	0,1973
9700	0,219	0,1951
9750	0,225	0,1993
9800	0,235	0,2019
9850	0,234	0,2022
9900	0,230	0,1998
9950	0,230	0,2002
10000	0,227	0,2005

L.1.3 HHG pada Temperatur 50 K, Nitrogen dan Oksigen

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
0	0,7242	0,7242
50	0,7283	0,7277
100	0,7777	0,7717
150	0,9148	0,8962
200	1,0483	1,0346
250	1,0521	1,0737
300	0,958	1,0117
350	0,8483	0,9101
400	0,7707	0,8181
450	0,7426	0,7595
500	0,7562	0,7418
550	0,787	0,7572
600	0,8131	0,7871
650	0,826	0,8141
700	0,828	0,8297
750	0,8246	0,834
800	0,8211	0,8316
850	0,8185	0,8273
900	0,8177	0,8233
950	0,8211	0,821
1000	0,8266	0,8213
1050	0,8236	0,8216
1100	0,814	0,8214
1150	0,8155	0,8219
1200	0,8239	0,823
1250	0,8215	0,8253
1300	0,8153	0,8217

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
1350	0,8168	0,8063
1400	0,8213	0,7955
1450	0,8224	0,8131
1500	0,8215	0,8407
1550	0,8212	0,836
1600	0,8194	0,8101
1650	0,8165	0,8095
1700	0,8119	0,8301
1750	0,8092	0,8359
1800	0,8135	0,8249
1850	0,8287	0,8169
1900	0,8546	0,8174
1950	0,884	0,8202
2000	0,8964	0,8203
2050	0,8609	0,8198
2100	0,7876	0,8193
2150	0,7416	0,818
2200	0,7394	0,8173
2250	0,741	0,8169
2300	0,761	0,8188
2350	0,8199	0,8255
2400	0,8697	0,8376
2450	0,8758	0,852
2500	0,8556	0,8601
2550	0,8319	0,8499
2600	0,8149	0,8113
2650	0,8078	0,7463

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
2700	0,8086	0,6769
2750	0,8126	0,6374
2800	0,8153	0,6513
2850	0,8169	0,7203
2900	0,8181	0,8405
2950	0,8185	0,9948
3000	0,8187	1,1134
3050	0,8146	1,1103
3100	0,8097	0,994
3150	0,8138	0,8519
3200	0,8236	0,7427
3250	0,8216	0,6822
3300	0,8135	0,6766
3350	0,8168	0,7187
3400	0,8226	0,7798
3450	0,8206	0,8289
3500	0,8173	0,852
3550	0,8171	0,8522
3600	0,8195	0,8404
3650	0,823	0,8272
3700	0,8266	0,8181
3750	0,8263	0,8144
3800	0,8153	0,8143
3850	0,7891	0,8149
3900	0,752	0,8156
3950	0,7234	0,8173
4000	0,7311	0,8178

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
4050	0,793	0,818
4100	0,9048	0,8167
4150	1,0354	0,8142
4200	1,1088	0,8179
4250	1,0385	0,8325
4300	0,8571	0,8439
4350	0,6948	0,8288
4400	0,6095	0,8008
4450	0,602	0,8026
4500	0,6753	0,8291
4550	0,7889	0,8327
4600	0,8728	0,8118
4650	0,8976	0,8042
4700	0,8792	0,8154
4750	0,847	0,8239
4800	0,8221	0,8231
4850	0,8112	0,8211
4900	0,8107	0,8209
4950	0,8147	0,82
5000	0,8178	0,8194
5050	0,8188	0,8189
5100	0,8191	0,8193
5150	0,8226	0,8219
5200	0,8269	0,8256
5250	0,8219	0,8287
5300	0,8123	0,8274
5350	0,815	0,8162

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
5400	0,8226	0,7917
5450	0,8186	0,757
5500	0,8128	0,7251
5550	0,8151	0,7161
5600	0,8174	0,7475
5650	0,8171	0,8241
5700	0,8169	0,9362
5750	0,8171	1,0549
5800	0,8191	1,1186
5850	0,8223	1,0629
5900	0,8268	0,9045
5950	0,829	0,7405
6000	0,8236	0,6348
6050	0,8071	0,591
6100	0,7841	0,6108
6150	0,769	0,6939
6200	0,7711	0,8009
6250	0,7833	0,8779
6300	0,8115	0,9035
6350	0,875	0,889
6400	0,9286	0,8574
6450	0,9048	0,8281
6500	0,8416	0,811
6550	0,8028	0,8064
6600	0,7841	0,8096
6650	0,7769	0,8151
6700	0,7884	0,8198

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
6750	0,8099	0,8227
6800	0,8261	0,8228
6850	0,8316	0,8222
6900	0,8295	0,8225
6950	0,825	0,8224
7000	0,8223	0,8241
7050	0,8207	0,8259
7100	0,8202	0,8223
7150	0,8212	0,8089
7200	0,821	0,7972
7250	0,8178	0,8101
7300	0,8135	0,8382
7350	0,8186	0,8395
7400	0,8275	0,8131
7450	0,8238	0,8064
7500	0,8163	0,8271
7550	0,8208	0,8367
7600	0,8263	0,8255
7650	0,8236	0,8158
7700	0,8203	0,8155
7750	0,82	0,8167
7800	0,8185	0,8177
7850	0,8154	0,8183
7900	0,8122	0,8186
7950	0,8138	0,8201
8000	0,8265	0,821
8050	0,8537	0,8214

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
8100	0,8896	0,82
8150	0,9136	0,8144
8200	0,8954	0,8041
8250	0,814	0,7917
8300	0,6914	0,7847
8350	0,5964	0,7937
8400	0,5798	0,8281
8450	0,6397	0,8888
8500	0,7739	0,9622
8550	0,9544	1,0189
8600	1,0668	1,0166
8650	1,0384	0,9244
8700	0,9301	0,7677
8750	0,8222	0,6265
8800	0,7544	0,5558
8850	0,7387	0,5525
8900	0,7616	0,6127
8950	0,7953	0,7411
9000	0,8195	0,8916
9050	0,8289	0,9834
9100	0,8281	0,9876
9150	0,8236	0,9354
9200	0,8201	0,8681
9250	0,8179	0,8142
9300	0,8183	0,787
9350	0,8227	0,785
9400	0,8267	0,7971

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
9450	0,8212	0,8115
9500	0,8129	0,8214
9550	0,8175	0,8251
9600	0,8247	0,8248
9650	0,8198	0,8235
9700	0,8147	0,8224
9750	0,8181	0,8206
9800	0,8222	0,8198
9850	0,8222	0,8195
9900	0,8212	0,818
9950	0,8207	0,8169
10000	0,8186	0,8198

L.1.4 HHG pada Temperatur 200 K, Nitrogen dan Oksigen.

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
0	0,7242	0,7242
50	0,7282	0,7278
100	0,7732	0,769
150	0,8714	0,8674
200	0,9123	0,9316
250	0,8486	0,8932
300	0,7682	0,8121
350	0,7297	0,7489
400	0,7327	0,7292
450	0,7479	0,7392
500	0,7572	0,7537
550	0,7597	0,761
600	0,7597	0,7631
650	0,7596	0,7641
700	0,76	0,7642
750	0,7603	0,764
800	0,7604	0,764
850	0,7605	0,7646
900	0,7607	0,7653
950	0,7606	0,7651
1000	0,7616	0,7647
1050	0,7637	0,7651
1100	0,7601	0,7651
1150	0,7593	0,7653
1200	0,7625	0,7652
1250	0,7601	0,7652
1300	0,7604	0,7655

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
1350	0,7614	0,7651
1400	0,7613	0,7601
1450	0,7615	0,7563
1500	0,7613	0,7698
1550	0,7613	0,7719
1600	0,7613	0,7598
1650	0,7614	0,7663
1700	0,7615	0,77
1750	0,7613	0,7655
1800	0,7603	0,7647
1850	0,7595	0,7647
1900	0,7623	0,7648
1950	0,7733	0,7648
2000	0,7901	0,7652
2050	0,7967	0,765
2100	0,7665	0,7649
2150	0,7181	0,7649
2200	0,7059	0,7651
2250	0,7167	0,7657
2300	0,7547	0,7657
2350	0,7899	0,7651
2400	0,79	0,7649
2450	0,7748	0,7654
2500	0,7629	0,7666
2550	0,7591	0,7685
2600	0,7597	0,7704
2650	0,7606	0,767

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
2700	0,7609	0,7461
2750	0,7607	0,705
2800	0,7607	0,6673
2850	0,7605	0,6764
2900	0,7607	0,749
2950	0,7607	0,8641
3000	0,7607	0,9585
3050	0,7608	0,9371
3100	0,7598	0,8226
3150	0,7581	0,7169
3200	0,7618	0,6683
3250	0,7624	0,6853
3300	0,7594	0,7311
3350	0,762	0,7631
3400	0,7615	0,7715
3450	0,7606	0,7682
3500	0,7607	0,7655
3550	0,7606	0,7647
3600	0,7604	0,7644
3650	0,7603	0,7641
3700	0,76	0,7645
3750	0,7596	0,7651
3800	0,7596	0,7649
3850	0,7595	0,7643
3900	0,7572	0,764
3950	0,7483	0,7643
4000	0,7321	0,7648

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
4050	0,7237	0,7647
4100	0,7528	0,7644
4150	0,8294	0,7646
4200	0,9108	0,7644
4250	0,9025	0,7651
4300	0,7709	0,7691
4350	0,6467	0,7733
4400	0,6148	0,7622
4450	0,6749	0,7569
4500	0,7598	0,7699
4550	0,7966	0,7654
4600	0,789	0,7592
4650	0,7722	0,7647
4700	0,7637	0,7659
4750	0,7622	0,7656
4800	0,7626	0,7653
4850	0,7628	0,7652
4900	0,7624	0,7654
4950	0,762	0,7652
5000	0,7619	0,7649
5050	0,7618	0,7647
5100	0,7616	0,765
5150	0,7614	0,7646
5200	0,7625	0,7638
5250	0,7638	0,7632
5300	0,76	0,7631
5350	0,7596	0,7632

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
5400	0,7625	0,7623
5450	0,7597	0,7595
5500	0,7603	0,7536
5550	0,7612	0,7418
5600	0,761	0,7289
5650	0,7608	0,7319
5700	0,7609	0,7721
5750	0,7609	0,8489
5800	0,7609	0,9241
5850	0,7609	0,9283
5900	0,7609	0,8203
5950	0,7612	0,6832
6000	0,7623	0,6095
6050	0,7628	0,6165
6100	0,7594	0,694
6150	0,7487	0,7762
6200	0,7357	0,808
6250	0,7357	0,798
6300	0,7538	0,7774
6350	0,7904	0,7672
6400	0,8257	0,766
6450	0,8012	0,7668
6500	0,7575	0,7664
6550	0,7356	0,7664
6600	0,7348	0,7668
6650	0,7499	0,7665
6700	0,7615	0,7658

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
6750	0,7641	0,7657
6800	0,7628	0,7662
6850	0,7616	0,7663
6900	0,7614	0,7662
6950	0,7615	0,7659
7000	0,7616	0,7661
7050	0,7618	0,766
7100	0,7615	0,7662
7150	0,7616	0,7653
7200	0,7617	0,7617
7250	0,7617	0,7578
7300	0,7605	0,7667
7350	0,7592	0,7747
7400	0,7631	0,7618
7450	0,7631	0,7629
7500	0,7602	0,7719
7550	0,7631	0,7662
7600	0,7623	0,7643
7650	0,7615	0,7647
7700	0,7617	0,7646
7750	0,7618	0,7648
7800	0,762	0,7645
7850	0,7621	0,7647
7900	0,7624	0,7646
7950	0,7628	0,765
8000	0,7629	0,7647
8050	0,7632	0,7645

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
8100	0,7662	0,7643
8150	0,7754	0,7647
8200	0,7896	0,7653
8250	0,793	0,7651
8300	0,7571	0,764
8350	0,6781	0,7628
8400	0,6186	0,7626
8450	0,6451	0,7674
8500	0,7549	0,7846
8550	0,8873	0,8185
8600	0,9139	0,8558
8650	0,8341	0,8587
8700	0,7516	0,7908
8750	0,7206	0,6713
8800	0,7332	0,5878
8850	0,752	0,5889
8900	0,7597	0,6693
8950	0,7604	0,7969
9000	0,7597	0,8751
9050	0,7596	0,8633
9100	0,76	0,8107
9150	0,7604	0,7668
9200	0,7605	0,7541
9250	0,7606	0,7598
9300	0,7607	0,7658
9350	0,7609	0,7664
9400	0,7622	0,7656

Waktu (fs)	Amplitudo	
	Nitrogen	Oksigen
9450	0,7632	0,7658
9500	0,7595	0,7657
9550	0,7598	0,7651
9600	0,7626	0,7648
9650	0,7597	0,7651
9700	0,7607	0,7658
9750	0,7615	0,7655
9800	0,7613	0,7647
9850	0,7615	0,7649
9900	0,7613	0,7649
9950	0,7613	0,7651
10000	0,7614	0,7649

Lampiran 2. Source code program MATLAB

```
function varargout = revival_simulation_v2_0(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn',
    @revival_simulation_v2_0_OpeningFcn, ...
    'gui_OutputFcn',
    @revival_simulation_v2_0_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
end

function revival_simulation_v2_0_OpeningFcn(hObject, eventdata,
handles, varargin)
handles.output = hObject;
set(handles.edit_dir,'String',cd)
handles.energy=str2double(get(handles.edit_energy,'string'));
handles.focus=str2double(get(handles.edit_focus,'string'));
handles.deltat_pulse=str2double(get(handles.edit_deltat_pulse,'string'
));

handles.I0=handles.energy*1e-6/(handles.deltat_pulse*1e-
15)/(pi*(handles.focus/2*1e-4).^2);
set(handles.edit_I0,'string',sprintf('%0.2e',handles.I0))

handles.mode_integration='interaction_integration';
handles.ext_file_choice='radiobutton_none';
handles.coeff3D_selection='';
guidata(hObject, handles);
```

end

```
function varargout = revival_simulation_v2_0_OutputFcn(hObject,  
eventdata, handles)  
varargout{1} = handles.output;  
end
```

```
function push_simulation_Callback(hObject, eventdata, handles)
```

```
filename=[get(handles.edit_dir,'string') filesep  
get(handles.edit_filename,'string') '.mat'];  
if exist(filename,'file')==2  
    choice = questdlg('Overwrite file?','Existing file!','Yes','No','No');  
    switch choice  
        case 'No'  
            return  
    end  
end  
end
```

```
handles.hc_J_2_nm_m1=6.626069*2.99792*1e-17;  
handles.Wcm_m2_2_fs_m1_nm_m3=1e-12/(6.626069*2.99792);  
handles.c=2.99792e2;%nm/fs  
handles.hbar=1/(2*pi*handles.c); %nm-1fs  
handles.kb=1.380648881e-23/handles.hc_J_2_nm_m1;%nm-1K-1
```

```
I0=str2double(get(handles.edit_I0,'string'));%W/cm-2  
handles.I0=I0*handles.Wcm_m2_2_fs_m1_nm_m3;%fs-1nm-3
```

```
handles.B0=str2double(get(handles.edit_B0,'string'))*1e-7;%nm-1  
handles.D0=str2double(get(handles.edit_D0,'string'))*1e-7;%nm-1  
handles.delta_alpha_v=str2double(get(handles.edit_delta_alpha_v,'st  
ring'))*1e-3;%nm3
```

```
handles.deltat_pulse=str2double(get(handles.edit_deltat_pulse,'string  
')); %fs
```

```
handles.tmin=str2double(get(handles.edit_tmin,'string')); %start  
time grid fs  
handles.tmax=str2double(get(handles.edit_tmax,'string')); %end  
time grid fs
```



```

handles.res_temp=str2double(get(handles.edit_res_temp,'string'));
%resolution fs
handles.time_grid=(handles.tmin:handles.res_temp:handles.tmax);

handles.temperature=str2double(get(handles.edit_temperature,'string'
));;%Kelvin

handles.g_even=str2double(get(handles.edit_g_even,'string'));
handles.g_odd=str2double(get(handles.edit_g_odd,'string'));

handles.j_ini_max=str2double(get(handles.edit_j_ini_max,'string'));
state_parity='even';
handles.j_ini=(0:handles.j_ini_max);
handles.j_max=(0:handles.j_ini_max+10);
if get(handles.checkcos2,'value')==1

handles.cos2jm=zeros(length(handles.time_grid),length(handles.j_ini
),length(handles.j_ini));
end
if get(handles.checkcos4,'value')==1

handles.cos4jm=zeros(length(handles.time_grid),length(handles.j_ini
),length(handles.j_ini));
end
if get(handles.checkcos6,'value')==1

handles.cos6jm=zeros(length(handles.time_grid),length(handles.j_ini
),length(handles.j_ini));
end

switch handles.coeff3D_selection
    case 'button_coeffs_on_int'

handles.coeff=zeros(length(handles.j_ini),length(handles.j_ini),lengt
h(handles.time_grid),length(handles.j_max));
        case 'button_coeffs_on_ext'

handles.coeff=zeros(length(handles.j_ini),length(handles.time_grid),l
ength(handles.j_max));

```

end

h = waitbar(0,'Elapsed time');

tic

ind=0;

handles.n_tot=length(handles.j_ini)*(length(handles.j_ini)+1)/2;

for ind_j=1:length(handles.j_ini)

handles.m_ini=(0:handles.j_ini(ind_j));

for ind_m=1:ind_j

ind=ind+1;

waitbar(ind/handles.n_tot,h);

handles.atjm0=zeros(size(handles.j_max));

handles.atjm0(ind_j)=1 handles.m=handles.m_ini(ind_m);

switch handles.mode_integration

case 'full_integration'

atjm=full_integration(handles);

case 'interaction_integration'

atjm=interaction_integration(handles);

end

if handles.m==0

if isfield(handles,'cos2jm')

handles.cos2jm(:,ind_j,ind_m)=fcosinus2(atjm,handles);

end

if isfield(handles,'cos4jm')

handles.cos4jm(:,ind_j,ind_m)=fcosinus4(atjm,handles);

end

if isfield(handles,'cos6jm')

handles.cos6jm(:,ind_j,ind_m)=fcosinus6(atjm,handles);

end

else

if isfield(handles,'cos2jm')

handles.cos2jm(:,ind_j,ind_m)=2*fcosinus2(atjm,handles);

end

if isfield(handles,'cos4jm')

handles.cos4jm(:,ind_j,ind_m)=2*fcosinus4(atjm,handles);

end

if isfield(handles,'cos6jm')

```

        handles.cos6jm(:,ind_j,ind_m)=2*fccosinus6(atjm,handles);
    end
end
switch handles.coeff3D_selection
case 'button_coeffs_on_int'
    handles.coeff(ind_j,ind_m,:)=atjm;
case 'button_coeffs_on_ext'
    handles.coeff(ind_m,:)=atjm;
end
end
switch handles.coeff3D_selection
case 'button_coeffs_on_ext'
    coeff=handles.coeff;
    save([get(handles.edit_dir,'string') '\
get(handles.edit_filename,'string') '_coeff_J_ini='
num2str(handles.j_ini(ind_j)) '.mat'],'coeff');

end

end
toc
close(h)
handles=statistic(handles);
handles=Fourier_transform(handles);
fsaving(handles)
guidata(hObject,handles)
end

% --- Executes when selected object is changed in uipanel_int_type.
function uipanel_int_type_SelectionChangeFcn(hObject, eventdata, handles)
handles.int_type=get(eventdata.NewValue,'tag');

switch handles.int_type
case 'radiobutton_numerical'
    handles.mode_integration='full_integration';
case 'radiobutton_analytical'
    handles.mode_integration='interaction_integration';
end

```

```
guidata(hObject,handles)
```

```
end
```

```
function savedata=fsaving(handles)
```

```
savedata.I0_edit=str2double(get(handles.edit_I0,'string'));%W/cm-2  
savedata.B0_edit=str2double(get(handles.edit_B0,'string'));%nm-1  
savedata.D0_edit=str2double(get(handles.edit_D0,'string'));%nm-1  
savedata.delta_alpha_v_edit=str2double(get(handles.edit_delta_alpha_v,'string'));%A3
```

```
savedata.deltat_pulse=str2double(get(handles.edit_deltat_pulse,'string')); %fs  
savedata.tmin=str2double(get(handles.edit_tmin,'string')); %start time grid fs  
savedata.tmax=str2double(get(handles.edit_tmax,'string')); %end time grid fs  
savedata.res_temp=str2double(get(handles.edit_res_temp,'string')); %resolution fs  
savedata.time_grid=(handles.tmin:handles.res_temp:handles.tmax);  
savedata.temperature=str2double(get(handles.edit_temperature,'string'));%Kelvin  
savedata.g_even=str2double(get(handles.edit_g_even,'string'));  
savedata.g_odd=str2double(get(handles.edit_g_odd,'string'));  
savedata.hc_J_2_nm_m1=handles.hc_J_2_nm_m1;  
savedata.Wcm_m2_2_fs_m1_nm_m3=handles.Wcm_m2_2_fs_m1_nm_m3;  
savedata.c=handles.c;  
savedata.hbar=handles.hbar;  
savedata.kb=handles.kb;  
savedata.B0=handles.B0;  
savedata.D0=handles.D0;  
savedata.I0=handles.I0;  
savedata.delta_alpha_v=handles.delta_alpha_v;  
savedata.j_ini=handles.j_ini;  
savedata.j_max=handles.j_max;  
savedata.j_ini_max=str2double(get(handles.edit_j_ini_max,'string'));
```

```

savedata.n_tot=handles.n_tot;
savedata.m_ini=handles.m_ini;
savedata.m=handles.m;

if isfield(handles,'cos2jm')
    savedata.cos2jm=handles.cos2jm;
end
if isfield(handles,'cos2stat')
    savedata.cos2stat=handles.cos2stat;
end

if isfield(handles,'cos4jm')
    savedata.cos4jm=handles.cos4jm;
end
if isfield(handles,'cos4stat')
    savedata.cos4stat=handles.cos4stat;
end
if isfield(handles,'cos6jm')
    savedata.cos6jm=handles.cos6jm;
end
if isfield(handles,'cos6stat')
    savedata.cos6stat=handles.cos6stat;
end
if isfield(handles,'omega_unit_B0')
    savedata.omega_unit_B0=handles.omega_unit_B0;
end
if isfield(handles,'fftcos2stat')
    savedata.fftcos2stat=handles.fftcos2stat;
end
if isfield(handles,'fftcos4stat')
    savedata.fftcos4stat=handles.fftcos4stat;
end
if isfield(handles,'fftcos6stat')
    savedata.fftcos6stat=handles.fftcos6stat;
end
switch handles.coeff3D_selection
    case 'button_coeffs_on_int'
        savedata.coeff=handles.coeff;
end
if isfield(handles,'theta')

```

```

    savedata.theta=handles.theta;
end
if isfield(handles,'proba_theta_stat')
    savedata.proba_theta_stat=handles.proba_theta_stat;
end
if isfield(handles,'proba_theta_j')
    savedata.proba_theta_j=handles.proba_theta_j;
end
if isfield(handles,'proba_theta_stat_norm')
    savedata.proba_theta_stat_norm=handles.proba_theta_stat_norm;
end
if isfield(handles,'proba_theta_j_norm')
    savedata.proba_theta_j_norm=handles.proba_theta_j_norm;
end
if isfield(handles,'proba_theta_stat_integrated')

savedata.proba_theta_stat_integrated=handles.proba_theta_stat_integ
rated;
end
if isfield(handles,'proba_theta_stat_norm_integrated')

savedata.proba_theta_stat_norm_integrated=handles.proba_theta_sta
t_norm_integrated;
end

save([get(handles.edit_dir,'string') '\
get(handles.edit_filename,'string') '.mat'],savedata');
end
function r = isfieldne(A, field)
r = false;
if ~isfield(A, field) return; end
r = ~isempty(eval(['A.', field]));
end
function N = randomfigurenumber()
N=round(abs(randn(1)*1e9));
if N >= 2147483646
    while N >= 2147483646
        N=round(abs(randn(1)*1e9));
    end
end
end

```

```

end
function SpinBox(hObject, eventdata, props)
handles = guidata(hObject);
if ~exist('props', 'var') props = struct(); end
if ~isfieldne(props, 'SmallStep') props.SmallStep = 1; end
if ~isfieldne(props, 'BigStep') props.BigStep = 10; end
if ~isfieldne(props, 'Min') props.Min = -Inf; end
if ~isfieldne(props, 'Max') props.Max = Inf; end
if ~isfieldne(props, 'Callback') props.Callback = []; end
if ~isfieldne(props, 'CallCallback') props.CallCallback = true; end
if ~isfieldne(props, 'Integer') props.Integer = false; end
c = str2double(get(hObject, 'String'));
if isempty(c) c = 0; end
CallCallback = false;
switch eventdata.Key
    case 'uparrow'
        c = c + props.SmallStep;
        CallCallback = true;
    case 'pageup'
        c = c + props.BigStep;
        CallCallback = true;
    case 'downarrow'
        c = c - props.SmallStep;
        CallCallback = true;
    case 'pagedown'
        c = c - props.BigStep;
        CallCallback = true;
end
c = max(min(c, props.Max), props.Min);
if props.Integer
    c = round(c);
end
set(hObject, 'String', c);
if ~isempty(props.Callback)
    props.Callback(hObject, eventdata, handles);
end
if props.CallCallback & CallCallback
    Callback = get(hObject, 'Callback');
    Callback(hObject, eventdata);
end
end

```

```

end
function y=normalise(x)
%y=normalise(x) normalise x to its maximum value
y=x/max(max(max(abs(x)))));
end
function number =
figure_docked(name,dockedyesno,imageviewer,pulseviewer)
if ~exist('dockedyesno','var')
    dockedyesno=1;
end
if ~exist('imageviewer','var')
    imageviewer=0;
end
if ~exist('pulseviewer','var')
    pulseviewer=0;
end

switch dockedyesno
    case 1
        windowst = 'docked';
    case 0
        windowst = 'normal';
end

number = findobj('name',name);
if isempty(number)
    number = randomfigurenumber;
    if imageviewer
        number = ImageViewer;
        set(number,'name',name,'windowstyle',windowst)
    elseif pulseviewer
        number = PulseViewer;
        set(number,'name',name,'windowstyle',windowst)
    else
        figure(number);
        clf;
        set(number,'name',name,'numbertitle','off')
        set(number, 'windowstyle',windowst)
    end
end
end

```



```
figure(number)
set(number,'windowstyle',windowst)
```

```
end
function E_eV=cm2eV(E_cm)
Cb=1.602176e-19; %coulomb
h=6.626068e-34; %J.s
c=2.997924586e8; %m.s-1
E_nm=1./(E_cm*1e2)*1e9;E_freq=c/E_nm*1e9;
E_J=E_freq*h;
E_eV=E_J/Cb;
end
```

```
function X=ft(x,dim)
if ~exist('dim','var')
    dim=2;
end
X=fftshift(fft(fftshift(x,dim),[],dim),dim);
end
```

```
function l = eV2nm(E)
eV = 1.6022e-019;c = 299792458;h = 6.6261e-034;nm = 1E-9;
l = (h*c)./(E*nm*eV);
end
```

```
function handles=statistic(handles)
%% statistic
handles=fcosstat(handles);
figure_docked('cos2stat');clf;hold on
plot(handles.time_grid,abs(handles.cos2stat),'r')
```

```
leg={'cos^2\theta'};
if isfield(handles,'cos4jm')
    plot(handles.time_grid,abs(handles.cos4stat),'k')
    leg=[leg;{'cos^4\theta'}];
end
if isfield(handles,'cos6jm')
    plot(handles.time_grid,abs(handles.cos6stat),'b')
```

```

leg=[leg;{'cos^6\theta'}];
end

[m,n]=size(handles.cos2stet);
a=zeros(m,n);
for i =1:m
    for j=1:n
        a(i,j)= str2double(get(handles.Konstanta1,'string'));
    end
end

%disp(sprintf('jumlah %d%',length(abs(handles.cos2stet))));
handles.HHG = 0.7*(a) + 0.5*abs(handles.cos2stet)+
0.8*abs(handles.cos4stet)+ 0.4*abs(handles.cos6stet)
plot(handles.time_grid, handles.HHG , 'm')
leg=[leg;{'HHG'}];
handles.HHG
legend(leg)
xlabel('time (fs)')
ylabel('Amplitude of <cos^2(\theta)>')
%ylim([0 2])
end

function handles=Fourier_transform(handles)
handles=fftcostat(handles);
figure_docked('fftcostat');clf;hold on
plot(handles.omega_unit_B0,abs(handles.fftcostat),'r')
leg={'cos^2\theta'};
if isfield(handles,'cos4jm')
    plot(handles.omega_unit_B0,abs(handles.fftcostat),'k')
    leg=[leg;{'cos^4\theta'}];
end
if isfield(handles,'cos6jm')
    plot(handles.omega_unit_B0,abs(handles.fftcostat),'b')
    leg=[leg;{'cos^6\theta'}];
end
[m,n]=size(handles.fftcostat);
a=zeros(m,n);
for i =1:m

```

```

for j=1:n
    a(i,j)= str2double(get(handles.Konstanta1,'string'));
end
end

handles.FHHG = 0.7*(a) + 0.5*abs(handles.fftcos2stat)+
0.8*abs(handles.fftcos4stat)+ 0.4*abs(handles.fftcos6stat);
plot(handles.omega_unit_B0,handles.FHHG , 'm')
leg=[leg;{'HHG'}];
legend(leg)
xlabel('energy (B_0)')
ylabel('Amplitude of the fft of <cos^2(\theta)>')
xlim([3 4*(4*handles.j_ini_max+6)])
grid on
%handles.FHHG

end

```

```

function handles=fftcosstat(handles)
N=length(handles.time_grid);
handles.omega=2*pi/max(handles.time_grid).*(-N/2+1/2:N/2-1/2);
handles.omega_unit_B0=handles.omega./(2*pi*handles.c/eV2nm(c
m2eV(handles.B0*1e7)));
handles.fftcos2stat=ft(handles.cos2stat,1);
if isfield(handles,'cos4jm')
    handles.fftcos4stat=ft(handles.cos4stat,1);
end
if isfield(handles,'cos6jm')
    handles.fftcos6stat=ft(handles.cos6stat,1);
end
end
end

```

```

function handles=fcosstat(handles)
gj=mod(handles.j_ini,2).*handles.g_odd;
gj=gj+(1-mod(handles.j_ini,2)).*handles.g_even;
Pj_norm=gj.*(2*handles.j_ini+1).*exp(-
fHzero(handles.j_ini,handles)./(handles.kb.*handles.temperature));
handles.Pj=gj.*exp(-
fHzero(handles.j_ini,handles)./(handles.kb.*handles.temperature));
if isfield(handles,'cos2jm')

```

```
handles.cos2stet=str2double(get(handles.Konstanta2,'string'))*(sum((
ones(length(handles.time_grid),1)*handles.Pj').*sum(handles.cos2jm
,3),2)./sum(Pj_norm));
```

```
    %handles.cos2stet=0.5*handles.cos2stat
```

```
handles.cos2stat=sum((ones(length(handles.time_grid),1)*handles.Pj
').*sum(handles.cos2jm,3),2)./sum(Pj_norm);
```

```
end
```

```
if isfield(handles,'cos4jm')
```

```
handles.cos4stet=str2double(get(handles.Konstanta3,'string'))*(sum((
ones(length(handles.time_grid),1)*handles.Pj').*sum(handles.cos4jm
,3),2)./sum(Pj_norm));
```

```
    %handles.cos4stet=0.8*handles.cos4stat
```

```
handles.cos4stat=sum((ones(length(handles.time_grid),1)*handles.Pj
').*sum(handles.cos4jm,3),2)./sum(Pj_norm);
```

```
end
```

```
if isfield(handles,'cos6jm')
```

```
handles.cos6stet=str2double(get(handles.Konstanta4,'string'))*(sum((
ones(length(handles.time_grid),1)*handles.Pj').*sum(handles.cos6jm
,3),2)./sum(Pj_norm));
```

```
    %handles.cos6stet=0.4*handles.cos6stat;
```

```
handles.cos6stat=sum((ones(length(handles.time_grid),1)*handles.Pj
').*sum(handles.cos6jm,3),2)./sum(Pj_norm);
```

```
end
```

```
end
```

```
function atjm=full_integration(handles)
```

```
options = odeset('RelTol',1e-5);
```

```
% fhandles('handles',handles);
```

```
[time,atjm_interaction] = ode45(@eq_schrodinger,[0
handles.tmax],handles.atjm0,options);
```

```
atjm=interp1(time,atjm_interaction,handles.time_grid,'spline',0);
```

```
%interpolation of
```

```
%the values on a fixed grid of time predefined
```

```
    function datjm=eq_schrodinger(time,atjm)
```

```

    % carefull I used circshift to create the vector atjp2m which brings
    % boudaries values on the other edge (which is fine if they have
similar
    % values) otherwise I will have to specify 0 for the 2 boudaries
value
    % being moved
    atjp2m=circshift(atjm,-2);atjp2m(end-1:end)=0;
    atjm2m=circshift(atjm,2);atjm2m(1:2)=0;
    if time<=2*handles.deltat_pulse
        datjm=-1i/handles.hbar*((fHzero(handles.j_max,handles)-
Uzero(time,handles).*fCjgm(handles.j_max,handles.m)).*atjm...
-
Uzero(time,handles).*fCjgp2m(handles.j_max,handles.m)).*atjp2m...
-
Uzero(time,handles).*fCjgm2m(handles.j_max,handles.m)).*atjm2m);
    else
        datjm=-1i/handles.hbar.*fHzero(handles.j_max,handles).*atjm;
    end
end
end
end

```

```

function atjm=interaction_integration(handles)
options = odeset('RelTol',1e-5);
[time,atjm_interaction] = ode45(@eq_schrodinger,[0
2*handles.deltat_pulse],handles.atjm0,options);
atjm=interp1(time,atjm_interaction,handles.time_grid,'linear',0);%int
erpolation of
%the values on a fixed grid of time predefined up to t=interaction
time
atjm_ini_free=interp1(time,atjm_interaction,2*handles.deltat_pulse,'l
inear',0); %inital condition of the free evolution at t=interaction time
atjm=atjm+(ones(size(handles.j_max)).*(handles.time_grid>2*handle
s.deltat_pulse)).*...
    (atjm_ini_free'*ones(size(handles.time_grid))).*...

exp(+1i/handles.hbar.*fHzero(handles.j_max,handles).*(handles.time
_grid-2*handles.deltat_pulse));%analytical integration during the
free evolution

```

```

function datjm=eq_schrodinger(time,atjm)

```

```

    atjp2m=circshift(atjm,-2);atjp2m(end-1:end)=0;
    atjm2m=circshift(atjm,2);atjm2m(1:2)=0;
    datjm=-1i/handles.hbar*((fHzero(handles.j_max,handles)-
    Uzero(time,handles).*fCjjm(handles.j_max,handles.m)).*atjm...
    -
    Uzero(time,handles).*fCjip2m(handles.j_max,handles.m).*atjp2m...
    -
    Uzero(time,handles).*fCjjm2m(handles.j_max,handles.m).*atjm2m);
end
end

```

```

function handles=revival_3D_calculation(handles)
h = waitbar(0,'Elapsed time');
theta_step=str2double(get(handles.edit_theta_step,'string'));
theta_min=str2double(get(handles.edit_theta_min,'string'));
theta_max=str2double(get(handles.edit_theta_max,'string'));
handles.theta=(theta_min:theta_step:theta_max)*pi/180;
ind=0;
handles.proba_theta_j=zeros(length(handles.time_grid),length(handles.theta),length(handles.j_ini));
handles.proba_theta_j_norm=zeros(length(handles.time_grid),length(handles.theta),length(handles.j_ini));

```

```

switch handles.ext_file_choice
case 'radiobutton_sym_function'
    fh=str2func(get(handles.edit_fexttheta,'string'));
    handles.observable=fh(handles.theta);
    %this case is set for Legendre sym function
    m=0+1;%this is for m=0 in the legendre function
    if size(handles.observable,1)>1
        handles.observable=handles.observable(m,:);
    end
case 'radiobutton_extangdistri'

handles.observable=interp1(handles.theta_ext,handles.ang_distri_ext,handles.theta);
otherwise
    handles.observable=ones(size(handles.theta));
end

```

```

for ind_j=1:length(handles.j_ini)
    handles.m_ini=(0:handles.j_ini(ind_j));
    switch handles.coeff3D_selection %how to save the coeffs atjm to
    proceed with qunatum calculation later on. int means all are saved in
    one variable coeff. For many j state can easily be out of memory.
    Need to save them as files externally
        case 'button_coeffs_on_ext'
            if isfield(handles,'filename_loaded')
                filepath=handles.filename_loaded;
            else
                filepath=[get(handles.edit_dir,'string') filesep
get(handles.edit_filename,'string') '.mat'];
            end
            load([filepath(1:end-4) '_coeff_J_ini='
num2str(handles.j_ini(ind_j))]);
            handles.coeff=coeff;
        end

    for ind_m=1:ind_j
        ind=ind+1;
        waitbar(ind/handles.n_tot,h);
        handles.m=handles.m_ini(ind_m);
        switch handles.coeff3D_selection %how to save the coeffs atjm
        to proceed with qunatum calculation later on. int means all are saved
        in one variable coeff. For many j state can easily be out of memory.
        Need to save them as files externally
            case 'button_coeffs_on_ext'
                handles.atjm=squeeze(handles.coeff(ind_m,:,:));
            otherwise
                handles.atjm=squeeze(handles.coeff(ind_j,ind_m,:,:));
            end
            [angular_proba,angular_proba_norm]=fangular_proba(handles);
            if handles.m==0

handles.proba_theta_j(:, :, ind_j)=handles.proba_theta_j(:, :, ind_j)+ang
ular_proba;

handles.proba_theta_j_norm(:, :, ind_j)=handles.proba_theta_j_norm(
,:, ind_j)+angular_proba_norm;
            else

```

```
handles.proba_theta_j(:,ind_j)=handles.proba_theta_j(:,ind_j)+2*angular_proba;
```

```
handles.proba_theta_j_norm(:,ind_j)=handles.proba_theta_j_norm(:,ind_j)+2*angular_proba_norm;
```

```
end
```

```
end
```

```
end
```

```
close(h)
```

```
handles=fproba_stat(handles);
```

```
end
```

```
function handles=fproba_stat(handles)
```

```
gj=mod(handles.j_ini,2).*handles.g_odd;
```

```
gj=gj+(1-mod(handles.j_ini,2)).*handles.g_even;
```

```
Pj_norm=gj.*(2*handles.j_ini+1).*exp(-
```

```
fHzero(handles.j_ini,handles)./(handles.kb.*handles.temperature));
```

```
handles.Pj=gj.*exp(-
```

```
fHzero(handles.j_ini,handles)./(handles.kb.*handles.temperature));
```

```
handles.proba_theta_stat=zeros(length(handles.time_grid),length(handles.theta));
```

```
handles.proba_theta_stat_norm=zeros(length(handles.time_grid),length(handles.theta));
```

```
for ind_theta=1:length(handles.theta)
```

```
handles.proba_theta_stat(:,ind_theta)=sum((ones(length(handles.time_grid),1)*handles.Pj').*squeeze(handles.proba_theta_j(:,ind_theta,:)),2)./sum(Pj_norm);
```

```
handles.proba_theta_stat_norm(:,ind_theta)=sum((ones(length(handles.time_grid),1)*handles.Pj').*squeeze(handles.proba_theta_j_norm(:,ind_theta,:)),2)./sum(Pj_norm);
```

```
end
```

```
%integration over theta for each time
```

```
handles.proba_theta_stat_integrated=zeros(size(handles.time_grid));
```

```
handles.proba_theta_stat_norm_integrated=zeros(size(handles.time_grid));
```



```

for ind_time=1:length(handles.time_grid)

handles.proba_theta_stat_integrated(ind_time)=trapz(handles.theta,h
andles.proba_theta_stat(ind_time,:).*sin(handles.theta));

handles.proba_theta_stat_norm_integrated(ind_time)=trapz(handles.t
heta,handles.proba_theta_stat_norm(ind_time,:).*sin(handles.theta));
end
handles.proba_theta_stat_integrated=handles.proba_theta_stat_integr
ated./mean(handles.proba_theta_stat_norm_integrated);

end

function
[angular_proba,angular_proba_norm]=fangular_proba(handles)
Yjm=spherical_harmo(handles);
angular_proba=zeros(length(handles.time_grid),length(handles.theta)
);
angular_proba_norm=zeros(length(handles.time_grid),length(handle
s.theta));

for ind_theta=1:length(handles.theta)

angular_proba(:,ind_theta)=sum(handles.atjm.*(ones(length(handles.
time_grid),1)*Yjm(:,ind_theta')),2).*handles.observable(ind_theta).*
sum(conj(handles.atjm).*(ones(length(handles.time_grid),1)*Yjm(:,i
nd_theta')),2);

angular_proba_norm(:,ind_theta)=sum(handles.atjm.*(ones(length(h
andles.time_grid),1)*Yjm(:,ind_theta')),2).*sum(conj(handles.atjm).
*(ones(length(handles.time_grid),1)*Yjm(:,ind_theta')),2);
end
end

function Yjm=spherical_harmo(handles)
costheta=cos(handles.theta);
m=handles.m;% mis here always positive because
atjm_ned=atjm_pos but we have to separate the cases here
Yjm=zeros(length(handles.j_max),length(costheta));

```

```

for ind_j=m+1:length(handles.j_max)
    j=handles.j_max(ind_j);
    leg=legendre(j,costheta);
    %the variation with phi is cancelled in the quantum calculation as
    well
    %as the prefactor  $(-1)^m$  for m negative.  $(-1)^m * (-1)^m = 1$ 
    %and  $\exp(1i*m*\phi)$  vanishes with its complex conjugate
    Yjm(ind_j,:)=sqrt((2*j+1). *factorial(j-
m)./(4*pi*factorial(j+m))).*leg(m+1,:);
end
end

```

```

% function cos2jm=fcosinus2(atjm,handles) %%with use of Cjjm
coeffs not general
% atjp2m=circshift(atjm,[0 -2]);atjp2m(:,end-1:end)=0;
%
cos2jm=sum(abs(atjm).^2.*(ones(size(atjm,1),1)*fCjjm(handles.j_m
ax,handles.m)')+...
%
2*real(conj(atjm). *atjp2m.*(ones(size(atjm,1),1)*fCjjp2m(handles.j
_max,handles.m)'))),2);
% end

```

```

function cos2jm=fcosinus2(atjm,handles) %%with use of general
coeffs
atjp2m=circshift(atjm,[0 -2]);atjp2m(:,end-1:end)=0;
j=handles.j_max;m=handles.m;
unitary=ones(size(atjm,1),1);
cos2jm=sum(...
    abs(atjm).^2.*(unitary*(coeffm1(j,m). *coeffp1(j-
1,m)+coeffp1(j,m). *coeffm1(j+1,m)'))+...
2*real(conj(atjm). *atjp2m).*(unitary*(coeffm1(j+2,m). *coeffm1(j+1
,m)')))...
    ,2);
end

```

```

function cos4jm=fcosinus4(atjm,handles) %%with use of general
coeffs
atjp2m=circshift(atjm,[0 -2]);atjp2m(:,end-1:end)=0;

```

```

atjp4m=circshift(atjm,[0 -4]);atjp4m(:,end-3:end)=0;
j=handles.j_max;m=handles.m;
unitary=ones(size(atjm,1),1);
cos4jm=sum(...
    abs(atjm).^2.*(unitary*...

(coeffp1(j,m).*coeffp1(j+1,m).*coeffm1(j+2,m).*coeffm1(j+1,m)+...
    coeffm1(j,m).*coeffm1(j-1,m).*coeffp1(j-2,m).*coeffp1(j-
1,m))+...

coeffp1(j,m).*coeffm1(j+1,m).*coeffp1(j,m).*coeffm1(j+1,m)+...
    coeffm1(j,m).*coeffp1(j-1,m).*coeffm1(j,m).*coeffp1(j-
1,m))+...
    coeffp1(j,m).*coeffm1(j+1,m).*coeffm1(j,m).*coeffp1(j-
1,m))+...
    coeffm1(j,m).*coeffp1(j-
1,m).*coeffp1(j,m).*coeffm1(j+1,m))')+...
    ...
    2*real(conj(atjm).*atjp4m).*(unitary*...

(coeffm1(j+4,m).*coeffm1(j+3,m).*coeffm1(j+2,m).*coeffm1(j+1,m
))')+...
    ...
    2*real(conj(atjm).*atjp2m).*(unitary*...

(coeffp1(j+2,m).*coeffm1(j+3,m).*coeffm1(j+2,m).*coeffm1(j+1,m)
+...
    coeffm1(j+2,m).*coeffm1(j+1,m).*coeffm1(j,m).*coeffp1(j-
1,m))+...

coeffm1(j+2,m).*coeffp1(j+1,m).*coeffm1(j+2,m).*coeffm1(j+1,m)
+...

coeffm1(j+2,m).*coeffm1(j+1,m).*coeffp1(j,m).*coeffm1(j+1,m))'..
.
    ,2);
end

function cos6jm=fcosinus6(atjm,handles) %%with use of general
coeffs

```

```

atjp2m=circshift(atjm,[0 -2]);atjp2m(:,end-1:end)=0;
atjp4m=circshift(atjm,[0 -4]);atjp4m(:,end-3:end)=0;
atjp6m=circshift(atjm,[0 -6]);atjp6m(:,end-5:end)=0;
j=handles.j_max;m=handles.m;
unitary=ones(size(atjm,1),1);
cos6jm=sum(...
    abs(atjm).^2.*(unitary*...

(coeffp1(j,m).*coeffp1(j+1,m).*coeffp1(j+2,m).*coeffm1(j+3,m).*c
oeffm1(j+2,m).*coeffm1(j+1,m))+...

coeffp1(j,m).*coeffp1(j+1,m).*coeffm1(j+2,m).*coeffp1(j+1,m).*co
effm1(j+2,m).*coeffm1(j+1,m))+...

coeffp1(j,m).*coeffp1(j+1,m).*coeffm1(j+2,m).*coeffm1(j+1,m).*co
effp1(j,m).*coeffm1(j+1,m))+...

coeffp1(j,m).*coeffp1(j+1,m).*coeffm1(j+2,m).*coeffm1(j+1,m).*co
effm1(j,m).*coeffp1(j-1,m))+...

coeffp1(j,m).*coeffm1(j+1,m).*coeffp1(j,m).*coeffp1(j+1,m).*coeff
m1(j+2,m).*coeffm1(j+1,m))+...

coeffp1(j,m).*coeffm1(j+1,m).*coeffp1(j,m).*coeffm1(j+1,m).*coef
fp1(j,m).*coeffm1(j+1,m))+...

coeffp1(j,m).*coeffm1(j+1,m).*coeffp1(j,m).*coeffm1(j+1,m).*coef
fm1(j,m).*coeffp1(j-1,m))+...
    coeffp1(j,m).*coeffm1(j+1,m).*coeffm1(j,m).*coeffp1(j-
1,m).*coeffp1(j,m).*coeffm1(j+1,m))+...
    coeffp1(j,m).*coeffm1(j+1,m).*coeffm1(j,m).*coeffp1(j-
1,m).*coeffm1(j,m).*coeffp1(j-1,m))+...
    coeffp1(j,m).*coeffm1(j+1,m).*coeffm1(j,m).*coeffm1(j-
1,m).*coeffp1(j-2,m).*coeffp1(j-1,m))+...
    ...
    coeffm1(j,m).*coeffm1(j-1,m).*coeffm1(j-2,m).*coeffp1(j-
3,m).*coeffp1(j-2,m).*coeffp1(j-1,m))+...
    coeffm1(j,m).*coeffm1(j-1,m).*coeffp1(j-2,m).*coeffm1(j-
1,m).*coeffp1(j-2,m).*coeffp1(j-1,m))+...

```

$$\begin{aligned}
& \text{coeffm1}(j,m).\text{coeffm1}(j-1,m).\text{coeffp1}(j-2,m).\text{coeffp1}(j-1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffm1}(j-1,m).\text{coeffp1}(j-2,m).\text{coeffp1}(j-1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffm1}(j,m).\text{coeffm1}(j-1,m).\text{coeffp1}(j-2,m).\text{coeffp1}(j-1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m)+\dots \\
& \text{coeffm1}(j,m).\text{coeffp1}(j-1,m).\text{coeffp1}(j,m).\text{coeffp1}(j+1,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m)))+\dots
\end{aligned}$$

$$\begin{aligned}
& \dots \\
& 2*\text{real}(\text{conj}(\text{atjm}).*\text{atjp2m}).*(\text{unitary}*\dots
\end{aligned}$$

$$(\text{coeffp1}(j+2,m).\text{coeffp1}(j+3,m).\text{coeffm1}(j+4,m).\text{coeffm1}(j+3,m).\text{coeffm1}(j+2).\text{coeffm1}(j+1,m),m)+\dots$$

$$\text{coeffp1}(j+2,m).\text{coeffm1}(j+3,m).\text{coeffp1}(j+2,m).\text{coeffm1}(j+3,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m)+\dots$$

$$\text{coeffp1}(j+2,m).\text{coeffm1}(j+3,m).\text{coeffm1}(j+2,m).\text{coeffp1}(j+1,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m)+\dots$$

$$\text{coeffp1}(j+2,m).\text{coeffm1}(j+3,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m).\text{coeffp1}(j,m).\text{coeffm1}(j+1,m)+\dots$$

$$\text{coeffp1}(j+2,m).\text{coeffm1}(j+3,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m).\text{coeffm1}(j,m).\text{coeffp1}(j-1,m)+\dots$$

$$\text{coeffm1}(j+2,m).\text{coeffp1}(j+1,m).\text{coeffm1}(j+2,m).\text{coeffp1}(j+1,m).\text{coeffm1}(j+2,m).\text{coeffm1}(j+1,m)+\dots$$

$\text{coeffm1}(j+2,m) \cdot \text{coeffp1}(j+1,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffp1}(j,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffp1}(j+1,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j-1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffp1}(j+1,m) \cdot \text{coeffp1}(j+2,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffm1}(j-1,m) \cdot \text{coeffp1}(j-2,m) \cdot \text{coeffp1}(j-1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j-1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j-1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j-1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j+1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffp1}(j,m) \cdot \text{coeffp1}(j+1,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffp1}(j,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffm1}(j,m) \cdot \text{coeffp1}(j-1,m) + \dots$

$\text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffp1}(j,m) \cdot \text{coeffm1}(j+1,m) \cdot \text{coeffp1}(j,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\dots$

$2 \cdot \text{real}(\text{conj}(\text{atjm}) \cdot \text{atjp4m}) \cdot (\text{unitary} \cdot \dots$

$(\text{coeffp1}(j+4,m) \cdot \text{coeffm1}(j+5,m) \cdot \text{coeffm1}(j+4,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+4,m) \cdot \text{coeffp1}(j+3,m) \cdot \text{coeffm1}(j+4,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+4,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffp1}(j+2,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

$\text{coeffm1}(j+4,m) \cdot \text{coeffm1}(j+3,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffp1}(j+1,m) \cdot \text{coeffm1}(j+2,m) \cdot \text{coeffm1}(j+1,m) + \dots$

```

coeffm1(j+4,m).*coeffm1(j+3,m).*coeffm1(j+2,m).*coeffm1(j+1,m)
.*coeffp1(j,m).*coeffm1(j+1,m)+...

coeffm1(j+4,m).*coeffm1(j+3,m).*coeffm1(j+2,m).*coeffm1(j+1,m)
.*coeffm1(j,m).*coeffp1(j-1,m))')+...

...
2*real(conj(atjm).*atjp6m).*(unitary*...

(coeffm1(j+6,m).*coeffm1(j+5,m).*coeffm1(j+4,m).*coeffm1(j+3,m)
).*coeffm1(j+2,m).*coeffm1(j+1,m))')...
,2);
end

function out=fHzero(j_ini,handles)
out=handles.B0.*j_ini.*(j_ini+1)-handles.D0.*(j_ini.*(j_ini+1)).^2;
end

function out=coeffp1(j,m)
out=sqrt(((j+m+1).*(j-m+1))./((2*j+1).*(2*j+3)));
end

function out=coeffm1(j,m) %to be noted: coeffm1(j,m)=coeffp1(j-
1,m)
out=sqrt(((j+m).*(j-m))./((2*j+1).*(2*j-1)));
end

function out=fCjjm(j,m)
out=(j+m+1).*(j-m+1)./((2*j+3).*(2*j+1))+(j+m).*(j-m)./((2*j-
1).*(2*j+1));
end

function out=fCjpp2m(j,m)
out=sqrt(((j+m+2).*(j-m+2).*(j+1+m).*(j+1-
m))./((2*j+5).*(2*j+3).*(2*j+3).*(2*j+1)));
end

function out=fCjjm2m(j,m)
out=sqrt(((j+m-1).*(j-m-1).*(j+m).*(j-m))./((2*j-3).*(2*j-1).*(2*j-
1).*(2*j+1)));

```

end

```
function out=Uzero(time,handles)
time=time.*(time<=2*handles.deltat_pulse);
out=2*pi/handles.c*handles.delta_alpha_v*handles.I0*sin(pi*time/(
2*handles.deltat_pulse)).^2;
end
```

```
function push_fft_Callback(hObject, eventdata, handles)
% hObject    handle to push_fft (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
handles=Fourier_transform(handles);
guidata(hObject,handles)
end
```

```
function push_3D_Callback(hObject, eventdata, handles)
% hObject    handle to push_3D (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
switch handles.coeff3D_selection %how to save the coeffs atjm to
proceed with qunatum calculation later on. int means all are saved in
one variable coeff. For many j state can easily be out of memory.
Need to save them as files externally
case 'button_coeffs_off'
    helpdlg('You havent saved all the rotational coefficients or
maybe there are saved externally!');
otherwise
    handles=revival_3D_calculation(handles);
    guidata(hObject,handles)
    fsaving(handles);
    structure.time_grid=handles.time_grid;
    structure.theta=handles.theta;
    structure.proba_theta_stat=handles.proba_theta_stat;

structure.proba_theta_stat_norm=handles.proba_theta_stat_norm;
```



```

structure.proba_theta_stat_integrated=handles.proba_theta_stat_integrated;

structure.proba_theta_stat_norm_integrated=handles.proba_theta_stat_norm_integrated;
end
switch handles.coeff3D_selection
    case 'button_coeffs_on_int'
        MF=findobj('name', '3D_revival:internal');
        if isempty(MF);
            revival_3D('Name','3D_revival:internal');structure.nameGUI='3D_revival:internal'; MF=findobj('name', '3D_revival:internal');end
        figure(MF)
        revival_3D('setdatafrommain',structure);
    case 'button_coeffs_on_ext'
        MF=findobj('name', '3D_revival:external');
        if isempty(MF);
            revival_3D('Name','3D_revival:external');structure.nameGUI='3D_revival:external'; MF=findobj('name', '3D_revival:external');end
        figure(MF)
        revival_3D('setdatafrommain',structure);
end
end

function edit_B0_Callback(hObject, eventdata, handles)
% hObject    handle to edit_B0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_B0 as text
%         str2double(get(hObject,'String')) returns contents of edit_B0
%         as a double
end

function edit_B0_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_B0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```

```

% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%    See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject, 'KeyPressFcn', @(hObject, eventdata)
SpinBox(hObject, eventdata, ...
    struct('SmallStep', 0.1, 'BigStep', 1)));
end

function edit_D0_Callback(hObject, eventdata, handles)
% hObject    handle to edit_D0 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_D0 as text
%    str2double(get(hObject,'String')) returns contents of edit_D0
as a double
end

function edit_D0_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_D0 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%    See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
end

```

```

function edit_delta_alpha_v_Callback(hObject, eventdata, handles)
% hObject   handle to edit_delta_alpha_v (see GCBO)
% eventdata reserved - to be defined in a future version of
MATLAB
% handles   structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_delta_alpha_v
as text
%         str2double(get(hObject,'String')) returns contents of
edit_delta_alpha_v as a double

end

```

```

function edit_delta_alpha_v_CreateFcn(hObject, eventdata, handles)
% hObject   handle to edit_delta_alpha_v (see GCBO)
% eventdata reserved - to be defined in a future version of
MATLAB
% handles   empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject, 'KeyPressFcn', @(hObject, eventdata)
SpinBox(hObject, eventdata, ...
    struct('SmallStep', 0.01, 'BigStep', 0.1)));
end

```

```

function edit_g_even_Callback(hObject, eventdata, handles)
% hObject   handle to edit_g_even (see GCBO)
% eventdata reserved - to be defined in a future version of
MATLAB
% handles   structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_g_even as text
%         str2double(get(hObject,'String')) returns contents of
edit_g_even as a double

```

end

```
function edit_g_even_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_g_even (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject, 'KeyPressFcn', @(hObject, eventdata)
SpinBox(hObject, eventdata, ...
    struct('SmallStep', 1, 'BigStep', 10)));
end
```

```
function edit_g_odd_Callback(hObject, eventdata, handles)
% hObject    handle to edit_g_odd (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_g_odd as text
%     str2double(get(hObject,'String')) returns contents of
edit_g_odd as a double
end
```

```
function edit_g_odd_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_g_odd (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject, 'KeyPressFcn', @(hObject, eventdata)
SpinBox(hObject, eventdata, ...
    struct('SmallStep', 1, 'BigStep', 10)));
end

```

```

function edit_temperature_Callback(hObject, eventdata, handles)
% hObject    handle to edit_temperature (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_temperature as
text
%          str2double(get(hObject,'String')) returns contents of
edit_temperature as a double
end

```

```

function edit_temperature_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_temperature (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all CreateFcns
called

```

```

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject, 'KeyPressFcn', @(hObject, eventdata)
SpinBox(hObject, eventdata, ...
    struct('SmallStep', 1, 'BigStep', 10)));
end

```

```

function popup_gaz_Callback(hObject, eventdata, handles)

```

```

% hObject   handle to popup_gaz (see GCBO)
% eventdata reserved - to be defined in a future version of
MATLAB
% handles   structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popup_gaz
contents as cell array
%         contents{get(hObject,'Value')} returns selected item from
popup_gaz
gaz=get(handles.popup_gaz,'string');
val=get(handles.popup_gaz,'Value');
current=gaz{val};
switch current
    case 'N2'
        handles.Bzero=1.989581;
        handles.Dzero=5.76E-6;
        handles.delta_alpha_v=0.93;
        handles.g_even=6;
        handles.g_odd=3;
        set(handles.edit_B0,'string',handles.Bzero);
        set(handles.edit_D0,'string',handles.Dzero);
        set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
        set(handles.edit_g_even,'string',handles.g_even);
        set(handles.edit_g_odd,'string',handles.g_odd);

    case 'CO'
        handles.Bzero=1.93128087;
        handles.Dzero=6.12147E-06;
        handles.delta_alpha_v=0.93;
        handles.g_even=6;
        handles.g_odd=3;
        set(handles.edit_B0,'string',handles.Bzero);
        set(handles.edit_D0,'string',handles.Dzero);
        set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
        set(handles.edit_g_even,'string',handles.g_even);
        set(handles.edit_g_odd,'string',handles.g_odd);

    case 'CO2'
        handles.Bzero=0.3902;
        handles.Bzero=0.391322464583567;%from Malte's experiment
        handles.Dzero=0.135E-6;

```

```

handles.delta_alpha_v=2.1;
handles.g_even=1;
handles.g_odd=0;
set(handles.edit_B0,'string',handles.Bzero);
set(handles.edit_D0,'string',handles.Dzero);
set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
set(handles.edit_g_even,'string',handles.g_even);
set(handles.edit_g_odd,'string',handles.g_odd);
case 'N2O'
handles.Bzero=1.93128087;
handles.Dzero=6.12147E-06;
handles.delta_alpha_v=0.93;
handles.g_even=6;
handles.g_odd=3;
set(handles.edit_B0,'string',handles.Bzero);
set(handles.edit_D0,'string',handles.Dzero);
set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
set(handles.edit_g_even,'string',handles.g_even);
set(handles.edit_g_odd,'string',handles.g_odd);
case 'O2'
handles.Bzero=1.4376766;
handles.Dzero=4.839E-6;
handles.delta_alpha_v=1.14;
handles.g_even=0;
handles.g_odd=1;
set(handles.edit_B0,'string',handles.Bzero);
set(handles.edit_D0,'string',handles.Dzero);
set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
set(handles.edit_g_even,'string',handles.g_even);
set(handles.edit_g_odd,'string',handles.g_odd);
case 'H2'
handles.Bzero=60.853;
handles.Dzero=0;
handles.delta_alpha_v=0.22;
handles.g_even=1;
handles.g_odd=3;
set(handles.edit_B0,'string',handles.Bzero);
set(handles.edit_D0,'string',handles.Dzero);
set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
set(handles.edit_g_even,'string',handles.g_even);

```

```

    set(handles.edit_g_odd,'string',handles.g_odd);
case 'D2'
    handles.Bzero=30.444;
    handles.Dzero=0;
    handles.delta_alpha_v=0.22;
    handles.g_even=6;
    handles.g_odd=3;
    set(handles.edit_B0,'string',handles.Bzero);
    set(handles.edit_D0,'string',handles.Dzero);
    set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
    set(handles.edit_g_even,'string',handles.g_even);
    set(handles.edit_g_odd,'string',handles.g_odd);

```

```

case 'CS2'
    handles.Bzero=0.109;
    handles.Dzero=1.176E-6;
    handles.delta_alpha_v=9.6;
    handles.g_even=1;
    handles.g_odd=0;
    set(handles.edit_B0,'string',handles.Bzero);
    set(handles.edit_D0,'string',handles.Dzero);
    set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
    set(handles.edit_g_even,'string',handles.g_even);
    set(handles.edit_g_odd,'string',handles.g_odd);

```

```

case 'C2H2'
    handles.Bzero=1.1766;
    handles.Dzero=0;
    handles.delta_alpha_v=2.69;
    handles.g_even=1;
    handles.g_odd=1;
    set(handles.edit_B0,'string',handles.Bzero);
    set(handles.edit_D0,'string',handles.Dzero);
    set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
    set(handles.edit_g_even,'string',handles.g_even);
    set(handles.edit_g_odd,'string',handles.g_odd);

```

```

case 'Benzene'
    handles.Bzero=0.1896;
    handles.Dzero=0;

```



```
handles.delta_alpha_v=-5.96;
handles.g_even=1;
handles.g_odd=1;
set(handles.edit_B0,'string',handles.Bzero);
set(handles.edit_D0,'string',handles.Dzero);
set(handles.edit_delta_alpha_v,'string',handles.delta_alpha_v);
set(handles.edit_g_even,'string',handles.g_even);
set(handles.edit_g_odd,'string',handles.g_odd);
end
guidata(hObject,handles)
end
```

Lampiran 3. Daftar Istilah

<i>Molekular alignment</i>	: Penyelarasan molekul dengan menggunakan pulsa laser non-resonan
Orientasi	: Pengarahan (molekul) yang diukur dengan $\cos \theta$
Alignmen	: Penyelarasan (molekul) yang diukur dengan $\cos^2 \theta$
HHG	: <i>High Harmonic Generation</i> / Pembangkitan Harmonik Tinggi, merupakan proses dimana atom gas mulia didorong oleh medan laser <i>intens</i> oleh frekuensi ω sehingga menghasilkan radiasi dari frekuensi yang lebih tinggi $\Omega = n\omega$, dimana n adalah angka ganjil, dikenal sebagai urutan harmonik.
<i>Kick Alignment</i>	: Merupakan nama lain dari dinamik Alignmen
FFT	: <i>Fast Fourier Transform</i> , merupakan metode pengolahan sinyal untuk merubah sinyal pada domain waktu ke domain frekuensi
Wilayah perturbative	: Merupakan penurunan sinyal harmonik secara eksponensial pada orde rendah
DSP	: <i>Digital Signal Processing</i> (DSP) merupakan proses analisa suatu sinyal
<i>Spectral minima</i>	: Amplitudo sinyal
<i>Switch-off Alignmen</i>	: Merupakan jenis alignmen yang menggunakan campuran metode adiabatik dan dinamik.
Rekombinasi	: Merupakan proses tertariknya elektron kembali pada inti

<i>Arbitrary unit</i>	: Merupakan angka sembarang untuk digunakan sebagai pembanding rasio kekuatan sinyal (dalam sinyal HHG)
<i>femtosecond</i>	: <i>femtosecond</i> (simbol fs) adalah sebuah satuan dalam sistem metrik yang menunjukkan faktor seperseributribiliun detik (10^{-15} atau 0,000000000000001 atau 1/1000.000.000.000.000)
<i>picosecond</i>	: <i>picosecond</i> (simbol ps) adalah sebuah satuan yang menunjukkan faktor sepertriliun detik (10^{-12} atau 0,000000000001 atau 1/1000.000.000.000) .