uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Justin Gagnon

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.Sc. (Physics)

GRADE / DEGREE

Department of Physics

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Laser Coulomb Explosion Imaging of Polyatomic Molecules

TITRE DE LA THÈSE / TITLE OF THESIS

Dr. P. Corkum

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Dr. A. Czajkowski

Dr. J. Giorgi

Dr. M. Vincter

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Laser Coulomb Explosion Imaging Of Polyatomic Molecules

By
Justin Gagnon

Directed by
Dr. Paul Corkum and Dr. Ravi Bhardwaj

A thesis submitted to the Faculty of Graduate Studies in partial fulfillment of the requirements for the degree of

Master of Science
in
Physics

UNIVERSITY OF OTTAWA

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-25775-3
Our file Notre référence
ISBN: 978-0-494-25775-3

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

© Justin Gagnon, Ottawa, Canada, 2006

Table Of Contents

I. Summary

Laser technology has steadily evolved over the last 50 years since its invention, and has generated a series of ramifications in experimental science. Particularly, lasers have enabled the creation of the shortest man-made event: a femtosecond pulse of electromagnetic radiation. Due to their unmatched spatial and temporal resolutions, femtosecond pulses have been used in a number of techniques to measure properties of individual molecules.

One of these techniques is *Coulomb Explosion Imaging* (CEI), whose purpose is to retrieve the structure of individual molecules. Unlike frequency domain spectroscopy (which is ill-suited to characterize the structure of large molecules due to their complex spectra) and diffraction techniques (which only work if molecules can be locked into a crystallization pattern), CEI provides a direct measurement of the properties of individual molecules, instead of measuring a sample as a whole. This novel technique was first introduced to study molecular structure by colliding a beam of highly energetic ions onto a thin foil. The version of CEI used in this work uses a beam of neutral molecules and replaces the thin foil with femtosecond optical pulses. The introduction of the laser has brought with it the ability to conduct time-resolved measurements of molecular processes (breaking of molecular bonds, internuclear motion, for example) on a femtosecond time scale using pump-probe techniques in conjunction with CEI. Furthermore, CEI is presently the only technique that can discriminate single molecules based on their handedness.

I have conducted a *Laser Coulomb Explosion Imaging* (LCEI) experiment using dicloromethane as a model polyatomic molecule. In order to perform LCEI, an intense femtosecond laser pulse is used to strip away electrons from a molecule and cause it to explode into smaller fragments. Imaging the molecule is done using data collected from its fragments. Thus, in practice LCEI can be seen as a technique comprising an experimental phase (Coulomb explosion) and an analytical phase (imaging). Dichloromethane was chosen for this study to prepare the techniques that are necessary for future experiments on chiral molecules.

The experimental setup used for this instance of LCEI is the PATRICK instrument, a combination of high-end vacuum, electronics and laser equipment, which will also be described herein. The rest of this thesis will focus on the results obtained from the computational tools I developed for imaging the CEI data and obtaining physical properties about the exploded molecules. In doing so I have also obtained the first geometrical reconstructions of five atom molecules from CEI data, which will also be given in this study.

Though LCEI is a general technique that can be exploited in a variety of different experiments, this particular project was built around the interest of imaging chiral molecules. Unlike mass, multipole moments, polarizabilities and other "conventional" physical properties of molecules, chirality arises solely from spatial symmetry considerations, making it more elusive. For example, in order to experimentally determine the properties of a molecule in the traditional manner, one proceeds by inferring molecular characteristics from

general spectroscopic data pertaining to a sample of molecules. In this manner, molecules are ascribed properties based on statistical measurements done on a population.

Although statistical methods are also used to measure the handedness of a sample of molecules, it is understood that these measurements yield information only about the sample, but not the individual molecules themselves. Indeed, chirality is not a property of a *type* of molecule, but of *individual* molecules, rendering LCEI very suitable to measure chirality. Accordingly, it is the ultimate goal of this thesis to set the stage for future experiments involving the measurement of the handedness of individual chiral molecules.

II. Background

Chirality, a concept rooted in geometry, is also applicable to physical molecules. In essence, a molecule could be treated as a geometrical structure made up of atoms (vertices) and bonds (axes). If a molecule is chiral, then it cannot be superposed onto its mirror image. Everyday objects such as shoes and gloves also embody this property. Alluding to these examples, people often speak of left-handed and right-handed molecules when distinguishing between the two mirror images, or enantiomers, of a species of molecule. Actually, the term chirality itself stems from the greek word for "hand". Perhaps William Thomson (Lord Kelvin) defined it most adequately in 1893 when he wrote[1]:

> *"I call any geometrical figure, or group of points, chiral, and say it has chirality if its image in a plane mirror, ideally realized, can not be brought to coincide with itself"*
>
> - William Thomson (Lord Kelvin)

Two enantiomers of a chiral substance have the same molecular formula, they are just mirror images of one another (Figure 0). Therefore they possess the same boiling and melting point, the same colour (absorption and emission spectrum), the same density and so on. It seems prima facie that there is no way to distinguish them from each other, let alone separate them! Yet the consequences of chirality have been known for some time.

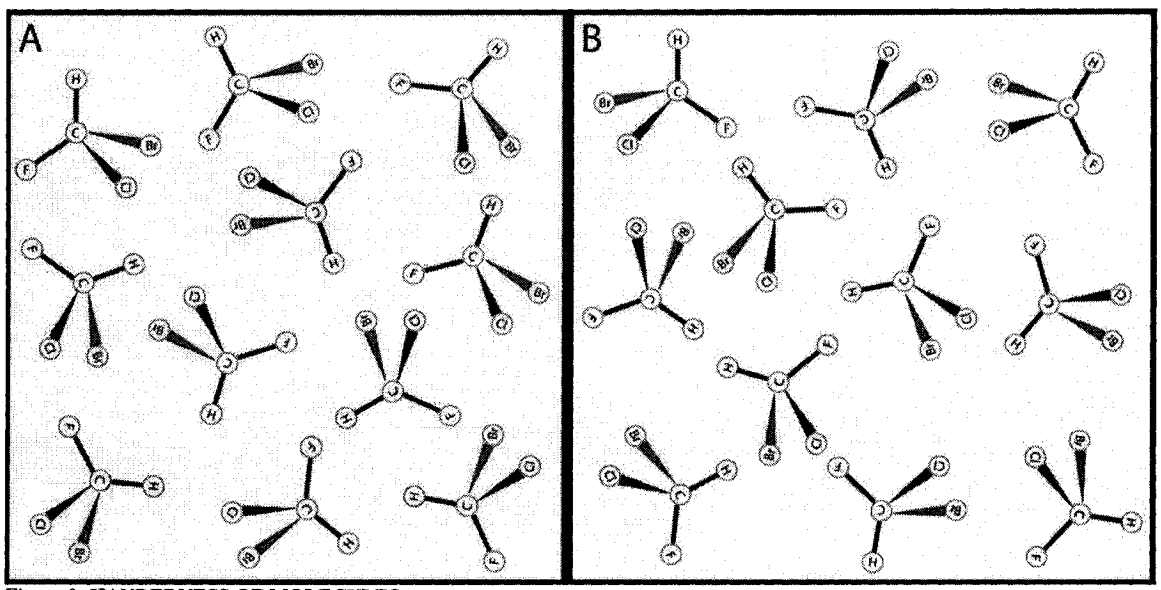


Figure 0: HANDEDNESS OF MOLECULES
Chirality can be illustrated by using objects that possess a helical geometry like these CHFClBr molecules, for example. By pointing your right thumb in the direction of the C-H bond in Figure 0-A, you will find that your fingers automatically curl along the F-Cl-BR atoms in a cyclic manner. This is true for any molecule in 0-A, regardless of its orientation. On the other hand, Figure 0-B's molecules are mirror images of the ones in 0-A. In this case, placing your left thumb along the C-H bond makes your fingers curl in the F-Cl-Br cyclic order. Thus, springs in 0-A can be called right-handed whereas the ones in 0-B are left-handed. The idea here is that handedness is preserved under rotation but not under a symmetry transformation. This distinction is especially important for molecules since they are constantly rotating, but never transform into their mirror image, which means that each molecule's handedness is conserved during its motion.

From the late 1950's to early 1960's the pharmaceutical industry was manufacturing the drug *thalidomide* for pregnant women who wanted to be relieved of morning sickness and other inconveniencing symptoms. It was synthesized in 1953 and marketed in 1957. Although teratogenic effects (birth defects) due to this drug first appeared in 1956, it wasn't until 1961 that the medical community realized that thalidomide was causing the birth defects and that Grunenthal Corp. did not perform adequate testing before releasing it to the market.

It turns out that the causes of the biological malformations are entirely chiral in nature. Thalidomide, being a chiral substance, comprises two enantiomers. In fact, one of the enantiomers does indeed reduce pregnancy-related illnesses but the other one triggers teratogenesis. This explains the birth defects, since thalidomide was administered to patients as a racemic mixture (a mixture that contains equal amounts of both enantiomers). Furthermore, thalidomide undergoes chiral inversion (where one enantiomer is converted into another) in the human body[2]. So even if a patient receives a dose of the "good" enantiomer, her body will automatically synthesize the teratogenic one. The underlying verity behind this whole ordeal is that chemical reactions involving two or more chiral molecules can depend on the handedness of the reactants. Or to put it bluntly, a right hand is easier to shake with a right hand. In point of fact, not only are most organic molecules chiral, but it seems that living matter only contains one enantiomer of each of its molecular constituents[3][4]. Consequently, living matter as a whole is chiral. For example, DNA is right-handed whereas amino acids are all left-handed. Apart from organic matter, a portion of the Earth's crust is chiral as a result of symmetry breaking processes during crystallization[5][6]. This means that chemical reactions in the soil can be highly dependent on the handedness of the reactants involved[7]. Since chirality can play an important role in several major areas of research, I will discuss the basis for measurement and quantification of this property of chemical substances.

First, if a beam of polarized light is passed through a chiral[1] solution the plane of polarization will be rotated clockwise or counter-clockwise depending on the handedness of the molecules in the solution (Figure 1). Through his helix model[8] Fresnel discovered that this phenomenon, referred to as *optical activity*, is the result of a medium having two different indices of refraction for left- and right- circularly polarized light (circular birefringence), due to the molecules exhibiting left- and right-handed helical morphology. Each substituent atom or group that is bound to the chiral center of a molecule has a different electronic polarizability, thereby resulting in slow and fast indices of refraction for different circular polarization directions. As a result, left and right circularly polarized light travel at different velocities inside the medium. Being a mix of equal amounts of left- and right-circular polarization states, linearly polarized light passing through an optically active sample will have its two circular components phase shifted with respect to each other, and the plane of linear polarization will be rotated by some angle β. In 1824 Fresnel made use of

[1] Actually optical activity also shows up under the Faraday effect, where a magnetic field oriented along the direction of propagation of light rotates the plane of polarization as if the magnetic field made the molecules become chiral.

this effect to introduce a quantity, the rotatory power, which describes the optical activity of a medium using the angle of rotation β per unit thickness for a given wavelength.

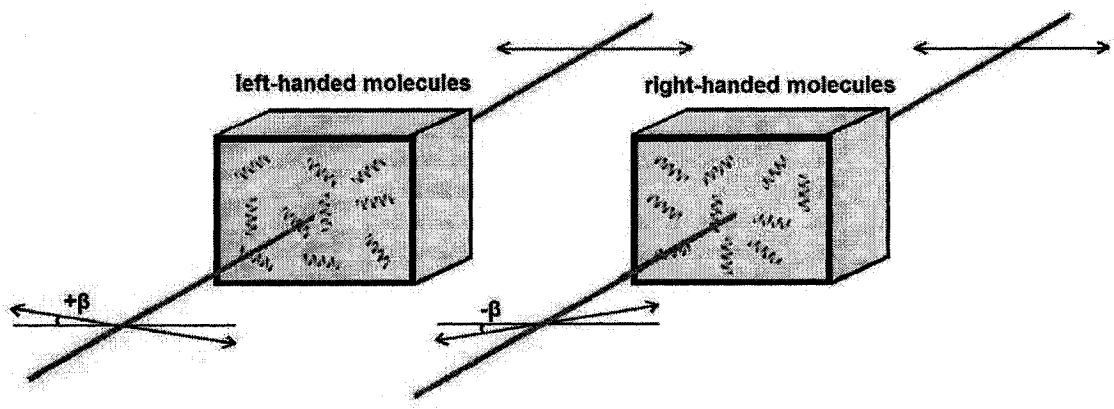


Figure 1: CIRCULAR BIREFRINGENCE
After passing through an optically active medium, linearly polarized light will be rotated by an angle β with respect to its initial direction. The orientation of β (clockwise or counter-clockwise) depends on the net handedness of the sample.

Drude[9] furthered Fresnel's model by relating optical activity to the helical motion of electrons under an oscillating electromagnetic field. Drude's model also incorporates the phenomenon of circular dichroism, which is the medium's differential absorption between left- and right-hand circularly polarized light. Just as right-hand (n_R) and left-hand (n_L) indices of refraction can be defined for optically active samples, right-hand (α_R) and left-hand (α_L) absorption coefficients can be defined for materials exhibiting circular dichroism. Actually, optical activity and circular dichroism together form the real and imaginary parts (respectively) of a material's complex circular birefringence. When the frequency of the electromagnetic wave is resonant with the natural frequency of the bound electrons, the molecule absorbs most of the radiation, rendering optical dichroism the dominating regime. On the other hand, when the wavelength of the radiation is far from resonance, the absorption is negligible and optical activity is the dominating process. An illustration of the effects of circular dichroism are shown in Figure 2.

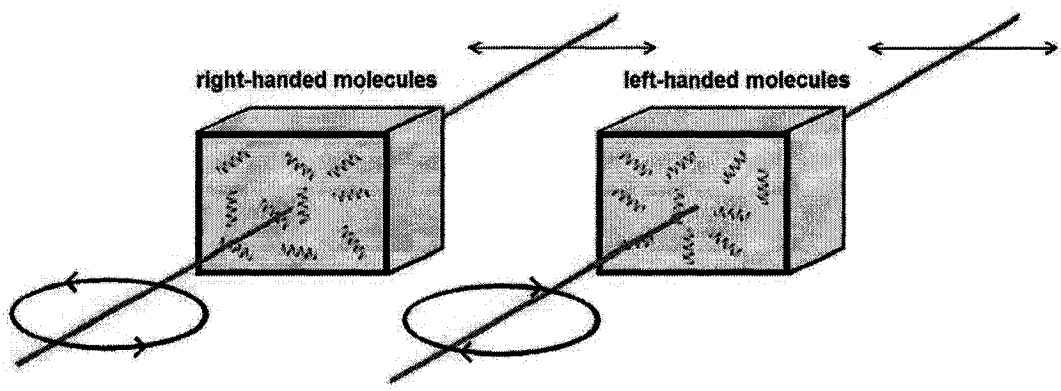


Figure 2: CIRCULAR DICHROISM
For example, a linearly polarized beam entering a circularly-dichroic medium will not be rotated, but will actually come out elliptically polarized with the major axis oriented in the direction of the incoming polarization plane. In this case, the handedness of the sample will reveal itself in the direction of the resulting elliptical polarization. That is, a sample of left (right)-handed molecules will create left (right)-handed elliptical polarization. As can be verified using Jones' calculus, a linearly polarized beam of light passing through a medium that is both optically active and circularly dichroic will come out elliptically polarized, but with the main elliptical axis rotated at some angle with respect to the direction of the incoming linear polarization. In the limit of zero differential absorption the ellipse degenerates into a straight line along its main axis, describing linear polarization as expected for an "ideal" optically active material. In most cases materials that are optically active concurrently show evidence of circular dichroism; this is analogous to birefringent media which usually possess two different absorption axes.

Now, as Louis Pasteur realized in 1848 by passing light through tartaric salts taken from wine production vessels[10], the angle of rotation of the plane of polarization is either clockwise (*dextrorotatory*) or counterclockwise (*levorotatory*) depending on which enantiomer is in the sample. If one enantiomer of a chiral molecule rotates the plane of polarization by an angle β, the other enantiomer of the same molecule will rotate it by $-\beta$. Naturally, if a sample contains equal amounts of both enantiomers (called a *racemic* mixture) it will not affect the polarization at all since the rotations cancel each other out. Optical activity is widely used for measuring and labeling enantiomers. Labels *L* or *D* are appended to the name of a chiral substance if it rotates light polarization in the counterclockwise or clockwise direction respectively. To further characterize a substance's optical activity, more sophisticated methods are used. For example, optical rotatory dispersion (ORD) measures the wavelength dependence of the rotatory power. This way, instead of a single value an entire spectrum can be used to characterize a particular chemical.

Over the years, a variety of schemes based on circular birefringence and/or circular dichroism combined with spectroscopic methods, NMR[11] and electron beam diffraction[12] (for enantiomorphic crystals) have been implemented to characterize enantiomeric compositions in solutions. As well, methods for separating enantiomers are well known[13][14][15][16][17][18]. For example, an analyte can be dissolved into a mobile phase and passed through a chiral medium which is in a stationary phase (a selector) that only lets through one enantiomer of the analyte. This is called chiral chromatography and is widely used for separating enantiomers from a mixture.

However, with the maturity of today's technology and scientific knowledge, experiments can now be devised to probe physics and chemistry on a molecular scale. It is

possible, for instance, to "watch" a chemical reaction unfold or to visualize the breaking of molecular bonds[19][20], and on the atomic scale to obtain a three dimensional picture of an electronic orbital[21]. Yet, present-day experiments are not limited to probing as it is also possible to control processes on molecular and atomic scales, such as the spatial alignment of molecules by controlling rotational wavepackets[22], vibrational modes in molecules[23], atomic confinement using optical lattices[24], and processes relating to and including electron recollision[25][26][27]. These advances make it possible to conduct experiments that can measure a single chemical reaction (time, energy and dynamics of a reaction) and the properties of the reactants and products (electronic orbitals, molecular *J*-states, energy levels).

Needless to say, these types of experiments would provide greater insight into the role that chirality plays in molecular and light-matter interactions if they were also able to identify the handedness of the individual molecules involved in each interaction. Techniques based on optical activity and circular dichroism, as well as diffractive and spectroscopic methods, rely on the statistical accumulation of an effect over a large population of molecules in order to provide a signal that can be measured in the laboratory. As such, they are ill-suited for identifying the handedness of individual molecules in a single chemical reaction. They provide knowledge of the net handedness of a group of molecules but are experimentally impractical for classifying individual left-handed and right-handed molecules. In view of the experimental difficulties in measuring optical activity on such a small scale with currently available technology, it is the objective of this research project to develop and implement a radically new scheme, first introduced by T. Kitamura *et al.*[28], involving Coulomb Explosion Imaging (CEI) to determine handedness on a molecule-per-molecule basis.

II. Technology

A) Coulomb Explosion Imaging

As its name implies, CEI is a procedure in which a molecule is stripped of its electrons and literally explodes into several positively charged fragments (these can be individual atoms or aggregates made up of several atoms), due to a deficit of negative charges. The imaging of this explosion is possible if it occurs inside a uniform electric field, which serves to direct the positive fragments onto a detector where they can be collected and identified. This method was first pioneered in 1979 by Z. Vager and E. P. Kanter[29][30][31]. In the early versions of CEI, an ionized molecule was accelerated to a near-relativistic kinetic energy of ~1 *MeV* and passed through a thin (~100 *Å*) foil during a time interval of ~0.1 *fs* (Figure 3). Such a short interaction time is on the scale of the electron motion within the molecule but is much shorter than the natural frequency of nuclear motion. Consequently, as they make their way through the foil several electrons are scattered away from their parent ion, in principle leaving the spatial configuration of nuclei unaltered. When they exit the foil the net charge on the ion is so great that Coulomb repulsion is the dominating force between its nuclei. The parent ion quickly breaks apart into a number of fragments (ideally the fragments are all individual nuclei, this makes experimental analysis less complicated).

These atomic fragments are treated as charged point particles (ignoring angular momentum, spin and vibrational states, as well as multipole moments of charge), which are collected by a detector that is placed along the center-of-mass velocity direction, and registers time-of-flight and two-dimensional spatial coordinates. From this data the reconstruction of molecular properties such as bond angles and bond lengths, among others, can be made by simply backtracking the trajectories of the fragments using Newtonian mechanics[2].

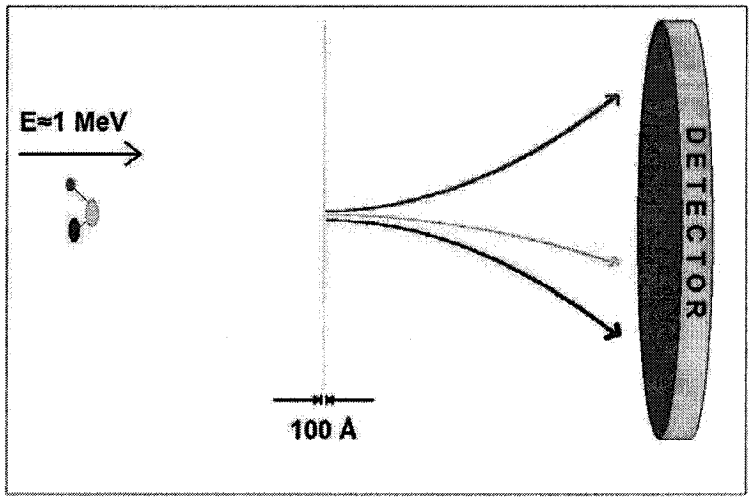


Figure 3: EARLY CEI
First version of Coulomb Explosion Imaging, in which a molecule is accelerated through a thin foil. After losing its electrons in the foil the positively charged atoms repel each other and fly onto a position and time sensitive detector.

Since Maiman's groundbreaking invention of the laser[32] in 1960, experimental scientists from many fields of research have embraced this new form of radiation and have incorporated it into their research in some way or another. In fact, lasers lend themselves particularly well to the study of the microscopic world because of their unequaled spatial coherence - making it possible to create a diffraction-limited focal volume small enough to enclose a single molecule – and also due to their extremely powerful output, enough to rip electrons off a molecule[33] with very high efficiency. More important than brute power, however, is the laser's flexibility. By adjusting laser intensity and wavelength one can control the number of electrons being removed[34][35][36], and thus the relative importance of the various ionization channels[37] occuring during the molecule-laser interaction. The polarization of light is a factor, albeit a more subtle one, that can be used to control finer processes like electron recollision[38] and molecular alignment[39]. Needless to say, these characteristic properties of laser radiation are quite suitable for use in CEI (Figure 4). As it happens, not only do lasers completely replace the thin foil in traditional CEI, they seamlessly merge CEI and pump-probe based experiments together, with the laser acting as both the pump and the probe[40]. Furthermore, by taking advantage of the inherent femtosecond precision of the pump-probe train, an extremely small time-of-flight mass spectrometer can effectively be constructed[41]. Evidently, if the laser is to replace the thin

[2] In fact, fragment velocities never reach relativistic speeds following the Coulomb explosion. Therefore, Newtonian mechanics is sufficient to map the (t, x, y) data into velocity vectors.

foil, the interaction must be made short enough so that the laser field doesn't thermalize or do mechanical work on the molecule[3]. Fortunately, modern laser technology is capable of delivering pulses of femtosecond duration coupled with PW/cm^2 peak intensities. And even though femtosecond interactions are temporally longer than those achieved using thin foils, they are nonetheless short enough to image molecular structure[42].

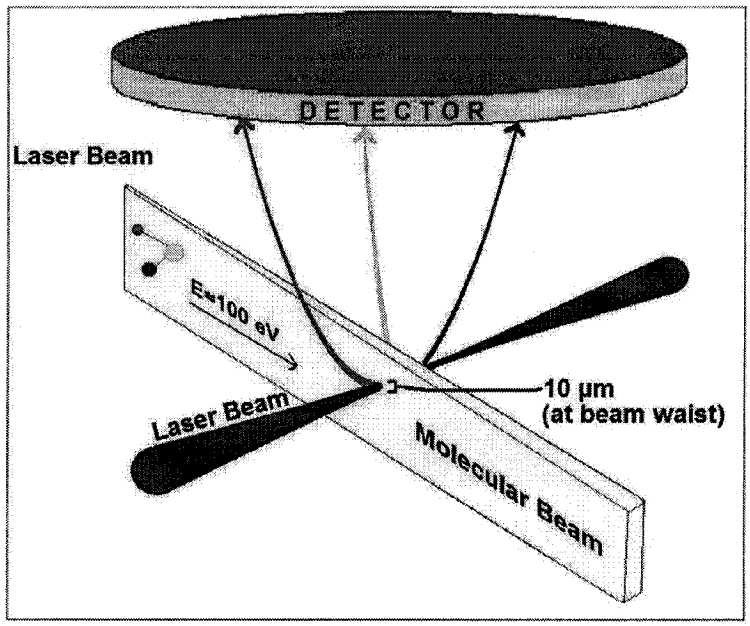


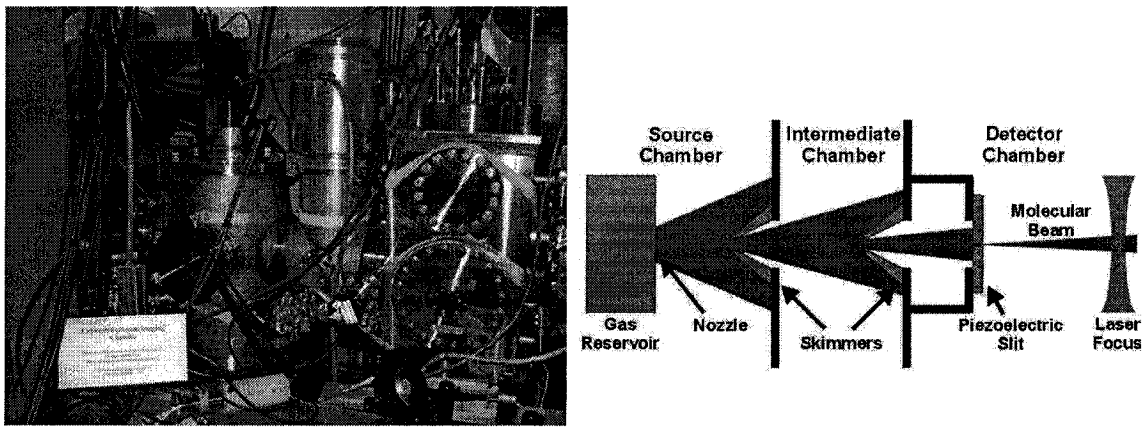
Figure 4: CONTEMPORARY CEI
Recent adaptations of CEI have replaced the thin foil with a powerful (~10^{15} W/cm^2) pulsed laser beam. Although the laser/molecule interaction time is typically longer than that of the molecule and thin foil (~5 *fs* instead of ~0.1 *fs*), it nevertheless permits the retrieval of the molecule's geometry[42]. The laser does provide a key advantage over the thin foil technique, which is the added support for pump-probe experiments.

B) The PATRICK Instrument[43]

As with thin foil CEI, the Coulomb explosion must happen inside a proper vacuum so that the fragments don't collide with background particles, as this would certainly spoil the coincidence analysis[4], and to avoid the Coulomb explosion of multiple molecules within the focal volume[5]. The vacuum and detection system used in this project consists of the PATRICK instrument, short for ***Position And Time Resolved Ion Correlation Kit***. It is the fruit of countless hours of hard labour from its creators, initiated by Dr. Christoph Ellert, and outfitted, completed, and fine-tuned by Dr. Patrick Dooley with the help of Dr. Ravi Bhardwaj, Dr. Igor Litvinyuk, David Joines, Bert Avery and John Parsons, and under the

[3] CEI was originally developed to image the structure of molecules. To this end, the action of ripping away the electrons must not modify its geometrical structure.
[4] A small amount of background is still required for calibration purposes, this will be explained in more detail later on.
[5] In general, it is preferable to explode multiple fragments at the same time to attain a faster data collection rate. Ambiguities arising from multiple explosions can be removed with proper concidence analysis of the data, detailed further on.

guidance of Research Officers Dr. Paul Corkum, Dr. David Villeneuve and Dr. David Rayner. The main elements comprised in the PATRICK apparatus are detailed below.



A *B*

Figure 5: VACUUM CHAMBERS
The three vacuum chambers of the PATRICK instrument. Pictured from left to right in Figure 5-A are the source, intermediate and detector chambers. The molecules entering the source chamber must be made into a fine beamwhen they reach the detector chamber, so that Coulomb explosion occurs only inside a very small volume defined by the intersection between the laser focus and the molecular beam width. This is accomplished by passing the initial intake of molecules through two skimmer before they reach the spectrometer (housed inside the detector chamber). The skimmers basically select only those molecules which have a low transverse velocity.

i) The Vacuum Chambers

A sequence of three vacuum chambers (Figure 5), each equipped with separate roughing and ultra-high vacuum pumping units provide an adequate environment for molecules to be exploded and detected. A gas manifold connected to the first chamber (called the *source chamber*) via a gas input flange (one of three ISO NW60 flanges on the source chamber) provides control of the amount of gas initially entering the system. Pressure readings in the manifold and in the gas reservoir embedded in the source chamber are given by a capacitance diaphragm gauge and a convection transducer (*MKS Instruments Series 907*, with *PDR 9000* readout) respectively.

The source chamber has a volume of roughly 20 liters with three *ISO NW160* side flanges and can be pumped down to a pressure of 6×10^{-8} Torr (monitored by a cold cathode inverted magnetron gauge) using a 2000 *l/s* diffusion pump (*BOC Edwards model Diffstak 250/2000M* with *Santovac 5* fluid) which is connected to the bottom flange. The diffusion pump itself is backed by a roots blower and a rotary pump. A manual butterfly valve is installed at the input of the diffusion pump to isolate it from the chamber when needed. As well, another manual valve is in place at the output of the diffusion pump to isolate it from the roughing line. The third flange interfaces with the second (*intermediate*) chamber. It houses an electroformed nickel skimmer (*Beam Dynamics model 1*) with a minimum diameter of 1 *mm*. The skimmer is also connected to a copper heater assembly which can heat it to eliminate condensation and clogging from the molecular beam.

The intermediate chamber is the smallest (it has a volume of 11 *l*) and simplest of the three. Its function is to create a "vacuum buffer" zone so that ultra-high vacuum (*UHV*) pumping in the final chamber (the *detector chamber*) can be performed more efficiently. It has an inverted *T* shape and also possesses three *ISO NW60* flanges, two that connect to the adjacent chambers and a top one that supports a 550 *l/s* turbomolecular pump (*Varian model V550*) backed by a rotary pump. This turbo is capable of pumping the intermediate chamber down to a pressure of about 8×10^{-9} Torr when no gas is inserted. Ultra-high vacuum pressure readings are accomplished using a Bayard-Alpert ionization gauge (*Varian* model *UHV-24* with a thorium-iridium filament). A gate valve is installed between the intermediate and detector chambers via an 8 inch *Conflat* flange onto which is mounted another 1 mm diameter skimmer to further narrow the molecular beam entering the detector chamber.

The detector chamber, as its name implies, is where the molecules are blown up with the laser pulse and detected. It encloses a volume of approximately 21 *l* and houses the main Coulomb Explosion Imaging hardware. First a controllable piezoelectric slit (*Piezosystem Jena* model *PZS INV O-101-51*, UHV compatible option with model *12V40* power supply), glued to a mounting plate with vacuum-compatible epoxy, sets the size of the molecular beam width[6] and thus the laser-molecule interaction volume (the region defined by the spatial overlap of the laser pulse and the molecular beam). The slit consists of two 22 *mm* long stainless steel edges that are driven symmetrically by a piezoelectric stack underneath them. Although the piezo slit increases turbulence in the molecular beam, thereby amplifying its divergence, it allows for manual control of the molecular beam width right before the laser focus. Vacuum connections on the detector chamber are actually all *Conflat* flanges since they provide the best seal for maintaining ultra-high vacuum. Thus the chamber can be evacuated to a pressure of 7×10^{-10} Torr with a 500 *l/s* turbomolecular pump (*Edwards BOC model EXT501*) mounted under the chamber onto an 8" *Conflat* flange. UHV readings are again given here by a Bayard-Alpert ionization gauge (*Varian UHV-24*). Along the molecular beam axis an 8" flange interfaces with the intermediate chamber and the opposite side houses an unused 6" flange. In the orthogonal direction an 8" flange holds a BK7 glass window (38.1 *mm* diameter and 2.0 *mm* thick), this is where the laser beam enters the chamber. Opposite to the laser input window is another 8" flange that supports a three dimensional translation stage (*Vacuum Generators* model *VZXYZ0515*) for the parabolic mirror inside the chamber. Finally, a 10" flange on top of the chamber supports the suspended spectrometer and detector, with feedthroughs leading to the time-to-digital converter (*Lecroy* model *3377*) which is outside of the chamber.

[6] The slit can be adjusted to a maximum separation of ~250 μm (creating a somewhat larger molecular beam width due to the thermal velocity spread), compared to a ~100 μm confocal parameter for the laser beam. However, a typical molecular beam width of 40-50 μm was used for this work.

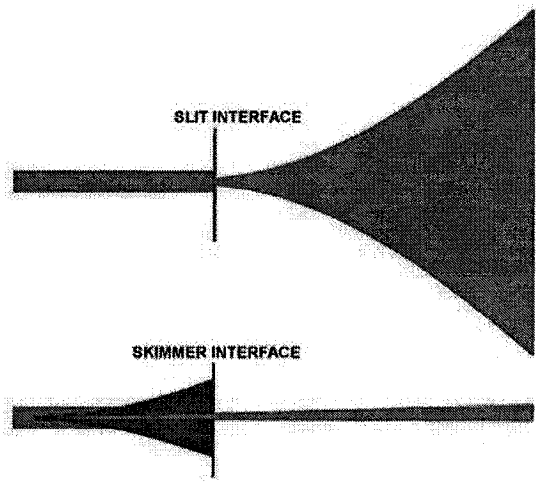


Figure 6: SKIMMER ADVANTAGE
The skimmers mounted between the chambers provide state of the art molecular beam collimation. Because of their geometry, the transverse velocity distribution of the molecules exhibits a much smaller spread compared to those passing through a simple slit. As a result of the reduced turbulence, the molecular beam coming out of the skimmer is much less divergent. This ensures that the laser-molecule interaction volume is kept at a minimum, a necessary condition for obtaining accurate CEI data.

A unique feature of the PATRICK instrument's double skimmer setup is the ability to produce a molecular beam as thin as 40 μm (with the piezo slit at minimal power), one of the thinnest currently available. The skimmers' design (Figure 6) ensures that the molecules that are not cold enough (in the transverse direction) are repelled away from the beam, and therefore do not collide with, and heat the incoming ones. Consequently, molecules coming out of the skimmers and the piezo slit are cold, and have an approximate transverse translational temperature of ~10 μK[7]. The small transverse speed coupled with the extremely small laser-molecule interaction volume (estimated at 125 μm^3 at a wavelength of 800 *nm*) greatly benefits experiments on single molecules.

ii) The Spectrometer

After passing through the piezoelectric slit in the detector chamber the molecules enter the time-of-flight mass spectrometer (Figure 7), which is suspended inside the detector chamber. Unlike conventional time-of-flight mass spectrometry, where ions go through a series of regions with differing electric fields, the CEI spectrometer does not use intermediate grids to accelerate ions. This avoids scattering of ions from the grids and thus reduces noise arriving at the detector and helps the data collection algorithm. Instead, an electrode stack is in place to ensure that a uniform electrostatic field permeates the region through which charged fragments fly onto the detector.

The electrode stack itself is made up of a series of 17 stainless steel ring electrodes 1.00 ± 0.01 *mm* apart. A consequence of this design is that molecular fragments are exploded under a uniform electric field. The geometrical dimensions of the electrode rings (14 *mm* thick, inner and outer diameters of 70 *mm* and 140 *mm* respectively) ensure that edge effects between the plates are minimized – modeling of the electric field has been performed with

[7] The temperature of the molecular beam can be measured by simply ionizing the molecules (preferably chosen to be monatomic) with a low-intensity laser pulse. Their initial velocity distribution (which directly gives the temperature) can then be calculated using the (t, x, y) data obtained from the detector.

the SIMION computer program before the spectrometer's construction[43]. Furthermore, steel grids that are placed at the top and bottom of the spectrometer, as well as 14 *mm* guard ring pairs for each electrode plate ensure that external electric fields do not find their way into the spectrometer.

Detection of the charged (Figure 8) fragments is done partially with a matched pair of *microchannel plates* (*Galileo* model *Long-Life*). The MCP's have a usable diameter of 75 *mm* and provide a 60:1 aspect ratio. Their purpose is to collect the charge from an incoming fragment, amplify it using secondary electron multiplication, and direct it to the delay line anode detector for spatial measurement of the signal. The two dimensional helical delay line anode detector (*RoentDek GmbH* model *DLD-80*) constitutes the basis for the CEI detection process. It is made up of two orthogonal copper wire helices wound about an aluminum plate, with one helix being nested inside the other. The delay lines are isolated from one another, and also from the aluminum plate. Actually, the delay lines consist of pairs of wires that are wound adjacent to each other, with one of them positively biased at 30 *V*. Hence a pulse is detected when the potential difference changes at one end of the delay line. Constant fraction discriminators (*Lecroy* model *2735DC*) at the endpoints of each delay line allow for the measurement of a coordinate value of the 2-D position of the localized charge packet received by the microchannel plate.

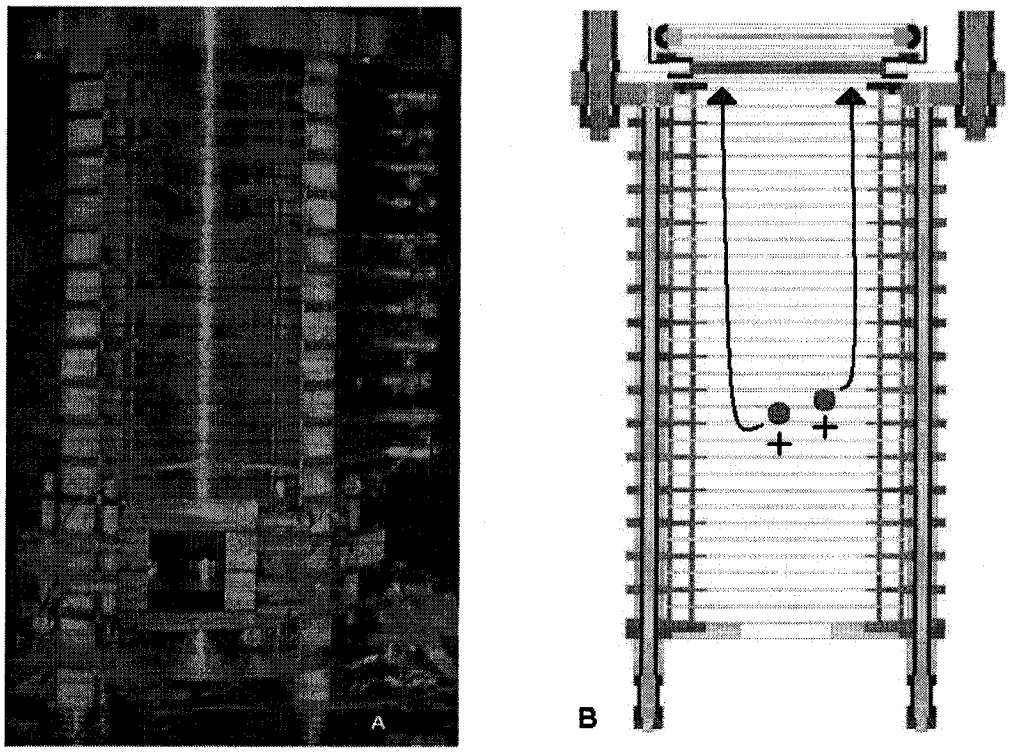


Figure 7: THE TIME-OF-FLIGHT MASS SPECTROMETER
Picture A shows a clear view of the electrode stack from the laser input side. The laser beam enters through the square opening at the bottom of the stack, exits through the other side (not displayed), reflects off a parabolic mirror (5 *cm* focal length) outside the spectrometer and is focused back inside where it crosses the molecular beam. As illustrated in Figure 7-B, molecules are ionized (and exploded) at the laser focus and fall towards the detector, mounted at the top of the stack, under a uniform electric field generated by the electrodes.

The MCP generates an avalanche of electrons in a localized region, constituting a charged electrical pulse of about 0.1 *pC* which is normally incident on the delay line. This generates two pulses traveling in opposite directions and uniform speed toward the extremities of the delay line. By measuring the arrival times t_1 and t_2 of the two pulses and knowing the speed of the pulses v_{DL} in the copper wire and the total length l_{DL} of the delay line it is possible to deduce the coordinate position s_{DL} where the pulse was generated on the delay line (the position $s_{DL} = 0$ is at the center of the delay line), and thus where the ion hit the MCP. The detection times t_1 and t_2 of the current pulses include the time-of-flight of the ion t_{TOF} (neglecting the TDC triggering time offset) added to the travel times of the pulses through the wire, hence

(1) $t_1 = t_{TOF} + \dfrac{\frac{l_{DL}}{2} - s_{DL}}{v_{DL}}$ for one of the pulses and

(2) $t_2 = t_{TOF} + \dfrac{\frac{l_{DL}}{2} + s_{DL}}{v_{DL}}$ for the other one.

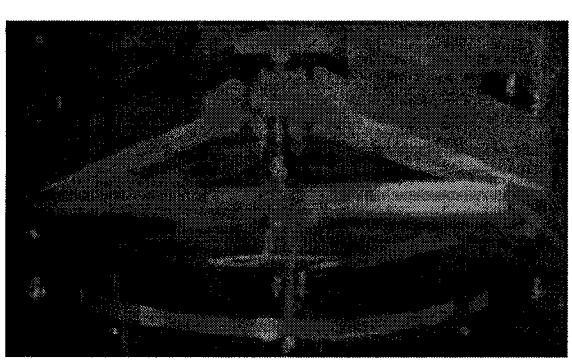
Therefore the position s_{DL} on the delay line is given by

(3) $s_{DL} = \frac{v_{DL}}{2}(t_1 - t_2)$,

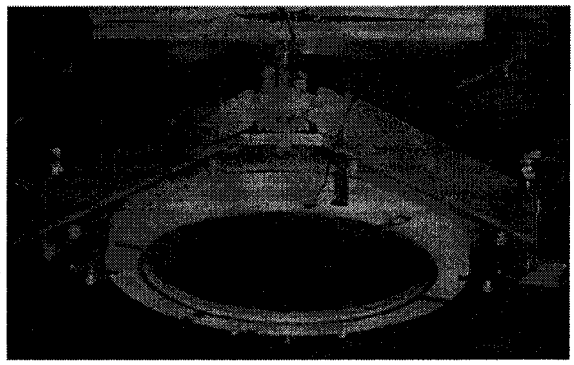
and the time-of-flight of the ion is

(4) $t_{TOF} = \dfrac{1}{2}\left(t_1 + t_2 - \dfrac{l_{DL}}{v_{DL}}\right)$

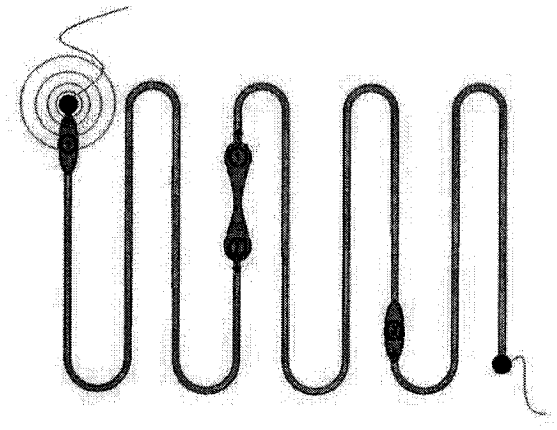
Of course, these formulas apply to both the x and the y delay lines, so that four measurements of time (t_{x1}, t_{x2}, t_{y1}, t_{y2}) are sufficient to determine the spatial and temporal coordinates of the ion impact along the delay lines.



A



B



C

Figure 8: THE DETECTOR
Picture A shows the two dimensional helical delay line anode detector, which collects the charge from the microchannel plate (Picture B). Due to its significant hygroscopy the MCP has to be placed in a dry environment during prolonged maintenance operations on the vacuum chambers. It is mounted right over the delay line.Figure 8-C is a schematic of the delay line anode detector.When a localized charge distribution hits the delay line it immediately divides itself into two parts traveling in opposite directions along the delay line until they reach the constant fraction discriminators (CFD's) at the endpoints. Once triggered by the electrical pulse the CFD sends a signal to the time-to-digital converter. These time values allow for the retrieval of the spatial location of the original charge distribution on the 2-D grid by looking at the uniform motion of the pulses in the delay line.

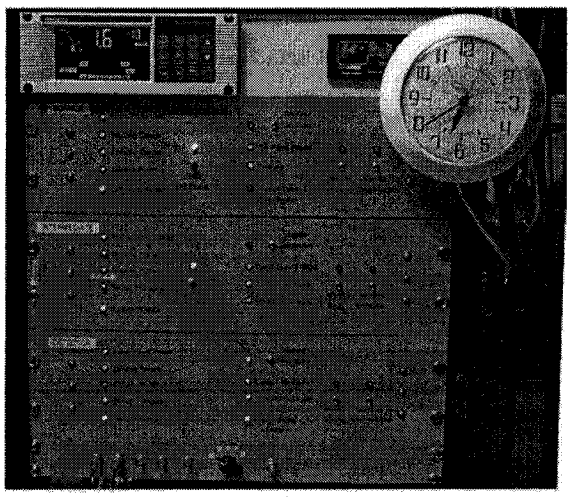
Because of the dispersion in the wires, the impact time t_{TOF} is given more accurately by picking off the signal directly from the MCP instead of deducing it from (4). However, the dispersive nature of the wires actually helps[8] the accuracy of the spatial measurement on the delay line[43]. As such, the spatial uncertainty is actually 250 microns instead of the 2 *mm* pitch of the helix turns.

iii) The Electronic Interlocks

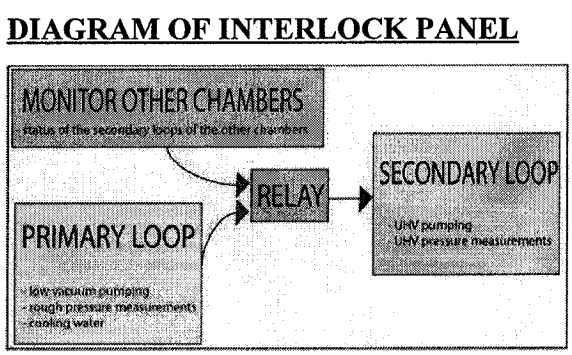
The PATRICK instrument is engineered to function non-stop for days and even weeks on end, therefore a reliable security system is in place to prevent unpredictable electrical and pump failures from harming the various mechanical and electronic pieces in and around the chamber. To this end, a panel of electronic interlocks, displayed in Figure 9, is set up to provide the chambers with a high level of self-awareness, although it still allows

[8] In practice, the group of electrons reaching the delay line will spill over several turns, and thus will create a train of electrical pulses that merge together (due to dispersion) into a long pulse. Therefore, the times at which these pulses trigger the CFD's at the endpoints of the delay line are not discretized in terms of the pitch of the delay line helices, but rather from the 0.5 *ns* bin size of the timing electronics, yielding a better spatial resolution than expected.

the experimenter to have tight control over the differential pumping levels and electrical functions of the PATRICK instrument thanks to its clear interface.



A



B

Figure 9: ELECTRONIC INTERLOCKS
This is the interface between the experimenter and the vacuum chambers. The interlocks are divided into three panels – one for each chamber (Figure 9-A). From top to bottom are the source, intermediate and detector chamber controls and LED's (the LED's show the status of their associated relay). The left and right columns of LED's monitor, respectively, the primary (rough vacuum and low-level pumping statuses) and secondary (ultra-high vacuum and pumping statuses) loop for each panel. Pressure measurements for the intermediate and detector chambers are displayed using a *Varian Multi-Gauge* controller (top-left), which is connected to the ionization and thermocouple gauges of their respective chambers. The source chamber's Pirani and inverted magnetron gauges are read with an *MKS PDR 9000* controller situated at the top right of the interlocks. Figure 9-B illustrates the principle of operation of the interlocks. Complete functionality of the chambers (power for the spectrometer, MCP, oven elements, and pulsed valve) is enabled provided that (1) the functions in the primary loop are active (all the leftmost LED's being turned on), and (2) the secondary loops of the other chambers being monitored are also active

The controls are divided into three separate panels – one for each chamber. Each panel consists of a primary and a secondary loop. The primary loops verify that the basic functions of each chamber – rough mechanical pumping and vacuum measurements on either side of the UHV pumps, as well as the flow of cooling water for each UHV pump – are without fault. If this is the case, then the the relays for the secondary loops are closed, enabling ultra-high vacuum pumping and measurement controls for each chamber to be manually activated. Once the UHV pumps are operating at a sufficient speed and the pressure readings from the UHV gauges are below their thresholds, additional functions can be activated. This allows the user to power up the spectrometer and MCP high voltage power supplies, as well as the oven heating elements and the pulsed valve (if these components are connected). Pump failures or gauge reading errors in any of the chambers will thus shut down the power from the turbomolecular and diffusion pumps, the UHV gauges and the spectrometer.

In addition, every chamber can be made to monitor the other two as an extra security measure. These extra monitoring capabilities are manually activated, and add an extra level of self-awareness to the PATRICK instrument. For example, on the source chamber's interlock panel there are "Monitor Detector" and "Monitor Intermediate" switches. If

"Monitor Intermediate" is turned on, then the source's secondary loop will rely on the intermediate's secondary loop to be activated. In the ideal case, all monitors are turned on, which means that each chamber (except one) depends on the other two to be activated, in addition to each chamber monitoring its own primary loop. In practice this is never done, as vacuum gauge and cooling malfunctions are common occurrences which would otherwise shut down the chamber frequently and needlessly.

iv) Femtosecond Laser Technology And Optics

Coulomb explosion of the molecules was accomplished with the help of a multi-gigawatt laser system built by Dr. David Villeneuve and other NRC staff members. It consists of a Kerr lens mode-locked Ti:sapphire oscillator that is pumped by a 5 *W* argon ion laser (*Coherent Innova 310*). The oscillator produces a train of pulses, each of a duration of 20 *fs* centered at 800 *nm* with a repetition rate of 80 *MHz* and an energy of 15 *nJ* per pulse (7.5 *nJ* of which is used in the CEI setup in this work), as well as an $M^2 = 1.1$[9].

The pulses are then amplified using a chirping scheme[44] to reduce the peak pulse intensity to protect the optical equipment from being damaged. Negative dispersion is first applied with the use of a grating pair which increases the pulse duration to 100 *ps*. After passing through a stretcher they are sent into a regenerative amplifier based on a Brewster cut Titanium sapphire crystal pumped by a 10 *W* frequency doubled Nd:YLF laser (a *Positive Light Merlin*). A Pockels cell at the input prevents most of the 100 *ps* seed pulses from entering the regenerative amplifier, keeping only one circulating through the cavity at any given time. After thirteen passes the pulse acquires 1.7 *mJ* and is switched out using another Pockels cell. Another grating pair is used to re-compress the amplified pulses back to 39 *fs*, with 900 μJ of energy before they enter the PATRICK instrument's detector chamber. The full beam profile of the pulses exiting the regenerative amplifier is then sent into the detector chamber to focus the beam as tightly as possible in the laser-molecule interaction volume. As previously mentioned, a parabolic mirror is mounted to an XYZ translation stage inside the detector chamber via a 406.4 *mm* long vented steel tube. The mirror is made of silver coated aluminum, has a diameter of 25 *mm* and a focal length of 50 *mm*. To preserve electric field homogeneity within the spectrometer the electric potential of the parabolic mirror and its mount are set to that of the nearest electrode in the stack.

IV. Experimentation, Analysis Techniques And Interpretations

This section will describe the various steps that were used to obtain the primary and seconday experimental data from the PATRICK instrument, including an overview of the software program written in the course of this work in order to analyze the raw data given by the time-to-digital converter (*Lecroy* model *3377*).

[9] The M^2 factor is a measure of the quality of a gaussian laser beam. In other words, it compares and quantifies the beam profile of a laser to that of a theoretical gaussian beam (for which $M^2=1$).

Immediately before a laser pulse is switched out, the Ti:sapphire regenerative amplifier's timing electronics send a trigger signal to the TDC via a commercial delay generator (*Stanford Research* model *DG-535*) coupled with a constant fraction discriminator (*Phillips Scientific* model *6915*). The TDC can then receive signals from the delay line anodes and MCP and record them in its internal buffer. A maximum of 16 ion hits can be recorded on each of the TDC's channels (one channel for each anode plus the MCP), at a resolution of 500 *ps*. After 32 μs the TDC shuts down and its buffer is emptied into the data acquisition computer.

It often happens that several of the anode signals do not trigger the CFD's, and thus are not detected and sent to the TDC. This can be a result of the dispersion in the delay lines or the initial spatial distribution of the electron pulse coming from the MCP, which would create a smeared out pulse arriving at the anode and prevent the triggering of the corresponding constant fraction discriminator (CFD). For example, consider two ions hitting the MCP in sequence. If both ions create flawless hits (a hit being a complete set of detection spatial and ion impact time values), then the ordered quintuplets - $(t_{MCP}^{(1)}, t_{x1}^{(1)}, t_{x2}^{(1)}, t_{y1}^{(1)}, t_{y2}^{(1)})$ and $(t_{MCP}^{(2)}, t_{x1}^{(2)}, t_{x2}^{(2)}, t_{y1}^{(2)}, t_{y2}^{(2)})$ - are properly interpreted by the computer. However if a CFD is not triggered during the first impact (say the $t_{y1}^{(1)}$ value) then the physical sequence $(t_{MCP}^{(1)}, t_{x1}^{(1)}, t_{x2}^{(1)}, t_{y2}^{(1)}, t_{MCP}^{(2)})$ will be understood as the quintuple of time values for the first ion impact. To prevent these mistakes from corrupting the data the acquisition computer is equipped with CEI2000, a set of software tools written by Dr. Igor Litvinyuk while Dr. Patrick Dooley was building the hardware and electronics for the PATRICK instrument. By using an appropriate discriminator for the x and y delay lines, CEI2000 effectively sifts through the TDC's buffer output after each laser pulse to identify and remove the incomplete hits from the raw data. A uniqueness constraint is also used in the algorithm to prevent multiple impacts (in different spatial positions of the MCP) from being associated to the same impact time. Once the initial raw data is filtered out and reordered it is recorded in binary form as a listmode file (lmf) on the data acquisition computer's hard disk drive.

Instead of recording the five individual anode times, data are organized in (t_{MCP}, x, y) triplets, where the x and y spatial coordinates are proportional to the time differences between anode signals. The lmf files order the data in "hits" and "events", according to Figure 10. A hit is a relevant (physical) (t_{MCP}, x, y) triplet, as determined by the data acquisition program CEI2000. This corresponds to all the necessary information that CEI provides about a single ion. An event is recorded when the TDC buffer is emptied after a laser pulse has done its job (a 32 μs time window is available to collect the hits once the laser pulse has passed through the detector chamber). In other words, an event contains all of the hits that resulted from a laser pulse interacting with molecules in the molecular beam.

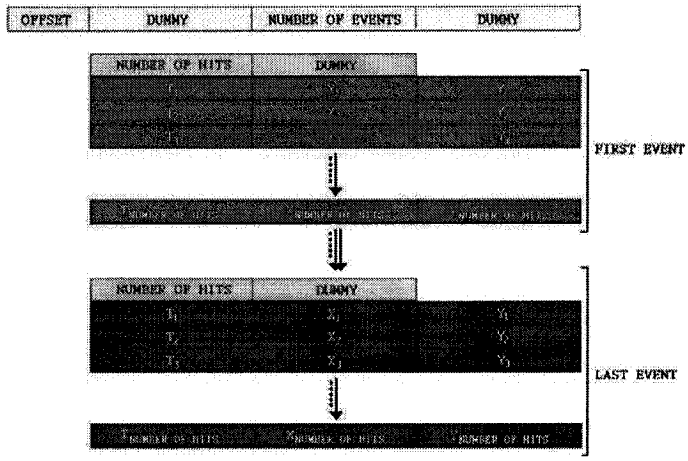


Figure 10: DATA ORDERING IN LMF FILES Listmode files contain the primary CEI data. When the acquisition program (CEI2000) disentangles the raw anode time signals it stores them into "hits" and "events" on the data acquisition computer's hard disk. A hit is a recognized (t, x, y) triplet, and a sequence of hits recorded after the passing of a laser pulse (during an interval of 32 μs) constitutes an event. These lmf files represent numbers in 8 bit precision, beginning with the file header (in yellow). The header consists of a 4 bit offset and an 8 bit "dummy" (this dummy is basically an empty space, it is not read by the acquisition software). After the dummy, the first significant value is the total number of events contained in the file, after which there is another 8 bit dummy. Following the file header is an event header (in green). The event header begins with the number of hits in the event, followed by an 8 bit dummy sequence. Essential experimental data are recorded immediately after the event header's dummy bits.

The main computational challenge following the preliminary data analysis amounts to understanding the hits and events contained in the lmf files and extracting from them the necessary information about the actual exploded molecules (3-D velocity vectors for each of the collected fragments). Indeed, the data contained in the lmf files are still quite rudimentary, as far as geometrical molecular reconstruction is concerned. To this end, a combinatorial coincidence imaging algorithm is implemented in order to retrieve velocity information about exploded molecules from primary (t, x, y) data. In fact, the secondary data analysis consists of

1) the combinatorial compilation of conjectural molecules from the raw (t, x, y) data, and

2) successive filtering of conjectural molecules based on calculated (from 3-D Coulomb explosion simulations) and empirically estimated physical properties.

A) The Dichloromethane Experiment

The CH_2Cl_2 molecule is the object of this study. Although it is not chiral, it embodies the main physical properties (molecular shape and size) necessary to devise appropriate Coulomb explosion imaging techniques to identify the handedness of small molecules. First, a small (~20 *mL*) glass flask containing the CH_2Cl_2 sample was inserted into a ¼ inch *Swagelok* tap on the source chamber's manifold. Its 400 Torr vapor pressure was sufficient to provide a continuous flow of gas through the manifold and all the way to the detector chamber, keeping in mind that only a very small number density of molecules is needed in the laser-molecule interaction volume, so that ~5 molecules should be contained in the molecules/laser interaction volume.

To accelerate the data acquisition process the slit opening and input pressure were adjusted so that an average of 2.256 hits were recorded per laser shot (a distribution of event sizes for this experiment is shown in Figure 11). Although this may seem like a low acquisition rate given that the CH_2Cl_2 molecule contains five atoms – leading to a premature conclusion that events will generally yield at most one complete molecule – it is important to keep in mind that most fragments are actually lost because many of the lighter fragments fly out of the spectrometer. Also, the CEI2000 program removes much of the data due to missed detections from the delay line anode CFD's. This is a result of each CFD's "dead time" of 16 ns FD after being triggered by an electronic pulse from the delay line, during which subsequent pulses are not registered. Since all of the CFD's must be triggered in order to register a hit, this means that fragments having a very close time-of-flight will generally not be recognized. This issue is even more apparent when dealing with molecules such as CH_2Cl_2, that are composed of many identical atoms. For these ones, certain spatial orientations will be partially suppressed.

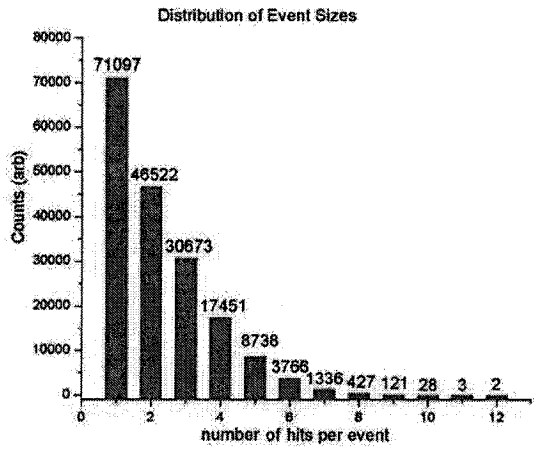


FIGURE 11: DISTRIBUTION OF EVENT SIZES
This histogram shows the distribution of event sizes for the current experiment on dichloromethane. The large proportion of events containing few (one and two) hits indicates that the ion detection was somewhat saturated. These events are completely ignored by ARBCORR, the data analysis program.

In the case of dichloromethane, a laser pulse that explodes a dozen molecules has the potential to generate 60 hits, but due to the low detection efficiency the vast majority of those are ignored. The resulting event could still contain several hits but is rendered useless because it contains only partial information about each molecule. In this case, the detector is said to be saturated.

Furthermore, exploding densely-packed molecules entails "space-charge" effects. This basically means that, in addition to the spectrometer's applied field, fragments also feel the electric field of other ionized particles in their vicinity and deviate from the simple parabolic projectile motion they would ideally follow, further corrupting the data. If too large, the molecular beam width can also have adverse effects on the experimental results. For example, if the molecular beam width is larger than the confocal parameter of the laser beam, the molecule-laser interaction volume is effectively increased and blurs the 3-D

velocity vector retrieval; and it also saturates the detection with low charge states[10] obtained from increased ionization at the low intensity regions of the laser beam. One thing that can be done to alleviate this problem is to reduce:

1) the piezoelectric slit opening – thinning the molecular beam;

2) the input pressure – lowering the number density of molecules;

3) the laser pulse intensity – effectively decreasing the laser-molecule interaction volume.

Such procedures will lift the aforementioned ambiguities but will also generate smaller (and many null) events, necessitating a longer acquisition time to build up proper statistics. For optimal results it is important to fine-tune these experimental parameters in order to strike an appropriate balance between the lucidity of the data and the rate of acquisition. Events containing ambiguous hits cannot be automatically dismissed, as this would result in an overwhelming loss of data. In order to extract the necessary information from them, a combinatorial "try all possibilities" strategy was developed and implemented in software for the data analysis, and will be described later on.

The spectrometer's internal electrostatic field was adjusted to approximately 21,000 *V/m* in order to capture the relatively light fragments (light fragments typically have greater velocities after the explosion and thus require a more powerful field to drag them towards the detector). Making the electric field too large can also have negative side effects:

1) Charged fragments from a molecule will hit the detector at closer time intervals (inversely proportional to electric field strength), reducing the number of detected hits due to dead times on the CFD's.

2) Exploded fragments will tend to fall closer to the center of the detector. This creates two ploblems:

   2.i) Spatial discrimination of impact positions is more difficult due to the detector's finite spatial resolution.

   2.ii) Molecular ions are those molecules ionized by the laser field but retain sufficient binding forces to prevent a Coulomb Explosion. They travel much more slowly than the fragments coming off a Coulomb explosion, and therefore hit closer to the center of the detector. Moreover, ionized oil molecules, that were back-streamed into the chambers from the rotary pumps, end up clogging the center

[10] The low charge states are not as useful because they don't recreate an ideal Coulomb repulsion. An ideal Coulomb explosion is one that agrees with the simplistic model of a spatial configuration of charges repelling each other according to Coulomb's force law, ignoring any attractive components resulting from higher-order electrical interactions.

of the MCP creating a spatial "low-sensitivity zone", where fragments are less likely to be recorded.

Again, it is important to make the electric field strong enough so that not too many data are lost, but weak enough to prevent obfuscating the data by localizing the spatial distribution of ion impacts onto a single area of the detector.

The laser pulses entering the spectrometer contained 136 μJ of circularly polarized light (45 *fs* duration, 8.37 PW/cm^2 peak intensity). Circular polarization was the preferred mode to prevent the orientational biasing of Coulomb explosions. That is, the dichloromethane molecule possesses an anisotropic polarizability tensor due to its particular geometrical shape. This means that the electron cloud displacement stimulated by an external electric field depends on the orientation between the field and the molecule. For extremely strong electric fields (such as the one from the laser pulse), Coulomb explosion occurs more effectively for molecules oriented in a certain direction with respect to the polarization direction of the laser pulse. Thus by randomizing the electric field direction of the laser pulse (by using circular polarization), Coulomb explosions are performed uniformly over all orientations.

Having chosen the appropriate polarization to use, we turn to laser pulse intensity. Peak pulse intensity is important, since the CEI data analysis assumes a simple classical model of repelling charges. Hence, CEI is properly performed when molecules are ionized to high enough charge states so that the force experienced by the exploding fragments is as close to the classical Coulomb force as possible. This also makes computer simulations of the experiment much simpler. This being said, it is also important to consider that increasing the laser intensity effectively makes the pulses temporally longer and spatially broader. For example, consider a laser pulse having a Gaussian intensity profile $I(t)=I_0 e^{-\left(\frac{t}{\tau}\right)^2 \ln 16}$, which theoretically never extinguishes to zero. One can measure its width by using a definition like the *Full Width at Half Maximum* (FWHM) τ or the standard deviation $\sigma=\sqrt{\langle t^2\rangle-\langle t\rangle^2}$. If the ionization potential of the molecules corresponds to a threshold intensity $\frac{I_0}{2}$ (the half maximum of this pulse), then ionization occurs during a time interval τ. If we double the laser intensity then the laser pulse is described by $I'(t)=2I_0 e^{-\left(\frac{t}{\tau}\right)^2 \ln 16}$. In this case the molecules begin to ionize earlier, and thus they feel a pulse of longer duration $\sqrt{2}\tau$. Since the pulse profile is TEM_{00} this argument also means that the spatial extension of the pulse increases with laser intensity (the pulse is approximately a Gaussian ellipsoid). As a matter of fact, the spatial and temporal parts of this effective pulse broadening have distinct consequences on the results obtained.

Ultrashort pulses, by virtue of their femtosecond time scale, possess the advantage of interacting with molecules without disturbing them. If the pulse is too long then it will start to physically deform and heat up the molecule before it explodes. This increases the

experimental error upon reconstruction of molecular bond lengths and bond angles (section IV-C-iii), and can lead to other undesirable effects such as molecular rotation[45] and enhanced ionization[46]. On the other hand, spatial broadening of the laser pulse is comparable to widening the molecular beam, in the sense that low charge states will be increasingly abundant in the collected data. The difference is that the pulse extends also outside of the molecular beam, ionizing more background molecules. As much as strong laser fields are needed to obtain the highest charge states, they still have to be moderated to ensure that the data remains transparent.

B) Primary Data Analysis

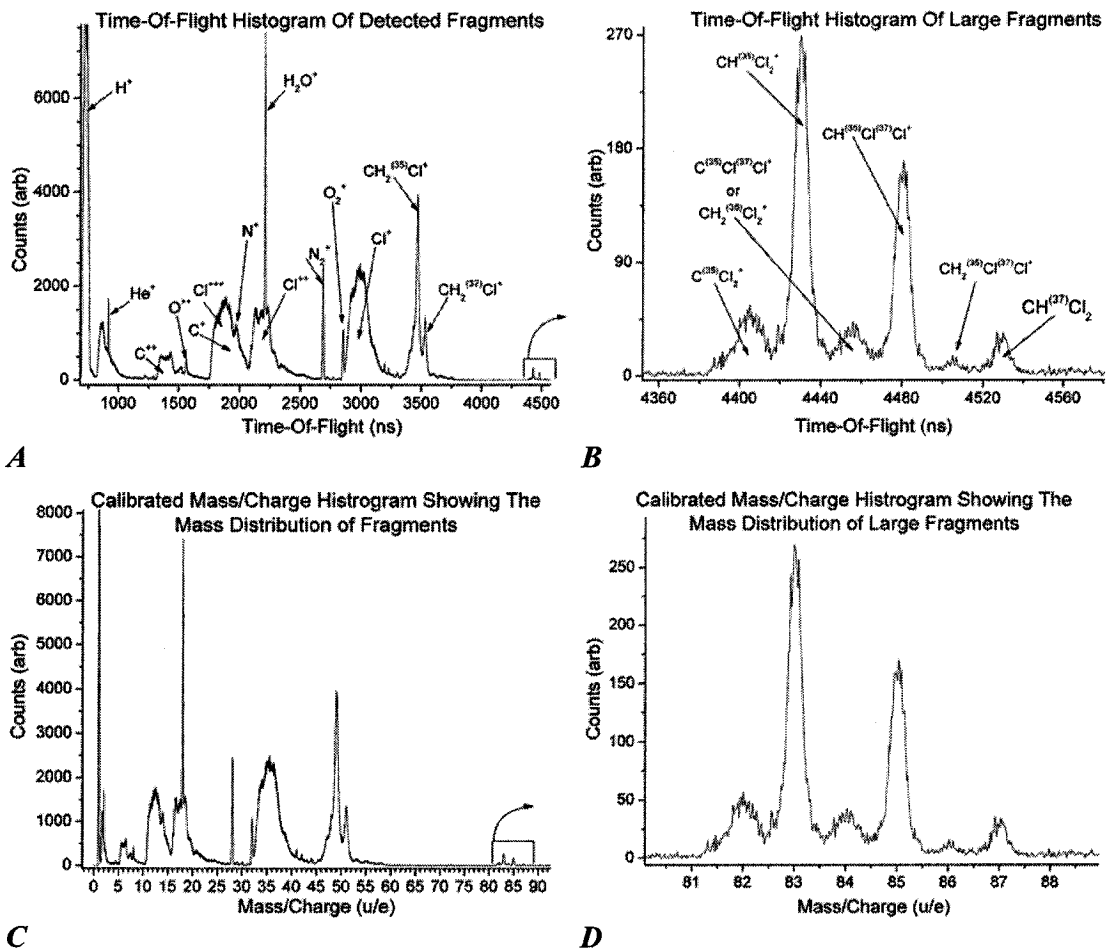


Figure 12: TIME-OF-FLIGHT HISTOGRAMS

Once the data are collected and stored the first strategy is to exploit the time-of-flight nature of the PATRICK apparatus. A histogram, like the one in Figure 12-A, is first compiled by collapsing the lmf data onto the time-of-flight values of each hit. This diagram displays two basic features – wide "humps", and narrow "peaks" - both essential to understand the data. As we will see, the peaks and humps are qualitatively different from one

another as they illustrate different kinds of laser-molecule interactions, the former being the result of ordinary molecular ionization and the latter originating from proper Coulomb explosion.

Because the molecules are not spatially pre-aligned before being exploded, they are randomly oriented in space when they meet the laser pulse. Since the time-of-flight of a fragment depends on its original velocity vector (both magnitude and orientation), unaligned Coulomb-exploded molecules produce wide humps in the TOF histogram (see Figure 12). The temporal width of the humps can be derived from Newtonian mechanics principles. In the ideal case of a diatomic molecule composed of identical atoms of mass m which are ionized to the same charge state q, Figure 13 shows the molecular orientation that defines the width $\Delta TOF = TOF_2 - TOF_1$ of the hump. In this case the two fragments of equal mass undergo uniformly accelerated motion according to

$$(5)\quad -h = -v_0 TOF_1 - \frac{1}{2}\frac{E}{(m/q)} TOF_1^2, \text{ and}$$

$$(6)\quad -h = v_0 TOF_2 - \frac{1}{2}\frac{E}{(m/q)} TOF_2^2,$$

where h is the distance along the spectrometer axis between the point of explosion and the point of impact on the detector, v_0 is the initial speed of the fragments along the spectrometer axis[11]; and E is the electric field permeating the spectrometer. Solving for TOF_1 and TOF_2 using (5) and (6) yields a simple expression for the temporal width of the hump:

$$(7)\quad \Delta TOF = \frac{2(m/q)v_0}{E}.$$

This argument explains the wider humps for fragments having a higher mass-to-charge ratio in Figure 12. Relation (7) can therefore be helpful to qualitatively identify the nature of a hump.

[11] Technically speaking, the initial speeds of exploded fragments are 0 (in the center-of-mass frame) since the first stage of the motion is the Coulomb repulsion between the charged fragments. This motion occurs on an attosecond (1E-18 *s*) time scale and is therefore not easily resolved in the laboratory. Analytically retrieved velocity vectors of the fragments shall hereafter be interpreted as the "asymptotic velocities"; in other words, the velocities of the fragments once they escape from Coulomb repulsion (in the asymptotic region of the Coulomb potential) and feel only the uniform electric field in the spectrometer.

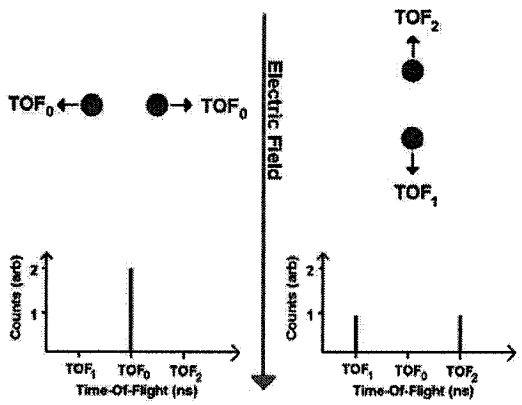


Figure 13: TOF SPREAD DUE TO COULOMB EXPLOSION
In the simplest case one can imagine a diatomic molecule undergoing Coulomb explosion. If the molecule is oriented perpendicularly with respect to the electrostatic field in the spectrometer (red arrow going down), then the molecular fragments have no initial velocity along the spectrometer axis. Thus they both reach the microchannel plate at the same time TOF_0. On the other hand, if the molecule is exploded when it is oriented along the electric field, its fragments reach the MCP at times TOF_1 and TOF_2. Naturally, TOF_1 and TOF_2 represent respectively the minimum and maximum time-of-flight for these fragments, and therefore constitute the boundaries for the hump generated by this particular Coulomb explosion. All other orientations of the molecule will contribute data lying in between TOF_1 and TOF_2.

The XYZ translation stage that controls the parabolic mirror provides the necessary mobility for fine-tuning the position of the laser focus, but at the same time makes the distance h impractical to directly measure with sufficient accuracy. Luckily, the analysis of the kinematics again provides a solution to this problem. Fragments exploding perpendicularly to the spectrometer axis (left-hand part of Figure 13) will generate data in the central time TOF_0 of a Coulomb explosion hump. Kinematics tells us that

$$(8)\quad h=\frac{1}{2}\frac{E}{(m/q)}TOF_0^2 .$$

Applying relation (8) to solitary humps (like the ones for C^{++} or Cl^{++} in Figure 12; the "++" notation denotes the excess of charge, in electron units, exhibited by the ion) allows for a precise value of h to be inferred.

However, relations (5) to (8) are useless without knowing the mass-to-charge ratio m/q of each hump. This calls for a calibration of recorded ion impact times to corresponding mass-to-charge ratios, for which the second important feature of the TOF histogram – the narrow peaks - comes in handy. Unlike exploded fragments, molecules ionized by the laser field travel much more slowly, according to the Maxwell speed distribution

$$(9)\quad f(v)=4\pi\left(\frac{M}{2\pi RT}\right)^{\frac{3}{2}}v^2 e^{-\frac{1}{2}\frac{Mv^2}{RT}}\text{, giving an average speed}$$

$$(10)\quad \bar{v}=\sqrt{\frac{8RT}{\pi M}},$$

where v, M, R and T are respectively the molecular speed, molar mass, universal gas constant and absolute temperature. For N_2, the most abundant background gas, this translates into an average molecular speed of $\bar{v}=471\,m/s$ at a temperature of 293 K. By contrast, Cl^{+++}

(molar mass of 35 *g/mol*) atoms coming off a Coulomb explosion travel at speeds up to 16,000 *m/s* and H^+ can attain relativistic speeds of up to 100,000 *m/s*.

Although some guess work is necessary for the calibration, an obvious choice is to assign the left-most peak to the H^+ ions since they are the lightest. Another peak of interest is the one centered at 2687 *ns*. It is a narrow peak and it is not invaded by other data. A first guess would be to say that it comes from ionized dinitrogen (N_2^+).

Another issue to consider is that the MCP impact times given by the TDC are not actually the times-of-flight of the ions. This is because there is a small but finite time difference between the start of the data acquisition (triggered by the regenerative amplifier's electronics) and the physical time of the ionization. Then, applying equation (8) to these two peaks and substituting the actual time-of-flight of the ion with a translated time $TOF_0 - t_{TDC}$ (where t_{TDC} represents the acquisition start delay and TOF_0 is the measured ion impact time) a proper time-of-flight calibration can be obtained. A calibration relation between mass-to-charge and time-of-flight is given by modifying (8) so that

$$(11) \qquad \sqrt{\frac{\kappa^{(N_2^+)}}{\kappa^{(H^+)}}} = \frac{TOF_0^{(N_2^+)} - t_{TDC}}{TOF_0^{(H^+)} - t_{TDC}},$$

where the κ's are the mass-to-charge ratios for their respective ions. Having solved for the TDC's acquisition delay time t_{TDC} using (11), it is then possible to calibrate the data set with

$$(12) \qquad \kappa^{(i)} = \kappa^{(N_2^+)} \left(\frac{TOF_0^{(i)} - t_{TDC}}{TOF_0^{(N_2^+)} - t_{TDC}} \right)^2,$$

where the index i ranges over the entire TOF spectrum. The calibration is good when all the peaks in the histogram fall on integer mass-to-charge values, in which case it should be possible to associate each one to a charge state of a known ion species. This in turn helps to identify the Coulomb explosion humps, which would otherwise be virtually impossible to do without the ion buoys.

Figures 12-C and 12-D on the next page show the calibration results of the initial time-of-flight data in 12-A and 12-B. In this case a secondary calibration was performed by replacing the H^+ peak with one of the rightmost peaks generated by $CH^{(35)}Cl_2^+$ from the molecular beam (12-B and 12-D). This provides a more accurate fit for the rest of the data. Interestingly, Figures 12-B and 12-D are in excellent agreement with the expected natural proportions of $CHCl_2^+$ for different chlorine isotopes. This is a testament to the extreme accuracy of the data provided by the PATRICK instrument.

$CHCl_2^+$ Variant	*Measured Amount (%)*	*Expected Amount (%)*
$CH^{(35)}Cl_2^+$	58.8	60.0
$CH^{(35)}Cl^{(37)}Cl^+$	35.0	34.9
$CH^{(37)}Cl_2^+$	6.2	5.1

Also of note is the instability of $CH_2Cl_2^+$ and CCl_2^+ aggregates as evidenced by the suppressed peaks (first, third and fifth from left) in Figures 13-B and 13-D. This suggests that the hydrogen atoms become loosely bound after being singly ionized , and are easily shaken off due to their small mass. When this happens, the second hydrogen atom no longer feels the positively-charged repulsion of its sibling (which was previously in an ionic bond with the central carbon atom), and tends to stay bound to the remaining charged conglomerate. This single hydrogen state is relatively more stable as can be seen by its large peaks (second, fourth and sixth from left) in Figures 13-B and 13-D. However if the second hydrogen leaves, the C-Cl bonds are further weakened and the molecule completely breaks apart into individual atoms, contributing to the large humps centered at 3000 ns (for Cl^+) and 1800 ns (for C^+).

For the purpose of reconstructing the secondary data, molecules must be represented in the molecular beam frame of reference instead of the laboratory frame of reference to avoid skewing the reconstructed velocity vectors (during the coincidence imaging phase of the data analysis). Thankfully, the production of the molecular beam largely eliminates the transverse motion of the molecules[43], such that knowledge of the molecular beam speed v_{MB} is sufficient to transfer our perspective into the molecular beam frame. In order to determine v_{MB} we turn to the 2-D spatial distribution of the ion impacts (Figure 14).

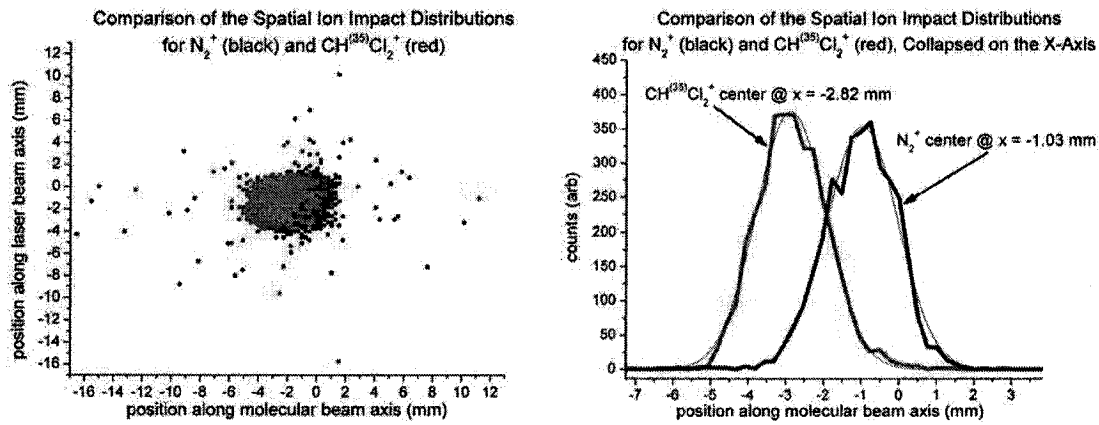


Figure 14: COMPARISON OF THE SPATIAL ION IMPACT DISTRIBUTIONS FOR N_2^+ (BLACK) AND $CH^{(35)}CL_2^+$ (RED)
Figure 14-A shows the ion impact distributions for N_2^+ from the background and $CH^{(35)}Cl_2^+$ ions from the molecular beam (from the graph's perspective the molecular beam is traveling in the negative *x*-axis direction). As expected the $CH^{(35)}Cl_2^+$ distribution is shifted in the direction of the molecular beam velocity. Although technically the $CH^{(35)}Cl_2^+$ aggregate is the result of a Coulomb explosion, a proton (H^+) accounts for merely 1% of dichloromethane's total mass, explaining why the two distributions in Figure 14-B have approximately the same spatial spread.

Using the entire data set would not reveal anything useful though – we would only see a giant circular blob about the size of the MCP. To determine v_{MB}, a proper choice of molecular beam ions from the TOF histogram, traveling at an average velocity v_{MB} in the molecular beam direction, must be plotted against background ions (whose average velocity is zero) on a 2-D spatial ion impact plot. N_2^+ was a convenient choice for background ions because it was not swamped in Coulomb explosion data (as can be seen in Figure 12-A and 12-C). The $CH^{(35)}Cl_2^+$ peak (in Figure 12-B and 12-D) was the only reasonable choice as it is the largest, and therefore contains just enough data for constructing healthy statistics. Strictly speaking it isn't an ion peak, since a proton (H^+) was ejected from the molecule; but due to its relatively small mass (~1% of the molecule's mass), hydrogen hardly transfers momentum to the remaining aggregate (actually this fact will be fruitfully exploited in the subsequent analysis). From the spatial separation Δ between the N_2^+ and $CH^{(35)}Cl_2^+$ distributions, an application of Newtonian mechanics $\left(v_{MB}=\Delta/\left(t^{(CH^{(35)}Cl_2^+)}-t_{TDC}\right)\right)$ leads to a molecular beam speed of $v_{MB} = 430.0$ *m/s*.

Having completely exhausted the benefits of ionization peaks, they must now be eliminated to avoid obfuscation of the secondary data created by the coincidence imaging algorithm developed during this work (we are interested in Coulomb explosion, not molecular ionization). Filtering out the ion data can be done in two ways:

1) temporal filtering – by directly removing hits lying within the temporal boundaries of the ion peak in the TOF histogram;

2) spatial filtering – by removing hits from the spatial ion impact distribution that are concentrated in the center of the 2-D spatial histogram.

Although the first method is straightforward, it needlessly eliminates the Coulomb explosion data by a significant amount, and also brings forth a temporal bias. Therefore the second method is the preferred one. After having eliminated the molecular ion data, one is left with a TOF histogram containing only Coulomb explosion humps without the ion peaks, as displayed in Figure 15.

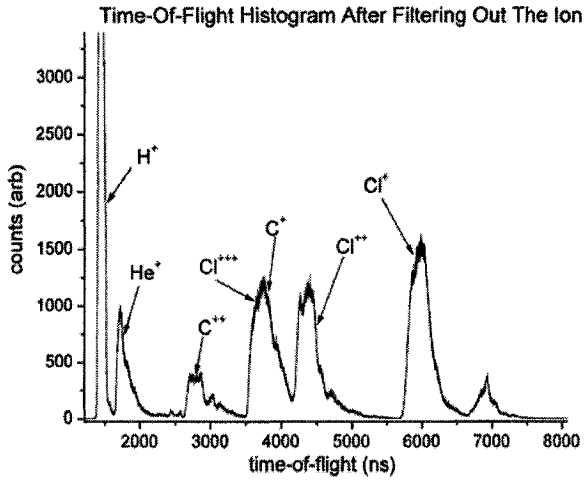


Figure 15: SPATIALLY FILTERED TOF HISTOGRAM OF COULOMB EXPLOSION DATA Removing the center of the spatial ion impact distribution effectively eliminates most of the ion data, as evidenced by Figure 15. In this case, data lying inside a circle of radius 16.0 *mm* about the center of the spatial ion impact distribution on the MCP have been removed.

This filtering strategy visibly eliminates some of the meaningful Coulomb explosion data (comparing Figure 15 with 12-C), but nevertheless is roughly unbiased with respect to time-of-flight. Moreover it helps to speed up the secondary analysis and disambiguate the secondary data.

C) The ARBCORR Coincidence Imaging Algorithm And Secondary Data Analysis

i) Coincidence Imaging And Retrieval Of 3-D Velocity Vectors

Once the TOF histogram and spatial ion impact distribution have been properly understood, the primary data given in the form of (t, x, y) triplets representing fragments hitting the detector and categorized into events corresponding to each laser shot, is ready to be used to build a list of conjectural molecules. These molecules are represented in terms of their constituent atoms. For example, a dichloromethane molecule would ideally generate the following data, in matrix form:

$v_x^{(H^1)}$	$v_y^{(H^1)}$	$v_z^{(H^1)}$	$m^{(H^1)}$	$q^{(H^1)}$
$v_x^{(H^2)}$	$v_y^{(H^2)}$	$v_z^{(H^2)}$	$m^{(H^2)}$	$q^{(H^2)}$
$v_x^{(C)}$	$v_y^{(C)}$	$v_z^{(C)}$	$m^{(C)}$	$q^{(C)}$
$v_x^{(Cl^1)}$	$v_y^{(Cl^1)}$	$v_z^{(Cl^1)}$	$m^{(Cl^1)}$	$q^{(Cl^1)}$
$v_x^{(Cl^2)}$	$v_y^{(Cl^2)}$	$v_z^{(Cl^2)}$	$m^{(Cl^2)}$	$q^{(Cl^2)}$

Figure 16: TABULAR REPRESENTATION OF SECONDARY DATA

According to this representation, the first three columns are the Cartesian components of each atom's velocity vector, and the last two are their mass and charge. The transformation from (t, x, y) fragment hits to (v_x, v_y, v_z) asymptotic fragment velocity vectors is again a straightforward application of ordinary kinematics:

$$(13) \qquad v_x = \frac{x - x_0}{TOF - t_{TDC}} - v_{MB},$$

$$(14) \qquad v_y = \frac{y - y_0}{TOF - t_{TDC}},$$

$$(15) \qquad v_z = \frac{1}{2}\frac{E}{m/q}\frac{\left((TOF_0 - t_{TDC})^2 - (TOF - t_{TDC})^2\right)}{TOF - t_{TDC}}.$$

Here, (x_0, y_0) is the center of the spatial ion impact data given by N_2^+ background ions (Figure 14-A); TOF is the ion impact time of the fragment in question, TOF_0 is the central time of its associated hump, and t_{TDC} the TDC acquisition offset time. When calculating v_x in the molecular frame, the beam speed v_{MB} must be taken into account. Aside from that, v_x and v_y derive from uniform motion in the xy-plane of the detector. For v_z, an equation that bypasses the flight distance h had to be constructed. As previously mentioned, h is difficult to directly measure in the laboratory due to the parabolic mirror's mobility. Instead, relation (8) shows that the central time TOF_0 of a Coulomb explosion hump is directly related to h, but is much more accurate because it is directly given by the detector.

From relations (13) and (14), the uncertainties on the timing signals from the delay lines (which are used for measuring the 2-D spatial impact coordinates of the fragment hit) are directly mapped to the v_x and v_y velocity components of the fragment in question. This means that the uncertainties on those velocity components are mostly due to the finite temporal resolution (0.5 *ns*) of the timing electronics. On the other hand, there are two sources of error that contribute to the uncertainty on the v_z velocity component. In addition to the time-of-flight signal of the fragment, the spectrometer's electric field (which is not strictly uniform due to technical limitations) appears explicitly in relation (15) and has an uncertainty associated to it as well. Essentially, the main contributors of experimental error on the velocities are the speed of the timing electronics and the uncertainty associated to the electric field. Error propagation, based on relations (13) to (15), yields uncertainties of ~100 *m/s* on the fragment velocities, translating into an uncertainty of $\sim 5\times10^{-24}$ *kg m/s* on fragment momentum.

A completely different type of experimental error originates from the fact that several molecules were exploded with each laser pulse. Due to the limitations of the detector and the timing electronics, a large part of the collected fragments do not form complete molecules. Rather, most of the primary data from the CEI chamber is made up of incomplete molecules. Nevertheless, it is possible to retrieve a set of molecules by using what is called *coincidence*

analysis. This technique consists of scanning through each event in the primary data and combinatorially mixing and matching the hits to build as many hypothetical molecules as can be allowed by the size of the event and previous knowledge about the times-of-flight of the fragments A number of conjectural molecules can be constructed from an event, depending on the richness of its content. However, most of these constructed molecules will prove to be unphysical due to momentum conservation requirements. The CEI data statistically contains all of the possible ionization and Coulomb explosion channels the laser pulse can provide. If one wishes to study a particular configuration of charge states of the molecule sent into the beam, a choice of time-of-flight intervals for each fragment must be made. This choice is not overwhelming since Figure 15 clearly demonstrates that there is an upper limit to the charge attained by any given fragment (this being $3e$ for the Cl^{+++} fragment in the present circumstances); and there is a finite number of fragments types depicted in Figure 15. Keeping in mind that higher charge states provide closer agreement with Coulomb repulsion, a requirement that is essential for analysis of the data based on classical mechanics, it is in our interest to feed the most highly-charged fragments into the secondary data analysis algorithm.

In order to make use of the primary (t, x, y) data I have developed an algorithm called *ARBCORR*, short for "arbitrary correlation", that reconstructs the molecules based on the form defined in Figure 16. Its implementation in the C programming language is given in Appendix A. It identifies exploded molecules in each event by correlating (t, x, y) data pertaining to individual fragments (themselves identified as real hits on the microchannel plate by the CEI2000 program), with respect to a user-specified set of fragments (Figure 17). In order for *ARBCORR* to function it requires the user to input an initial array of TOF values deduced empirically from the CEI2000 acquisition software's primary data.

The mass spectrum displayed in Figure 15 is sufficient to identify the fragments based on their charge states, at least for the present experiment. It contains well-separated charge states based on time-of flight data. However, there is an exception: the C^{+} and Cl^{+++} fragments are contained in the same Coulomb explosion hump due to their similar mass-to-charge ratios (12.01 and 11.82 respectively). Also, chlorine naturally abounds in two significant isotopes: $^{(35)}Cl$ (77.5%) and $^{(37)}Cl$ (22.5%). This mass discrepancy further blurs the Cl^{+++} TOF hump. Not much can be done in this case since all these fragments are the result of a Coulomb explosion, and therefore occupy similar areas of the spatial ion impact distribution (making spatial filtering impossible). Given that we are interested in the highest possible charge states for each fragment we tentatively search for the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{+++}$) molecule[12] despite the previous ambiguities.

At present the ARBCORR algorithm can only construct conjectural molecules based on the time-of-flight humps of each fragment (atom or otherwise), neglecting the spatial ion impact distribution altogether. Therefore the experimenter is responsible to completely

[12] Here we make use of the fact that hydrogen minimally affects the Coulomb explosion due to its small mass. Therefore, molecules will be constructed solely based on their carbon and chlorine constituents, keeping in mind that the momentum vector sum of the fragments will not add to zero, but some finite value provided by a computer simulation of the Coulomb explosion.

understand the content of the TOF histogram. To retrieve the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{+++}$) correlated group from the primary data the following array of values have to be fed to ARBCORR:

$t_1^{(C^{++})}$	$t_2^{(C^{++})}$	$m^{(C^{++})}$	$q^{(C^{++})}$
$t_1^{(^{(35)}Cl^{+++})}$	$t_2^{(^{(35)}Cl^{+++})}$	$m^{(^{(35)}Cl^{+++})}$	$q^{(^{(35)}Cl^{+++})}$
$t_1^{(^{(35)}Cl^{+++})}$	$t_2^{(^{(35)}Cl^{+++})}$	$m^{(^{(35)}Cl^{+++})}$	$q^{(^{(35)}Cl^{+++})}$

Figure 17: INPUT VALUES FOR THE ARBCORR ALGORITHM

The first two columns are the empirically determined time-of-flight intervals for each fragment, and the last two columns are their masses and charges. Although the input data in Figure 17 contains two identical rows, there is no redundancy here. It simply tells ARBCORR to look for three fragments, two of which have identical mass-to-charge ratios and therefore fall inside the same time-of-flight interval $\left[t_1^{(^{(35)}Cl^{+++})}, t_2^{(^{(35)}Cl^{+++})}\right]$. Alternately, by providing only the first two rows, ARBCORR would understand a request for molecules giving off the (C^{++}, $^{(35)}Cl^{+++}$) fragment group. In general, the time-of-flight ranges for different fragments fed to ARBCORR are allowed to overlap, as this would otherwise impose artificial (unphysical) temporal boundaries for the data analysis and antagonize the "don't waste data" motto. The price to pay for adopting this approach is an increase in algorithmic complexity for the coincidence imaging due to more involved combinatorics in treating the data situated in the overlapping regions between temporal ranges.

The ARBCORR algorithm essentially goes through the primary (t, x, y) data in the lmf files and attempts to reconstruct as many molecules as it can based on the user-defined quantities such as those in Figure 17. The reconstructed molecules form the secondary data, consisting of a list of arrays that are represented in the form given in Figure 16. These conjectural molecules are recorded as a continuous list, since the experimental parameters (laser pulse intensity, molecular beam size and density, spectrometer field strength, etc.) are held constant throughout.

For an event to be considered by ARBCORR it must contain at least one fragment falling between each of the user-defined temporal ranges. If this is not the case then a complete molecule cannot be found, and the entire event is ignored. If the event contains exactly one potential molecule in it, then this potential molecule is recorded as part of the secondary data and no further analysis is performed on the event. If the event is large enough to contain several molecules then the analysis quickly becomes messy. Consider for example the following situation:

Fragment Species	*User-Defined Time Intervals (ns)*
C^{++}	[1297, 1479]
$^{(35)}Cl^{+++}$	[1752, 2101]
$^{(35)}Cl^{+++}$	[1752, 2101]

A

B

Conjectural Molecules Reconstructed Under ARBCORR (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{+++}$)
$\{h_3, h_1, h_5\}$
$\{h_3, h_1, h_6\}$
$\{h_3, h_5, h_6\}$
$\{h_4, h_1, h_5\}$
$\{h_4, h_1, h_6\}$
$\{h_4, h_5, h_6\}$

C

Event Contents	*Ion Impact Times Only (ns)*
h_1	1861
h_2	883
h_3	1349
h_4	1441
h_5	1917
h_6	1941

Figure 18: POSSIBLE SITUATION ENCOUNTERED BY ARBCORR
Table 18-A displays the user-defined time intervals for each requested fragment. Table 18-C represents an unusually large event in an lmf file. The real nature of the hits h1...h6 has yet to be determined by applying physical constraints on the coincidence analysis data of Table 18-B.

Table 18-C potentially contains parts of two molecules (there is only one carbon atom in dichloromethane, but two hits fall inside the C^{++} TOF window) due to the presence of two hits, h_3 and h_4, falling inside the C^{++} interval. However, only one (at most) of these can be properly constructed since merely three hits are identified as chlorine fragments (hits h_1, h_5 and h_6). Nonetheless, a total of 2 C(3,2) = 6 molecules will be conjectured and added to the secondary data set as a result of the "don't waste data" motto. In the worst case, this event could be completely useless if all three chlorine fragments come from different parent molecules, exposing the danger of overwhelming the data acquisition.

What's more, the exemplary event in Figure 18-C would yield 2 P(3,2) = 24 conjectural molecules if a search for the isotopically different (C^{++}, $^{(35)}Cl^{+++}$, $^{(37)}Cl^{+++}$) fragment group was performed. In this case the chlorine atoms are non-identical but close enough in mass to meld under the same hump in the TOF histogram. For the sake of rigor the ARBCORR algorithm then has to make each of the hits h_1, h_5 and h_6 assume the nature of both chlorine isotopes, further expanding the combinatorics. Thus, the coincidence analysis is made increasingly complex when three or more fragment species coincide in the same TOF hump, which also means that the data is more prone to be spoiled by false coincidences. Choosing a heterogeneous fragment group with well separated TOF distributions, such as (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$), would alleviate this problem. Of course, the $^{(35)}Cl$ isotope is preferred over $^{(37)}Cl$ due to its greater natural abundance (77.5% and 22.5% respectively).

The previous example illustrates the difference between conjectural molecules (making up the false coincidences in the secondary data) and factual molecules (the ones that

are shown to be consistent with physical laws, namely momentum conservation). These molecules are indiscriminately listed one after another, in the format given by Figure 16, as ARBCORR crunches through the primary lmf data. Since the velocities of the molecules are given in the molecular beam frame, it is sensible to build a histogram (Figure 19) compiling their momentum magnitudes, calculated by vector-summing that of their constituent fragments.

ii) Momentum Histograms And Molecular Properties

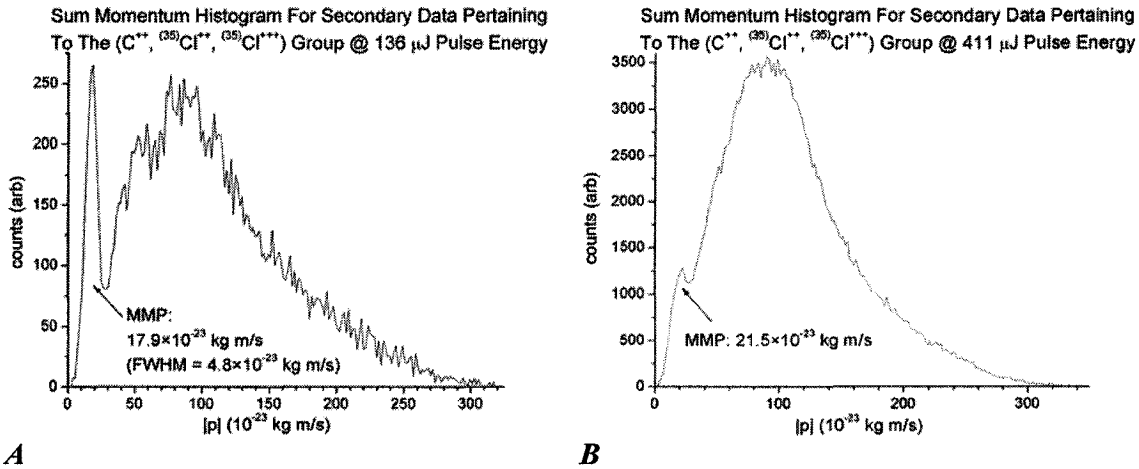


Figure 19: MOMENTUM SUM HISTOGRAM FOR (C^{++}, $^{35}Cl^{++}$, $^{35}Cl^{+++}$) WITH 136 μJ AND 411 μJ PULSES
Momentum conservation provides the defining test that distinguishes between real and artificial molecules in the secondary data set. In Figure 19-A the sharp peak centered at 1.79×10^{-22} *kg m/s* represents the total momentum (in magnitude) of the factual molecules, piercing through the overwhelming amount of false coincidences making up the rest of the distribution (in this case 89.7% of the total data represents false coincidences). If further results are to be obtained from data in this peak, additional filters can be employed to reduce the amount of background data (~33% of the peak in 19-A) resulting from false coincidences. These additional filters can consist of plots of other velocity-related quantities, such that the false coincidences are visibly separated from the true data. This strategy can be especially useful when false coincidences become overwhelming, as in Figure 19-B. Here, the molecules were exploded using 411 μJ circularly polarized laser pulses (~25 PW/cm^2). Due to a heavy data acquisition rate and previously mentioned (section IV-A) shortcomings arising from the use of excessively powerful pulses, the relevant secondary data is swamped in unwanted false coincidences, and so does not yield clear information about the molecule. Although there is a vast quantity of data in total, false coincidences occupy 95.3% of the data, rendering it virtually unworkable.

The resulting histogram displays a sharp peak representing the physical molecules (this peak will henceforth be called the *minimal momentum peak*, abbreviated as MMP), and a broad distribution containing most of the data. Our next objective is to show that the MMP contains the actual molecules, whereas the rest of the distribution is erroneous.

Because the molecules are incomplete their momenta add to a finite value (the remaining 1.8×10^{-22} *kg m/s* is carried away by the hydrogen ions, in agreement with the numerical CEI propagation program in Appendix B). To further understand the nature of the MMP a momentum histogram can be constructed from ARBCORR's coincidence analysis by looking for a complete dichloromethane molecule based on several different charge state configurations. Unfortunately, this approach does not yield a significant amount of secondary data due to a lack of large events (Figure 11), particularly those which involve the C^{++} ion (as

evidenced by Figures 12-A and 12-C)[13]. As a general rule, ions chosen from the smallest TOF hump set the bottleneck for the amount of secondary data that can be generated (if only 20 C^{++} ions have been collected, then no more than 20 parent molecules can be constructed). Given this constraint I was forced to choose lower charge states for the five-fragment coincidence analysis, also keeping in mind that choosing ions from distinctive TOF humps helps to simplify ARBCORR's combinatorics and thus reduce the number of false coincidences.

Figure 20 shows the momentum sum histograms yielding the complete dichloromethane molecule. Despite the overwhelming number of false coincidences, a small peak can be observed at the leftmost side of each graph. As a matter of fact, the data falling under these MMP's will be used for the molecular geometry retrieval given in section IV-C-iii. As expected, the momentum peaks for the full CH_2Cl_2 retrieval are located at a much smaller value compared to those in Figure 19 - the vector-summed momentum is simply that of the molecule itself. Furthermore, the spatial spread in ionized $CH^{(35)}Cl_2^+$ (going back Figure 14) can be used to estimate the velocity spread in the molecular beam. In section IV-B, the center of this spatial ion impact distribution was used to calculate the molecular beam speed. Because this fragment is almost a molecular ion, the spread in the $CH^{(35)}Cl_2^+$ spatial distribution gives an approximate value of $|p| = 4.2\times10^{-23}$ *kg m/s* for the average momentum of dichloromethane in the molecular beam.

The discrepancy between the measured total momentum and its ideally null value stems from the thermal velocity spread in the molecular beam. That is, the use of v_{MB} to translate our perspective into the molecular beam frame of reference gives a zero momentum sum in the beam direction only for molecules traveling at precisely the mean molecular beam speed. In actual fact, the lateral and longitudinal speeds of molecules in the beam both follow a distribution given by the kinetic theory of gases (formula (9)). In other words, most molecules are not stationary in the lateral direction and do not travel at a velocity v_{MB} in the longitudinal direction. Hence the momentum sum histogram (which is a distribution of momentum *magnitudes*) should produce a small but finite momentum peak for the true coincidences. Moreover, the finite value of this peak is directly related to the spread in velocity vectors – a larger velocity spread makes the MMP shift to a higher value. This discrepancy can be used to gauge the experimental error on the momentum sum for the coincidence analyses of incomplete molecules (Figure 19-A and 19-B).

[13] For the purpose of determining molecular handedness, merely three fragments need be collected, so the approach given by Figure 19 is quicker and provides much better statistics. The reason for five-fragment coincidence analysis given here is to underline the nature of the MMP.

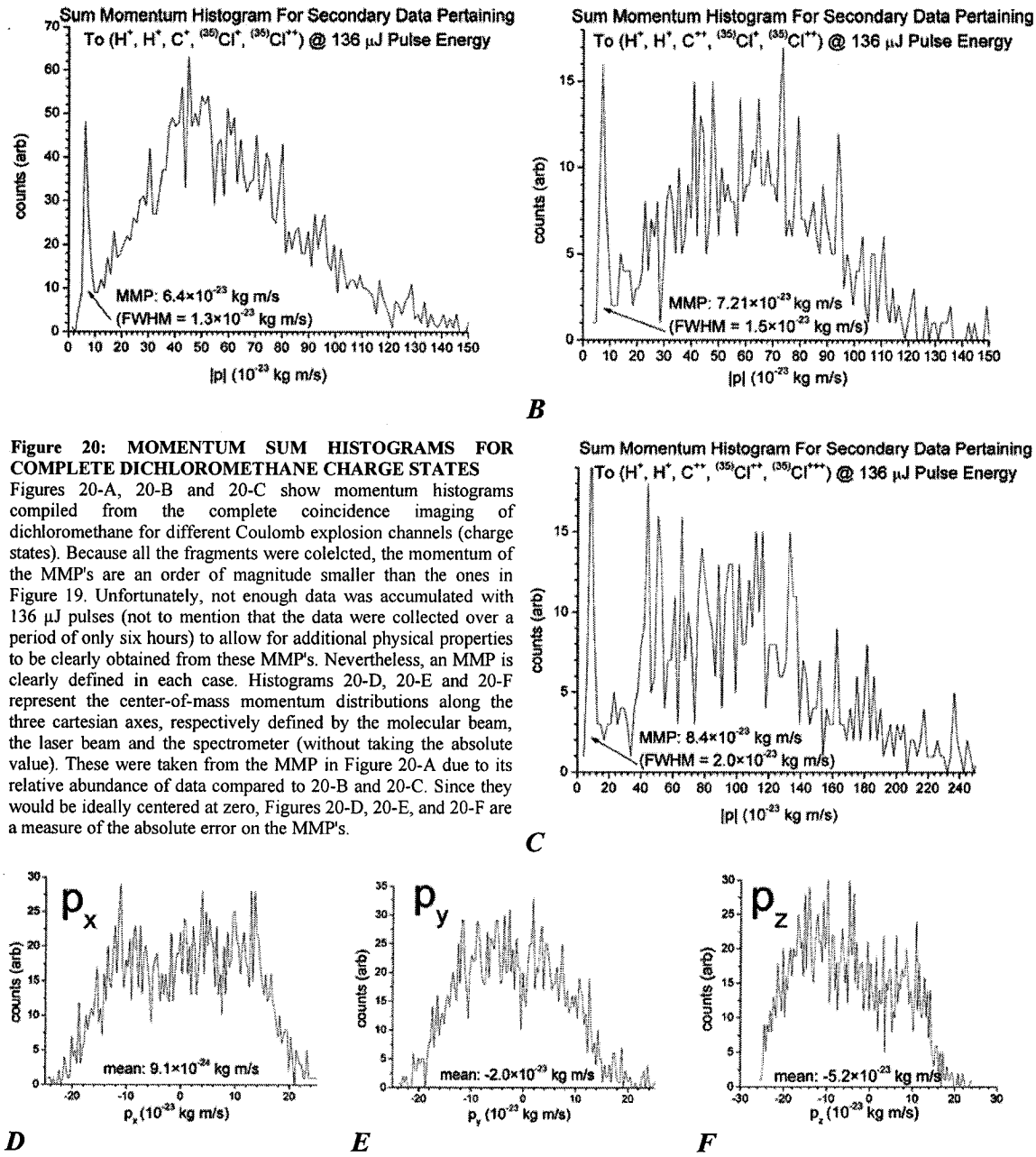


Figure 20: MOMENTUM SUM HISTOGRAMS FOR COMPLETE DICHLOROMETHANE CHARGE STATES
Figures 20-A, 20-B and 20-C show momentum histograms compiled from the complete coincidence imaging of dichloromethane for different Coulomb explosion channels (charge states). Because all the fragments were colelcted, the momentum of the MMP's are an order of magnitude smaller than the ones in Figure 19. Unfortunately, not enough data was accumulated with 136 μJ pulses (not to mention that the data were collected over a period of only six hours) to allow for additional physical properties to be clearly obtained from these MMP's. Nevertheless, an MMP is clearly defined in each case. Histograms 20-D, 20-E and 20-F represent the center-of-mass momentum distributions along the three cartesian axes, respectively defined by the molecular beam, the laser beam and the spectrometer (without taking the absolute value). These were taken from the MMP in Figure 20-A due to its relative abundance of data compared to 20-B and 20-C. Since they would be ideally centered at zero, Figures 20-D, 20-E, and 20-F are a measure of the absolute error on the MMP's.

To find out if this choice of experimental error is justified we compare the momentum sum (only for the MMP) to a theoretical value obtained by a deterministic computer simulation of classical Coulomb explosion dynamics (Appendix B). According to the simulation, the charge state configuration (C^{++}, $^{(35)}Cl^{++}$, $^{(35)}Cl^{+++}$) should possess a total momentum of 2.42×10^{-22} *kg m/s* in the molecular center-of-mass frame. This value was obtained after numerically propagating the classical Coulomb explosion over 100,000 iterations, using a time step of 2.7 femtoseconds, and ending the simulation with final

interatomic distances of tens of nanometers. Thus the absolute experimental error due to the momentum spread in Figure 19-A (as compared to the calculated value) is $\Delta p = |2.42 \times 10^{-22} - 1.80 \times 10^{-22}| = 0.62 \times 10^{-22}\ kg\,m/s$, which is reasonably close to the MMP value in Figure 20-A ($0.64 \times 10^{-22}\ kg\ m/s$).

Furthermore, the MMP's in Figure 20 provide an adequate measure of the temperature of the molecules in the beam, following their Coulomb explosion. Due to the molecular beam's cylindrical symmetry (in velocity space, not coordinate space), it is convenient to think of this temperature as being composed of a longitudinal component T_{LONG} and a transverse component T_{TRAN}, each associated to their respective speed distributions,

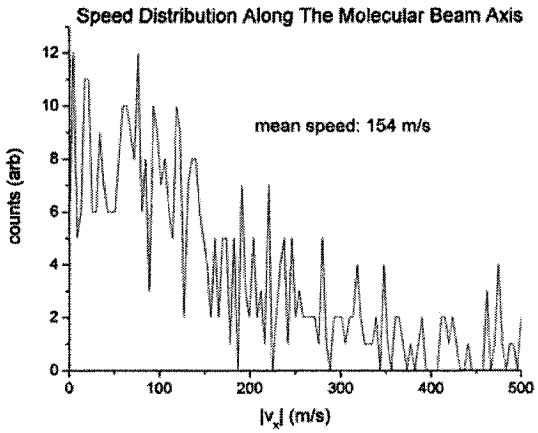
$$(16) \quad T_{LONG} = \frac{\pi M \bar{v}_{LONG}^{\,2}}{8R} \qquad (17) \quad T_{TRAN} = \frac{\pi M \bar{v}_{TRAN}^{\,2}}{8R}$$

$$(18) \quad f_{LONG}(v_{LONG}) = 4\pi \left(\frac{M}{2\pi R T_{LONG}} \right)^{\frac{3}{2}} v_{LONG}^2 e^{-\frac{1}{2}\frac{M v_{LONG}^2}{R T_{LONG}}}$$

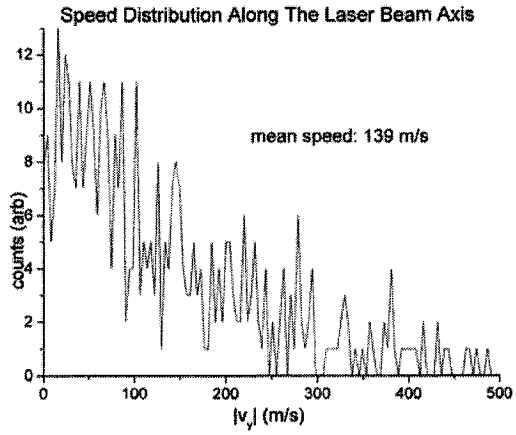
$$(19) \quad f_{TRAN}(v_{TRAN}) = 4\pi \left(\frac{M}{2\pi R T_{TRAN}} \right)^{\frac{3}{2}} v_{TRAN}^2 e^{-\frac{1}{2}\frac{M v_{TRAN}^2}{R T_{TRAN}}} .$$

It must be noted that these temperatures do not necessarily represent the molecular beam's temperature, they represent the temperature of the molecules after the passing of the laser pulse. In the limit of extremely short and intense pulses, electrons are ejected by tunneling through the barrier due to the laser field acting on the Coulomb potential of the molecule. The subsequent drift momentum acquired by the electrons from the ponderomotive force of the laser field is on the order of $\sim 0.24 \times 10^{-23}\ kg\ m/s$ (corresponding to an energy of about 20 eV). Being also charged, the parent ion could have also acquired a similar drift momentum due to its motion in the laser field, before the event of Coulomb explosion. Furthermore, since the ponderomotive energy depends on the electric charge $(U_p \propto q^2)$, this could explain the systematic increase in momentum values for greater charge states, from Figures 20-A to 20-C.

Nevertheless, approximate experimental values for the translational temperatures can be obtained by plotting the speed distributions for molecules in the MMP's along the axes defined by the molecular beam, the laser beam and the spectrometer (these are respectively the x, y and z axes), and by making use of the kinetic theory relations (16) and (17).

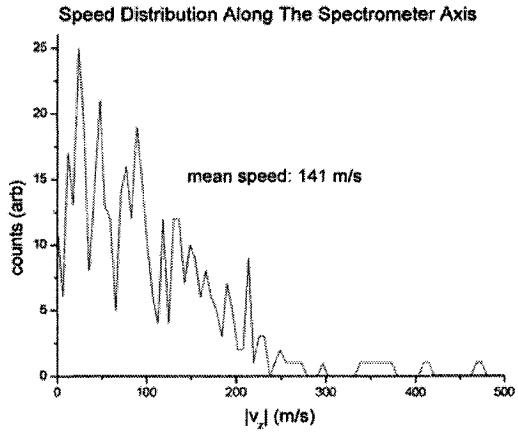


A



B

Figure 21: MOLECULAR SPEED DISTRIBUTIONS For The (H^+, H^+, C^{++}, $^{(35)}Cl^{++}$, $^{(35)}Cl^{+++}$) SPECIFICATION
Molecules being accelerated into a vacuum experience a sudden decrease in pressure. The ensuing drop in temperature makes their transverse and longitudinal velocity distributions more compact.



C

Figures 21-A, 21-B and 21-C were calculated from the MMP of a previous experiment performed with 250μJ pulses over twelve hours, and yield temperatures of T_x = 94.1 *K*, T_y = 76.7 *K* and T_z = 78.9 *K* respectively. Due to the molecular beam's cylindrical symmetry, T_y and T_z have similar values (the laser beam and spectrometer axes lie in the molecular beam's cross-sectional plane). These temperatures are much higher than those measured by Dr. Dooley (~12 *mK*), the PATRICK instrument's creator, and H.J. Lewandowski *et al.*[47] (~1.3 *K*), which would suggest that the laser pulse did affect the translational motion of the molecules during the Coulomb explosion. Although the rotational and vibrational temperatures were not measured, it is possible to estimate their values based on knowledge of molecular dynamics. For one, since rotational energy levels are spaced close together, molecules have no trouble accessing higher rotational states. This means that translation→rotation interactions are just as likely to happen as translation→translation interactions. Consequently, the rotational temperature should be similar to the translational temperature. On the other hand, vibrational energy levels are spaced at larger intervals. Therefore, high vibrational states are more difficult to occupy, and so translation/rotation→vibration interactions are less likely to happen. As a result, the

vibrational temperature, although still correlated with rotational and translational temperatures, should be much lower.

Turning back to the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$) specification, due to its clean segregation of conjectural and factual molecules (Figure 19-A), we can proceed to the evaluation of kinematic properties of the individual fragments, which is sufficient for determining handedness (as will be described in section IV-C-iv). Thanks to the ordering of secondary data into lists of fragment groups, calculating molecular information based on fragment velocity vectors is quite straightforward.

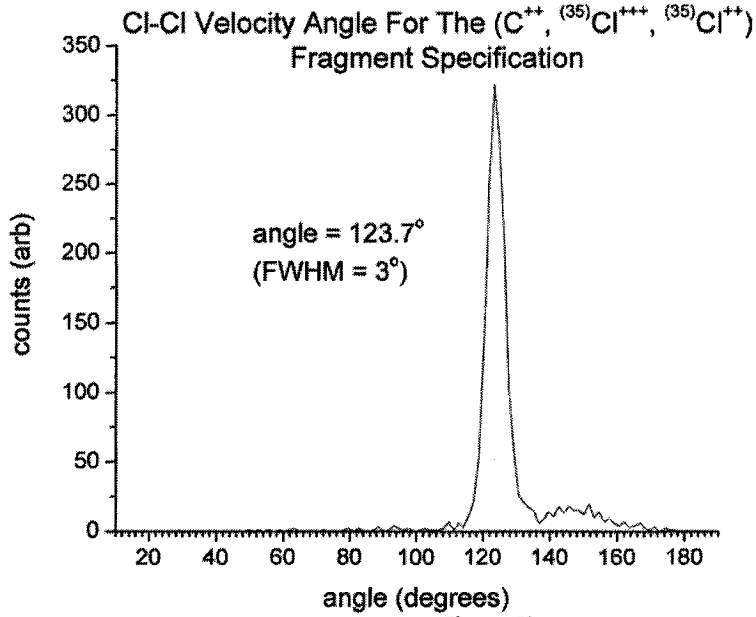
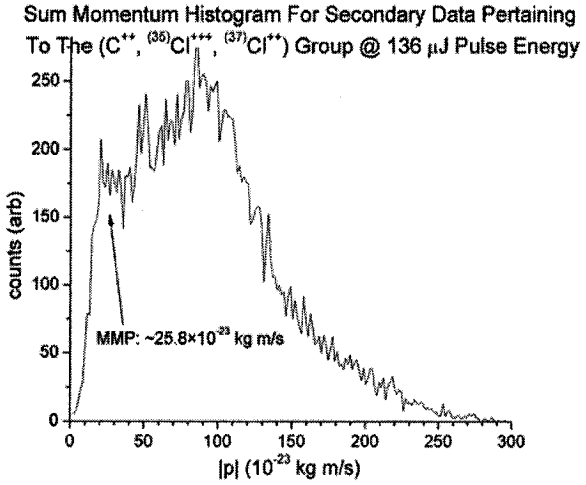


Figure 22: Cl-Cl VELOCITY ANGLE FOR (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$)
Indeed, the histogram in Figure 19-A yields a precise (FWHM = 6°) and accurate value for the Cl-Cl velocity angle (the computer simulated Coulomb explosion gives a velocity angle of 123.0°).

To see if the secondary data makes any sense at all, a histogram of some kinematic properties of the fragments – kinetic energy, angle between velocity vectors, etc. - can be constructed. For example, a histogram of Cl-Cl velocity angles (Figure 22) is made by selecting the fragment groups under the MMP of Figure 19-A and calculating the angle between their chlorine velocity vectors using $\theta_{Cl^{++},Cl^{+++}}=\arccos\left(\frac{\vec{v}_{Cl^{++}}\cdot\vec{v}_{Cl^{+++}}}{\left|\vec{v}_{Cl^{++}}\right|\left|\vec{v}_{Cl^{+++}}\right|}\right)$.

The origin of the small hump centered at ~149° in Figure 22 is somewhat equivocal. Perhaps it is a manifestation of the finite amount of $^{(37)}Cl$ isotopes that found their way into the final momentum-filtered data. More specifically, since these two isotopes share the same TOF hump, both of them are assigned a mass of 35 when a coincidence analysis under the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$) specification is performed. The relative mass difference between $^{(37)}Cl$ and $^{(35)}Cl$ is small enough that the momentum sum of molecules containing the $^{(37)}Cl$ isotope

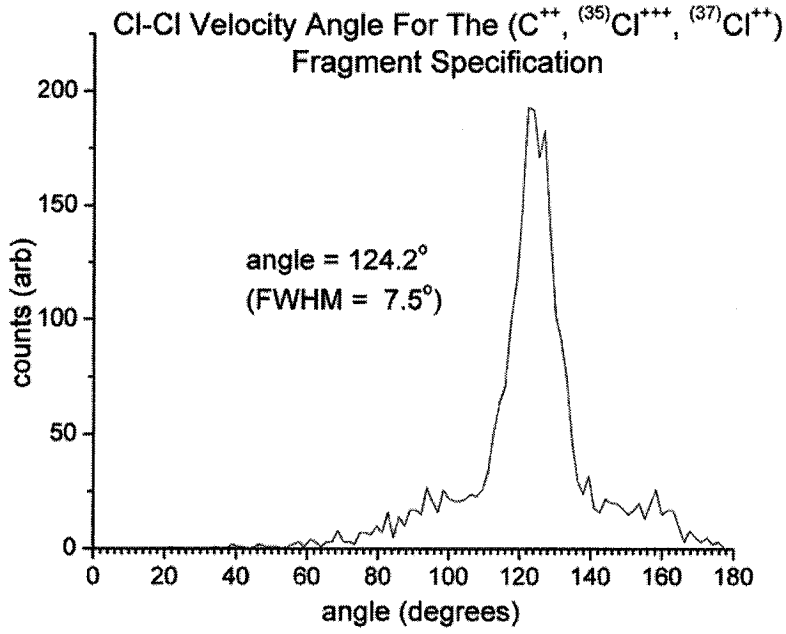
is close enough to the momentum of (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$) that it effectively melds in with the MMP. To verify if this hypothesis is valid, the same analysis can be carried out based on the slightly heavier, but less common, (C^{++}, $^{(35)}Cl^{+++}$, $^{(37)}Cl^{++}$) charge state configuration (Figure 23).



A

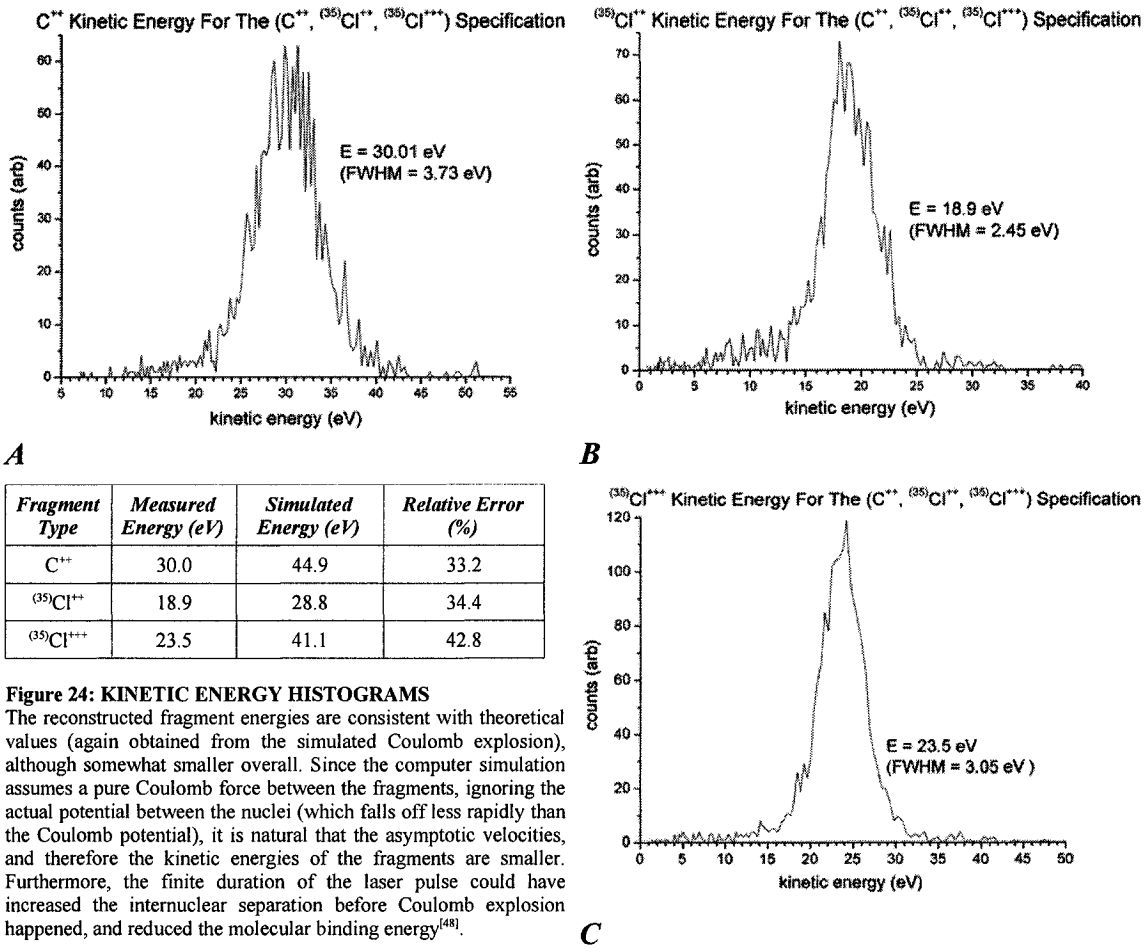
Figure 23: Cl-Cl ANGLE MEASUREMENT FOR THE (C^{++}, $^{(35)}Cl^{+++}$, $^{(37)}Cl^{++}$+) CHARGE STATE

Figure 23-A vaguely shows an MMP centered at roughly 2.58×10^{-22} *kg m/s* (the associated theoretical value is 2.30×10^{-22} *kg m/s*). It is flat at the top and slightly wider than the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{+++}$) MMP. This seems to be the result of ARBCORR assigning a mass of 37 to fragments which were actually $^{(35)}Cl^{++}$. Since $^{(35)}Cl^{++}$ travels faster than $^{(37)}Cl^{++}$ after a Coulomb explosion, such a false mass assignment would over-estimate its momentum (by a theoretical amount of 0.32×10^{-23} *kg m/s* in this case). As a consequence, these false coincidences create an artificial momentum peak right next to the MMP. And since no clear distinction exists between these two, both are allowed to pass through the momentum filter, resulting in the "noisy" angle distribution of Figure 22-B. Nevertheless, the measured Cl-Cl angle is in good agreement with the calculated one (123.3°).



B

Apart from velocity angles, fragment kinetic energies can be readily acquired from the secondary data as well. Reverting back to the (C^{++}, $^{(35)}Cl^{+++}$, $^{(35)}Cl^{++}$) specification for its exemplary data (Figure 19-A), these are displayed in Figure 24.



Fragment Type	*Measured Energy (eV)*	*Simulated Energy (eV)*	*Relative Error (%)*
C^{++}	30.0	44.9	33.2
$^{(35)}Cl^{++}$	18.9	28.8	34.4
$^{(35)}Cl^{+++}$	23.5	41.1	42.8

Figure 24: KINETIC ENERGY HISTOGRAMS
The reconstructed fragment energies are consistent with theoretical values (again obtained from the simulated Coulomb explosion), although somewhat smaller overall. Since the computer simulation assumes a pure Coulomb force between the fragments, ignoring the actual potential between the nuclei (which falls off less rapidly than the Coulomb potential), it is natural that the asymptotic velocities, and therefore the kinetic energies of the fragments are smaller. Furthermore, the finite duration of the laser pulse could have increased the internuclear separation before Coulomb explosion happened, and reduced the molecular binding energy[48].

The relative importance of Coulomb explosion channels depends on the intensity of the field driving the molecules. A very strong field will produce many highly charged fragments falling at low impact times, whereas a weak field will favor the lower charge states. If the field is small enough, the molecules will break apart into larger conglomerates made up of several atoms, whose lifetimes depend on their bond structures and bond energies. Such weak Coulomb explosions happen in the very low-intensity regions of the molecule-laser interaction volume; examples of them are the CH_2Cl^+ and $CHCl_2^+$ humps in the rightmost regions of Figure 12. To see if these hits can be correlated to a molecule we can attempt to calculate some properties of the molecules created from the ($CH_2{}^{(35)}Cl^+$,$^{(35)}Cl^+$) fragment specification (Figure 25).

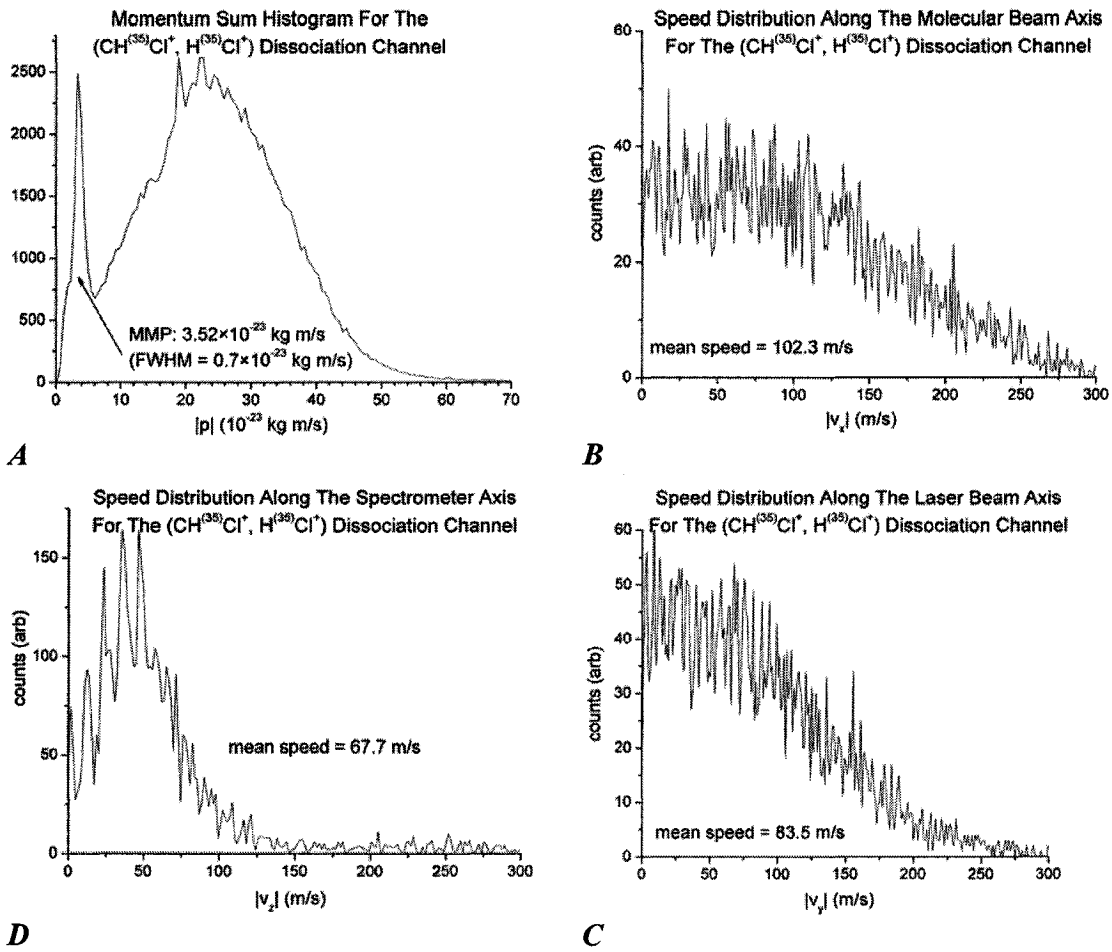


Figure 25: MOMENTUM SUM HISTOGRAMS FOR WEAK COULOMB EXPLOSIONS
These graphs show the speed distributions obtained from the weak Coulomb explosion of $CH^{(35)}Cl^+$ and $H^{(35)}Cl^+$ fragments. They are to be compared to those obtained with respect to a "complete" Coulomb explosion, as displayed in Figure 21.

Figures 25-B, 25-C and 25-D yield lower temperatures (41.5 *K*, 27.7 *K*, and 18.2 *K* respectively) than those found in Figure 21 (95 *K*, 75 *K*, and 78.9 *K*). This is because the molecules that underwent a weak Coulomb explosion were those those that were exposed to the weaker part of the EM field, leaving them less disturbed after their interaction with the laser pulse. It must be said that, unlike atomic ions, charged fragments coming from a weak Coulomb explosion do have noticeable internal structure. This means that part of the molecule's binding energy can be given to them in the form of vibrational and rotational motion. Treating these polyatomic fragments as point particles clearly overestimates their translational kinetic energy. A better analysis of weak Coulomb explosions should take into account the internal structure of the fragments.

iii) Reconstruction Of Molecular Geometry

The above quantities – velocity angles, kinetic energies and temperatures – were directly given by the fragment velocities in the secondary data through coincidence imaging. Yet, the objective of CEI is to deliver actual information about molecular structure, such as bond angles and bond distances. A molecular reconstruction has been successfully achieved with SO_2 and D_2O[42], but this endeavor quickly becomes a hassle when trying to obtain information about large molecules. This is mainly due to the fact that the retrieval process is not unique, i.e. classical motion for a system of N particles is ill-defined if less than $6N$ coordinates are specified. As a result, there are potentially many initial configurations of charged fragments that would satisfy a particular set of velocity vectors. Fortunately, an approximate notion of the geometry of the molecule can be used to guess an initial shape, which can then be refined by feeding it into a suitable global optimization algorithm.

To tackle this problem with respect to the CH_2Cl_2 molecule, a simulated annealing (SA) algorithm has been implemented in C (Appendix C) to carry out the global optimization of the molecular configuration. First however, it is necessary to choose a quantity to be minimized. In this case, we want to find an initial spatial configuration whose calculated asymptotic velocities correlate well with the empirical ones. Therefore, the quantity to be minimized by the SA algorithm must be a measure of the discrepancy between the calculated and measured asymptotic velocity vectors. For the purpose of this analysis I chose to minimize the maximum difference between theoretical and empirical fragment velocities, i.e. $\Delta = Max\left(\frac{\left|\vec{v}_{theo}^{(i)} - \vec{v}_{emp}^{(i)}\right|}{\left|\vec{v}_{theo}^{(i)}\right|}\right)$, where i is an index ranging over the fragments. Although this condition is quite stringent (for example, one could use the average velocity difference instead of the maximum difference), it allows for peace of mind as it provides an upper limit for the uncertainty of the optimized molecular configuration. The empirical velocities are taken from a particular molecule in the secondary data of Figure 20-B[14]. A reasonable candidate was one of the molecules lying at the center of the MMP, since these ones are good representatives of the true coincidences. As in most probabilistic algorithms, the next thing to do is make a guess about the initial positions of the atoms.

Unfortunately, we are dealing with a simple molecule (compared to larger organic molecules containing hundreds of atoms, for example), making it easy to guess a shape that will allow the algorithm to converge quickly to the optimized configuration. In order to remain unbiased, one can choose the initial positions systematically from the measured velocity vectors instead of relying on previous knowledge of the shape of the molecule (for simple molecules the velocity vectors, when properly scaled, do a pretty good job of estimating the initial atomic position vectors). This way, no intuition-driven guess-work

[14] Unlike the previous calculations that were shown earlier in this section, the SA method will provide direct structural information about one individual molecule (the one chosen from the secondary data). This is in contrast to the tabular Monte Carlo scheme employed by *F. Légaré et al.*[37], which provides a statistical distribution of reconstructed molecules in space, thereby allowing for a direct evaluation of the uncertainty associated to the reconstruction, and promptly generating a distribution of molecular distributions.

trivializes the analysis. As previously discussed, the individual molecules chosen for this examination probably were not traveling longitudinally at the molecular beam speed, and were not transversely stationary. So despite our efforts to translate our perspective into the molecular beam frame, we are nevertheless observing individual molecules in a frame of reference that was statistically determined (the molecular beam frame is not the same as the individual molecule's frame due to a thermal spread of velocities). The strategy to use asymptotic velocities as an approximation for initial coordinates is more fruitful when velocities are expressed in the molecule's center-of-mass reference frame. This is done by calculating the center-of- mass velocity of the molecule based on the individual fragment velocities.

When a systematically calculated guess of the initial configuration has been made, the SA algorithm attempts to minimize Δ by letting the atoms participate in Brownian motion with a variable Gaussian-distributed step size. At each iteration of the algorithm a numerical simulation of a Coulomb explosion is calculated from the instantaneous atomic positions. Once the difference Δ has been calculated by comparing the simulated asymptotic velocities to the empirical ones, the rest of the SA algorithm makes use of artificial thermodynamic cooling and heating gimmicks to artificially control the convergence of the optimization. From their initially guessed positions, the SA algorithm allows the atoms to explore the space surrounding them, all the while maintaining controlled constraints ensuring their increasing agreement with the empirical velocities. An exhaustive search of molecular configurations is terminated when the algorithm stagnates. This generally means that a local minimum (perhaps not the global one) of sufficient depth (given by the tolerance of Δ as defined by the user) has been detected and recognized as being the closest approximation of the ideal reconstruction of the molecule, given the accuracy of the initial configuration. Examples of molecules recovered with this algorithm are listed in the table of Figure 27, and the SA algorithm's calculations are illustrated in Figure 26.

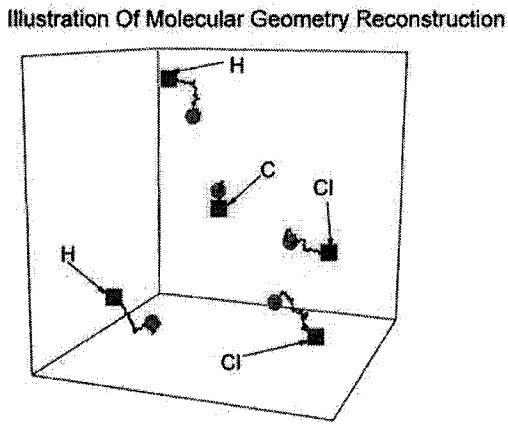


Figure 26: MOLECULAR GEOMETRY RECONSTRUCTION
This diagram demonstrates the 3-D stochastic motion imposed on the CH_2Cl_2 atoms (small black dots), from their initially guessed positions (red circles) until a satisfactory molecular configuration has been found (blue squares). In this case, a value of Δ = 2.32% has been achieved.

Molecule	*Structural Property*	*Retrieved Value*	*Starting Value*
A ***Δ = 0.25%***	Cl^{++}-C^{++}-Cl^{+++} bond angle	122.5°	39.2°
	$H^{+}_{(1)}$-C^{++}-$H^{+}_{(2)}$ bond angle	101.8°	130.2°
	Cl^{++}-C^{++} bond length	2.03 Å	0.77 Å
	Cl^{+++}-C^{++} bond length	3.08 Å	0.78 Å
	$H^{+}_{(1)}$-C^{++} bond length	2.98 Å	1.72 Å
	$H^{+}_{(2)}$-C^{++} bond length	2.93 Å	1.61 Å
	angle between (Cl^{++}-C-Cl^{+++}) and ($H^{+}_{(1)}$-C-$H^{+}_{(2)}$) planes	70.5°	79.3°
B ***Δ = 0.10%***	Cl^{++}-C^{++}-Cl^{+++} bond angle	39.0°	38.0°
	$H^{+}_{(1)}$-C^{++}-$H^{+}_{(2)}$ bond angle	131.0°	138.1°
	Cl^{++}-C^{++} bond length	4.73 Å	0.75 Å
	Cl^{+++}-C^{++} bond length	2.72 Å	0.77 Å
	$H^{+}_{(1)}$-C^{++} bond length	2.26 Å	1.62 Å
	$H^{+}_{(2)}$-C^{++} bond length	2.74 Å	1.77 Å
	angle between (Cl^{++}-C-Cl^{+++}) and ($H^{+}_{(1)}$-C-$H^{+}_{(2)}$) planes	102.2°	102.1°
C ***Δ = 0.13%***	Cl^{++}-C^{++}-Cl^{+++} bond angle	47.1°	42.3°
	$H^{+}_{(1)}$-C^{++}-$H^{+}_{(2)}$ bond angle	20.3°	87.8°
	Cl^{++}-C^{++} bond length	2.65 Å	0.85 Å
	Cl^{+++}-C^{++} bond length	1.92 Å	0.94 Å
	$H^{+}_{(1)}$-C^{++} bond length	4.07 Å	1.40 Å
	$H^{+}_{(2)}$-C^{++} bond length	2.19 Å	1.59 Å
	angle between (Cl^{++}-C-Cl^{+++}) and ($H^{+}_{(1)}$-C-$H^{+}_{(2)}$) planes	101.8°	104.0°
D ***Δ = 0.12%***	Cl^{++}-C^{++}-Cl^{+++} bond angle	43.8°	37.7°
	$H^{+}_{(1)}$-C^{++}-$H^{+}_{(2)}$ bond angle	171.3°	160.6°
	Cl^{++}-C^{++} bond length	4.03 Å	0.78 Å
	Cl^{+++}-C^{++} bond length	2.49 Å	0.84 Å
	$H^{+}_{(1)}$-C^{++} bond length	2.07 Å	1.82 Å
	$H^{+}_{(2)}$-C^{++} bond length	2.22 Å	2.30 Å
	angle between (Cl^{++}-C-Cl^{+++}) and ($H^{+}_{(1)}$-C-$H^{+}_{(2)}$) planes	110.6°	91.1°
E ***Δ = 0.02%***	Cl^{++}-C^{++}-Cl^{+++} bond angle	65.0°	39.3°
	$H^{+}_{(1)}$-C^{++}-$H^{+}_{(2)}$ bond angle	93.4°	116.0°
	Cl^{++}-C^{++} bond length	2.88 Å	0.72 Å
	Cl^{+++}-C^{++} bond length	3.03 Å	0.81 Å
	$H^{+}_{(1)}$-C^{++} bond length	2.86 Å	1.64 Å
	$H^{+}_{(2)}$-C^{++} bond length	2.33 Å	1.57 Å
	angle between (Cl^{++}-C-Cl^{+++}) and ($H^{+}_{(1)}$-C-$H^{+}_{(2)}$) planes	98.6°	98.4°

Figure 27: SEVERAL RECONSTRUCTED MOLECULAR GEOMETRIES

This table lists the reconstructed geometries of five molecules chosen under the MMP of Figure 20. The third column shows the bond angles and bond lengths computed by the SA algorithm. The fourth column shows the initial configurations that were used to start the minimization. These values were directly obtained from the velocity vectors of the MMP of a complete coincidence analysis (using all five fragments) of dichloromethane.

Property	*Theory* *from Paul J. Schupf computational chemistry lab*[49]	*Experiment*
Cl-C-Cl bond angle	141.6°	63.5°
H-C-H bond angle	112.4°	103.6°
Cl-C bond length	1.805 *Å*	2.956 *Å*
H-C bond length	1.098 *Å*	2.665 *Å*
angle between (Cl-C-Cl) and (H-C-H) planes	90.0°	96.7°

Figure 28: COMPARISON BETWEEN THEORETICAL AND EXPERIMENTALLY RECONSTRUCTED GEOMETRY
The experimental values listed in this table represent geometry averaged over the five molecules in Figure 27. The theoretical values are based on preliminary molecular structures obtained from J. J. P. Stewart's MOPAC with the PM3 Hamiltonian. From there, a DeFT program from Alain St-Amant is applied, followed by further calculations using the GAMESS program (M. W. Schmidt *et al.*). Orbitals are finally given with the NBO code by E. D. Glendenign *et al.*

The retrieval of dichloromethane's geometry is roughly consistent with its theoretical equilibrium structure, in that it embodies the tetrahedral shape of the molecule. However, there are some visible quantitative discrepancies. First of all, the internuclear separations are all over-estimated. This artifact can be related to the dissociation dynamics in two ways (which are not dissimilar):

i) Since the interatomic repulsion was not *purely* Coulombic, the actual asymptotic velocities reached by the fragments are lower than expected from the theoretical model. By using the classical model to correlate atomic configurations to the measured velocities, the molecule's dimensions are thereby overestimated.

ii) Even worse, if the molecule attained its charge state too slowly under the laser field, it could have slightly expanded and reconfigured itself into a temporary equilibrium state before completely being exploded, further diminishing the accuracy of the retrieval process.

Also, the retrieved Cl-C-Cl and H-C-H angles vary considerably, with the Cl-C-Cl angle being significantly underestimated. If point ii) above is realistic, the error on these angles can be due to the deformation, and induced vibrative motion of the molecule under the action of the intense laser field, before being exploded. More specifically, since carbon is three times lighter than chlorine, it moves around much more easily than the chlorine atoms. This naturally means that the Cl-C-Cl bond angle would have decreased before Coulomb explosion occurred. On the other hand, results for the H-C-H angle are not as conclusive. As evidenced in Figure 27, hydrogen atoms are very light, and therefore are easily accelerated by the laser electric field, which in turn spreads out the retrieved velocity distribution for the hydrogen fragments. Indeed, a different selection of molecules from the MMP could have just as well yielded an overestimation of the H-C-H bond angle.

As can be verified by the classical explosion model, an average nuclear displacement of 0.07 *Å* from equilibrium can lead to a value of $\Delta = 17\%$. Therefore, a small molecular deformation can have a large impact on the direction of the asymptotic velocity vectors for

all fragments, in turn affecting the retrieved molecular configuration. It must also be mentioned that initial positions for the simulated explosions are calculated under the premise that the atoms' initial velocities are all zero at the moment of dissociation. This is a very restricting assumption, given that the molecules had finite vibrational and rotational temperatures. Experimentally, one can achieve lower vibrational and rotational (as well as translational) temperatures by increasing the gas pressure at the input of the source chamber. And as the previous thermal measurements suggest, it seems that the laser pulse duration used throughout these experiments (~45 *fs*) was not short enough to prevent laser-induced nuclear motion. This is especially true for the hydrogen fragments. Therefore, molecular structures can be obtained more accurately by using shorter (~5 *fs*) laser pulses in order to better freeze the internuclear motion[42][50]. In addition to these experimental adjustments, a proper reconstruction algorithm should allow the molecule to participate in a random walk both in configuration space as well as in velocity space.

Considering these limitations, it must be noted that the accuracy of the given reconstructions is determined by the measured velocity vectors. Since the LCEI measurement technique provides only the asymptotic velocity vectors of each fragment after the explosion, an analytical or numerical method designed to reconstruct molecular geometry must ultimately rely on these empirical velocity values and their uncertainties. Therefore, the accuracy of the geometry reconstructions is based on the minimized values of the chosen discriminant[15]. This discriminant is independent of the minimization scheme that is used, in that it detects the maximum deviation between the measured velocities and the ones given by the reconstruction algorithm. More advanced algorithms will be able to attain minimized values of Δ over less iterations; however, the calculated values of Δ presented here should speak for themselves in terms of their ability to measure the fidelity of the reconstruction. Thus, the discrepancies between the measured and the theoretically calculated equilibrium geometries should be seen as limitations of the experimental parameters instead of deficiencies in the reconstruction algorithm.

Although the SA algorithm is useful for the retrieval of the spatial configuration of individual molecules, it can also be applied to give statistical structural information about an ensemble of molecules, given sufficient computational speed and time - the results presented in Figures 26 to 28 required ~2000 iterations of the SA algorithm, with each iteration executing a 20,000-step propagation of a classical Coulomb explosion. By using a shorter step size and propagating the Coulomb explosion over more iterations, the calculated asymptotic velocities would be more accurate. However, the current parameters were verified to preserve the kinetic energy to within one percent (of the initial potential energy of the molecule).

---

[15] In this case, $\Delta = Max\left(\frac{\left|\vec{v}_{theo}^{(i)} - \vec{v}_{emp}^{(i)}\right|}{\left|\vec{v}_{theo}^{(i)}\right|}\right)$ was minimized. However, a minimization of $\chi^2 = \sum \frac{(\vec{v}_{theo}^{(i)} - \vec{v}_{emp}^{(i)})^2}{\sigma_{theo}^{(i)\,2} + \sigma_{emp}^{(i)\,2}}$ is perhaps a better choice since it also makes use of the errors (σ^2) for the empirical and test velocities.

iv) Analytical Recipe For Determining Molecular Handedness

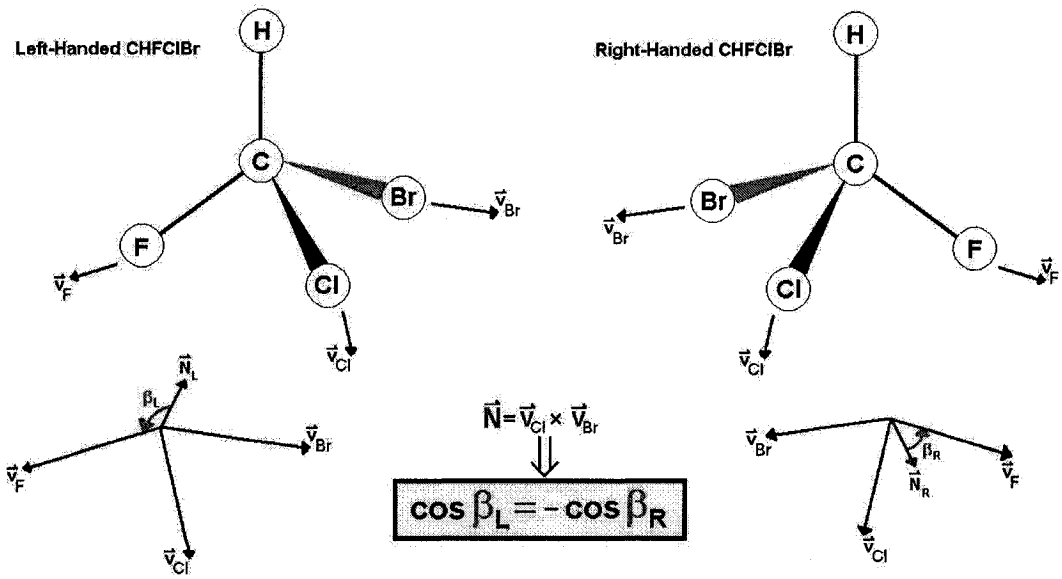


Figure 29: HANDEDNESS DISCRIMINANT FOR BROMOCHLOROFLUOROMETHANE
As an example of a chiral molecule, consider bromochlorofluoromethane (CHFClBr). In order to find a quantity that changes sign under a symmetry transformation we turn to the good old cross product. First, a pseudovector N is defined by the cross product between chlorine and bromine asymptotic velocity vectors. The cosine of the angle β between N and fluorine's velocity vector v_F changes sign under a reflection through a plane of symmetry and consequently provides an adequate qualitative measure of the molecule's handedness. Of course, a different set of fragments' velocity vectors could have been chosen to construct an equally valid discriminant. The choice made here is convenient as it is conceptually (and geometrically) simple. Also, because of fluorine's relatively small mass and its position in the molecule, the resulting angle β would turn out to be larger (making it a better discriminant).

For the purpose of measuring molecular handedness, geometrical reconstruction of molecules is pretty much overkill, only the velocity vectors are required here. One can calculate a discriminant, which will distinguish between left- and right-handed molecules, from the velocity vectors according to the prescription described under Figure 29.

To substantiate the validity of this particular discriminant, let the vectors $\vec{N}$, $\vec{v}_{Cl}$, $\vec{v}_{Br}$, and $\vec{v}_F$ be defined in terms of their Cartesian coordinates:

$$\begin{aligned}\vec{N}&=\vec{v}_{Cl}\times\vec{v}_{Br}=(N_x,N_y,N_z)\\ \vec{v}_{Cl}&=(v_{Clx},v_{Cly},v_{Clz})\\ \vec{v}_{Br}&=(v_{Brx},v_{Bry},v_{Brz})\\ \vec{v}_F&=(v_{Fx},v_{Fy},v_{Fz})\end{aligned} \qquad (20)$$

The angle β is then given by

$$\cos\beta=\frac{\vec{N}\cdot\vec{v}_F}{|\vec{N}||\vec{v}_F|} \qquad (21)$$

Now, under a reflection about the yz-plane (implying a change of handedness) the vectors $\vec{N}, \vec{v}_{Cl}, \vec{v}_{Br}$, and $\vec{v}_F$ become

$$(22) \qquad \begin{array}{l} \vec{N}\,'=(N_x,-N_y,-N_z) \\ \vec{v}_{Cl}\,'=(-v_{Clx}, v_{Cly}, v_{Clz}) \\ \vec{v}_{Br}\,'=(-v_{Brx}, v_{Bry}, v_{Brz}) \\ \vec{v}_F\,'=(-v_{Fx}, v_{Fy}, v_{Fz}) \end{array} ,$$

which means that the corresponding angle β' is related to β by the relation

$$(23) \qquad \cos\beta' = \frac{\vec{N}\,'\cdot\vec{v}_F\,'}{|\vec{N}\,'||\vec{v}_F\,'|} = -\frac{\vec{N}\cdot\vec{v}_F}{|\vec{N}||\vec{v}_F|} = -\cos\beta \quad .$$

Hence, direct information about the enantiomeric proportions of a sample of chiral molecules can be obtained – without having to undertake statistically-based measurements such as optical rotatory dispersion and specific rotation - by plotting a histogram of the discriminant (chosen to befit the the experiment in question) calculated from the secondary data.

V. Outlook And Acknowledgments

Even though most of the results presented above were displayed in a statistical manner, it must be stressed that the secondary data constructed by coincidence imaging allows for the evaluation of information on a molecule-per-molecule basis. Therefore, the techniques presented in this work make it possible not only to bypass indirect statistical measurements of molecular handedness, but they also open up an entirely new line of experiments involving or based on molecular chirality. For example, experiments in femtosecond pump-probe spectroscopy can be performed in conjuction with Coulomb explosion coincidence imaging to study the relation between handedness and some other molecular property that is probed by the experiment.

The prospect of studying larger molecules using CEI is also on the horizon thanks to constant developments in technology. As it was shown, the difficulties in performing coincidence imaging on large molecules are purely technological in nature. As detectors and electronics become faster and laser pulses become shorter, coincidence imaging of larger (>5 atoms) molecules will be possible. In the meanwhile, "lightly" Coulomb-exploding large molecules into several of their subgroups can constitute a reasonable compromise. For example, the alanine molecule ($C_3H_7NO_2$) contains 13 atoms, but it could be exploded into a weak charge state that would keep its subgroups (CH_3, $CH(H_2N)$, OH, COOH) together until they reach the detector. Coincidence imaging could then be carried out using the subgroups as fragments, while also taking into account the effect of their finite geometrical structure on the explosion dynamics. Therefore, the current technology should be sufficient for obtaining

a measurement of chirality, even for large organic molecules, thus reducing the risks of the medical usage of chiral substances.

During the course of this work I have received invaluable advice and assistance from several members of the scientific and technical staff at the National Research Council. I would like to express gratitude of the utmost sincerity to technological officers Mr. David Joines, Mr. Bert Avery and Mr. John Parsons for contributing their support and their wealth of knowledge and experience to the table. I am also very thankful to chemist Dr. David Rayner, whose powerful insight originated the procedure for determining molecular handedness described in section IV-C-iv, and helped me to grasp molecular dynamics. Also of fundamental importance were the enlightening discussions I had with University of McMaster PhD candidate Mr. Kevin Lee, postdoctoral fellow Dr. Domagoj Pavicic, University of Ottawa professor Dr. Ravi Bhardwaj and NRC research officer Dr. David Villeneuve. I commend these people for their sophisticated understanding of cutting edge science and technology, and for helping me to understand the key elements that were at play during the CEI experiment, not to mention the experience I gained from their coaching. And last but not least, I am indebted to Femtosecond Group leader Dr. Paul Corkum for bringing about this project, and providing me with a role in science and a satisfying challenge.

VI. References

[1] W. Thomson, *Lectures on Molecular Dynamics and the Wave Theory of Light* (in Baltimore), C. J. Clay & Sons, London, 1904. Appendix H, p. 602.
[2] Chem Res Toxicol. 1998 Dec; 11(12): 1521-8
[3] G. E. Tranter, Nature, **318**, 172-173, (1985)
[4] Y. J. Yamagata, J Theor Biol; 11(3), 495–498 (1966)
[5] C. Viedma, Phys. Rev. Lett. **94**, 065504 (2005)
[6] Y. Saito *et al.*, J. Phys. Soc. Jpn. **74**, **535-537** (2005)
[7] M. Wedyan *et al.*, Environ Sci Technol, 1; 39 (7): 2115-9 (2005)
[8] A. Fresnel, Mem. Acad. Sci., 1827, 7, 45
[9] P. Drude, Ann. d. Physik, 48, 536 (1896); *The Theory of Optics*, translation by C.R. Mann and R.A. Millikan, (1902) , Dover reprint, New York (1959) p.400.
[10] L. Pasteur, *Sur la dissymétrie moléculaire*, C.R. Hebd. Séanc, 26, 535 (1848)
[11] J Chromatogr A. 2001 Jul 6;921(2):161-7
[12] Acta Crystallogr B. 2003 Dec;59(Pt 6):802-10. Epub 2003 Nov 25
[13] *J. Chromatogr.*, 475 (1989) 23
[14] *J. Chromatogr.*, 638 (1993) 85
[15] *J. Chromatogr.*, 666 (1994) 111
[16] *J. Chromatogr.*, 666 (1994) 617
[17] *J. Chromatogr. A*, 679 (1994) 23
[18] *J. Chromatogr. A*, 755 (1996) 299-307
[19] F. Légaré *et al.*, Phys. Rev. A **72**, 052717 (2005)
[20] O. Geßner *et al.*, Science **311,** 5758, pp. 219 – 222 (2006)
[21] J. Itatani *et al.*, Nature **432**, 867-871 (2004)
[22] K. F. Lee *et al.*, Phys. Rev. Lett. **93**, 233601 (2004)
[23] H. Niikura *et al.*, Phys. Rev. A **73**, 021402 (2006)
[24] A. Kastberg *et al.*, Phys. Rev. Lett. **74**, 1542-1545 (1995)
[25] P. B. Corkum, Phys. Rev. Lett. **71**, 1994-1997 (1993)
[26] V. R. Bhardwaj *et al.*, Phys. Rev. Lett. **87**, 253003 (2001)
[27] M. Weckenbrock *et al.*, Phys. Rev. Lett. **92**, 213002 (2004)
[28] T. Kitamura *et al.*, J. Chem. Phys., Vol. 115 Issue 1 (2001)
[29] Z. Vager *et al.*, Phys. Rev. Lett. **57**, 2793-2795 (1986)
[30] Z. Vager *et al.*, Phys. Rev. Lett. **55**, 638 (1985)
[31] A. Belkacem, E. P. Kanter, Phys. Rev. Lett. **63**, 2555-2558 (1989)
[32] T. H. Maiman, Phys. Rev. Lett. **4**, 564-566 (1960)
[33] D. T. Strickland *et al.*, Phys. Rev. Lett. **68**, 2755-2758 (1992)
[34] M. Ivanov *et al.*, Phys. Rev. A **54**, 1541 (1996)
[35] D. T. Strickland *et al.*, Phys. Rev. Lett. **68**, 2755 (1992)
[36] S. M. Hankin *et al.*, Phys. Rev.A **64**, 013405 (2001)
[37] F. Légaré *et al.*, Phys. Rev. Lett. **91**, 093002 (2003)
[38] V. R. Bhardwaj *et al.*, Phys. Rev. Lett. **87**, 253003 (2001)
[39] P. W. Dooley *et al.*, Phys. Rev. A **68**, 023406 (2003)

[40] A. H. Zewail, *Femtochemistry — Ultrafast Dynamics of the Chemical Bond, Vols. I and II*, World Scientific, New Jersey, Singapore (1994)
[41] P. W. Dooley, Optically Timed Sub-Millimetre Mass Spectrometry *(Ph. D. Thesis, Chapter 3)*, 2003
[42] F. Légaré *et al.*, Phys. Rev. A **71**, 013415 (2005)
[43] P. W. Dooley, *Molecular Imaging Using Femtosecond Laser Pulses (Ph. D. Thesis)*, 2003
[44] D. Strickland *et al.*, Opt. Commun. **56,** 219 (1985)
[45] D. M. Villeneuve *et al.*, Phys. Rev. Lett. **85**, 542–545 (2000)
[46] D. M. Villeneuve *et al.*, Phys. Rev. A **54**, 736-741 (1996)
[47] H.J. Lewandowski *et al.*, Chem. Phys. Lett. 395 53–57 (2004)
[48] P. H. Bucksbaum et al., Phys. Rev. Lett. 64, 1883-1886 (1990)
[49] http://www.colby.edu/chemistry/webmo/mointro.html
[50] S. Chelkowski *et al.*, Phys. Rev. A **65**, 023403 (2002)

VII. Appendices

A) ARBCORR C Implementation

```
#include "arb.h"

/*****************************************************************************
Copyright (C) 2006 Justin Gagnon

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.
*****************************************************************************/

/***Constants***/
#define AMU             1.66054E-27
#define NS              1.0E-9
#define QE              1.601177E-19

/***Experimental Parameters***/
#define EE              20833.333
#define XSCALE          0.0002545454
#define YSCALE          0.00026718

/***Calibration Parameters***/
#define ALPHA           1.20E-6
#define CT              536.363
#define CX              (-4.062)
#define CY              (-4.195)
#define DATA            "D:\\!!! work\\136 uJ"
#define FRAGMENTS       "frags_136uJ_CH2(35)Cl+_(37)Cl+.txt"
#define VMB             (-430.0)

long CorrelateEvent(double **,double **,double ***,FILE *,long);
/*****************************************************************************
Arguments:
I.   array of hits (t,x,y)
II.  array of fragment specifications (tmin,tmax,t0,[properties])
III. array of combinations of fragment specifications (created by
     FragmentCombinations)
IV.  output file stream
```

Output:
number of fragment combinations calculated from I

Description:
finds all the possible combinations of fragments from I using the specifications in II and III and writes them to IV
**********/
void CorrelateFiles(**char** *,**char** *);
/**********CorrelateFilesCorrelateFiles**********
Arguments:
I. folder path of (t,x,y) data to be correlated
II. name of the fragment specifications file

Output:
none

Description:
loads individual events from I and fragment specifications from II, correlates them using CorrelateEvent and writes the correlated data to III; (t,x,y) files in I must be named "0 (0).txt", "0 (1).txt", "0 (2).txt" and so on
**********/
int EquivalentEvents(**double** **,**double** **,**long** **);
/**********
Arguments:
I. array of fragments (t,x,y,[properties])
II. array of fragments (t,x,y,[properties])
III. array of permutations created by Permutations

Output:
1 if I and II are equivalent
0 otherwise

Description:
if I and II have the same (unordered) set of rows of (t,x,y,[properties]) then they are equivalent and EquivalentEvents returns 1; if they differ in at least one row of (t,x,y,[properties]) then EquivalentEvents returns 0
**********/
double **FilterHits(**double** **,**double** **);
/**********
*FilterHits(double **hits,double **frags)*
Arguments:
I. array of hits (t,x,y)
II. array of fragment specifications (tmin,tmax,t0,[properties])

Output:
array of hits (t,x,y)

Description:
creates and returns a new array of (t,x,y) hits that contains only those in I for which the time of flight t is inside at least one time interval defined in II

```
**********************************************************************************/
double **FormatFragSpecs(double **);
/**********************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of fragment specifications (tmin,tmax,t0,[properties])

Description:
    separates I into mutually exclusive time intervals and stores them into a
    new array of (tmin,tmax,t0,[properties]) fragment specifications; each
    mutually exclusive time interval is associated to every possible set of
    properties as defined in I
**********************************************************************************/
double ***FragmentCombinations(double **);
/**********************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of combinations of fragment specifications

Description:
    formats I using FormatFragSpecs, and creates from this larger set of
    fragment specifications all the combinations of valid fragment
    specifications as determined by TestFragmentCombination
**********************************************************************************/
double ***GenerateSameIntervalPermutations(double **);
/**********************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of arrays of fragment specifications (tmin,tmax,t0,[properties])

Description:
    returns an array of arrays of (tmin,tmax,t0,[properties]) fragment
    specifications for which the fragment specifications having the same time
    interval are permuted with eachother using SameIntervalPermutations
**********************************************************************************/
double **LabelHits(double **,double **);
/**********************************************************************************
Arguments:
I.  array of hits (t,x,y)
II. array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of fragments (t,x,y,[properties])

Description:
    creates and returns an array of fragments (vx,vy,vz,[properties]) made
    from (t,x,y) hits associated to the corresponding properties of
```

```
    (tmin,tmax,t0,[properties]) fragment specifications
**********************************************************************************/
double **LoadEvent(FILE *);
/**********************************************************************************
Arguments:
I.  reference to an open file

Output:
    array of hits (t,x,y)

Description:
    reads the binary (t,x,y) content of an lmf file and creates and returns
    a corresponding array of (t,x,y) hits
**********************************************************************************/
double NumberOfOccurrences(double *,double **);
/**********************************************************************************
Arguments:
I.  properties
II. array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    number of occurrences of a set of properties I

Description:
    finds the number of occurrences of properties I from the ones in II
**********************************************************************************/
double **NumberOfOccurrencesTable(double **);
/**********************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of property occurrences ([properties],number of occurrences)

Description:
    finds the number of occurrences of the properties of each fragment
    specification in I and creates and returns a
    ([properties],number of occurrences) array
**********************************************************************************/
double ***RemoveRedundantEvents(double ***,long **);
/**********************************************************************************
Arguments:
I.  array of fragment arrays (t,x,y,[properties])
II. array of permutations created by Permutations

Output:
    array of fragment arrays (t,x,y,[properties])

Description:
    looks for equivalent (t,x,y,[properties]) fragment arrays in I and copies
    only one of each into a new array of (t,x,y,[properties]) fragment arrays,
    which is then returned
**********************************************************************************/
```

```
long **SameIntervalPermutations(double **);
/*****************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    array of permutations

Description:
    isolates the (tmin,tmax,t0,[properties]) fragments specifications that
    have the same time interval but different properties and puts them in a
    separate array; permutes the elements of each array and constructs a new
    set of fragment specifications arrays with appropriately permuted elements
*****************************************************************************/
int TestFragmentCombination(double **,double **);
/*****************************************************************************
Arguments:
I.  array of fragment specifications (tmin,tmax,t0,[properties])
II. array of property occurrences ([properties],number of occurrences)

Output:
    1 if I is a valid fragment specifications array
    0 otherwise

Description:
    a fragment specifications array is valid if and only if the number of
    occurrences of each property set of I is the same as the corresponding
    property set in II
*****************************************************************************/
int TestHitCombination(double **,double **);
/*****************************************************************************
(double **hits,double **frags)
Arguments:
I.  array of hits (t,x,y)
II. array of fragment specifications (tmin,tmax,t0,[properties])

Output:
    1 if I is a valid hit array
    0 otherwise

Description:
    a hit array I is valid if and only if every time of flight t in I
    simultaneously fits into a time interval in II
*****************************************************************************/


int main(){

 CorrelateFiles(DATA,FRAGMENTS);
 //while(1);
 return 0;
}
/*****************************************************************************
```

```
*****************************************************************************/
long CorrelateEvent(double **hits,double **frags,double ***fragcombs,
            FILE *outstream,long cseed){

 long i,j,nfrags,ncombsf,ncombsh,colh,colf,**hitcombarray,nhits,nwritten;
 double **filteredhits,**hitcomb,**test;

 filteredhits=FilterHits(hits,frags);
 nfrags=(long)frags[1][0];
 nhits=(long)filteredhits[1][0];
 colh=(long)filteredhits[0][0];
 colf=(long)frags[0][0];
 ncombsf=(long)fragcombs[0][0][0];
 nwritten=0;

 if(nhits>=nfrags){
  hitcombarray=Combinations(nhits,nfrags);
  ncombsh=(long)hitcombarray[1][0];
  for(i=1;i<=ncombsf;i++){
   for(j=1;j<=ncombsh;j++){
    hitcomb=GenerateRowCombination(filteredhits,hitcombarray,j);
    if(TestHitCombination(hitcomb,fragcombs[i])==1){
     test=LabelHits(hitcomb,fragcombs[i]);
     if(Ran2(&cseed)<=1.0){
      WriteOneMatrix(outstream,test);
      nwritten++;
     }
     FreeMatrix(test);
    }
    FreeMatrix(hitcomb);
   }
  }
  FreeMatrixL(hitcombarray);
 }
 FreeMatrix(filteredhits);
 return nwritten;
}
/*****************************************************************************
*****************************************************************************/
void CorrelateFiles(char *datafolder,char *fragfile){

 FILE *fd,*fout;
 long i,j,nfiles,nfrags,dummy,nevents,nwritten,ldoi;
 double **frags,***fragcombs,**hits;
 int stop;
 char *fname1,*fname2,*fname3,*sj,*dataout,*_correlated;

 dataout=(char *)malloc((strlen(fragfile)+11+1)*sizeof(char));
 ldoi=strlen(fragfile)+11;
 frags=LoadMatrix(fragfile);
 nfiles=0;
 nfrags=(long)frags[1][0];
 j=0;
```

```
stop=0;
nwritten=0;
_correlated="_correlated";

dataout[ldoi]='\0';
i=0;
while(fragfile[i]!='.'){
 dataout[i]=fragfile[i];
 i++;
}
while(j<11){
 dataout[i+j]=_correlated[j];
 j++;
}
dataout[i+j]='.';
j=strlen(fragfile);
i=0;
while(fragfile[j-i]!='.'){
 dataout[ldoi-i]=fragfile[j-i];
 i++;
}
j=0;
fout=fopen(dataout,"w");
for(i=1;i<=nfrags;i++){
 frags[1][i]-=0.1;
 frags[2][i]+=0.1;
}
fragcombs=FragmentCombinations(frags);
while(stop==0){
 i=1;
 sj=StringL(j);
 fname1=Concat("\\0 (",sj);
 fname2=Concat(fname1,").lmf");
 fname3=Concat(datafolder,fname2);
 fd=fopen(fname3,"rb");
 if(fd==NULL) stop=1;
 else{
  fseek(fd,4,SEEK_CUR);
  fread(&dummy,sizeof(long),1,fd);
  fread(&nevents,sizeof(long),1,fd);
  fread(&dummy,sizeof(long),1,fd);
  do{
   hits=LoadEvent(fd);
   nwritten+=CorrelateEvent(hits,frags,fragcombs,fout,-100*j-10);
   FreeMatrix(hits);
   i++;
  }while(i<=nevents);
  printf("\r");
  printf("file %d complete\n",j);
  fclose(fd);
  free(sj);
  free(fname1);
  free(fname2);
```

```
    free(fname3);
    j++;
   }
  }
  for(i=1;i<=100;i++) fprintf(fout,"\n");
  fprintf(fout,"%d",nwritten);
  fclose(fout);
}
/*****************************************************************************
*****************************************************************************/
int EquivalentEvents(double **m1,double **m2,long **permarray){

  long i,nperms;
  double **temp;
  int test;

  nperms=(long)permarray[1][0];
  test=0;

  for(i=1;i<=nperms;i++){
   temp=GenerateRowPermutation(m1,permarray,i);
   if(EqualMatrices(temp,m2)==1){
    test=1;
    i=nperms+1;
   }
   FreeMatrix(temp);
  }
  return test;
}
/*****************************************************************************
*****************************************************************************/
double **FilterHits(double **hits,double **frags){

  long i,j,m,nfrags,colf,colh,nhits,ntofilter;
  int *tofilter,filtertest;
  double **filtered;

  nhits=(long)hits[1][0];
  nfrags=(long)frags[1][0];
  colf=(long)frags[0][0];
  colh=(long)hits[0][0];
  tofilter=VectorI(nhits);
  ntofilter=0;
  m=1;

  for(i=1;i<=nhits;i++) tofilter[i]=0;
  for(i=1;i<=nhits;i++){
   filtertest=1;
   for(j=1;j<=nfrags;j++){
    if(hits[1][i]>=frags[1][j] && hits[1][i]<=frags[2][j]){
     filtertest=0;
     j=nfrags+1;
    }
```

```
    }
    if(filtertest==1){
     tofilter[i]=1;
     ntofilter++;
    }
   }
   if(ntofilter!=nhits){
    filtered=Matrix(colh,nhits-ntofilter);
    for(i=1;i<=nhits;i++){
     if(tofilter[i]==0){
      for(j=1;j<=colh;j++) filtered[j][m]=hits[j][i];
      m++;
     }
    }
   }
   else{
    filtered=Matrix(1,1);
    filtered[1][1]=0.0;
   }
   free(tofilter);
   return filtered;
}
/*****************************************************************************
*****************************************************************************/
double **FormatFragSpecs(double **frags){

 long i,j,k,m,nfrags,colf,nintervals,npfrags;
 double *separators,**intervals,**possiblefrags;

 nfrags=(long)frags[1][0];
 colf=(long)frags[0][0];
 separators=Vector(2*nfrags);
 m=1;
 npfrags=0;

 for(i=1;i<=nfrags;i++){
  separators[m]=frags[1][i];
  separators[m+1]=frags[2][i];
  m+=2;
 }
 RemoveDuplicates(&separators);
 Sort(separators);
 nintervals=(long)separators[0]-1;
 intervals=Matrix(2,nintervals);
 for(i=1;i<=nintervals;i++){
  intervals[1][i]=separators[i];
  intervals[2][i]=separators[i+1];
 }
 for(i=1;i<=nintervals;i++){
  for(j=1;j<=nfrags;j++){
   if(MutuallyExclusive(intervals[1][i],intervals[2][i],frags[1][j],
   frags[2][j])==0) npfrags++;
  }
```

```
 }
 m=1;
 possiblefrags=Matrix(colf+1,npfrags);
 for(i=1;i<=nintervals;i++){
  for(j=1;j<=nfrags;j++){
   if(MutuallyExclusive(intervals[1][i],intervals[2][i],frags[1][j],
   frags[2][j])==0){
    possiblefrags[1][m]=intervals[1][i];
    possiblefrags[2][m]=intervals[2][i];
    possiblefrags[3][m]=sqrt(frags[3][j]/frags[4][j]/ALPHA)+CT;
    for(k=1;k<=colf-2;k++) possiblefrags[3+k][m]=frags[2+k][j];
    m++;
   }
  }
 }
 free(separators);
 FreeMatrix(intervals);
 return possiblefrags;
}
/*****************************************************************************
*****************************************************************************/
double ***FragmentCombinations(double **frags){

 long i,m,nfrags,ntfrags,colf,**combarray,ngoodcombs,ncombs;
 int *goodcombs;
 double ***fragcombs,***goodfragcombs,**formattedfrags,**nocctable,***temp1,
     ***temp2;

 nfrags=(long)frags[1][0];
 colf=(long)frags[0][0];
 formattedfrags=FormatFragSpecs(frags);
 ntfrags=(long)formattedfrags[1][0];
 combarray=Combinations(ntfrags,nfrags);
 ncombs=combarray[1][0];
 nocctable=NumberOfOccurrencesTable(formattedfrags);
 goodcombs=VectorI(ncombs);
 fragcombs=MatrixArray(ncombs);
 ngoodcombs=0;
 m=1;

 for(i=1;i<=ncombs;i++) goodcombs[i]=0;
 for(i=1;i<=ncombs;i++){
  fragcombs[i]=GenerateRowCombination(formattedfrags,combarray,i);
  if(TestFragmentCombination(fragcombs[i],nocctable)==1){
   ngoodcombs++;
   goodcombs[i]=1;
  }
 }
 if(ngoodcombs!=0){
  goodfragcombs=MatrixArray(ngoodcombs);
  for(i=1;i<=ncombs;i++){
   if(goodcombs[i]==1){
    goodfragcombs[m]=CopyMatrix(fragcombs[i]);
```

```
      m++;
     }
    }
    FreeMatrixArray(fragcombs);
    temp1=GenerateSameIntervalPermutations(goodfragcombs[1]);
    for(i=2;i<=ngoodcombs;i++){
     temp2=GenerateSameIntervalPermutations(goodfragcombs[i]);
     AppendMatrixArray(&temp1,&temp2);
     FreeMatrixArray(temp2);
    }
   }
   else{
    goodfragcombs=MatrixArray(1);
    goodfragcombs[1]=Matrix(1,1);
    goodfragcombs[1][1][1]=0.0;
   }
   FreeMatrixArray(goodfragcombs);
   FreeMatrixL(combarray);
   free(goodcombs);
   FreeMatrix(formattedfrags);
   FreeMatrix(nocctable);
   return temp1;
}
/*****************************************************************************
*****************************************************************************/
double ***GenerateSameIntervalPermutations(double **frags){

 long **perms,nperms,i;
 double ***total;

 perms=SameIntervalPermutations(frags);
 nperms=perms[1][0];
 total=MatrixArray(nperms);

 for(i=1;i<=nperms;i++) total[i]=GenerateRowPermutation(frags,perms,i);
 FreeMatrixL(perms);
 return total;
}
/*****************************************************************************
*****************************************************************************/
double **LabelHits(double **hits,double **frags){

 long i,j,m,nfrags,nhits,colf,colh;
 int *fragmentslot;
 double **labeled,xx,yy,tt,t0,mass,charge;

 nfrags=(long)frags[1][0];
 colf=(long)frags[0][0];
 nhits=(long)hits[1][0];
 colh=(long)hits[0][0];
 labeled=Matrix(colh+colf-3,nfrags);
 fragmentslot=VectorI(nfrags);
 m=1;
```

```
 for(i=1;i<=nfrags;i++) fragmentslot[i]=0;
 for(i=1;i<=nhits;i++){
  for(j=1;j<=nfrags;j++){
   if(fragmentslot[j]!=1 && hits[1][i]>=frags[1][j]
   && hits[1][i]<=frags[2][j]){
     xx=(hits[2][i]-CX)*XSCALE;
     yy=(hits[3][i]-CY)*YSCALE;
     tt=(hits[1][i]-CT)*0.5*NS;
     t0=(frags[3][j]-CT)*0.5*NS;
     mass=frags[4][j]*AMU;
     charge=frags[5][j]*QE;
     labeled[1][m]=xx/tt-VMB;
     labeled[2][m]=yy/tt;
     labeled[3][m]=0.5*charge*EE*(t0*t0-tt*tt)/tt/mass;
     labeled[4][m]=mass;
     labeled[5][m]=charge;
     /*labeled[1][m]=hits[1][i];
     labeled[2][m]=hits[2][i];
     labeled[3][m]=hits[3][i];
     labeled[4][m]=frags[4][j];
     labeled[5][m]=frags[5][j];*/
     m++;
     fragmentslot[j]=1;
     j=nfrags+1;
    }
   }
  }
  free(fragmentslot);
  return labeled;
}
/*****************************************************************************
*****************************************************************************/
double **LoadEvent(FILE *ff){

  long k,nhits,dummy,tt,xx,yy;
  double **hits;

  fread(&nhits,sizeof(long),1,ff);
  hits=Matrix(3,nhits);
  fread(&dummy,sizeof(long),1,ff);
  for(k=1;k<=nhits;k++){
   fread(&tt,sizeof(long),1,ff);
   fread(&xx,sizeof(long),1,ff);
   fread(&yy,sizeof(long),1,ff);
   hits[1][k]=(double)tt;
   hits[2][k]=(double)xx;
   hits[3][k]=(double)yy;
  }
  return hits;
}
/*****************************************************************************
*****************************************************************************/
```

```
double NumberOfOccurrences(double *props,double **frags){

 long i,j,nfrags,ncols,nprops;
 int test;
 double nocc;

 nfrags=(long)frags[1][0];
 ncols=(long)frags[0][0];
 nocc=0;
 nprops=(long)props[0];

 for(i=1;i<=nfrags;i++){
  test=1;
  for(j=1;j<=nprops;j++){
   if(frags[j+3][i]!=props[j]) test=0;
  }
  if(test==1) nocc++;
 }
 return nocc;
}
/*****************************************************************************
*****************************************************************************/
double **NumberOfOccurrencesTable(double **frags){

 long i,j,nfrags,ncols,nprops;
 double **nocctable,*props;

 nfrags=(long)frags[1][0];
 ncols=(long)frags[0][0];
 nprops=(long)(frags[0][0]-3.0);
 props=Vector(nprops);
 nocctable=Matrix(nprops+1,nfrags);

 for(i=1;i<=nfrags;i++){
  for(j=1;j<=nprops;j++){
   nocctable[j][i]=frags[j+3][i];
   props[j]=frags[j+3][i];
  }
  nocctable[nprops+1][i]=NumberOfOccurrences(props,frags);
 }
 free(props);
 return nocctable;
}
/*****************************************************************************
*****************************************************************************/
double ***RemoveRedundantEvents(double ***arb,long **permarray){

 long i,size,m,*toberemoved,ntorem,ncombs,**combs;
 double ***caca;

 size=(long)arb[0][0][0];
 combs=Combinations(size,2);
 ncombs=(long)combs[1][0];
```

```
 toberemoved=VectorL(size);
 ntorem=0;
 m=1;

 for(i=1;i<=size;i++) toberemoved[i]=0;
 for(i=1;i<=ncombs;i++){
  if(EquivalentEvents(arb[combs[1][i]],arb[combs[2][i]],permarray)==1){
   if(toberemoved[combs[1][i]]==0) toberemoved[combs[1][i]]=2;
   if(toberemoved[combs[2][i]]==0){
    toberemoved[combs[2][i]]=1;
    ntorem++;
   }
  }
 }
 if(ntorem!=size){
  caca=MatrixArray(size-ntorem);
  for(i=1;i<=size;i++){
   if(toberemoved[i]!=1){
    caca[m]=CopyMatrix(arb[i]);
    m++;
   }
  }
 }
 else{
  caca=MatrixArray(1);
  caca[1]=Matrix(1,1);
  caca[1][1][1]=0.0;
 }
 free(toberemoved);
 FreeMatrixL(combs);
 return caca;
}
/*****************************************************************************
*****************************************************************************/
long **SameIntervalPermutations(double **fragcomb){

 long i,j,nfrags,colf,m,ncombs,**combs,*flags,**permute,nextflag,**temp1,
    **temp2,**temp3;
 double *props1,*props2;

 nfrags=(long)fragcomb[1][0];
 colf=(long)fragcomb[0][0];
 combs=Combinations(nfrags,2);
 ncombs=(long)combs[1][0];
 m=1;
 flags=VectorL(nfrags);
 nextflag=1;
 props1=Vector(colf-3);
 props2=Vector(colf-3);

 for(i=1;i<=nfrags;i++) flags[i]=0;
 for(i=1;i<=ncombs;i++){
  for(j=1;j<=colf-3;j++){
```

```
    props1[j]=fragcomb[j+3][combs[1][i]];
    props2[j]=fragcomb[j+3][combs[2][i]];
   }
   if(fragcomb[1][combs[1][i]]==fragcomb[1][combs[2][i]] &&
   fragcomb[2][combs[1][i]]==fragcomb[2][combs[2][i]] &&
   EqualVectors(props1,props2)==0){
    if(flags[combs[1][i]]==0 && flags[combs[2][i]]==0){
     flags[combs[1][i]]=nextflag;
     flags[combs[2][i]]=nextflag;
     nextflag++;
    }
    else{
     if(flags[combs[1][i]]>flags[combs[2][i]])
      flags[combs[2][i]]=flags[combs[1][i]];
     else flags[combs[1][i]]=flags[combs[2][i]];
    }
   }
  }
  permute=(long **)malloc(nextflag*sizeof(long *));
  for(i=0;i<nextflag;i++){
   permute[i]=VectorL(NOccL(i,flags));
   m=1;
   for(j=1;j<=nfrags;j++){
    if(flags[j]==i){
     permute[i][m]=j;
     m++;
    }
   }
  }
  if(permute[0][0]>0){
   temp1=MatrixL(permute[0][0],1);
   for(i=1;i<=permute[0][0];i++) temp1[i][1]=permute[0][i];
   i=1;
  }
  else{
   temp1=GeneratePermutationsL(permute[1]);
   i=2;
  }
  while(i<nextflag){
   temp2=GeneratePermutationsL(permute[i]);
   temp3=CombineMatricesL(temp1,temp2);
   FreeMatrixL(temp1);
   temp1=CopyMatrixL(temp3);
   FreeMatrixL(temp3);
   FreeMatrixL(temp2);
   i++;
  }
  for(i=0;i<nextflag;i++) free(permute[i]);
  free(permute);
  FreeMatrixL(combs);
  free(props1);
  free(props2);
  free(flags);
```

```
 return temp1;
}
/*****************************************************************************
*****************************************************************************/
int TestFragmentCombination(double **fragcomb,double **nocctable){

 long i,j,k,colf,nfrags,nprops;
 double *props;
 int test,test2;

 nfrags=(long)fragcomb[1][0];
 colf=(long)fragcomb[0][0];
 test=1;
 nprops=(long)(fragcomb[0][0]-3.0);
 props=Vector(nprops);

 for(i=1;i<nfrags;i++){
  for(j=1;j<=nfrags;j++){
   test2=1;
   for(k=1;k<=nprops;k++){
    if(fragcomb[k+3][i]!=nocctable[k][j]){
     test2=0;
     k=nprops+1;
    }
   }
   if(test2!=0){
    for(k=1;k<=nprops;k++) props[k]=nocctable[k][j];
    if(NumberOfOccurrences(props,fragcomb)!=nocctable[nprops+1][j]){
     test=0;
     i=nfrags+1;
    }
    j=nfrags+1;
   }
  }
 }
 free(props);
 return test;
}
/*****************************************************************************
*****************************************************************************/
int TestHitCombination(double **hits,double **frags){

 long i,j,nfrags,nhits;
 int testhit,test,*fragmentslot;

 nfrags=(long)frags[1][0];
 nhits=(long)hits[1][0];
 fragmentslot=VectorI(nfrags);

 for(i=1;i<=nfrags;i++) fragmentslot[i]=0;
 if(nhits<nfrags) test=0;
 else{
  test=1;
```

```
  for(i=1;i<=nhits;i++){
   testhit=0;
   for(j=1;j<=nfrags;j++){
    if(fragmentslot[j]!=1){
     if(hits[1][i]>=frags[1][j] && hits[1][i]<=frags[2][j]){
      testhit=1;
      fragmentslot[j]=1;
      j=nfrags+1;
     }
    }
   }
   if(testhit==0){
    test=0;
    i=nhits+1;
   }
  }
 }
 free(fragmentslot);
 return test;
}
```

B) C Program For Simulating A Classical Coulomb Explosion

```
#include  <stdio.h>
#include  <stdlib.h>
#include  <math.h>
#include  "arb.h"

#define MS 1.66053886E-27     // mass in atomic units
#define CS 1.60217653E-19     // charge in units of elementary charge
#define LS 1.0E-10            // length in angstroms
#define ES 2.30707725173E-18 // energy scaled from Coulomb interaction
#define VS 3.72740468511E4    /* velocity scaled from unit k=1/(4*PI*epsilon)
                                 and energy conservation of two particle
                                 Coulomb interaction*/
#define TS (LS)/(VS)          // time scaled from length and velocity
#define PS (MS)*(VS)          // momentum scaled from mass and velocity

/*****************************************************************************
This program computes trajectories and velocities of an ensemble of ions.
Initial velocity and position coordinates, mass and charge values for each of
the ions are required (specified in the Initialize function). Externally
applied time-dependent electric and magnetic fields can also be specified
(ee and bb respectively), as well as an external potential field UU.
Propagation in time is accomplished using classical mechanics and
electrodynamics using a third-order Runge-Kutta type of scheme
(see Propagate). At the output, the trajectories of each ion are recorded in
a text file (first string argument of Propagate) and their final
velocities and positions as well as other relevant quantities are written to
another text file (second string argument of Propagate).
*****************************************************************************/
/*****************************************************************************
Copyright (C) 2006 Justin Gagnon
```

```
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301, USA.
*****************************************************************************/

// define the characteristics of a particle (atom)
typedef struct{

  double mass;   // mass
  double charge; // charge
  double *pos;   // position vector
  double *vel;   // velocity vector
}particle;

// calculate the acceleration of an atom
double *aa(long,particle *,double **,double);
// specify an external time-dependent magnetic field
double *bb(double);
// specify an external time-dependent electric field
double *ee(double);
// initialize the ions
void Initialize(particle *,double);
// compute the total kinetic energy of the system
double kk(particle *);
// allocate memory for an array of particles
particle *PArray(long nn);
// compute the total momentum of the system
double *pp(particle *);
// propagate in time according to classical electrodynamics
void Propagate(particle *,long,double,char *,char *);
// specify an arbitrary external potential function
double UU(double *);

int main(){

  //  declare variables
  //  natoms:  number of atoms (ions) in the cluster
  //  ntimes:  number of iterations for time propagation
  //  ht:      time step
  //  i,j,k:   dummy indices
  //  *atoms:  1-D array containing information about each ion
```

```
 long natoms,ntimes,i,j,k;
 double ht,hand,**vel;
 char *f1,*f2;
 particle *atoms;

 natoms=5;
 ntimes=20000;
 ht=0.01;
 hand=1.0;
 atoms=PArray(natoms);
 f1="Traj.txt";
 f2="Final_Coords.txt";

 Initialize(atoms,hand);
 Propagate(atoms,ntimes,ht,f1,f2);
 vel=Matrix(5,natoms);
 for(i=1;i<=natoms;i++){
  for(j=1;j<=3;j++) vel[j][i]=atoms[i].vel[j];
  vel[4][i]=atoms[i].mass;
  vel[5][i]=atoms[i].charge;
 }
 WriteMatrix("vel0.txt",vel);
 //while(1);
 return 0;
}

double *aa(long n0,particle *atoms,double **pos,double t){

 long i,j,natoms;
 double *acc,rd,rd3,*ef,*bf1,*bf,*uf;

 natoms=(long)atoms[0].mass;
 acc=Vector(3);
 for(i=1;i<=3;i++) acc[i]=0.0;
 ef=ee(t);
 bf1=bb(t);
 bf=CrossP(atoms[n0].vel,bf1);
 uf=Grad(UU,pos[n0],0.001);

 for(j=1;j<=natoms;j++){
  if(j!=n0){
   rd=Distance(pos[n0],pos[j]);
   rd3=rd*rd*rd;
   for(i=1;i<=3;i++)
    acc[i]+=atoms[n0].charge*atoms[j].charge*(pos[n0][i]-pos[j][i])/rd3;
  }
 }
 for(i=1;i<=3;i++){
  acc[i]+=atoms[n0].charge*(ef[i]+bf[i])-uf[i];
  acc[i]/=atoms[n0].mass;
 }
 free(bf);
 free(ef);
```

```
 free(bf1);
 free(uf);
 return acc;
}

double *bb(double t){

 double *arb;

 arb=Vector(3);
 arb[1]=0.0;
 arb[2]=0.0;
 arb[3]=0.0;

 return arb;
}

double *ee(double t){

 double *arb;

 arb=Vector(3);
 arb[1]=0.0;
 arb[2]=0.0;
 arb[3]=0.0;

 return arb;
}

void Initialize(particle *atoms,double hand){

 long i,j,natoms;

 natoms=(long)atoms[0].mass;

 for(i=1;i<=natoms;i++){
  atoms[i].pos=Vector(3);
  atoms[i].vel=Vector(3);
 }
 for(i=1;i<=3;i++){
  for(j=1;j<=natoms;j++) atoms[j].vel[i]=0.0;
 }
 // C
 atoms[1].pos[1]=0.08505;
 atoms[1].pos[2]=0.43139;
 atoms[1].pos[3]=-0.14471*hand;
 // H2
 atoms[2].pos[1]=0.2517;
 atoms[2].pos[2]=0.97725;
 atoms[2].pos[3]=0.52253*hand;
 // H5
 atoms[3].pos[1]=0.04882;
 atoms[3].pos[2]=0.87147;
```

```
 atoms[3].pos[3]=-0.89389*hand;
 // Cl3
 atoms[4].pos[1]=0.9409;
 atoms[4].pos[2]=-1.31727;
 atoms[4].pos[3]=-0.05861*hand;
 // Cl4
 atoms[5].pos[1]=-1.1454;
 atoms[5].pos[2]=-1.03774;
 atoms[5].pos[3]=0.205*hand;
 // C
 atoms[1].mass=12.0;
 atoms[1].charge=2.0;
 // H2
 atoms[2].mass=1.0;
 atoms[2].charge=1.0;
 // H5
 atoms[3].mass=1.0;
 atoms[3].charge=1.0;
 // Cl3
 atoms[4].mass=35.0;
 atoms[4].charge=2.0;
 // Cl4
 atoms[5].mass=35.0;
 atoms[5].charge=3.0;
}

double kk(particle *at){

 long i,nn;
 double kt,vi;

 nn=(long)at[0].mass;
 kt=0.0;

 for(i=1;i<=nn;i++){
  vi=Norm(at[i].vel);
  kt+=0.5*at[i].mass*vi*vi;
 }
 return kt;
}

particle *PArray(long nn){

 particle *part;

 part=(particle *)malloc((nn+1)*sizeof(particle));
 part[0].mass=(double)nn;
 return part;
}

double *pp(particle *at){

 long i,j,nn;
```

```
 double *pt;

 nn=(long)at[0].mass;
 pt=Vector(3);

 for(i=1;i<=3;i++) pt[i]=0.0;
 for(i=1;i<=3;i++){
  for(j=1;j<=nn;j++) pt[i]+=at[j].vel[i]*at[j].mass;
 }
 return pt;
}

void Propagate(particle *atoms,long ntimes,double ht,char *tname,char *vname){

 FILE *f1=fopen(tname,"w");
 FILE *f2=fopen(vname,"w");
 FILE *f3=fopen("kinetic_energy.txt","w");
 long i,j,k,natoms;
 double *acc,**prevvel,**prevpos,**currpos,nextvel,*tempmo;

 natoms=(long)atoms[0].mass;
 prevvel=Matrix(natoms,3);
 prevpos=Matrix(natoms,3);
 currpos=Matrix(natoms,3);

 for(i=1;i<=natoms;i++){
  for(j=1;j<=3;j++){
   currpos[i][j]=atoms[i].pos[j];
   prevvel[i][j]=0.0;
   prevpos[i][j]=atoms[i].pos[j];
  }
 }

 for(i=1;i<=ntimes;i++){
  for(j=1;j<=natoms;j++){
   for(k=1;k<=3;k++) fprintf(f1,"%.9f ",currpos[j][k]);
   fprintf(f1,"\n");
  }
  fprintf(f3,"%E %E\n",(double)(i-1)*ht,kk(atoms));
  for(j=1;j<=natoms;j++){
   acc=aa(j,atoms,currpos,(double)(i-1)*ht);
   for(k=1;k<=3;k++){
    nextvel=prevvel[j][k]+2.0*ht*acc[k];
    atoms[j].pos[k]+=0.5*ht*(nextvel+atoms[j].vel[k]);
    prevvel[j][k]=atoms[j].vel[k];
    atoms[j].vel[k]=nextvel;
   }
   free(acc);
  }
  for(j=1;j<=natoms;j++){
   for(k=1;k<=3;k++) currpos[j][k]=atoms[j].pos[k];
  }
 }
```

```
fclose(f1);
tempmo=pp(atoms);

fprintf(f2,Concat(vname,"\n\n"));
fprintf(f2,"C  final position:\n");
fprintf(f2,"r = (%E,%E,%E) m\n",
atoms[1].pos[1]*LS,atoms[1].pos[2]*LS,atoms[1].pos[3]*LS);
fprintf(f2,"|r| = %E m\n\n",Norm(atoms[1].pos)*LS);

fprintf(f2,"H2  final position:\n");
fprintf(f2,"r = (%E,%E,%E) m\n",
atoms[2].pos[1]*LS,atoms[2].pos[2]*LS,atoms[2].pos[3]*LS);
fprintf(f2,"|r| = %E m\n\n",Norm(atoms[2].pos)*LS);

fprintf(f2,"H5  final position:\n");
fprintf(f2,"r = (%E,%E,%E) m\n",
atoms[3].pos[1]*LS,atoms[3].pos[2]*LS,atoms[3].pos[3]*LS);
fprintf(f2,"|r| = %E m\n\n",Norm(atoms[3].pos)*LS);

fprintf(f2,"Cl3 final position:\n");
fprintf(f2,"r = (%E,%E,%E) m\n",
atoms[4].pos[1]*LS,atoms[4].pos[2]*LS,atoms[4].pos[3]*LS);
fprintf(f2,"|r| = %E m\n\n",Norm(atoms[4].pos)*LS);

fprintf(f2,"Cl4 final position:\n");
fprintf(f2,"r = (%E,%E,%E) m\n",
atoms[5].pos[1]*LS,atoms[5].pos[2]*LS,atoms[5].pos[3]*LS);
fprintf(f2,"|r| = %E m\n\n\n",Norm(atoms[5].pos)*LS);

fprintf(f2,"C  final velocity:\n");
fprintf(f2,"v = (%E,%E,%E) m/s\n",
atoms[1].vel[1]*VS,atoms[1].vel[2]*VS,atoms[1].vel[3]*VS);
fprintf(f2,"|v| = %E eV\n\n\n",
0.5*atoms[1].mass*Sqr(Norm(atoms[1].vel))*ES/CS);

fprintf(f2,"H2  final velocity:\n");
fprintf(f2,"v = (%E,%E,%E) m/s\n",
atoms[2].vel[1]*VS,atoms[2].vel[2]*VS,atoms[2].vel[3]*VS);
fprintf(f2,"|v| = %E eV\n\n\n",
0.5*atoms[2].mass*Sqr(Norm(atoms[2].vel))*ES/CS);

fprintf(f2,"H5  final velocity:\n");
fprintf(f2,"v = (%E,%E,%E) m/s\n",
atoms[3].vel[1]*VS,atoms[3].vel[2]*VS,atoms[3].vel[3]*VS);
fprintf(f2,"|v| = %E eV\n\n\n",
0.5*atoms[3].mass*Sqr(Norm(atoms[3].vel))*ES/CS);

fprintf(f2,"Cl3 final velocity:\n");
fprintf(f2,"v = (%E,%E,%E) m/s\n",
atoms[4].vel[1]*VS,atoms[4].vel[2]*VS,atoms[4].vel[3]*VS);
fprintf(f2,"|v| = %E eV\n\n\n",
0.5*atoms[4].mass*Sqr(Norm(atoms[4].vel))*ES/CS);
```

```
  fprintf(f2,"Cl4 final velocity:\n");
  fprintf(f2,"v = (%E,%E,%E) m/s\n",
  atoms[5].vel[1]*VS,atoms[5].vel[2]*VS,atoms[5].vel[3]*VS);
  fprintf(f2,"|v| = %E eV\n\n\n",
  0.5*atoms[5].mass*Sqr(Norm(atoms[5].vel))*ES/CS);

  fprintf(f2,"total momentum = (%E,%E,%E) kg*m/s\n",
  tempmo[1]*PS,tempmo[2]*PS,tempmo[3]*PS);
  fprintf(f2,"total kinetic energy = %E eV\n\n",kk(atoms)*ES/CS);
  fprintf(f2,"time = %E s\n",(double)ntimes*ht*TS);

  fclose(f2);
  free(tempmo);
  FreeMatrix(prevvel);
  FreeMatrix(prevpos);
  FreeMatrix(currpos);
}

double UU(double *x){

  return 0.0;
}
```

C) Simulated Annealing Algorithm For Molecular Reconstruction

```
#include  <stdio.h>
#include  <stdlib.h>
#include  <math.h>
#include  "arb.h"

#define MS 1.66053886E-27
#define CS 1.60217653E-19
#define LS 1.0E-10
#define ES 2.30707725173E-18
#define VS 3.72740468511E4
#define TS (LS)/(VS)
#define PS (MS)*(VS)

typedef struct{

  double mass;
  double charge;
  double *pos;
  double *vel;
}particle;

/*****************************************************************************
```

This program attemps to recover the geometrical structure of a molecule based on measured velocity vectors. These velocity vectors, along with an initial guess of the molecule's configuration are required. The OptimizeStructure function, the meat-and-potatoes of this program, then attempts to minimize the discrepancy between the empirical velocity vectors and the ones (calculated via a Coulomb explosion simulation carried out by the Propagate function) corresponding to the

test configuration. OptimizeStructure makes the atoms move around according to Brownian motion and for each new configuration a discriminant is calculated and compared to the previous one. New configurations are accepted according to the rules for simulated annealing. In addition, the step size of the Brownian motion is appropriately adjusted to suit the changing landscape of the multidimensional configuration space. If the algorithm gets stuck in a local minimum, the step size is briefly and quickly increased (heating) to kick it out and make it explore new configurations.
**********/
/**********
Copyright (C) 2006 Justin Gagnon

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.
**********/

```
double *Acceleration(long,particle *,double **,double);
double Discriminant(particle *,double **);
void Initialize(double **,particle *);
void OptimizeStructure(double **,double **,char *);
particle *PArray(long);
void PrintParticles(particle *);
void Propagate(double **,particle *,long,double);
void WriteAtms(FILE *,double **);

int main(){

 long i,j;
 double **atms=LoadMatrix("guess0.txt");
 double **vel=LoadMatrix("experimental_velocities_4_test.txt");
 char *fname="experimental_velocities_0_test_optimized.txt";
 OptimizeStructure(atms,vel,fname);
 //while(1);
 return 0;
}


// calculate the acceleration of an atom
double *Acceleration(long n0,particle *atoms,double **pos,double t){

 long i,j,natoms;
 double *acc,rd,rd3;
```

```
 natoms=(long)atoms[0].mass;
 acc=Vector(3);
 for(i=1;i<=3;i++) acc[i]=0.0;

 for(j=1;j<=natoms;j++){
  if(j!=n0){
   rd=Distance(pos[n0],pos[j]);
   rd3=rd*rd*rd;
   for(i=1;i<=3;i++)
    acc[i]+=atoms[n0].charge*atoms[j].charge*(pos[n0][i]-pos[j][i])/rd3;
  }
 }
 for(i=1;i<=3;i++) acc[i]/=atoms[n0].mass;
 return acc;
}

double Discriminant(particle *atoms, double **vel){

 long i,j,k,natoms;
 double discr,*velv,*dvel,*test;

 natoms=(long)vel[1][0];
 test=Vector(natoms);

 for(i=1;i<=natoms;i++){
  velv=Vector3(vel[1][i],vel[2][i],vel[3][i]);
  dvel=SubV(atoms[i].vel,velv);
  test[i]=Norm(dvel)/Norm(velv);
  free(dvel);
  free(velv);
 }
 discr=Max(test);
 free(test);
 return discr;
}

void Initialize(double **atms,particle *atoms){

 long i,j,natoms;

 natoms=(long)atms[1][0];

 for(i=1;i<=natoms;i++){
  atoms[i].pos=Vector(3);
  atoms[i].vel=Vector(3);
 }

 for(i=1;i<=3;i++){
  for(j=1;j<=natoms;j++){
   atoms[j].pos[i]=atms[i][j];
   atoms[j].vel[i]=atms[i+3][j];
  }
```

```
 }
 for(i=1;i<=natoms;i++){
  atoms[i].mass=atms[7][i];
  atoms[i].charge=atms[8][i];
 }
}

void OptimizeStructure(double **atms,double **vel,char *fnout){

 long i,j,k,seed,natms,nsteps,m,nt,stale,lost,max,maxlost;
 double **atms1,kT,**tryatms,discr1,discr2,*steps,stepsize1,stepsize,ht,
      cooling,tolerance,heating,statstepsize,vstepsize1,vstepsize,
      statvstepsize;
 particle *testp;
 FILE *fout;

 nt=20000;
 ht=0.02;
 seed=-7188;
 stepsize=2.0;
 vstepsize=0.0;
 cooling=1.005;
 heating=0.002;
 max=500;
 maxlost=100;
 tolerance=0.05;
 nsteps=6000000;
 natms=(long)atms[1][0];
 steps=BoxMuller(nsteps,seed);
 m=1;
 testp=PArray(natms);
 do{
  fout=fopen(fnout,"w");
  k=0;
  stepsize1=stepsize;
  statstepsize=stepsize;
  vstepsize1=vstepsize;
  statvstepsize=vstepsize;
  atms1=CopyMatrix(atms);
  tryatms=CopyMatrix(atms);
  stale=0;
  lost=0;
  Propagate(tryatms,testp,nt,ht);
  discr1=Discriminant(testp,vel);
  kT=0.1*discr1;
  do{
   for(i=1;i<=natms;i++){
    for(j=1;j<=3;j++){
     tryatms[j][i]=atms1[j][i]+stepsize1*steps[m];
     if(m++>nsteps) m=1;
     tryatms[j+3][i]=atms[j+3][i]+vstepsize1*steps[m];
     if(m++>nsteps) m=1;
    }
```

```
    }
    Propagate(tryatms,testp,nt,ht);
    discr2=Discriminant(testp,vel);
    if(discr2<discr1){
     FreeMatrix(atms1);
     atms1=CopyMatrix(tryatms);
     stepsize1=statstepsize;
     stepsize1*=discr2/discr1;
     vstepsize1=statvstepsize;
     vstepsize1*=discr2/discr1;
     kT*=Min2(1.0/cooling,discr2/discr1);
     discr1=discr2;
     stale=0;
     lost=0;
     printf("%d %.6f %.6e %.6e %.6e :) good\n",
     k,discr1*100.0,kT,stepsize1,vstepsize1);
    }
    else{
     if(Ran2(&seed)<exp(-(discr2-discr1)/kT)){
      FreeMatrix(atms1);
      atms1=CopyMatrix(tryatms);
      stepsize1=statstepsize;
      stepsize1*=discr2/discr1;
      vstepsize1=statvstepsize;
      vstepsize1*=discr2/discr1;
      discr1=discr2;
      lost++;
      printf("%d %.6f %.6e %.6e %.6e :{ bad\n",
      k,discr1*100.0,kT,stepsize1,vstepsize1);
     }
     else{
      stale++;
      if(stale==1){
       statstepsize=stepsize1;
       statvstepsize=vstepsize1;
      }
      stepsize1+=heating*stepsize1;
      vstepsize1+=heating*vstepsize1;
     }
    }
    WriteAtms(fout,atms1);
    k++;
   }while(stale<=max && lost<=maxlost);
   fprintf(fout,"DELTA=%.2f percent after %ld iterations\n",discr1*100.0,k);
   printf("stale\n");
   fclose(fout);
  }while(discr1>=tolerance);
  fclose(fout);
}

particle *PArray(long nn){

  long i;
```

```
  particle *part;

  part=(particle *)malloc((nn+1)*sizeof(particle));
  part[0].mass=(double)nn;
  for(i=1;i<=nn;i++){
   part[i].pos=Vector(3);
   part[i].vel=Vector(3);
  }
  return part;
}

void PrintParticles(particle *atoms){

  long natoms,i;

  natoms=(long)atoms[0].mass;

  for(i=1;i<=natoms;i++){
   printf("PARTICLE %d:\n\n",i);
   printf("FINAL POSITION:\nr = (%E,%E,%E) m\n",
   atoms[i].pos[1]*LS,atoms[i].pos[2]*LS,atoms[i].pos[3]*LS);
   printf("|r| = %E m\n\n",Norm(atoms[i].pos)*LS);
   printf("FINAL VELOCITY:\nv = (%E,%E,%E) m/s\n",
   atoms[i].vel[1]*VS,atoms[i].vel[2]*VS,atoms[i].vel[3]*VS);
   printf("|v| = %E m/s\n\n",Norm(atoms[i].vel)*VS);
   printf("CHARGE: q = %f e\n",atoms[i].charge);
   printf("MASS: m = %f amu\n\n\n",atoms[i].mass);
  }
}

void Propagate(double **atms,particle *atoms,long ntimes,double ht){

  long i,j,k,natoms;
  double *acc,**prevvel,**prevpos,**currpos,nextvel;

  natoms=(long)atms[1][0];
  prevvel=Matrix(natoms,3);
  prevpos=Matrix(natoms,3);
  currpos=Matrix(natoms,3);

  Initialize(atms,atoms);
  for(i=1;i<=natoms;i++){
   for(j=1;j<=3;j++){
    currpos[i][j]=atoms[i].pos[j];
    prevvel[i][j]=0.0;
    prevpos[i][j]=atoms[i].pos[j];
   }
  }
  for(i=1;i<=ntimes;i++){
   for(j=1;j<=natoms;j++){
    acc=Acceleration(j,atoms,currpos,(double)(i-1)*ht);
    for(k=1;k<=3;k++){
     nextvel=prevvel[j][k]+2.0*ht*acc[k];
```

```
      atoms[j].pos[k]+=0.5*ht*(nextvel+atoms[j].vel[k]);
      prevvel[j][k]=atoms[j].vel[k];
      atoms[j].vel[k]=nextvel;
     }
     free(acc);
    }
    for(j=1;j<=natoms;j++){
     for(k=1;k<=3;k++) currpos[j][k]=atoms[j].pos[k];
    }
  }
  FreeMatrix(prevvel);
  FreeMatrix(prevpos);
  FreeMatrix(currpos);
}

void WriteAtms(FILE *f,double **table){

  long i,j,cols,rows;

  cols=(long)table[0][0];
  rows=(long)table[1][0];

  for(i=1;i<=rows-1;i++){
   for(j=1;j<=cols-1;j++) fprintf(f,"%.12E ",table[j][i]);
   fprintf(f,"%.12E\n",table[j][i]);
  }
  for(j=1;j<=cols-1;j++) fprintf(f,"%.12E ",table[j][i]);
  fprintf(f,"%.12E\n",table[j][i]);
}
```