Improvements to detection efficiency and measurement accuracy in Coulomb Explosion Imaging experiments

by

Benjamin Wales

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Science
in
Physics

Waterloo, Ontario, Canada, 2011

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

An algorithm for extracting event information from a Coulomb Explosion Imaging (CEI) position sensitive detector (PSD) is developed and compared with previously employed schemes. The PSD is calibrated using a newly designed grid overlay and validates the quality of the described algorithm. Precision calculations are performed to determine how best the CEI apparatus at The University of Waterloo can be improved. An algorithm for optimizing coincidence measurements of polyatomic molecules in CEI experiments is developed. Predictions of improved efficiency based on this algorithm are performed and compared with experiments using a triatomic molecule. Analysis of an OCS targeted CEI experiment using highly charged Argon ions to initiate ionization is performed. The resulting measurements are presented using a variety of visualization tools to reveal asynchronous and sequential fragmentation channels of OCS^{3+}.

Acknowledgments

I would like to acknowledge my supervisor Dr. Joseph Sanderson whose wisdom and wit encouraged taking my research to eleven. Insights into laser phenomena provided by Dr. Donna Strickland and the phenomenon of operating such systems by Ph.D. candidate Reza Karimi brought *completion* to finished projects.

For keeping the reigns on my horses throughout the seasons I must acknowledge the brine, brews, and boisterous laughter shared with my family and closest friends. Also, Supertramp.

Table of Contents

List of Figures

List of Tables

1   Introduction

1.1   Background

Since Eadweard Muybridge first observed the rapid gallop of a horse, the technique of photography, or more generally, imaging, has become a tool of scientific inquiry. Beginning with this set of 1ms snapshots of an object several meters in breadth, imaging has evolved magnificently improving by over ten orders of magnitude in space (eg Atomic Force Microscopy) and time (femtosecond laser initiated Coulomb Explosion Imaging (CEI)). The latter technique is a developing field and the focus of this work. The scope of this imaging method is to make measurements of single molecule geometry on a timescale faster than molecular motion. Ultimately these images can be recorded in sequence to image the dynamics of a molecule which has been set in my motion by a controllable process such as laser initiated ionization. The method involves removing many electrons from a molecule which then explodes under Coulomb repulsion. By detecting all of these fragments in coincidence and measuring their complete momentum it is possible to measure molecular structure. Methods of initiating such a multiple ionization process range from accelerating a molecular ion to several MeV and colliding it with a thin foil [1,2], colliding a highly charged atomic ion with a neutral molecule [3-5], or exposing a molecule to the intense electric field of a femtosecond laser pulse [6,7]. The highly controllable nature of the laser pulse makes imaging of dynamics possible through pump-probe experiments [8] where a single pulse initiates a specific state, and a second pulse causes controlled explosion moments later. Such a system is very similar to Muybridge's experiment but, with current laser technologies, images dynamics on the order of femtoseconds.

CEI experiments in recent years have focused on diatomic molecules such as D_2 [9,10]and triatomic molecules such as OCS [11], CO_2 [12], and CS_2 [13,14]. The former experiments used few cycle laser pulses to initiate fragmentation of low charge ions yielding insight into the structure of the potential well by measuring fragment ion energies. The studies have provided further insight into the interaction between an ionized electron and the laser field by observing electron rescattering as a means of multiple ionization and enhanced ionization. The latter studies have focused on the nuclear geometry of triatomic molecules and the extent to which they are affected by laser fields and the ionization process. It has been shown that high charge state ionization of a triatomic molecule through either highly charge ion interaction [5] or high intensity laser interaction [6] produces ionic fragments with momenta characteristic of the ground state molecular geometry. Low charge state fragmentation has revealed molecular bond stretching, bending, and sequential fragmentation where a metastable ion is formed before complete breakup occurs [3].

The CEI apparatuses used in the above experiments all combine high resolution time-of-flight (TOF) mass spectrometry with position-sensitive detectors (PSD) [12,15-17] to measure the momenta of

all fragments from a Coulomb explosion in coincidence. Due to the numerous fragmentation pathways, specific channels are defined using the convention (a,b,c) where the following process has occurred:

$$ABC^{q+} \rightarrow A^{a+} + B^{b+} + C^{c+} \qquad (q = a + b + c)$$
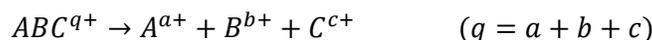
Once a fragmentation channel has been determined and fragment momenta have been recorded, all the necessary information to reconstruct the molecular geometry is available [12,18,19]. Figure 1 below shows a photo of the CEI apparatus out of vacuum used at The University of Waterloo, while Figure 2 shows a schematic of the detection process. It has been shown [20] how geometry reconstruction can be optimized using a simplex algorithm to match measured asymptotic momenta with the results of a classical trajectory calculation.

1.2   Project Description

The role of this project is to characterize the limits of current CEI technology at Waterloo and optimize the computational components on two fronts: data acquisition and analysis. To achieve improvements applicable in all CEI systems, several experiments were performed at a variety of institutions.

a) To begin with, data acquisition from the PSD is first improved upon through the implementation of a new detection scheme and tested with recorded data of laser initiated CS_2 fragmentation from the CEI apparatus at The University of Waterloo (UW). To assess the improvement, the following chapter presents a new method for calibrating a PSD. The calibration is performed at UW using N_2 as a target molecule in a CEI experiment.

b) Momentum precision – The limit of CEI resolution is calculated and demonstrated in a pair of experiments. The first uses N_2 as a target molecule at UW, while the second uses OCS as a target molecule in a similar CEI apparatus at the Advanced Laser Light Source (ALLS) [21].

c) Improvements to the analysis of CEI data sets are made by implementing a sophisticated algorithm capable of optimizing coincidence count rates. This chapter follows a paper accepted for publication [22] and uses the same OCS data collected at ALLS described above. Pertinent addendums to this paper are included in the appendix of which the first uses measurements made with N_2O as a target at ALLS.

d) An experiment which implements the analysis technique described in chapter 5 to produce several physical metrics of OCS fragmentation is described. This data was recorded at Tokyo Metropolitan University (TMU) using highly charged ions to ionize target OCS molecules. The work described in this chapter has been submitted for publication in a modified form.
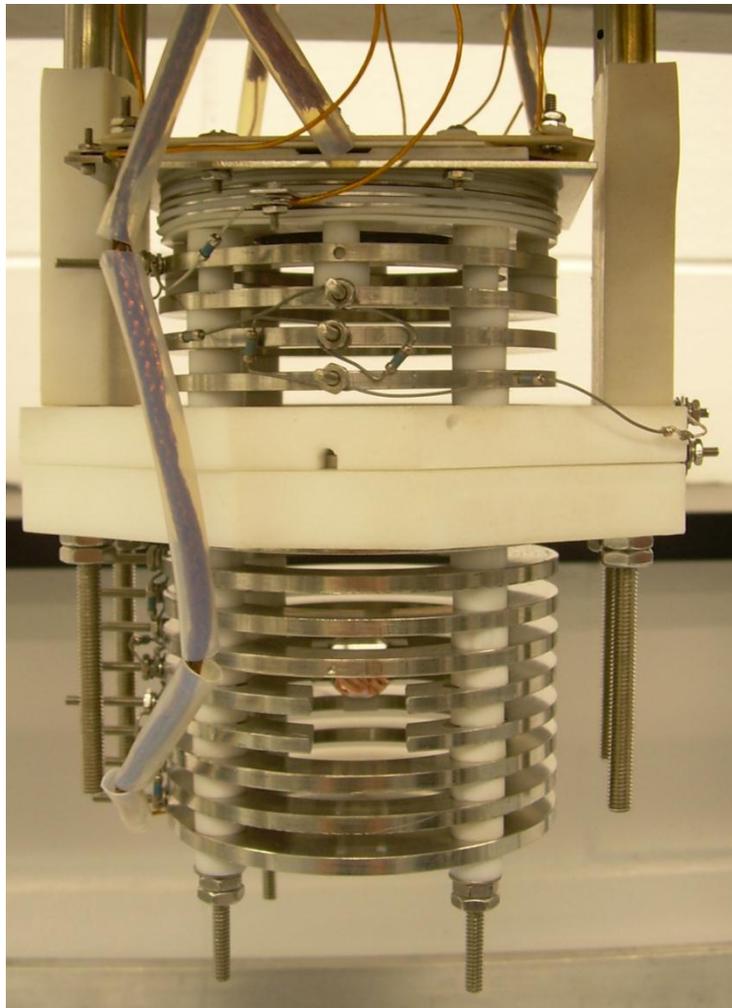
Figure 1 – Photo of CEI Apparatus
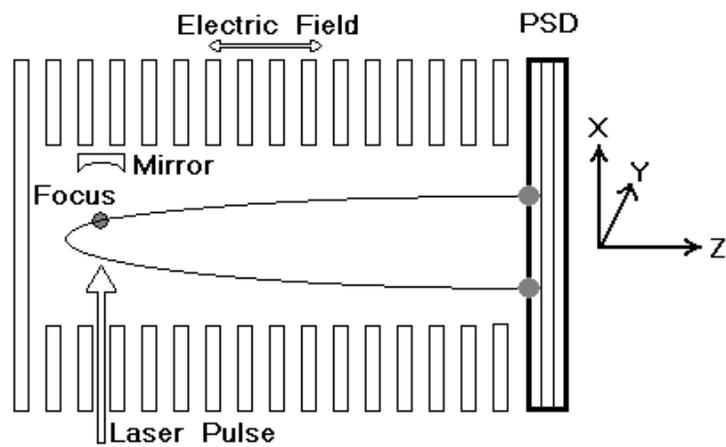
Figure 2 – Schematic of CEI apparatus in operation

2 Waveform Analysis Algorithm

2.1 Motivation

CEI experiments using laser pulse initiated ionization produce several ions per laser shot all detected with a single PSD. The system demands fast measurements of event position and time with the capability of presenting recorded data on the fly. Delay line anodes [23] and modified backgammon weighted-coupling capacitor [24] detectors theoretically meet the resolution needs for CEI. The latter requires digitizing the voltage output and analyzing the waveform to extract event position and time. Waveform analyzing algorithms [7] have been successfully used but produce inaccurate measurements with high count rate experiments. A new waveform analyzing algorithm is developed to produce high accuracy measurements and discriminate against false events while maintaining computational demands low enough for on-the-fly readout.

2.2 Physical Structure of PSD

The time and position sensitive anode in these experiments is a modified backgammon weighted-coupling capacitor (MBWC) [24] (Figure 3). This modification on the general wedge and strip design uses capacitive coupling allowing faster time response. The anode receives a charge cloud produced from a pair of microchannel plates (MCPs) when an ion strikes the PSD surface. The charge cloud from the MCPs deposits on several of the anode's wedges, and travels left and right towards the PSD terminals. The ratio of the charge escaping left to right provides a measurement of the location of the electron cloud in one dimension (x-axis). The charges located at either end of the anode are subsequently divided by two sets of wedges capacitively coupled to readouts at the top and bottom of the anode. In similar fashion to determining the location of the charge cloud along the x-axis, the ratio of the charge deposited from the top to the bottom is used to measure the location of the electron cloud along the y-axis.
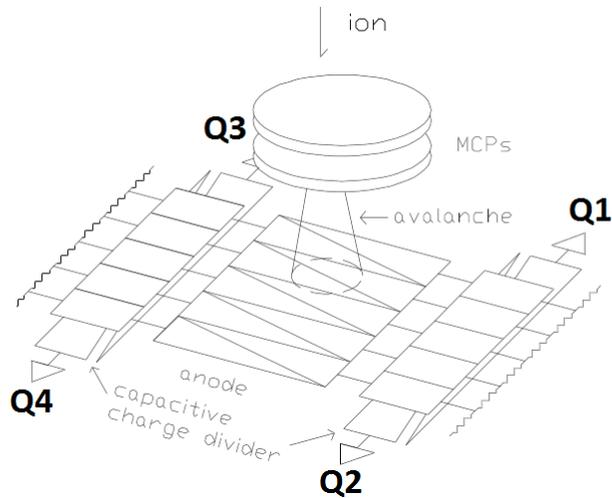
**Figure 3 – Modified Weighted Backgammon Capacitor anode used as the CEI PSD at UW
Charge outputs Q1 through Q4 are labeled.**
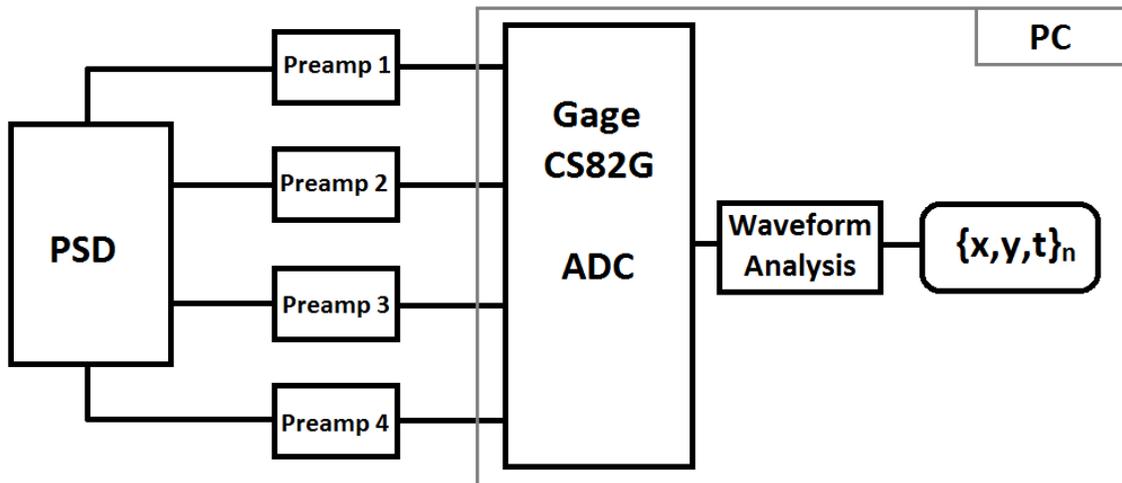
Figure 4 – Schematic of PSD readout system

The voltage created by the charge deposition is amplified at the four outputs with fast pre-amps (Ortec 142B), then sent to a four channel ADC to digitize the voltages (Figure 4). The position of an electron cloud (and so the position of the detected ion) is calculated by the ratios between voltage steps as follows:

$$X = \frac{Q_1 + Q_2}{Q_1 + Q_2 + Q_3 + Q_4} = \frac{Q_1 + Q_2}{Q_T}$$

$$Y = \frac{Q_1 + Q_3}{Q_1 + Q_2 + Q_3 + Q_4} = \frac{Q_1 + Q_3}{Q_T}$$

2.3 Waveform Characterization

A typical set of waveforms is shown in Figure 5. The time axis ranges from 0 to 1023 and counts the number of data points. Depending on the digitizing rate selected for the ADCs, each data point may be spaced between 1ns and 5ns. The voltage on the y-axis is the raw digitized value ranging from 0 to 255 for each preamp channel. Since the waveform analysis of position readout is independent of calibrated voltage measurements, reduced voltages – that is, the raw digitized value – are used and annotated as *du* (digitized unit). Similarly, measurements of time are annotated as *du* when the acquisition rate has not been used for calibration. Without any processing to the data set presented in Figure 5, flat areas, steps, and ringing are easily identified. The ringing was previously identified [7] as an inherent feature of the system and has been observed in other PSDs making use of the same anode design. Though the ringing changes shape from step to step and channel to channel, the shape resulting from the sum of the channels (shown in the inset) is a well-defined exponentially decaying sine function. A fit to the sum waveform within the interval of the step (solid overlay) confirms that the ringing is of the form:

$$y = BL_1 + (BL_2 - BL_1)e^{\frac{-t}{\tau_{decay}}}\cos(2\pi\frac{t}{T})$$

where BL_1 is the baseline voltage before the step, BL_2 is the baseline voltage after the step, τ_{decay} is the decay time constant, and T is the ringing period.
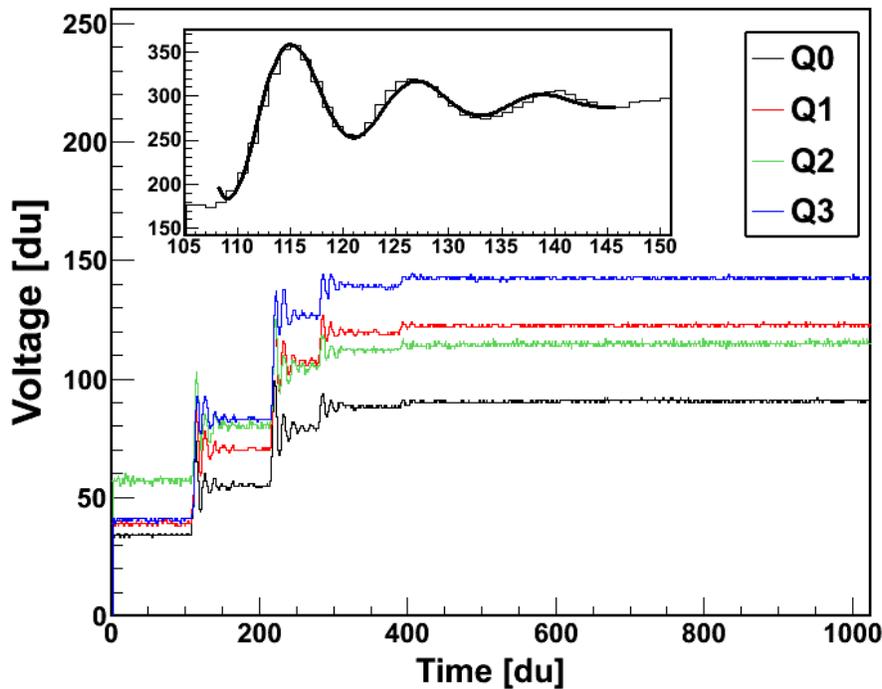
Figure 5 – Sample PSD waveforms from four channels.
Inset shows the sum waveform fitted with decaying sine function.

The accuracy of position measurements depends exclusively on how well the height of each baseline (flat region) between each step can be resolved. The accuracy of time measurements, however, requires a consistent scheme for determining when a voltage jump starts. Both of these measurements are complicated by the ringing structure of each voltage step.

A previously employed scheme for measuring step heights and step time uses the following steps:

1. Apply a triangle filter to the sum waveform.
2. Starting from t=0, calculate the derivative at each point.
3. If the derivative is large enough, begin to calculate voltage step, otherwise continue step 2.
4. Average the voltages for a number of points leading and following the high derivative point for each waveform and the sum waveform.

From step 4, enough information is collected to calculate the event position and time. The benefit to this algorithm is its simplicity and in turn the speed with which event information can be calculated. However, several faults are visible with this scheme. The derivative triggering for the beginning of a voltage step is likely to trigger on the second upward rise of a ringing structure leading to false event detection. The averaging scheme for calculating end baselines very poorly approximates the final baseline (in previous work, this led to skew in position measurements which were partly adjusted for by including a skew matrix correction). When two events occur shortly after the first ringing peak is reached, they are considered one event, and a false position is calculated.

In contrast, a group at Tokyo Metropolitan University has employed a much more sophisticated algorithm to read out voltage baselines and step times [Hayakawa Thesis, TMU 2010]. Essentially, the sum waveform is projected onto the voltage axis as a histogram (See Figure 6).
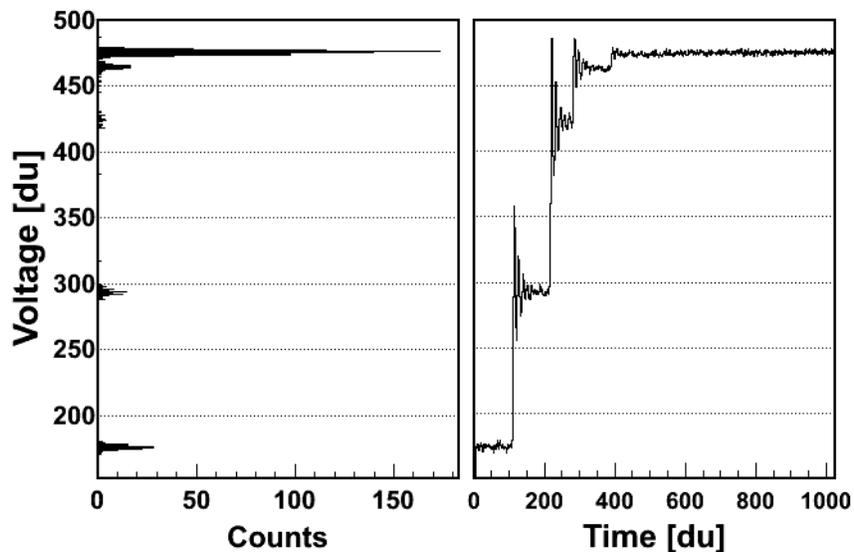
Figure 6 – Sample waveform and projected histogram for alternative analysis algorithm

A peak finding algorithm making use of positive and negative derivative calculations is used to find the voltages at which the projected histogram spikes. The difference between peak voltages of successive spikes determines the voltage steps and in turn is used to calculate event positions. Event time is determined by searching through the channel sum waveform for the time at which the voltage peak value is reached. This technique is an improvement on the simple scheme but has its own faults. As can be seen in Figure 6, the third peak starting from the bottom is muddled due to excessive ringing. This can also occur if a second event causes the voltage to jump before the ringing dies down. If the peak is misread, two voltage steps are miscalculated and two event positions are skewed. If the peak is ignored all together, two events are identified as one and a false position is calculated. Lastly, in determining the event time, the second peak in the ringing structure of a step can signal a false event time if it rises above the subsequent voltage step.

In situations where ringing is non-existent, the first scheme is applicable while the more sophisticated algorithm is appropriate for waveforms with moderate ringing. Since the anode in use at UW causes substantial ringing, an algorithm was designed to actively analyze the waveform sets exploiting the exponentially decaying sine step behavior.

2.4   Ringing Characterization

The sinusoidal waveforms appear consistent in Figure 5 and Figure 6 and suggest that there is a well-defined ringing shape based solely on the voltage step height. If this is indeed the case for all single hit events, it is then justifiable to use an algorithm which discriminates against those ringing shapes which do not conform. Ideally a fit of an exponential decaying sinusoid would be done at every step as a form of discrimination, but such efforts are computationally heavy and therefore not feasible for real-time acquisition. Since, however, the exponentially decaying sine formula requires that the first peak and the second peak in the ringing be proportional to the full step voltage, these values can be used as measures of single-event step Quality.

To check that the relationship between peak step heights and full step height holds water, analysis of a data set with low count rates was performed. By keeping the count rates low (<0.1 per laser shot) it is ensured that each detected event is in fact a single hit event. For this experiment, background data was used from operating the CEI apparatus at low pressures with 300μJ laser pulses.
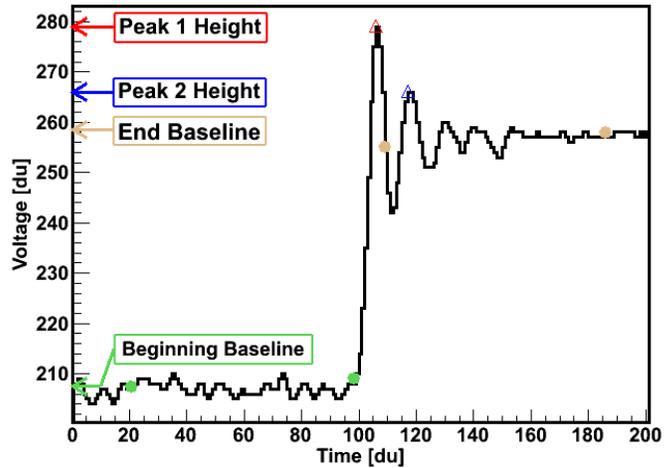
Figure 7 – Sample sum waveform from single hit event
Red and blue triangles indicate the first and second peaks respectively. Green and brown circles indicate the
ranges over which the voltages are averaged to calculate the beginning and end baselines respectively.

To begin with, each waveform is processed to identify peaks and the location of the beginning of a step. This is done by using the peak and baseline finding algorithm described in the following section. The beginning voltage baseline is calculated by averaging the 80 data points before the step. The final baseline is calculated by using the 80 data points following half way between the first peak and the first trough. The difference between these values gives the full voltage step. Since ringing reduces to background levels by 40 data points, using 80 points to measure a baseline ensures an accurate reading (see Figure 7). The difference between the first peak and the voltage baseline before the step begins gives the first peak voltage step while the difference between the second peak and the voltage baseline before the step begins gives the second peak voltage step. Figure 8 and Figure 9 present the peak voltage step versus the full voltage step for steps above 10du (as steps smaller than 10du cannot be used in experiment for rendering accurate positions).

9
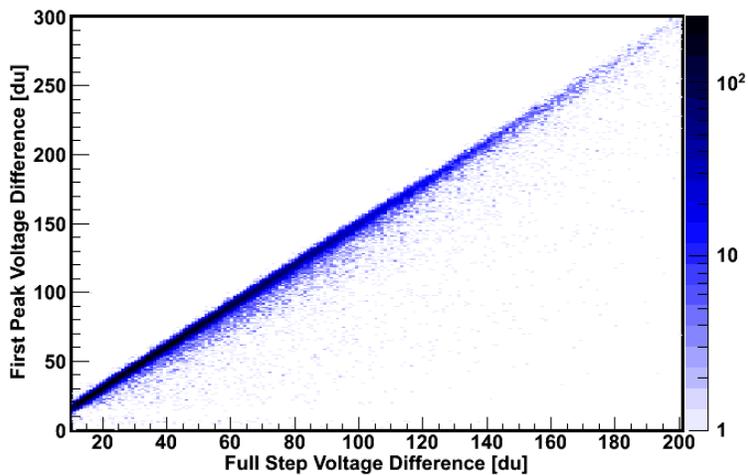
Figure 8 – First voltage peak correlation with full voltage step
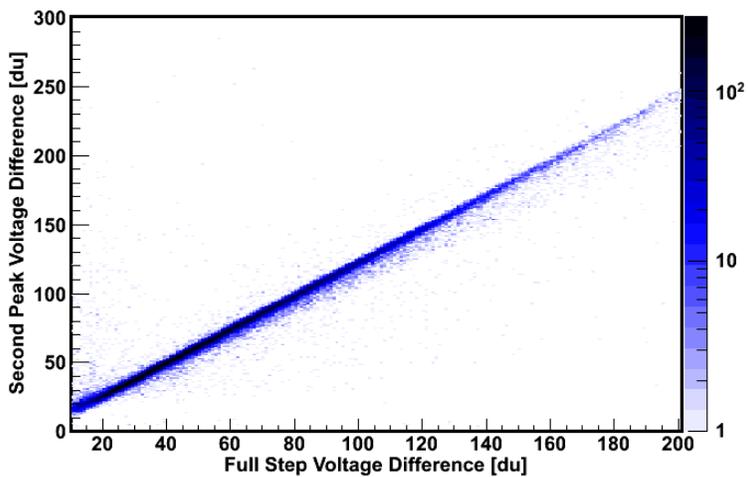
Figure 9 – Second voltage peak correlation with full voltage step

The continuity of peaks in these figures creates a well-defined line, confirming that single events produce ringing waveforms whose first and second peaks are proportional to the full voltage step. Furthermore, the distribution of points about the visible line gives an estimate of uncertainty. 89% of the data points in the Figure 8 are within 5du of the expected peak voltage for a given full step voltage. The other 11% scattered outside this region are likely the result of multiple hits or noise interfering with the ringing structure. In case of the former, position information could not be correctly resolved and, in fact, false events would be detected – exactly the events to be excluded. In the latter case, position information could be reconstructed properly if a following event occurs late enough such that an accurate baseline could be measured. Otherwise, inaccurate measurements of position and time would be made, and it is appropriate to discriminate against such an event.

Using the slope from the first figure, it is concluded that an estimate of the full voltage step from a measurement of the first peak can be calculated by multiplying the peak step height by 0.67. Furthermore, an estimate of the second peak height can be made from measurement of the first peak height by multiplying the first peak height by 0.82. Waveforms which do not meet these estimates by +-5du can be said to have occurred from multiple hit events or noisy signals and should be disregarded.

Lastly, Figure 10 shows the distribution of time between the first and second peaks (the period T in equation above). The sharp spike indicates that single events all have the same time constant for the sinusoid (about 11du between peaks). The small peak located at 6du accounts for <5% of the data and is likely due to noise or multi-hit events. It is reasonable then to use this time constant as a discriminator against multi-hit events.
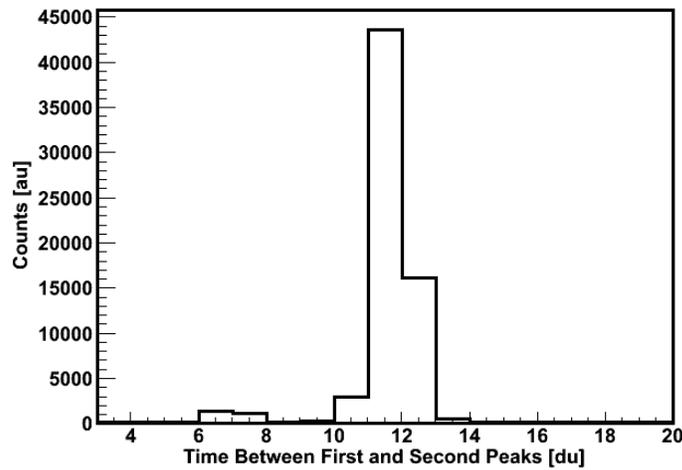
Figure 10 – Characteristic time between first and second voltage peaks

2.5 New Algorithm Description

In a broad scope, the sum of the four channels is used to identify events by examining the plateaus and peaks throughout the waveform. The event hit time can be read accurately from the sum waveform even if it has been modified slightly with an averaging routine. The voltage steps, however, are calculated using the raw data from each channel. This technique allows one to most accurately discriminate against false events while measuring event position with the highest accuracy.

The sum of the four channels is produced for identifying events. To reduce noise, a triangle filter is used which roughly averages a voltage measurement with its neighbouring voltage measurements. Explicitly, the following formula is used:

$$moddata[i] = \frac{1}{9}(sumData[i-2] + 2\ sumData[i-1] + 3\ sumData[i] + 2\ sumData[i+1] + sumData[i+2])$$

where the square brackets are used as indexes for the array of waveform data (A-I smoothData). The event finding algorithm proceeds using the triangle filter waveform from the sum of the four channels. The first task is to identify flat regions and stepping regions (A-II findPeaks). The algorithm assumes the waveform begins at a plateau. This is strictly true for datasets with laser triggering since events that strike the PSD will occur sometime after the laser shot, causing any steps in voltage to occur after the beginning of the waveform. From the first data point, the waveform is sampled forward, searching for high derivatives (Figure 11 section A). If the derivative of a point is above a threshold (*DivThreshold* (A-II findPeaks)), the point is flagged as a potential rising step. If the following *RisingThreshold* (A-II findPeaks) many points also have derivatives above *DivThreshold*, then the first point is flagged as an actual rising step ; otherwise the potential rising step flag is dropped and the algorithm continues searching for a streak of high derivatives. Once a streak of high derivatives is found, the beginning of the streak is registered as the end of a plateau (Figure 11 section B). The peak of the rising voltage is then found by looking for a point whose voltage drops below that of the previous point. Once such a data point is found, it is registered as a peak (Figure 11 section C).

Since a rise in voltage is expected to peak then drop again, the algorithm searches for the following trough using the same criteria as before (find the point whose following voltage is higher than its own (Figure 11 section D)). The beginning of a plateau is placed half way between this low trough and the previous high peak. This location is chosen as it generally allows for a better calculation of the plateau baseline. The algorithm then begins from whence it started – searching for a rising streak in voltage.

This simple processing produces the location of the beginning and end of plateaus, as well as the major positive peaks. Another routine (A-III processPeaks) processes these peaks to identify which steps can be resolved accurately, and which events occurred too closely to one another to be correctly resolved.
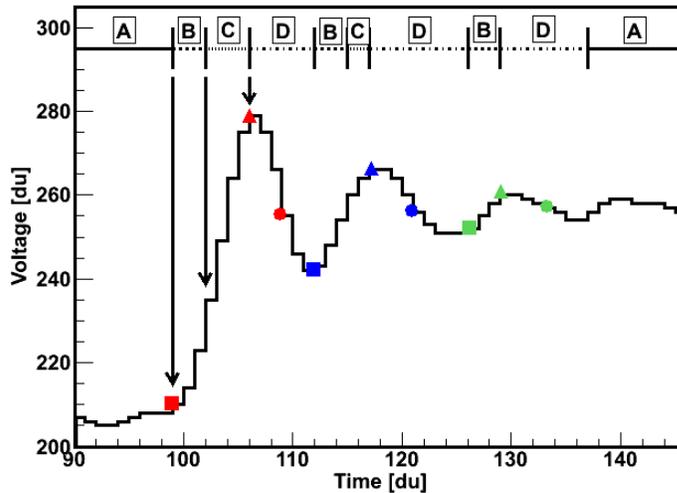
Figure 11 – Ringing algorithm processing stages
Circles and squares show beginning and end of baseline regions. Triangles show peaks. See text for description of processing sections A through D.

To begin with, a baseline is calculated for the beginning and end of each plateau. This is done by averaging the voltages after and before the beginning and ends respectively. The number of points used is the smallest of the following: the parameter *NBaselineAvg* (A-III processPeaks) or the number of points between the beginning and end of the plateau. By using the former, the algorithm safely uses voltages more local to a step in case there was a slow rise or drop in voltage not registered as peaks in the previous processing scheme. The latter limits the points to those roughly outside the initial voltage step such that the final average voltage more accurately represents the final step voltage. The peak voltages are also registered. This collection of values is shown as the solid red curve in Figure 12 where the peaks have been registered at the triangle markers and the baselines averaged between the beginning (red circle) and end (blue circle) of a local baseline region.

13
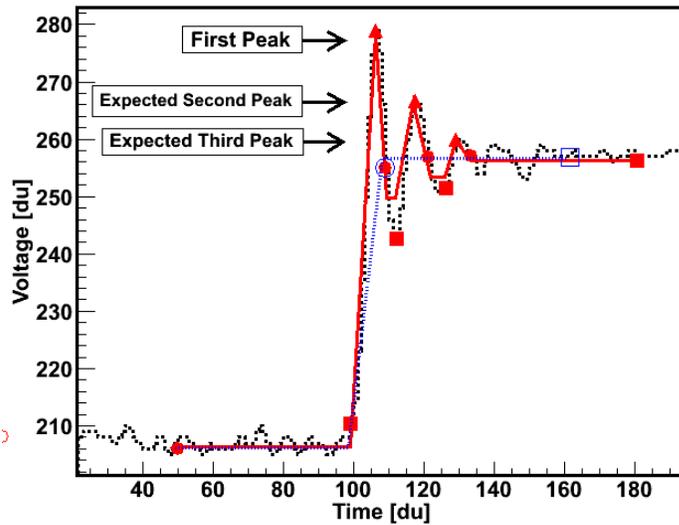
Figure 12 – Sample voltage step with identified peaks (triangles) and baseline regions (circles to squares)
The arrows point to the voltage height predicted for the second and third peaks using the first baseline and the first peak. The curve indicates the calculated baselines (averaged from the open blue circle to square) once the ringing filter has removed the peaks.

With the baselines and peak voltages measured, the algorithm applies a cut using the expected sinusoidal decay waveform. Starting with the first step (from the first baseline and the first peak), the expected voltage height of the following peak is calculated if it were to come from ringing. If the voltage of this peak is below this value (theoretical + *RingDiffThreshold* (A-III processPeaks)), it is considered to have come from the ringing structure of the first step. If, however, the peak is higher than this value, this peak is considered the first peak in a new step. The same voltage ringing discrimination is then done using this new step peak as the reference.
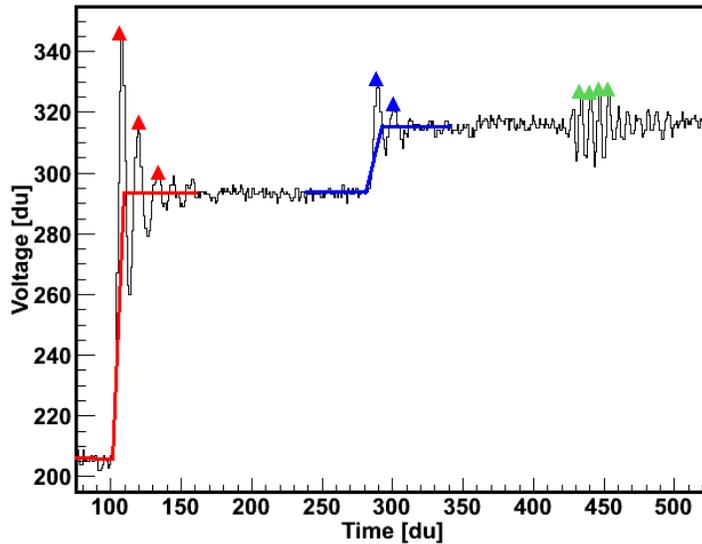
Figure 13 – Sample waveform with two steps and a noise pattern
The First blue peak is above the expected ringing height calculated using the first red peak and therefore is identified as a new step. The voltage steps calculated about the green peaks are too small to be considered a step.

Once completed, the new list of baselines and peaks will be free of those generated from ringing. The voltage step before and after a peak is calculated using the measured baselines (blue curve in Figure 12). Voltage steps which are less than a small cutoff (*BaselineCutoff* (A-III processPeaks)) are removed from the list of peaks and baselines. This discrimination is effective at removing false event triggers caused by noise worms which have little effect on accurately measuring baselines, but frequently trigger peaks (See noise sample in Figure 13).

This reduced set of peaks and baselines is then discriminated using the deadtime parameter *DeadtimeThresh* (A-III processPeaks). If the time difference between subsequent peaks – and thus event TOFs – is smaller than *DeadtimeThresh*, the two events are thrown out. This discrimination reduces the statistics of high event waveforms but greatly improves the signal to noise ratio of TOF spectra (see following section). For example, a low value for *DeadtimeThresh* will register 5 events in Figure 14 below. A slightly higher value will eliminate events 2 and 3, while a deadtime threshold of approximately 25 will only leave event 1.
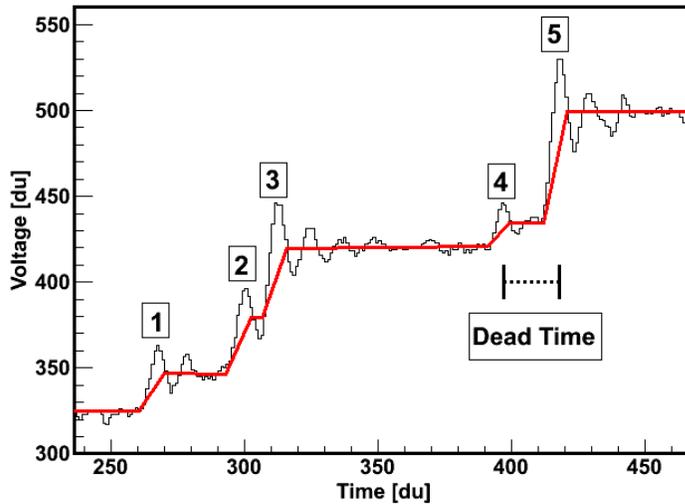
Figure 14 – Sample sum waveform from high count rate dataset
Red overlay shows calculated baselines.

Lastly, multi-hit events are discriminated by calculating the expected voltage step based on the first voltage peak. If the measured step is more than the expected + *RingDiffThreshold*, it can be concluded that two events occurred at nearly the same time. The slight time difference causes a phase offset in the ringing structure, reduces the first peak, but results in the same high voltage step (see Figure 15).
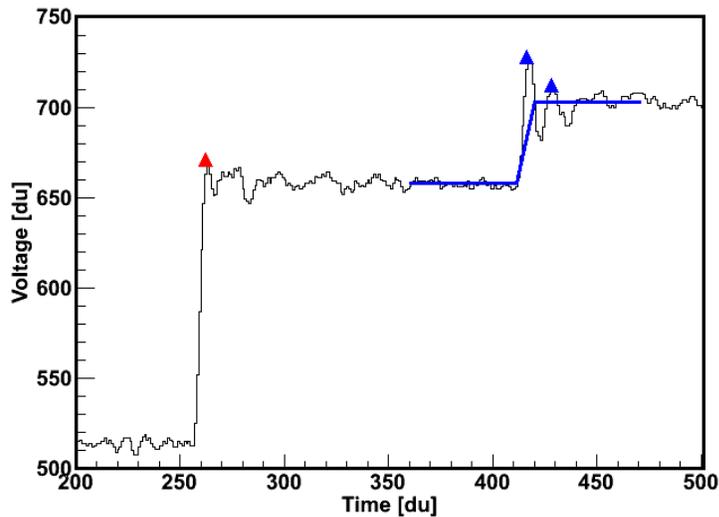
Figure 15 – Sample sum waveform with multi-hit event
The first peak (red triangle) is a result of two events occurring slightly after one another. The phase difference in the ringing structure eliminates the subsequent ringing and reduces the height of the first peak, unlike the single hit event in blue.

16

This battery of discrimination against voltage steps, event times, and comparisons with theory, eliminates multi-hit events that would otherwise be registered as single events, and records only those events whose position and TOFs can confidently and accurately be measured. By adjusting the discriminating parameters, better signal to noise ratio can be achieved at the cost of lowered count rates and vice versa. The positions are calculated using the ratio of baselines passing the discrimination tests. The event time requires further processing due to the large rise time of a voltage step. Beginning with the point that initiates a rising voltage, the algorithm searches for the point at which half the voltage step has occurred. Using the voltage points just below and just above the mid step voltage, a linear interpolation is performed to determine the time at which half the voltage step has occurred (see Figure 16):

$$ t_{hit} = \frac{V_{hit} - V_1}{V_2 - V_1}(t_2 - t_1) + t_1 $$
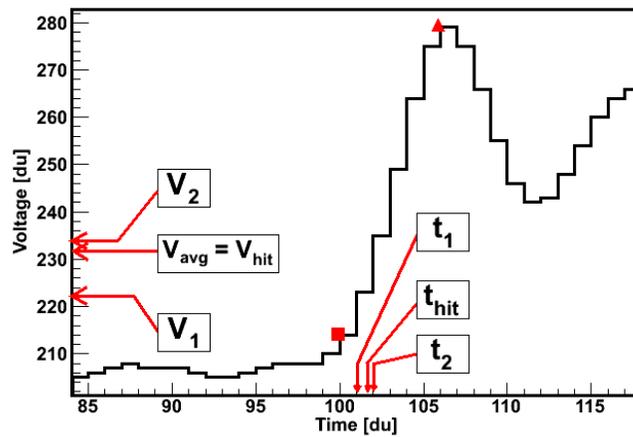
Figure 16 – Variables used for linear interpolation of the average hit time

2.6 Comparison

A comparison between the developed algorithm and the simple scheme is performed using CS_2 data recorded in the CEI apparatus at UW in 2008. The dataset is high in count rates producing many false events. The ringing structures are also large enough to cause the simple algorithm to detect new events when ringing peaks raise high enough. Figure 17 shows the TOF distribution received with the previously employed scheme [7] and Figure 18 shows the TOF distribution received with the developed algorithm. The red circles in Figure 17 indicate false events detected from the second peak in the ringing structure. These ubiquitous artifacts clutter the distribution falsely portraying a high event yield. The number of events acquired in the simple scheme is 1513524 while in the developed scheme it is 1242656. Though there are fewer counts with the developed scheme, those which are detected are more likely real events. The purple diamond in Figure 17 highlights are particular ion. With the simple scheme, the

17

statistics for this ion are much lower than that in Figure 18, indicating that the developed scheme both discourages false events and acquires more true events. To quantify this, coincidence events from the fragmentation channel $CS2+ \rightarrow CS+ + S+$ were extracted from both the simple and developed datasets. The total yield for the developed scheme was double that of the simple scheme. This improvement is expected to increase further for triple coincidence fragmentation channels.
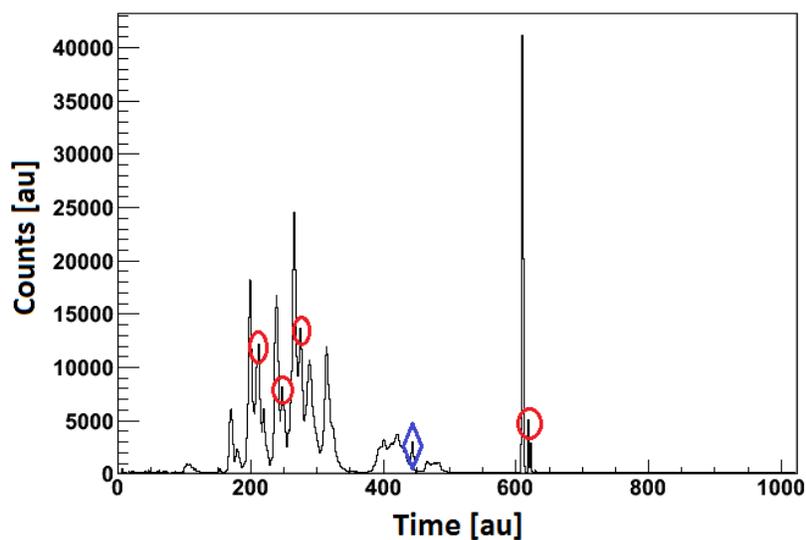
Figure 17 – Time of flight distribution for CS$_2$ using the simple scheme to extract event information Red circles highlight false events. Purple diamond indicates low counts for a specific ion.
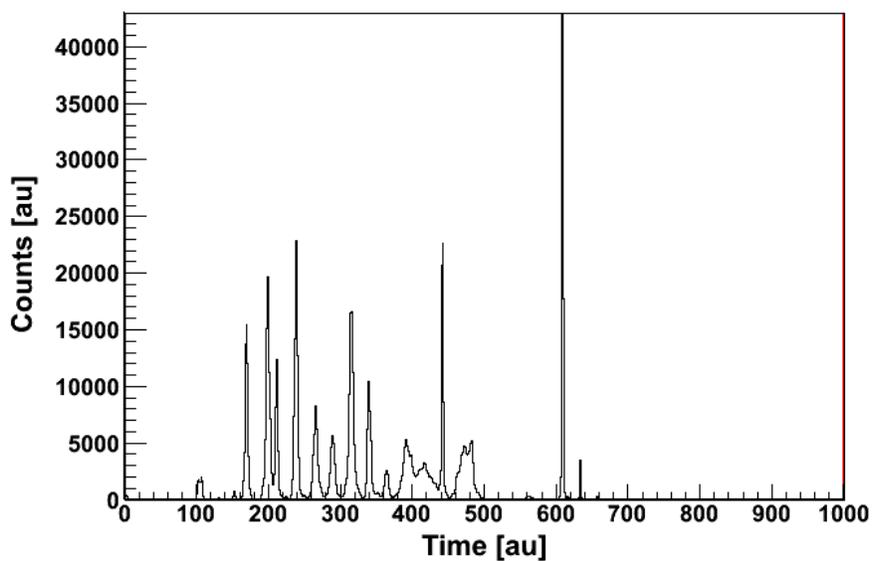
Figure 18 – Time of flight distribution for CS$_2$ using the developed scheme to extract event information

3 PSD Calibration

3.1 Mask Design

To calibrate the PSD, a mask was designed with a series of holes that would produce a scalable image when the detector was illuminated with ions. Such a technique was previously used to calibrate the CEI apparatus [7]. The mask in previous calibrations was a sheet of stainless steel featuring a grid of holes with mostly one size (Figure 19). The installation required dissembling the entire detector such that the mask could sit nearly flush to the first MCP's front surface. Though this method places the mask as close as possible to the detector, and therefore produces the highest resolution images, the installation overhead and the risk involved in damaging the detector or perturbing the extraction field rings suggest a safer method be employed. For these reasons, a new mask was designed to be inserted in front of the detector between the extraction field rings without any manipulation of the PSD mount.
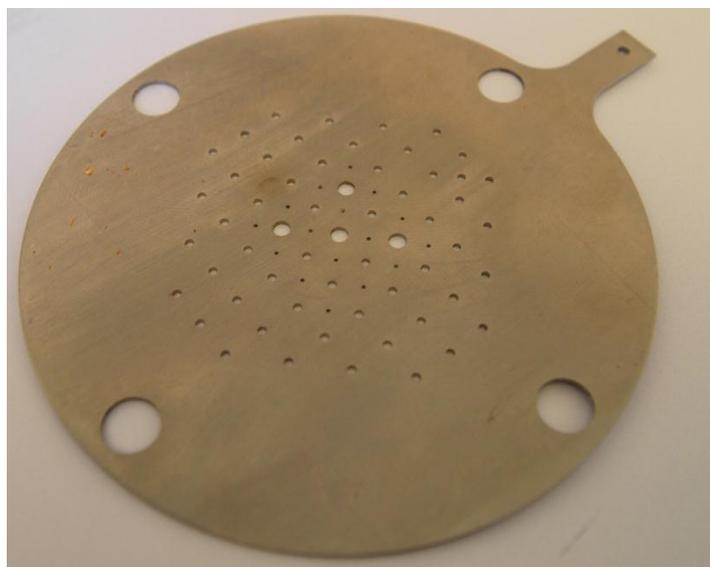
Figure 19 – Photo of previous calibration mask

The region in which the mask is inserted is encumbered by several elements. The stainless steel rings used to establish the extraction field and between which the mask is to be inserted are spaced 3.8mm apart. The four threaded rods supporting the entire apparatus and to which the field rings are fastened are spaced equally around the 76.2mm diameter rings and padded with Teflon rings 10.1mm in diameter. Lastly, the rings located closest to the PSD are spaced with a second set of four Teflon disks 6.8mm in diameter (see top left of Figure 20). As a result of these elements, any opening through which a mask could be inserted is less than the diameter of the imaging area on the PSD. To circumvent this issue, a transformable mask was designed that, when closed, could fit between the extraction field rings and Teflon spacers and later fan open to cover the area of the detector. The system is a three piece set

19

of thin metal slats all connected at a single axle. When collapsed, the fan measures 17.6mm by 95.2mm and can fit between the Teflon spacers separating the rings. Once fanned open, a grid of 1mm diameter holes extending the area of the detector is formed. The holes are spaced 5mm apart to produce approximately 50 resolvable points. For orientation recognition, three larger holes were produced in an 'L' shape. To ensure that the system fans out to the correct angles, alignment holes were placed on the end which remains protruding from the apparatus. Once fully extended, the holes align, and a peg is inserted to maintain the fan shape throughout the calibration experiment.
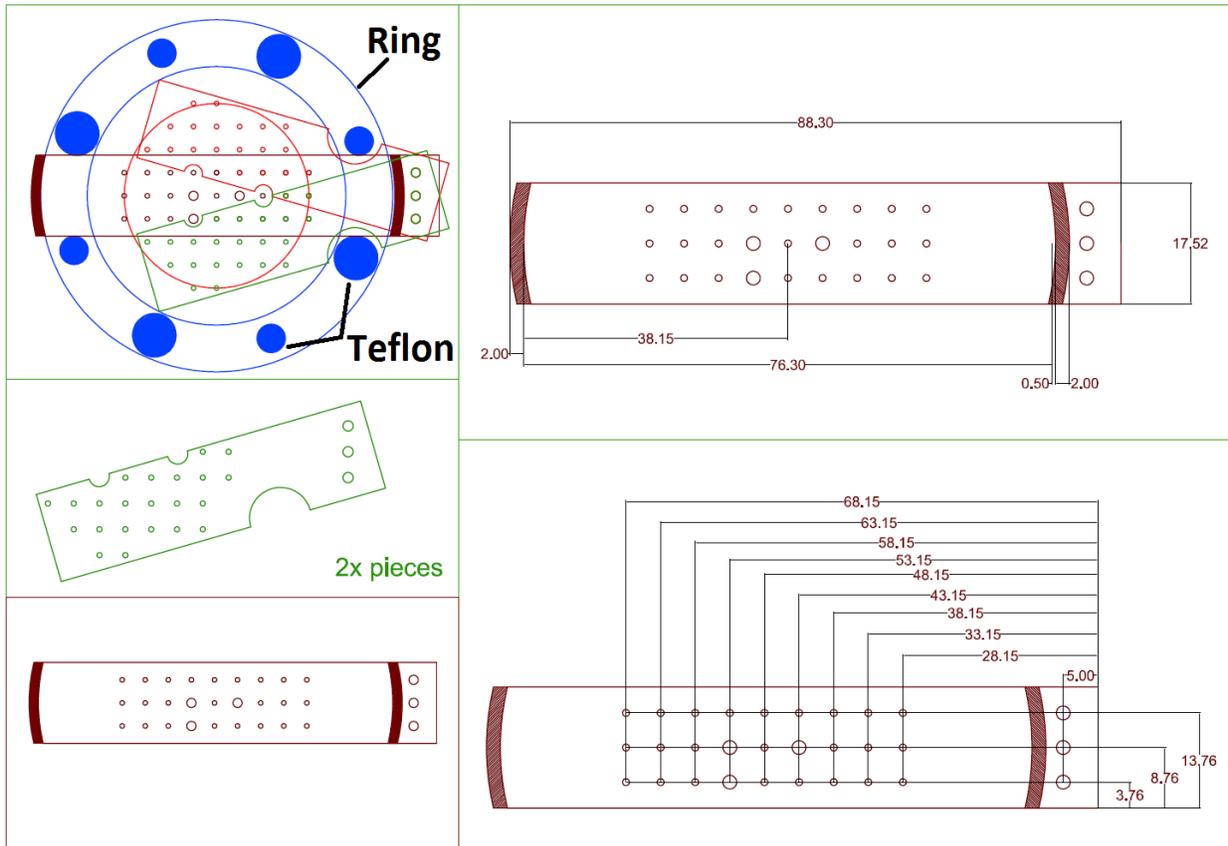
Figure 20 – Calibration mask schematic
Top left shows extraction field ring with Teflon spacers.
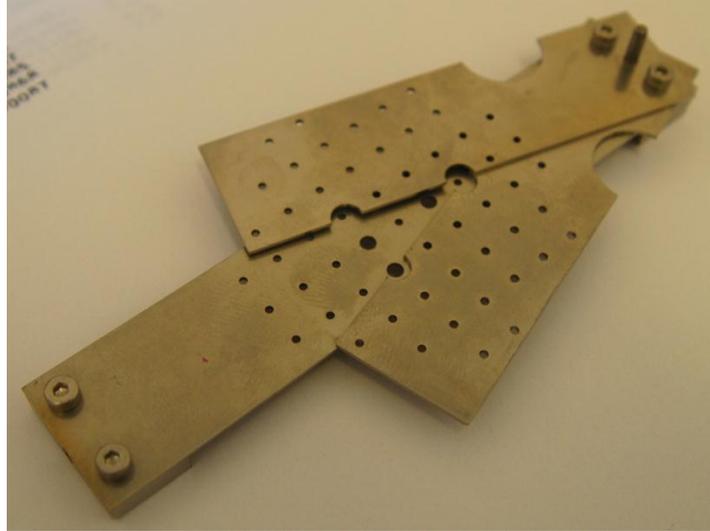
Figure 21 – Photo of calibration mask

As discussed in the following chapter, the position resolution of the detector is on average 0.5mm. The 1mm and 2mm holes should therefore be resolvable and identifiable. The machine precision used to drill the mask was sufficiently high to produce a stable grid once the mask was fully fanned open. Any jitter amongst the three fan pieces is less than the resolution of the PSD and therefore negligible for the calibration.

The size of the holes also puts a limit on the feasibility of imaging the mask grid. With approximately 50 holes 1mm in diameter, 95% of the detector surface is blocked, requiring a high number of ions to perform the calibration. For precise position statistics, more than 1000 events should be acquired for each grid point. When the 50% detector efficiency and 95% ion yield factors are considered for the array of ~50 points, it is estimated that at least 2 million ions should be produced to successfully resolve the mask. The simplest way to produce so many ions in the CEI chamber is with the laser initiated process used in the CEI technique. Usually the ion count rate for this setup is about one ion per laser shot which would produce the necessary amount of data in about a half hour with a 1kHz laser. This is an appropriate length of time to perform a CEI experiment and therefore qualifies the choice of grid hole size.

A final concern with the new mask arrangement is the space between the grid and the first MCP's front surface. Ions produced through standard CEI operation are generated at a central point and reach the extents of the PSD due to their initial velocities parallel to the detector surface. The space between the mask and the detector will therefore skew the grid image if the radial velocity of ions is substantial enough to strike far enough from the projection of the grid hole on the PSD to be resolvable. If it is assumed the velocity of the particles are perpendicular to the TOF direction, the largest distance an ion will strike from a grid hole, r_{Diff}, can be easily calculated (see Figure 22):

21
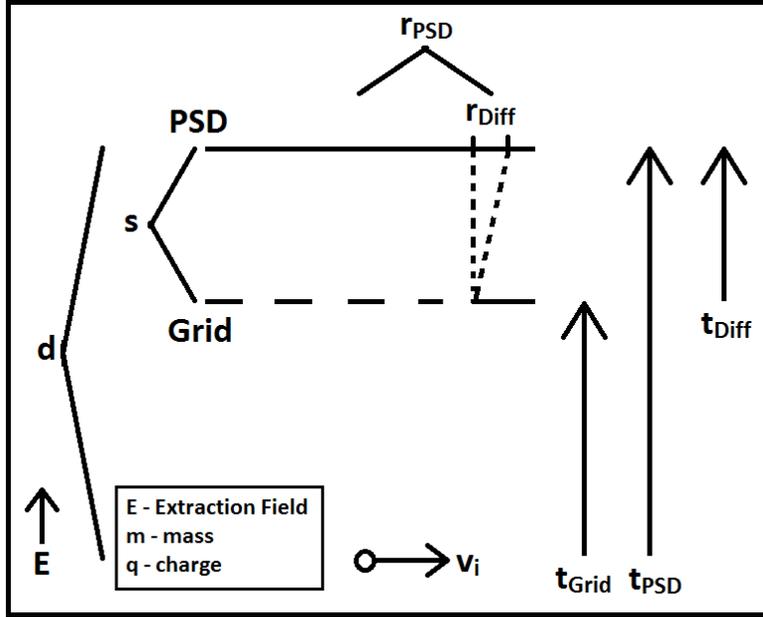
Figure 22 – Schematic for calculating ion drift between the calibration mask and PSD

$$d - s = \frac{1}{2} a\, t^2$$

$$t_{Grid} = \sqrt{\frac{2(d-s)m}{E\,q}}$$

$$t_{PSD} = \sqrt{\frac{2\,d\,m}{E\,q}}$$

$$t_{Diff} = t_{PSD} - t_{Grid}$$

$$r_{Diff} = t_{Diff}\, v_i$$

$$v_i = \frac{r_{PSD}}{t_{PSD}}$$

$$r_{Diff} = r_{PSD}\left(1 - \sqrt{\frac{d-s}{d}}\right) \tag{3-1}$$

The MCP is approximately 20mm in radius [7], the ion flight length 89.3mm, and the mask placed about 5mm from the opening. As a result, the distance the ions land from a grid point is at most 0.5mm. Since this length is nearly resolvable, the calibration will be used with ions striking close to the center. If only the grid points within 10mm of the source point are used, the spread from the ions reduces the below resolution limit.

3.2   Experiment

With the calibration mask in place, position data was taken to create an image of the grid on the PSD. Normal CEI operation was used with low pressure N_2 gas as the target. Since N_2 tends to ionize and subsequently fragment along the laser polarization, the laser was polarized parallel to the detector such that ions would strike the entirety of the PSD.

This polarization of laser pulses generates radially directed N^+ ions and zero velocity N_2^{2+} ions with identical time of flights. The former is detected with 2.5% efficiency due to mask screening while the latter is detected for more efficiently as the ion source is located above a large grid hole. The result of the high detection efficiency of N_2^{2+} is that N^+ and N_2^{2+} events detected at the same time are registered as a single event and skew the overall mask image. The previous calculation of acquisition time is therefore an underestimate as it assumes each ion is detected with the same efficiency. To circumvent this, a lower ion production rate was used (approximately 0.1 ions per laser shot) and several hours of data was recorded.

The raw data from the experiment is shown in Figure 23 with the z scale as a logarithm and the x and y positions rescaled from [0, 1] to [-1, 1]. The grid structure is clearly visible as a collection of 46 islands. The largest island indicates the ion center while the collection of the three largest islands indicates the L-shape drilled into the grid mask. Between each island are visible streaks directed radially towards the ion source position. These streaks result from multiple ion detection described above. If the voltage outputs from the anode are well behaved, the positions of such multiple events are averaged and weighted towards the position with higher charge output. If the sum of the voltage jumps for both ions is not ideal, the waveform analyzing algorithm poorly calculates a position, resulting in the collection of events to the left of the grid (this data was extracted using a simpler form of the above described waveform analyzer which was sensitive to such multi-hit events and has since been improved upon). Lastly, the islands located closest to the edge of the MCP are considered unreliable for two reasons. Firstly it is possible only part of the grid hole is imaged and the other part does not cover the MCP, producing an island image closer to the center than the grid hole. It is also possible that the electron cloud produced by the MCP extends past the edge of the anode, thus produces a charge ratio across the four voltage outputs that measure a position further inward than where the event occurs.
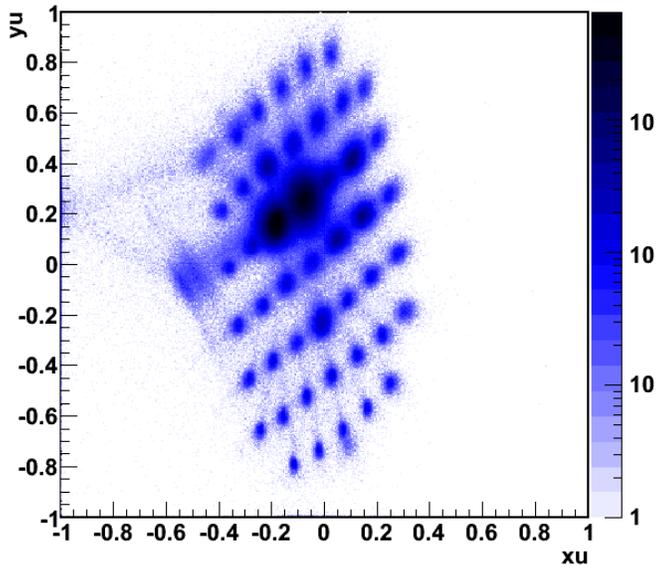
Figure 23 – Raw position data scaled to the ranges [-1,1]

Figure 24 displays the positions for events which produced at least 100du of total charge. By discriminating against low Q_T events, higher precision position of hole centers is yielded. The 100du cutoff leaves approximately 1000 events at most hole centers allowing for resolvable islands. The largest island has disappeared as this high traffic region of the MCP has lost efficiency so that it produces a small electron cloud. The streaks between islands are still visible as they result from a voltage sum, but should not affect the following analysis substantially as the event numbers in comparison with the islands are small.
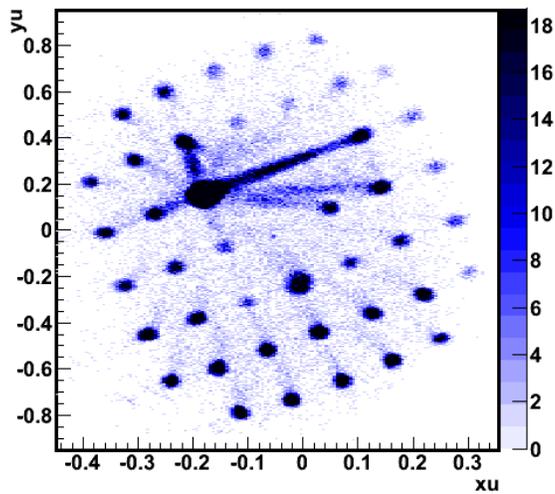
Figure 24 – Raw data for events measured with at least 100du of total charge

3.3   Calibration Algorithm and Results

Several fits are performed to identify the following parameters: grid angle, grid center (Cx,Cy), and grid stretch (Lx, Ly). Due to the radial expansion of hole images described in the previous section, grids of various sizes at various locations are fit to the mask image and their resulting fit parameters are displayed in the table below. For the purposes of CEI, the two grid stretch parameters are the only ones needed. The consistency of the other three parameters confirms the validity of each parameter set.

Holes Fit	Angle	Cx	Cy	Lx	Ly
4	21.2958	-0.0434516	0.0097855	0.02039	0.0482984
16	21.0427	-0.044148	0.00927874	0.0201816	0.048249
33	21.0329	-0.0425099	0.0112194	0.0202591	0.0480768
Average:	21.1 +- 0.8%	-0.0433 +- 2%	0.010 +-11%	0.0203 +- 0.6%	0.0482 +- 0.3%

Table 1 – Calibration results using 4, 16, and 33 imaged holes

To fit the grid image to the theoretical grid hole positions, a fitness parameter is calculated and maximized. Previously a peak finding algorithm is used on the above 2D histogram to identify hole image centers by locating the peaks in histogram bin content. This set of image centers was then fit with the theoretical positions. This technique is not used for two reasons: the uncertainty in such a peak finding algorithm would be amplified in the subsequent fit algorithm; and the events detected across the 1mm diameter holes are statistically flat, resulting in poor resolution of hole image centers. Instead, a fitness parameter is used which one: emphasizes the island nature of the hole images and two: encourages the fitting procedure to give equal favour to all hole images.

The first part is accomplished by counting the number of events within a 0.5mm radius range of a guessed hole center. Since the raw data is of course not calibrated, an approximation of 0.5mm is made by estimating the diameter of an island image. This value is then divided by the number of events between a 0.5mm and 1.0mm radius donut at the same center. The ratio of these values – $hFitValue_i$ (Hole Fit Value for hole *i*) - effectively measures the solidarity of an island image and increases for those positions with high event interiors and low event exteriors. Imaged islands with large streaks are measured to have high exterior events and therefore consistently return a low $hFitValue_i$, no matter how well centered the guessed grid hole may be. Well-defined islands with less statistics should produce similar $hFitValue_i$ for well-defined islands with high statistics.

$$hFitValue_i = \frac{\sum events\ within\ 0.5mm\ radius}{\sum events\ betwen\ 0.5mm\ and\ 1.0mm\ radius}$$

The net fitness of a set of guessed grid hole centers is therefore the sum of $hFitValue_i$ across the set of grid holes. If only the numerator were used – that is, the sum of the events detected within each guess grid hole – the fitting procedure would show preference for islands with high statistics. Though this ratio calculation largely homogenizes such a preference, further encouragement to find approximate locations for each grid hole is performed by calculating the product of $hFitValue_i$ for the set of grid hole positions instead of the sum. The net fit parameter is thus:

$$mFitValue = \prod_{i=1}^{Number\ of\ Holes} hFitValue_i$$

With appropriate estimates for the 5 fit parameters (Cx, Cy, Lx, Ly, and angle) determined by examination, a computer algorithm maximizes *mFitValue* by manipulating the 5 parameters. Table 1 shows the fit results from sets of grid holes of variable number. The resulting fits superimposed with a logarithm of the >100du dataset used are shown below where the transparent circles represent those used in the fitting algorithm, and the grey circles those which were not.
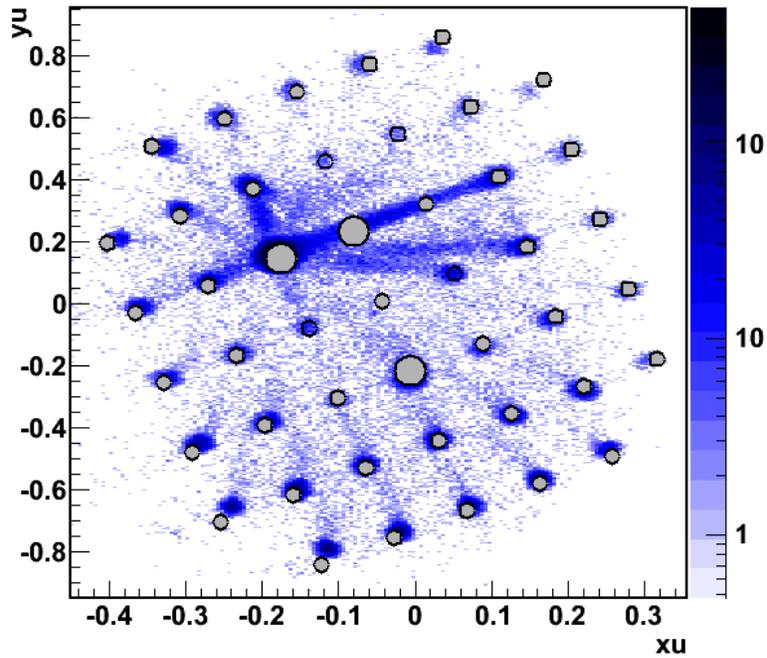
Figure 25 – Raw data superimposed with mask grid fit with 4 holes
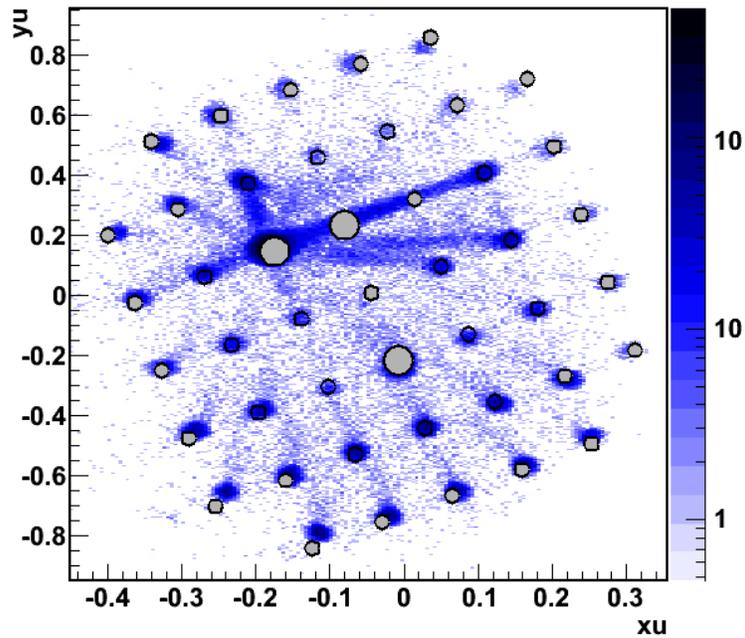
Figure 26 – Raw data superimposed with mask grid fit with 16 holes
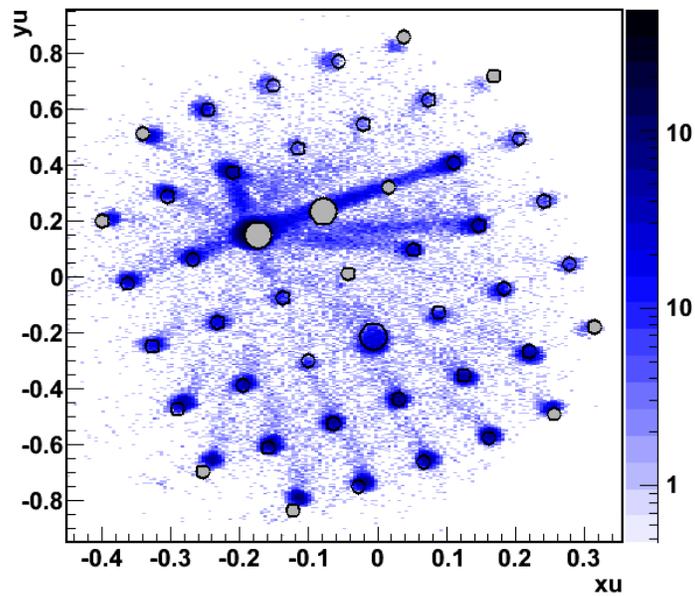
Figure 27 – Raw data superimposed with mask grid fit with 33 holes

A visual confirmation of the parameter sets is performed by comparing the fit mask pattern for low (4), medium (16), and high (33) grid hole sets. Figure 25, Figure 26, and Figure 27 show the results. The fit parameters for the 4 hole set produce a mask image where each grid hole is located above the

image hole and nearly at its center. The same is true for the 16 hole set save for two hole images to the top left largely effected by the central large diameter hole events. The parameters from the 33 hole set produce grid hole positions covering each image. However, the center of each hole is not located above the image hole center. Close examination of a row of holes reveals that the difference in centers is not a statistical uncertainty which would otherwise produce image centers in random directions from mask hole centers. Instead, there appears to be a consistent skew of each row of holes with respect to the image of the holes. This difference is especially noticeable for holes located at the edge of the image. Since this skew only exists for the large number case and not the medium, it is likely that the fringe holes are being imaged incorrectly. Such an error may be from the aforementioned image spread expected to occur from the space between the mask and the PSD. Since the ion source is close to the top, it is expected that the image spread be large at the bottom. However, the vertical distance between holes is less at the bottom than the top. Similarly, the horizontal spacing between image holes is larger to the left than to the right despite being equidistant from the ion source. It is clear then that there is a positive correlation between the image position and image spread. Such skew is likely to come from errors in the PSD voltage output or the waveform algorithm – the latter of which has been updated for more precise position measurements since the time of this experiment.

4   PSD Precision

4.1   Motivation

The goal of the CEI technique is to accurately image the geometry of a single molecule under various conditions by putting the molecule in a state which causes complete fragmentation into its constituent ionized atoms. Assuming a Coulombic potential, the initial geometry of the molecule is encoded in the asymptotic momenta of the ions which are measured via the PSD. Solving for the geometry is a non-trivial problem for which a simplex based algorithm is used. Despite the robustness of the reconstruction technique, the end result is highly sensitive to the precision of the measurements of momenta. To improve upon this problem, larger detectors (TMU), longer time of flight chambers (TMU and ALLS), and supersonic molecular beams (ALLS, UW in development) have been employed in CEI apparatuses. The former two improvements increase the overall precision in momentum by enlarging the space over which position and time measurements are made thereby reducing the relative uncertainty (while maintaining the same absolute uncertainty in position and time). The latter augmentation reduces the initial thermal motion of a molecule from room temperature down to few Kelvin [Dooley], further improving upon the assumption that molecules explode from rest thereby narrowing the discrimination thresholds. Such improvements are vital to the success of the CEI technique. In this section, the factors contributing to the precision of the PSD are explored and it is determined which improvement will best improve precision.

4.2   Contributors to Uncertainty

The three dimensional momentum vectors are calculated from measurements of the position (x,y) and time (t) an ion strikes the detector. Using the coincidence algorithm described later, the mass and charge {m,q} of an event is determined. Using the projection of the ion source position on the PSD (x_0, y_0) – that is, the location at which an ion will arrive if zero momentum is imparted post ionization – and the expected time of flight of a zero momentum ion (t_0), the momentum is calculated as follows:

$$P_x = m\frac{x - x_0}{t} \qquad\qquad \textbf{(4-1)}$$

$$P_y = m\frac{y - y_0}{t} \qquad\qquad \textbf{(4-2)}$$

$$\textbf{(4-3)}$$

$$P_z = \frac{qE}{2}\frac{t_0^2 - t^2}{t} = \frac{qV}{2l}\frac{t_0^2 - t^2}{t}$$

where the electric field E is calculated from the extraction field chamber length and applied voltage ($V /$ l). The uncertainty in momenta clearly has several contributors. The mass m, and charge q, however are well defined and contribute nothing. Uncertainty in the zero momentum ion positions and time (x_0, y_0, t_0) comes from the size of the laser focal volume (ignorable since it is tens of micrometers in size). The PSD measurables (x, y, t) and the extraction field (E) therefore are the main sources of error. The relative contributions on the net uncertainty are explored next.

4.3 Error Propagation

To calculate the error in momentum, the linear error propagation method is used [25]. For any relation:

$$y = f(a, b, c)$$

the uncertainty in y is given by:

$$dy = \sqrt{\left(\frac{\partial y}{\partial a}da\right)^2 + \left(\frac{\partial y}{\partial b}db\right)^2 + \left(\frac{\partial y}{\partial c}dc\right)^2}$$

where 'da' is the absolute uncertainty in 'a'. The uncertainty in the above three momenta equations therefore becomes:

$$\frac{\partial P_x}{\partial x} = \frac{m}{t}$$

$$\frac{\partial P_x}{\partial t} = m\frac{x - x_0}{t^2}$$

$$dP_x = \sqrt{\left(\frac{m}{t}\right)^2 dx^2 + \left(m\frac{x - x_0}{t^2}\right)^2 dt^2}$$

$$dP_x = m\frac{x - x_0}{t}\sqrt{\left(\frac{dx}{x - x_0}\right)^2 + \left(\frac{dt}{t}\right)^2} \tag{4-4}$$

$$dP_y = m\frac{y - y_0}{t}\sqrt{\left(\frac{dy}{y - y_0}\right)^2 + \left(\frac{dt}{t}\right)^2} \tag{4-5}$$

For the TOF direction:

$$\frac{\partial P_z}{\partial V} = \frac{q}{2l} \frac{t_0^2 - t^2}{t}$$

$$\frac{\partial P_z}{\partial l} = \frac{qV}{2l^2} \frac{t_0^2 - t^2}{t}$$

$$\frac{\partial P_z}{\partial t} = -\frac{qV}{2l} \left(1 + \left(\frac{t_0}{t}\right)^2 \right)$$

$$dP_z = \sqrt{\left(\frac{q}{2l} \frac{t_0^2 - t^2}{t}\right)^2 dV^2 + \left(\frac{qV}{2l^2} \frac{t_0^2 - t^2}{t}\right)^2 dl^2 + \left(-\frac{qV}{2l} \left(1 + \left(\frac{t_0}{t}\right)^2\right)\right)^2 dt^2}$$

$$dP_z = \frac{qV}{2l} \frac{t_0^2 - t^2}{t} \sqrt{\left(\frac{dV}{V}\right)^2 + \left(\frac{dl}{l}\right)^2 + \left(\frac{t_0}{t_0^2 - t^2} dt\right)^2 + \left(\frac{t}{t_0^2 - t^2} dt\right)^2} \qquad \text{(4-6)}$$

The uncertainty in time, dt, is given by half the digitizing unit of the analog to digital converters (ADCs). The ADCs (Gage CS82G Master/Slave) typically operate at 500MHz to 1GHz acquisition rates yielding uncertainties of between 1ns and 0.5ns respectively. The uncertainty in extraction voltage is less than 0.1V from the power supply control (Fluke 415B). Though extraction field length *l* was measured with 0.1mm precision, this length remains constant throughout the experiment and is implicitly calculated when the TOF mass spectrometer is calibrated. Therefore, this term contributes nothing to the overall uncertainty. The uncertainty in position, *dx* and *dy*, are functions of the total charge deposited on the anode and the total voltage digitized by the ADCs. These are now calculated. For a single event:

$$X = \frac{Q_1 + Q_2}{Q_1 + Q_2 + Q_3 + Q_4} = \frac{Q_1 + Q_2}{Q_T}$$

$$Y = \frac{Q_1 + Q_3}{Q_1 + Q_2 + Q_3 + Q_4} = \frac{Q_1 + Q_3}{Q_T}$$

The partial derivatives are

$$\frac{\partial X}{\partial Q_1} = \frac{1}{Q_1} - \frac{Q_1 + Q_2}{Q_T^2} = \frac{1 - X}{Q_T}$$

$$\frac{\partial X}{\partial Q_2} = \frac{1}{Q_1} - \frac{Q_1 + Q_2}{Q_T^2} = \frac{1-X}{Q_T}$$

$$\frac{\partial X}{\partial Q_3} = \frac{Q_1 + Q_2}{Q_T^2} = \frac{X}{Q_T}$$

$$\frac{\partial X}{\partial Q_4} = \frac{Q_1 + Q_2}{Q_T^2} = \frac{X}{Q_T}$$

The full uncertainty is therefore

$$dX = \sqrt{\left(\frac{\partial X}{\partial Q_1}dQ_1\right)^2 + \left(\frac{\partial X}{\partial Q_2}dQ_2\right)^2 + \left(\frac{\partial X}{\partial Q_3}dQ_3\right)^2 + \left(\frac{\partial X}{\partial Q_4}dQ_4\right)^2}$$

$$dX = \frac{1}{Q_T}\sqrt{(1-X)^2 dQ_1{}^2 + (1-X)^2 dQ_2{}^2 + X^2 dQ_3{}^2 + X^2 dQ_4{}^2}$$

The uncertainty in a measurement of charge is given by half the division of the digitizing unit *du* so $dQ_1 = dQ_2 = dQ_3 = dQ_4 = 0.5$, resulting in:

$$dX = \frac{1}{Q_T}\sqrt{X^2 - X + 0.5}$$

Likewise, the uncertainty in the Y-axis is given by:

$$dY = \frac{1}{Q_T}\sqrt{Y^2 - Y + 0.5}$$

As expected, the uncertainty in position decreases as the amount of digitized charge – Q_T (step size) – increases. Depending on where a measurement is made between the range [0,1], the uncertainty will vary between:

$$\frac{0.5}{Q_T} < \{dX, dY\} < \frac{0.7}{Q_T}$$

where the lower bound results from measurements made near the center (X = 0.5) and the upper bound at the edges (X = 0 or X = 1). The range is quite narrow so an average uncertainty is appropriate for continuing this analysis: $dX = dY = 0.6 / Q_T$. The ADCs are capable of 8-bit digitization resulting in values of Q_i from 0 to 255 and Q_T from 0 to 1020 depending on the charge deposition per event and the voltage range for which the ADCs are set. The ringing peak, multiple hits, voltage offset, and maximum charge deposition largely reduce the upper bound of the Q_i range such that 50du on average is measured for Q_T with typical operating parameters. The above analysis shows there is a 1% uncertainty across the full range of X and Y. The PSD calibration performed in the previous chapter reveals that approximately half of the X range is covered by the PSD. Once calibrated, a measurement in X will have an uncertainty approximately 2.4 times larger than a measurement in Y. The calibrated uncertainties are $dX = 60 / Q_T$ [mm] and $dY = 25 / Q_T$ [mm].

Returning to the uncertainty in momentum in the X direction (Equation (4-4)), the second term contributing to the overall uncertainty can be bounded by noting that the earliest arriving ion, H+, has a TOF of approximately 260ns. Using dt = 1ns, an upper bound of 1.5e-5 is placed on this term. A lower bound for the first term can similarly be calculated by using the largest value of (x-x0) – which is half the width of the PSD (20mm) since the ion source center, x_0, is located in the middle of the PSD. Using dX = 60 / Q_T, this first term is approximately 9 / Q_T^2. For these terms to be comparable, Q_T would have to be 775du – more than 15 times the average digitizing value. At Q_T = 50, the first term is 3.6e-3 (240 times larger than the second), so it is appropriate to ignore the second term in equations (4-4) and (4-5).

The voltage term in the P_z uncertainty has been calculated to be $(0.1 / 3000)^2$ = 1e-9; while the length term has been dropped. For the last two terms, approximations are made by considering typical TOF measurements. Under normal operating conditions, N^+ has a TOF of approximately 1µs with a 15ns spread in time when fragmenting from a low charge state channel. Since t_0 = 1000ns and t = 1015ns, the last two terms are approximately the same and, furthermore, sum to 2e-3 when the 500MHz is set for the digitizing rate. For 1GHz acquisition, the sum yields one quarter the above: 5e-4. Clearly these terms dominate the uncertainty of P_z, allowing one to simplify the calculation to two times the last term. The approximate uncertainties for momentum in each dimension are as follows:

$$dP_x \approx \frac{m}{t} \frac{0.06}{Q_T}$$ (4-7)

$$dP_y \approx \frac{m}{t} \frac{0.025}{Q_T}$$ (4-8)

$$dP_z \approx 0.705 qE \, dt$$ (4-9)

where the extraction field has replaced V/l in equation (4-9). For a typical ion (N^+) with an average Q_T of 40du digitizing with 2ns resolution and an extraction field generated from 3000V across a 113.2mm chamber, the uncertainties of momentum in x, y, and z are 0.34e-22 kgm/s, 0.15e-22 kgm/s, and 0.03e-22 kgm/s respectively. For comparison, low charge state fragmentation produces ions with momenta on the order of 2e-22 kgm/s yielding relative errors of up to 17%. It is clear from the equations (4-7) and (4-8) the importance of operating with a high digitization precision in order to acquire large values of Q_T. In terms of ADC operation, this can be achieved by either optimizing the digitizing voltage range to match the PSD voltage output, or increasing the digital resolution (from 8bit to 16bit in the case of UW ADCs). Similarly, equation (4-9) suggests operating the ADCs at the highest acquisition rate. The same improvement can be obtained by reducing the extraction field strength via the applied voltage. Doing so, however, elongates the flight time of the ions causing them to strike further across the PSD until the limits of the detector are reached. Before this threshold, uncertainties in P_x and P_y will reduce. Increasing the flight time requires a proportional increase in the size of the detector which in turn increases the calibration factors used to calculate equations (4-7) and (4-8). Since the TOF and flight distance along the x and y dimensions are linearly proportional, neither dP_x nor dP_y will reduce as a result of increased detector size.

The values of uncertainty calculated above result from approximations of the PSD and ADC operation parameters. In practice, triggering delay, voltage fluctuations in anode output, and the waveform analyzing algorithm contribute uncertainties in measurement of event position and time. The extent to which the above calculations are accurate is now assessed by examining the momentum measurements from a simple N_2 experiment using the CEI apparatus.

4.4 Comparison with N_2 experiment

In this experiment, the laser initiated explosion of N_2 into multiply charged ions was performed using 300uJ sub 100fs laser pulses. The fragments were detected in coincidence using The University of Waterloo's CEI apparatus described above. The ADCs operated at 250MHz acquisition rate with voltage measurements set such that the average Q_T for the entire dataset was 30du. The laser pulses were polarized in the TOF direction, producing ions with momenta largely in the Z direction. The recorded coincident momentum measurements were analyzed using the coincidence algorithm described in the following chapter to extract fragmentation coincidences into the (2,1) channel.

To examine the precision of the PSD, the sum of the momenta of the N^{2+} and N^+ ions in the (2,1) channel are plotted for all three dimensions in Figure 28. The sharp edges result from momentum discrimination allowing the coincidence algorithm to select the true coincidences from false coincidences. The half width at half maximum for each distribution is measured using a Gaussian fit to retrieve the values (0.56, 0.35, 0.28) [10^{-22} kg m/s] for momentum in the x, y, and z directions respectively. To compare with the theory above, the uncertainty in each direction is calculated for N^{2+} and N^+ using equations (4-7) through (4-9) to be (0.35, 0.15, 0.12) [10^{-22} kg m/s] and (0.27,0.11,0.06) [10^{-22} kg m/s] respectively. The uncertainty in the sum of the momenta is given by the root of the sum of the squares of the uncertainty for each ion resulting in a net uncertainty in the momentum sum of (0.44, 0.19, 0.13) [10^{-22} kg m/s] for (ΣP_x, ΣP_y, ΣP_z). These theoretical values underestimate the measured uncertainty but are consistent with the relative uncertainty between each dimension – that momentum in the TOF direction is more highly resolvable than the Y direction which is more highly resolvable than the X direction.
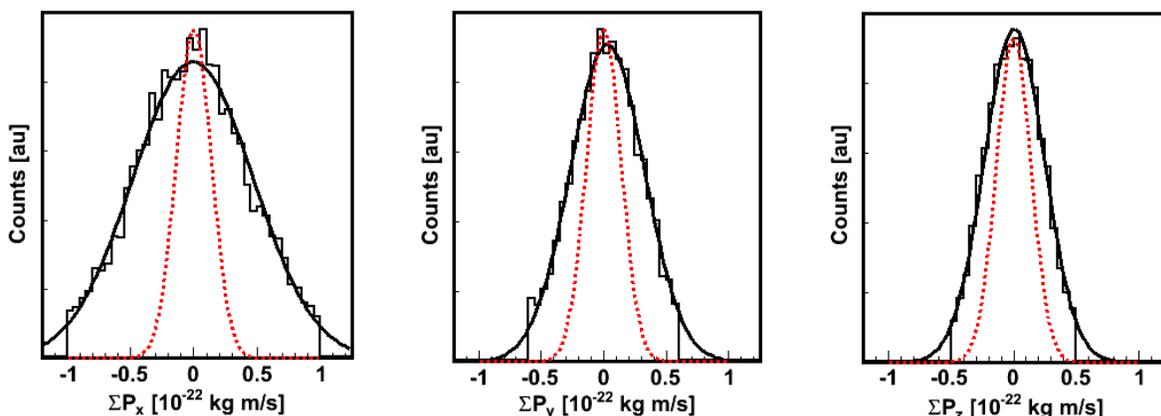
Figure 28 – Momentum sum of both ions in the (2,1) channel
Black lines are Gaussian fits to the data. Red dashed lines show Maxwell Boltzman thermal distribution for T=298K N_2 molecule.

Accurate measurements of momenta in the CEI technique rely on the assumption that target molecules are at rest prior to fragmentation. The limit to which this is true places a lower bound on the precision with which momentum measurements can be made. To quantify the capacity of the CEI apparatus at the University of Waterloo, the above momentum distributions are superimposed with Maxwell Boltzmann distributions of room temperature N_2 using the probability distribution:

$$f(p_z) = \sqrt{\frac{m}{2\pi kT}} \exp\left[-\frac{P_z^2}{2kTm}\right]$$

where P_z is the momentum in the Z direction, m the mass of N_2, k the Boltzmann constant, and T the temperature. The distribution of momentum in the Z direction is closest to the thermal limit (HWHM of 0.16×10^{-22} kgm/s) suggesting an increase in timing resolution (from 250MHz acquisition to 1GHz) may require cooling the target molecules to allow for higher momentum precision. Alternatively, lowering the extraction field voltage and increasing the detector size to compensate for the larger lateral flight distance could also bring the momentum resolution to the thermal limit. By doubling the acquisition rate or halving the extraction field, equation (4-9) predicts the measured HWHM in ΣP_z of 0.28×10^{-22} kgm/s would breach the thermal limit. The distributions in the X and Y directions suggest that larger charge output, Q_T, is the first step to improving momentum resolution. To confirm this hypothesis, the same distributions are made selecting coincident (2,1) ions with Q_T larger than 30du. Nearly 80% of the coincident events are removed and the resulting distributions are shown in Figure 29. The Gaussian fits measure HWHM values of (0.40, 0.26, 0.27) [10^{-22} kg m/s], indicating ~25% improvement in resolution in both the X and Y directions. A slight improvement in the TOF direction is apparent and increases if the Q_T limit also increases. This suggests that the voltage step rise time is a factor limiting the precision of P_z since the hit time can be more highly resolved as the step Q_T increases. These results validate the calculations of uncertainty performed above, and suggest an increase in charge digitization and time digitization rates could push the CEI apparatus to require a cold molecule source.

For comparison, plotted as well in Figure 29 are results recorded at ALLS using a super sonic molecular beam of OCS as the target molecule. The combination of few Kelvin source molecules and a high resolution PSD results in momentum resolution below the room temperature thermal limit. The 0.25ns uncertainty in time measurements yield the finest resolution measurements while lack of cooling in the beam direction yield courser measurements of momentum along the Y axis.
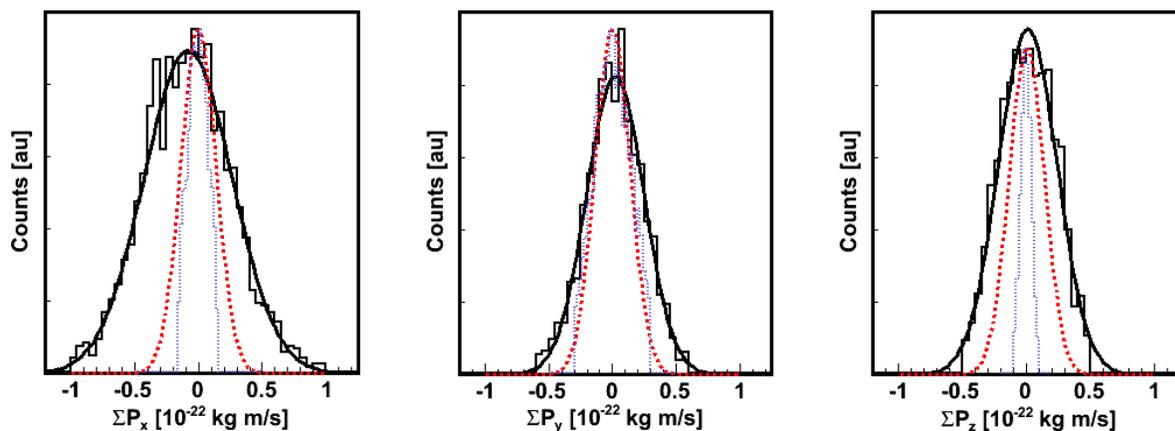
Figure 29 – Momentum sum of both ions in the (2,1) channel with Q_T > 30du
Black lines are Gaussian fits to the data. Red dashed lines show Maxwell Boltzman thermal distribution for T=298K N_2 molecule. Blue dotted lines are the momentum sum of (OC^{2+}, S^+) recorded at ALLS using a supersonic molecular beam.

5   Coincidence Algorithm

This section is adapted from the article "A coincidence detection algorithm for improving detection rates in Coulomb Explosion Imaging" accepted for publication in Nuclear Instruments and Methods in Physics Research July 2011. Two addendums have been added to the appendix regarding issues with degeneracy and the benefits of dimensional momentum discrimination pertinent to the success of coincidence momentum imaging.

5.1   Motivation

The methodology of coincidence analysis is largely undocumented in this field. However, it is crucial to the successful reconstruction of molecular geometry and it is essential for the development of Coulomb imaging as a mature and reliable technique that the efficiency of coincidence detection be maximized. This chapter will concentrate on the method of laser induced multiple ionization which has the big advantage that it can promptly initiate the ionization of a controllable number of molecules per laser shot by adjusting the target molecule density. This allows the detection efficiency restrictions (50% per ion) to be countered. However, it has the drawback that false coincidences may result - that is, ions from more than one molecule may be detected and interpreted incorrectly. The simplest method to ensure that only genuine coincidences are recorded is by having no more than one molecule in the laser focus per laser shot. By way of Poisson statistics, this criterion can be met within 5% by having on average $\lambda = 0.35$ molecules in the laser focus per shot. This value has been stretched but maintained below 1 [15,26]. Such count rates have been adopted by the majority of experimenters allowing them to keep the ratio of true to false coincidences overwhelmingly high. The cost for this guarantee is in the final detection rate of true coincidences. In the case of triatomic molecules and 50% ion detection efficiency (typical of MCP detectors), the successful coincidence detection rate is approximately 0.05 molecules per laser shot. In order to investigate polyatomic molecules using detectors with such efficiencies, overall coincidence detection efficiency decreases exponentially [27] and so high count rates become a necessity.

An important aspect of measuring genuine coincidences is the identification of ionic species through the TOF technique. This has been accomplished by considering molecules with fragments having a combination of distinguishable charge to mass ratios and distinct orientation dependent TOF such as CS_2, and CO_2, and by avoiding ions with degenerate charge to mass ratios (OCS: O^{n+} & S^{2n+}) or several identical atoms (N_2O). In these latter cases, sophisticated analysis techniques are required for coincidence analysis even when low count rates are employed.

In this chapter, an algorithm is developed to handle high count rates and charge to mass ratio degeneracy and it is compared with a simple and intuitive algorithm used in previous studies [12,18].

The effectiveness of these algorithms is simulated for diatomic, triatomic, and polyatomic (6 atom) molecules.

5.2 General Analysis Technique

The CEI apparatus is, at its most fundamental, a time of flight mass spectrometer combined with a position sensitive detector (PSD). A molecule in the laser focus, stripped of many electrons, can fragment due to Coulombic repulsion into atomic ions with charge to mass ratio (q/m). By measuring the location of impact on the PSD (x, y) and the time of impact (t), the asymptotic momentum vectors can be calculated using the equations

$$P_x = \frac{m\,(x - x_0)}{t}$$

$$P_y = \frac{m\,(y - y_0)}{t}$$

$$P_z = \frac{qE}{2}\left(\frac{t_0^2 - t^2}{t}\right)$$

where x_0, y_0, and t_0 are the zero momentum positions and time of flight. E is the electric field (constant for the experiment), while q and m are the charge and mass of the impact ion. For TOF systems with a constant accelerating field, ions can be identified by their mass to charge ratio using the relationship

$$t_0 = \sqrt{\frac{2d}{E}\left(\frac{m}{q}\right)}$$

where d is the ion flight length defined by the laser focus location. Unlike traditional mass spectroscopy, ions arrive earlier and later (forwards and backwards) with respect to t_0 due to the explosion momentum imparted in the TOF direction. As a result, wide TOF windows are needed to identify detection events as specific ions. Frequently – and especially in the case of degenerate charge to mass ratio species – these windows overlap, causing uncertainty in identifying events with the TOF technique.

Though it is not possible to say with certainty that a single event has been correctly identified as a specific ion, it is possible to test a collection of events detected in a single laser shot by considering the net momentum. Assuming that the parent ion is at rest to begin with, the momentum sum of the fragment ions should be zero (to within a thermal limit). In most systems, the initial momentum is anisotropically Maxwellian, so the complete condition is the union of the coordinate sum:

$$\bigcap_j^{x,y,z}\left\{\left|\sum_i^{Ions}\overrightarrow{P_j^i}\right| < PT_j\right\} \qquad\qquad \textbf{(5-1)}$$

where PT_j is the momentum threshold for the j'th axis. In experiment, PT_j is limited by either the target molecule thermal distribution (~ 3e-23 kg m/s for room temperature systems down to 1e-27 kg m/s for those employing super-sonic molecular beams) or detector resolution (~ 1e-23 kg m/s).If a subset of the detected events passes this test, it can be considered that the ions were correctly identified as a coincidence event - in other words, they were produced from the same molecule.

5.3   Coincidence Technique

For situations where the number of detected events rarely exceeds the number of expected ions from fragmentation (i.e. λ = 0.5), a simple algorithm can be used. CO_2, SO_2, and CS_2 have all been studied using an approximation of the algorithm that follows [12,15,18] so, we will consider only the triatomic case. In CEI, data is extracted and collected in terms of fragmentation channels. For example, OCS -> O^{2+} + C^+ + S^{4+} is characterized as the (2,1,4) channel. Consequently, a channel is first defined:

$$\text{MQTW} = \{\{\text{mass}, \text{charge}, \text{low TOF limit}, \text{high TOF limit}\}_i\} \qquad 1 \leq i \leq Number\ of\ ions$$

The first two items in this set define the i'th ion via mass and charge (MQ) while the last two items define the TOF window (TW) used to identify a detected event as the i'th ion. In addition to these standard TOFMS definitions is the set $\{\{x_0, y_0, t_0\}_i\}$ used to calculate the i'th ion momentum.

The following algorithm assumes that three or more events were detected. Each event is identified through the index *EventNum* and processing begins with the event which happens last (i.e. the arrival of the final ion at the detector). *IonNum* is an index that begins at 1 and tracks which ion is being identified (and runs up to 3 in the case of the triatomic molecule). An event is successfully identified as an ion if the event TOF falls within the ion TOF window defined in *MQTW*.

1. If event *EventNum* is in the TOF range of ion number *IonNum*, increase *IonNum*. Increase *EventNum*.
2. Loop step 2 until *IonNum* > 3 or *EventNum* reaches the number of events collected in the laser shot. In the former case, go to step 3, in the latter case, the apparatus failed to detect ions in coincidence.
3. With the list of successful ion identities, perform momentum discrimination by applying condition (1). If this is successful, then the 3 events have been correctly identified as specific ions coming from the molecule under investigation.

The simplicity of this approach is clear and it can be very successful provided the ions arrive in the order that the TOF windows are defined. Systems such as CO_2 and SO_2 are therefore ideal targets.

In response to the limitations of the simple treatment described above, an algorithm was developed which can identify combinations of ions whose identity cannot be simply determined by their TOF. With the previous definition of terms, a list of potential ion identities is generated by iteratively processing each event against each MQTW entry. The collection of these lists forms a potential ion matrix (PIM). As an example, consider the data in the first two columns of Table 2. Testing this data against the ion TOF windows defined for O^+, C^+, and S^+ (see table caption), the PIM (last three columns of Table 2) is generated. After the creation of the PIM, the following actions are performed.

1. The first element in the PIM is selected and a potential match is searched for in the following event entries.
2. If an ion in the PIM is found that is different from the first, a third ion is searched for from the beginning of the following row. Once a third unique ion is found, the collection of PIM indices – and thus ion identities – is recorded.
3. The search for the third ion continues with the following row. Once the search for the third ion has exhausted the PIM, the algorithm returns to search for a second suitable ion.
4. Once the search for the second ion has exhausted the PIM, the algorithm returns to step 1 and selects the next element in the PIM as the first ion.

Though this example is limited to triatomic ions, a general recursive function exists. Finally, the list of potential molecules is tested against inequality (5-1). If two correct molecules have been generated, the entire laser shot is ignored.

The obvious benefit of this algorithm is that the entire space of possibilities is explored. If the ions from from one molecule are detected – and any reasonable number of false coincidences occurs – the fragmentation channel will be correctly identified. Furthermore, the order of event detection and time of flight ranges do not affect the identification of the molecule. Unexpected exotic channels can therefore be detected, maximizing the impact of the data. Though the cost is clearly computational overhead, an implementation of this algorithm in the C++ based ROOT framework (0 -

Coincidence Analysis) has been shown to work at real-time acquisition speeds on a conventional PC.

Event #	TOF [us]	Ion 1	Ion 2	Ion 3
1	20	**C**	-	-
2	40	**C**	**O**	-
3	60	**C**	**O**	-
4	80	**O**	-	-
5	100	-	-	-
6	120	-	-	-
7	140	**S**	-	-
8	160	**S**	-	-

Table 2 – Example of single laser shot data and PIM for OCS (1,1,1)
Potential Ion Matrix (PIM) are the last three columns. Potential ion identities are generated by comparing the TOFs to the following windows: C+[10 - 70], O+[30 – 90], S+[130 – 170] (MQTW).

5.4 Efficiency Comparison

To compare the efficiency of the advanced algorithm against the simple treatment described above, coincidence analysis is performed for various fragmentation channels of the triatomic molecule OCS. A CEI experiment was performed using the systems described in reference [8]. The ALLS system employs sub 10fs, ~100uJ, 800nm pulses in a CEI apparatus including a supersonic beam source. The anisotropic thermal distributions produced from this source are handled using thresholds of 1.5e-23kgm/s, 3e-23 kgm/s, 1e-23 kgm/s in the x, y, and TOF directions respectively for equation (5-1). The supersonic beam is directed in the y direction resulting in larger momentum uncertainty while the TOF axis has the least momentum uncertainty due to the detector's high time resolution.

To perform a fair comparison between the two algorithms, it is critical that the TOF windows be large enough to accept the whole spectrum of events but small enough that the simple algorithm is not swamped with false events. Figure 30 shows typical event yield as a function of window width about the zero TOF for both algorithms. The advanced algorithm quickly increases to a maximum as expected while the simpler treatment reaches a maximum earlier and decreases thereafter. Choosing the maximum yield in the simple scheme discriminates against high energy events at the edge of the TOF window and therefore causes inhomogeneous distributions of physical measurements like energy. Since the advanced algorithm effectively detects all coincidence events, the edge of the TOF window is chosen such that only 1% of all events are excluded by the simple method. The TOF windows for each ion in each channel of interest are shown in Figure 31 above a typical OCS TOF spectrum.
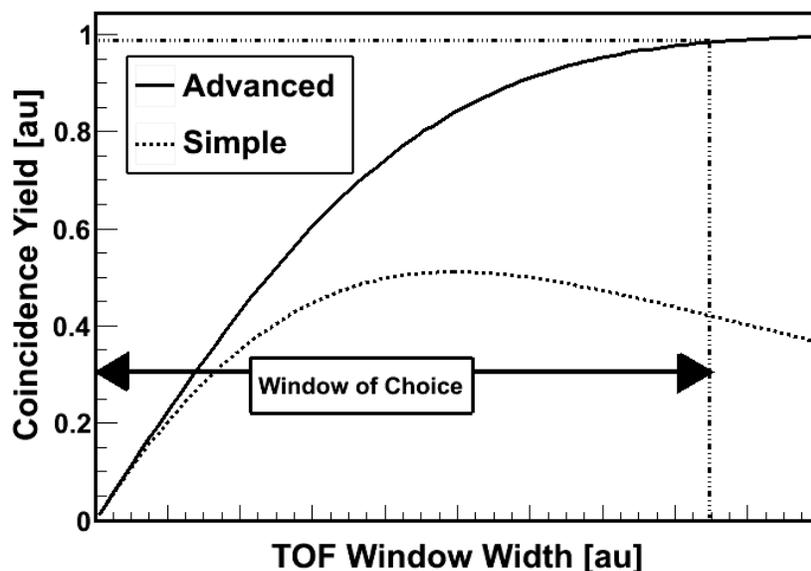
Figure 30 – Coincidence yield for both algorithms with different TOF window widths
The statistical TOF distribution is considered to be Gaussian about the zero momentum TOF t_0.
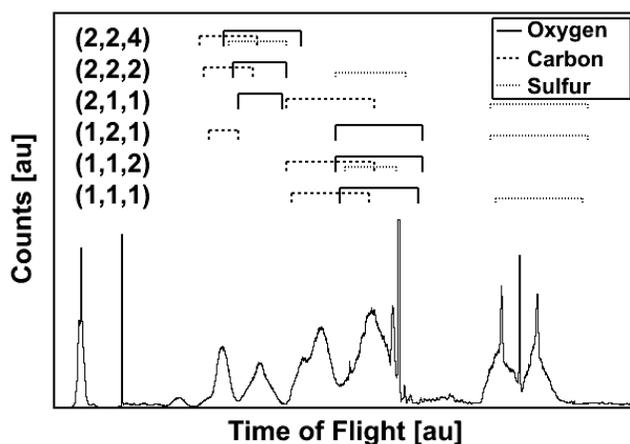
Figure 31 – OCS TOF and fragment ion TOF Windows for a selection of channels

The full windows of Figure 31 encompass ions arriving earlier and later (forwards and backwards) with respect to the zero momentum TOF t_0. This range is larger than necessary as the molecular orientation with respect to the TOF axis causes high charge ion fragments to arrive far from t_0. Furthermore, the tendency of OCS to align along the TOF axis causes either the oxygen and sulfur ions to propel forwards and backwards or vice versa. By segregating these arrangements, the TOF windows can be shrunk without discriminating against high energy events to increase the yield of the simple treatment. To do so, the TOF windows of sulfur are split at the t_0 mark to bifurcate the channel into (x,x,f) and (x,x,b) for sulfur forwards and backwards. This segregation is in fact necessary for the simple treatment when the TOF windows of fragment ions exactly overlap (O^+ and S^{2+} in the (1,1,2), and (2,2,4)

channels) as the order of arrival for fragment ions is changed. (x,x,f+b) denotes the sum yield of the bifurcated channel - that is, (x,x,f) + (x,x,b).

Since the oxygen ion (and somewhat the carbon ion) in the (x,x,f) channel is expected to propel backwards, the full window for these fragment ions is also unnecessarily large. (b,b,f) represents the channel where the TOF window of the oxygen and carbon ions have been shrunk to select just the backwards range (by using the same 1% condition as for the full window selection). The opposite is done for (f,f,b) and their sum yield is given by (f+b, f+b, f+b).

This decomposition of channels into forwards and backwards elements is effective in the case where ions are well separated in time. However, when multiple ions overlap (as is the case for the (1,1,2) channel), (x,x,b) and (x,x,f) sub channels need to be further segregated to accommodate carbon and oxygen switching the order of arrival in the TOF. Such efforts were not performed for this comparison.

5.5   Results

The results for OCS are shown in Figure 32. When considering the full explosion channel, the advanced algorithm far exceeds the simple treatment. This is expected when the event count rates are high and the TOF windows large – as in all the cases listed. Bifurcating the channels and adjusting the TOF windows for fragment ions ((x,x,f+b) and (f+b, f+b, f+b)) improves the yield for the simple algorithm up to ~50% that of the advanced scheme. As expected, these changes in window widths have little effect on the advanced algorithm yield. This highlights the reduction in analytic overhead and a priori knowledge of fragmentation energy required to select proper TOF windows.
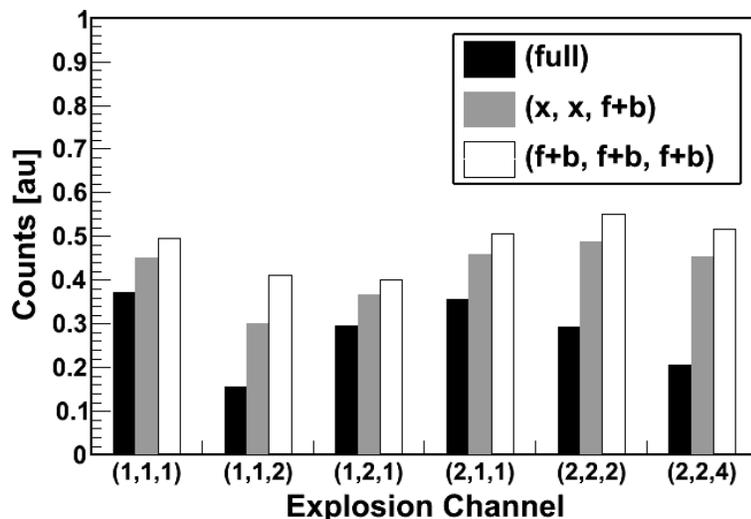
Figure 32 – The number of true coincidence events retrieved using the simple algorithm normalized to the number of events retrieved using the advanced algorithm for selected channels (Full) (black) result from using full window widths for the three fragment ions. (x,x,f+b) (grey) results from the sum of (x,x,f) and (x,x,b). (f+b, f+b, f+b) (white) result from the sum of (f,f,b) and (b,b,f). See text.

In order to demonstrate the overall improvement from implementing the developed algorithm in a CEI experiment, a physical metric common in the field was measured. The kinetic energy release (KER) is a sum of the ion kinetic energies produced during fragmentation. Figure 33 displays the distributions acquired for these measurements of the (2,2,4) channel using both the simple and developed algorithms. Not only is the expected improvement in statistical uncertainty visible, but the simple algorithm systematically underestimates signal from high energy events (141eV to 155eV).
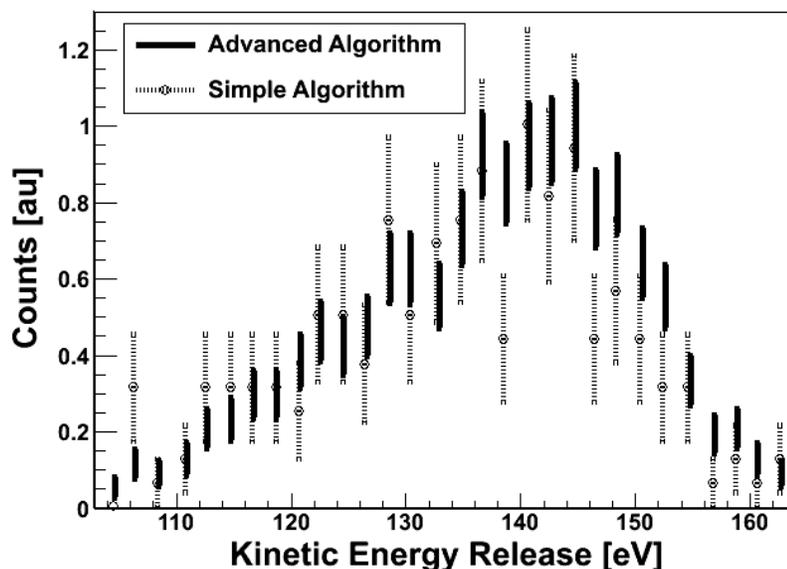
Figure 33 – Total kinetic energy release for OCS channel (2,2,4) generated with sub 10fs pulses
The distributions resulting from the advanced and simple algorithms are shown as solid and dashed lines respectively.

The experimental results represent a snapshot of the effectiveness of the new algorithm for a specific count rate, but its effectiveness should be assessed over a range of conditions and for a range of molecular sizes. In order to do this the apparatus has been simulated using a Monte Carlo method to determine the efficiency of the simple and improved algorithms. In this simulation, only the focal volume producing a specific channel (e.g. (1,1,1)) is considered. By defining the average number of molecules in the laser focal volume, λ, a number of molecules are generated following Poisson statistics. The fragment ions from each generated molecule are detected with 50% efficiency. Additionally, a typical dead time of 10% of the TOF window is simulated by reducing the fragment ion detection efficiency by 10% for each previously detected ion of the same type. Thus, the first detected sulfur ion must pass a 50% detection test, the second a 45% test, the third a 40% test, etc. This scheme produces a list of fragment ions for each ion type. The simple algorithm selects one combination of detected fragments as the "coincident ions" while the advanced algorithm searches for ions coming from the same parent. If the former's choice correctly chooses ions from the same parent, its yield is increased, while the latter's yield increases if it discovers only one combination of ions matching the same parent. Since the measured three-dimensional momentum space is so large, it is assumed that ions from different parents could not appear as coming from the same parent. This fact is reflected in the large TOF windows and justifies ignoring the distributions that would otherwise place more fragment ions closer together in time and increase the effect of the dead time and momentum uncertainty. Such scarcity makes it unnecessary to perform a classical trajectory calculation and simulate momentum measurements with appropriate uncertainty.

The number of molecules successfully detected by each algorithm for a range of λ is plotted in Figure 34. The ratios between the rates of success for both algorithms are consistent with the OCS experiment and previous work done on N_2 where $\lambda \approx 1$ in both cases.
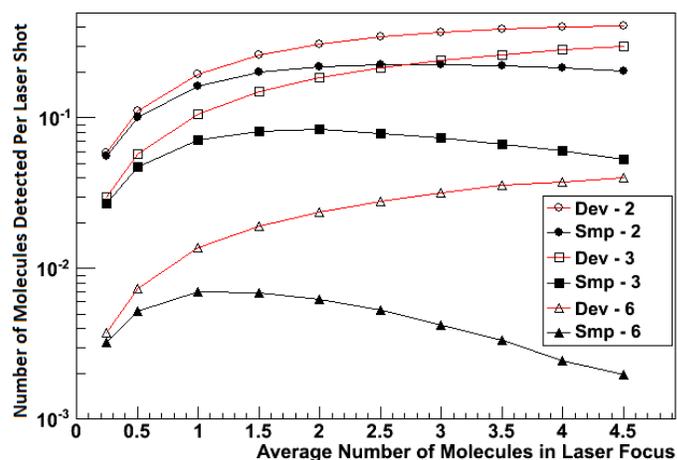
**Figure 34 – Simulation results of diatomic (circle), triatomic (square), and 6 atom molecules (triangle)
Simple algorithm rates are shown with solid markers while advanced rates with open markers**

Both algorithms have peaks in their success rates but the advanced algorithm only begins to decrease at high count rates ($\lambda \geq 5$) as a result of multiple molecule detection (the upper count rate of model is conservatively set at $\lambda =5.5$). Regardless, such count rates are likely to breach experimental feasibility due to an increase in computation time and detector resolution. Within the plotted range, a factor of 3-4 increase in detection rate is available by increasing the number of molecules in the focus from 1 to 3.5 and employing the advanced algorithm. This improvement grows as the number of atoms in the molecule increases. In the 6 atom case the advanced algorithm gives a factor of 5 increase in detection rate.

5.6 Conclusion

It has been shown that the simple treatment of coincidence data in CEI experiments used up to now is sufficient for low count rate data sets on small molecules made up of easily identifiable ions. For experiments involving ion identity ambiguity or the high count rates required for polyatomic reconstruction, the advanced algorithm has been shown (through experiment and simulation) to extract more true coincidences while relaxing the TOF window definitions. With such a scheme, detection yield for polyatomic molecules becomes significantly enhanced, making it possible to use Coulomb explosion as a diagnostic of experiments which seek to demonstrate coherent control of molecular geometry or promote selective bond breaking in complex systems. Currently this algorithm only analyzes one ionization channel at a time but it may be possible to further improve it in order to handle the detection of multiple channels simultaneously.

6 OCS Fragmentation

6.1 Motivation

The experimental study of multiple ionization and complete breakup of small molecules in collision with highly charged ions is a field of considerable interest now [1] and has been since the development of the multiple coincidence method using time and position sensitive detection [2]. The technique relies on the ability to detect every fragment ion produced by a single molecule in coincidence. This allows the dependence of final ionization state parameters such as the total energy release and the angles between the momentum vectors of the fragments, to be determined as a function of the projectile energy and charge state [3,4]. These parameters can then in turn be used to determine how closely the dissociation reaction can be described by a purely Coulombic potential and to what extent the bonds break simultaneously in a concerted dissociation reaction or one at a time in a stepwise manner [1].

The multiple ionization and dissociation of the atmospherically significant molecule OCS [5] from collisions with Ar^{8+} and Ar^4 is investigated. OCS is of particular interest because it has major properties which are close to those of the heavily investigated CO_2. Though similar, the substitution of one oxygen atom by a sulfur atom introduces significant differences in terms of an asymmetry in bond length and mass distribution as well as making the molecule polar. These properties allow serve as an assessment of how asymmetry effects the molecular breakup during multiple ionization, in comparison with CO_2 [3,4]. Previous ion impact results have only been carried out using Ar^+, but have revealed a range of channels. OCS^+ to OCS^{4+} were generated from Ar^+ impact with subsequent fragmentation into two and three-body pathways [6], with selected analysis of two-ion pathways. Here the concentration is on channels which result in triple ion molecular breakup, concentrating on ionization states from 3+ to 6+.

6.2 Experimental

The present experimental setup was similar to that described earlier [4] except for a newly constructed recoil ion extractor in which a stack of 30 electrodes generate a uniform electric field (about 15V/mm). Two experiments were performed using Ar^{8+} and Ar^{4+} ions from the TMU ECR ion source (TMUECRIS) accelerated to energies of 120keV and 60keV respectively. The beam was trimmed with a 0.5mm aperture and crossed a target gas beam of OCS introduced through a multicapillary plate. Typical operating pressures were $7x10^{-5}$ to $9x10^{-5}$ Pa whereas the base chamber pressure was on the order of 10^{-7} Pa. Ejected electrons were accelerated in the opposite direction and passed through a 1mm hole as well as an array of capillaries before being detected by a channel electron multiplier. The detection of these electrons produces the trigger for the data acquisition system. The fragment ions drifted through

a 204 mm TOF region until striking a pair of 120mm diameter microchannel plates in Chevron formation. The electron avalanche from the second microchannel plate was collected by a backgammon type anode [2,7]. The four output signals from the anode were amplified by charge-sensitive preamplifiers (Ortec: 142 B) and sent to a four channel ADC equipped PC. The four channel spectra were analyzed with a sophisticated algorithm to extract the x-y position and time of flight of each detected event in coincidence [8]. Approximately 23 hours and 88 hours of data were collected for the Ar^{8+} and Ar^{4+} projectiles respectively.

6.3   Analysis

Due to the number of possible triple ion fragmentation channels, and the fact that there is a charge to mass ratio degeneracy between O^{a+} and S^{2a+}, the data sets were first parsed for each channel in terms of their ion states. For this article, a fragmentation channel labeled (a,b,c) is defined by the charges on the fragment ions O^{a+}, C^{b+}, and S^{c+}. The details of the algorithm used to analyze the data are described in detail elsewhere [9] but, in brief, data extraction has been optimized by considering that each detected event could be any possible ion and any combination of ions (O^{a+}, C^{b+}, and S^{c+}) might result from a single OCS parent molecule. The validity of each hypothetical molecule is discriminated using conservation of momentum by calculating the theoretical collision center and accepting only those within a narrow range of high probability (the location range is approximately a cubic centimeter centered at the highest point of collision statistics).

The count rates for each channel as a function of projectile are shown in Figure 35. The electron capture ionization process allows only significant formation of fragmentation channels whose total charge is up to 8 and 4 for Ar^{8+} and Ar^{4+} respectively. Furthermore, based on the assumption that the electron ejection process is largely isotropic and independent of the final ion charge distribution, the distribution of the fragment ion momentum directions exhibits isotropy. This means that in certain cases it is possible to choose a restricted solid angle over which to observe a channel, for example 45 degrees to the TOF axis. This is necessary when the fragmentation channel contains ions with similar times of flight for certain molecular orientations, resulting in reduced count rate for those orientations due to detector deadtime. As a result, channels with exact ion TOF overlap (those with O^{n+} and S^{2n+} ions) are detected with significantly less efficiency then those with no overlap ((1,1,1), (1,2,1), etc) or little time overlap ((1,1,3), (2,1,3), (2,2,3), etc.). It is for this reason for instance that the (2,1,2) channel appears more dominant than the (1,2,2).
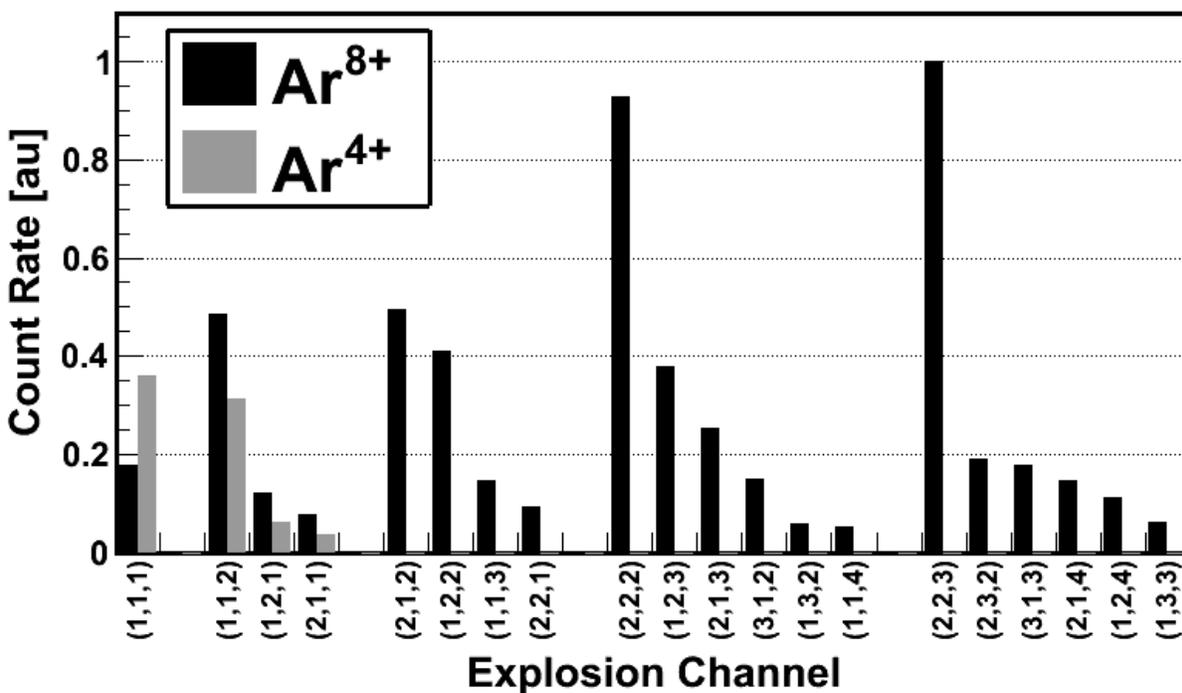
Figure 35 – Count rates as a function of fragmentation channel and projectile ion

6.4 Results

The energy distributions for the four channels with the lowest charge states are very similar whether generated by Ar^{4+} or Ar^{8+} this is in line with previous results for CO_2 using several extremely different ion projectiles and the phenomenon was attributed to the presence of many excited ionic states from which the molecule can dissociate [3]. For Ar^{4+} impact only low channels can be achieved but statistics are best for the 3+ state. Figure 36 shows the total kinetic energy release (KER) for various channels along with arrows and lines indicating the theoretical release energies calculated from equilibrium geometry distributions assuming a Coulomb potential. For OCS^{4+} and lower charge states all channels exhibit energy release which peaks at lower values than expected from Coulomb explosion. The lowest ratio, 89%, is observed for the 3+ state. Recent measurements of this charge state in work on CO_2 collisions with Ar^{8+} ions [1] has shown the energy to peak at a value around 70% of that expected from Coulomb explosion, and this has also been observed for femtosecond laser ionization measurements [10] using few cycle pulses. These results have been attributed to the presence of a partially bound ground state in CO_2^{3+} [11]. This pattern is repeated in OCS which also shows a shallow bound region in the ground and excited states of the 3+ ion [12], followed by a Coulombic behavior at longer bond lengths. A similar energy release ratio would be expected for OCS, which means this result is somewhat high. However, earlier work on CO_2 using Xe^{18+} and Xe^{43+} (5.9MeV u−1) and He+ (250 keV) [3] measured the peak energy release form the 3+ ion to be around 115% of the predicted Coulombic value. For other

49

low channels (2,1,1) and (1,1,2), the ratio is higher at around 95% of the expected value which is understandable as these charge states are likely to be completely repulsive. The peaks are lower in energy than for the corresponding channels in CO_2 [3], which are close to 130% of the Coulombic energy. These higher energies were attributed to the effect of insufficient electronic screening at close range for these channels implying that ground and excited states might give higher energy release than from a purely point particle calculation. The current result is consistent with this picture as the S-C arm is already stretched compared to the C-O arms of CO_2. The (2,1,2) channel does release more energy than expected from purely point charge Coulombic consideration in agreement with the measurements on CO_2 but to a lesser extent with around 110% compared to 120% for CO_2 [3]. This result is also consistent with the screening picture. For (2,2,2) the peak energy position is closest to the calculated value (99%) indicating the Coulombic picture is increasingly valid with higher charge states - a trend also seen in CO_2 [3,4].
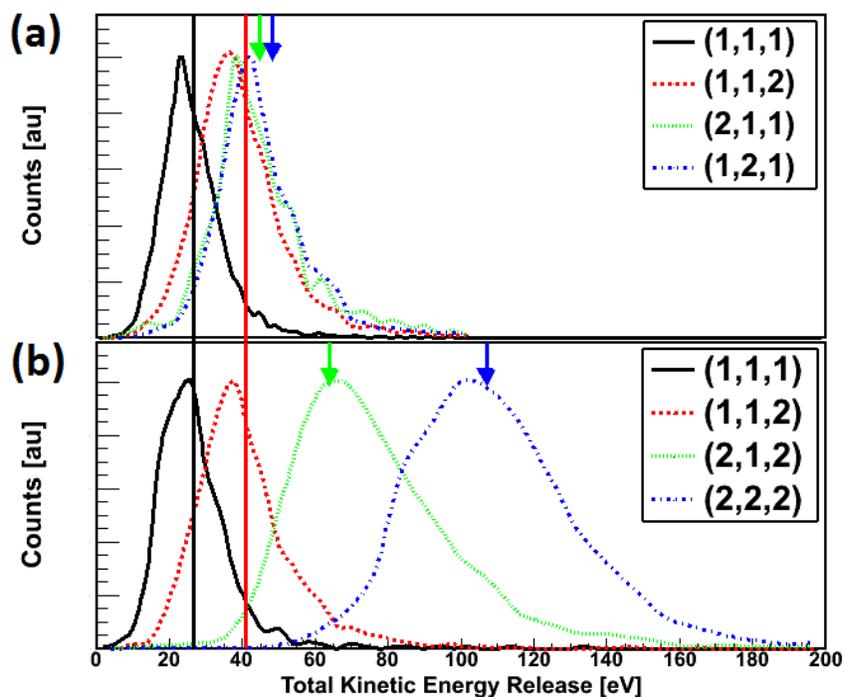
Figure 36 – Total kinetic energy release (KER) spectra for selected channels using Ar[4+] (a) and Ar[8+] (b) Arrows indicate theoretical KER calculated from the OCS equilibrium geometry
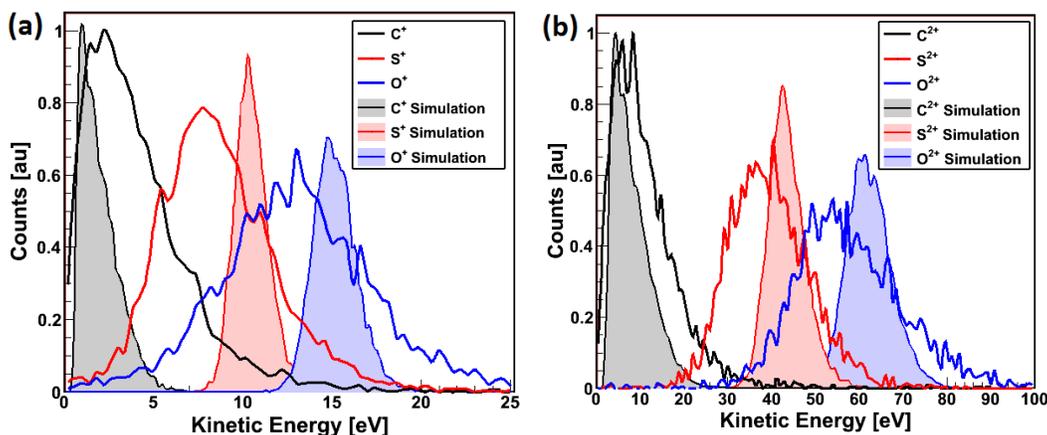
Figure 37 – Kinetic energy for fragment ions from (a) (1,1,1) channel and (b) (2,2,2) channel

The analysis is continued on the Ar[4+] generated (1,1,1) and Ar[8+] generated (2,2,2) channels as these channels represent the extremes of the agreement with a purely Coulombic picture and were measured with the highest isotropic statistics. Figure 37 plots the individual energies of the fragment ions compared to distributions calculated using the electronic structure program package GAMESS [13]. The peak of the terminal ion energies are lower than expected for ground state geometries as would be expected from the total energy release distributions. The widths of the distributions are considerably larger than calculated and comparable to those measured for CO_2 indicating the importance of excited states. The peaks of the sulfur and oxygen ion energies are 75% and 90% respectively of that calculated for the (1,1,1) channel while for the (2,2,2) channel they are 85% and 85% respectively. The carbon ion in both cases is emitted with a peak energy higher than calculated 220% for (1,1,1) and 175% for (2,2,2). These values indicate the significant differences with CO_2 where, in particular, good agreement was found between the peak positions of the measured and calculated carbon ion energy distributions. This discrepancy is an indication that the OCS energy release distribution is dramatically influenced by the effect of the bending experienced by the molecule during ionization. To investigate this, the coincidence parameters are considered in greater detail by first measuring the simplest coincidence parameter available θ_v - the angle between the momentum vectors of the outer sulfur and oxygen ions (see Figure 38 inset). The geometry of a triatomic molecule is defined by two bond lengths (between the terminal atoms and the central atom) and a bend angle α (defined as the angle between the two bond lengths). Though the geometry of the OCS molecule is not directly measured here, the momentum vectors give some indication as to its form. The relationship between θ_v and the bend angle α is non-trivial. However, in the extreme linear case, it is clear that both the bond angle α and θ_v are 180 degrees. As the molecule bends away from 180 degrees, so too does θ_v. Therefore, a spread in θ_v does give an indication of the range of bent geometries for OCS.
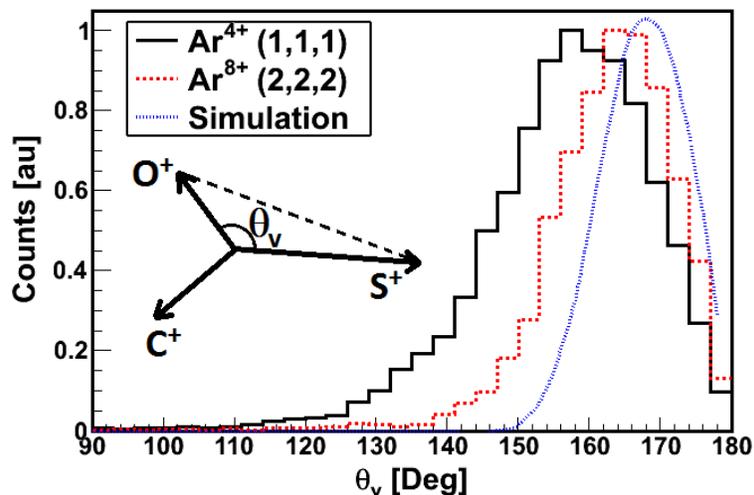
Figure 38 – θ_v distribution for (solid black) Ar[4+] generated (1,1,1) and (dashed red) Ar[8+] generated (2,2,2) Simulated distribution is the same for both channels (dotted blue). Inset shows the calculation of θ_v

The distribution of θ_v for the (1,1,1) and (2,2,2) channels exhibit the general behavior expected after considering the carbon energy distributions. The 3+ state experiences considerably more bending than the 6+ state with a long tail stretching down to 110 degrees compared to 140 degrees. For comparison, a simulated θ_v distribution was produced from the ground state bond angle distribution assuming a Coulomb potential (see above). The peak and FWHM of the (2,2,2) distribution is close to the theoretical distribution (5% lower) confirming that the channel is closely described by the pure Coulomb interaction. Although agreement with calculation is not as close as in the case of CO_2 [3,4] it is perhaps better than would be expected at first glance from the carbon energy distribution, which is an indication that the kinetic energy distributed to the carbon ion is very much enhanced during bending by the asymmetric masses of the terminal atoms.

In addition to the importance of concerted breakup through excited states, in the case of the 3+ state there is the possibility of stepwise processes leading to small θ_v values. Stepwise processes involving CO^{2+} and CS^{2+} metastable species have been observed in the ion impact ionization of CO_2 [1] and the femtosecond laser ionization of CS_2 [13]. For OCS both channels are of course possible with partial dissociation of OCS^{q+} into $OC^{m+*} + S^{(q-m)+}$ or $O^{(q-n)+} + CS^{n+*}$, the resulting diatomic fragment rotates as it moves away conserving angular momentum, and subsequently explodes into $O^{r+} + C^{p+}$ or $S^{r'+} + C^{p'+}$ where r+p=m and r'+p'=n. In order to investigate the possibility of stepwise processes the χ parameter is measured as illustrated in Figure 39. χ measures the angle between the momentum of the middle carbon ion and the difference in momentum between the two outer ions. In the case of symmetric CO_2 or CS_2, it is clear that for a symmetric ionization channel bending without asymmetric stretching results in a χ measurement of 90 degrees for any bond angle. The amount of signal at angles far higher or lower than 90 degrees is considered to be a measure of the contribution of stepwise processes in the molecular breakup.
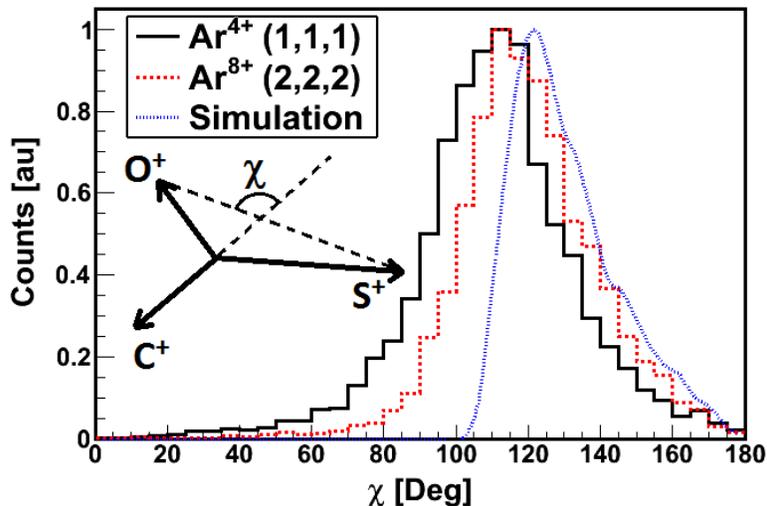
Figure 39 – χ distribution for (solid black) Ar^4+ generated (1,1,1) and (dashed red) Ar^8+ generated (2,2,2) Simulated distribution is the same for both channels (dotted blue). Inset shows the relationship of fragment momentum vectors to χ.

Again, the measurement is compared with a simulation for concerted breakup of the molecule from a distribution of ground state bond lengths and bond angles. For OCS the concerted process in which the bonds break simultaneously does not give a single angle as it does for symmetric molecules such as CO_2 and CS_2 but instead results in a distribution. This means that depending on the range of bend angles allowed, even concerted breaking of the two bonds will result in a value of χ different from 90 degrees. This is due to the asymmetric mass distribution of OCS which leads to the oxygen and carbon ions gaining momentum predominantly in the same direction opposite the motion of the sulfur ion. The χ distributions for the (1,1,1) and (2,2,2) channels peak at close to the same value (115 degrees compared to 120 for the calculation) once again the higher channel is more like the simulation indicating a higher degree of concertedness. The discrepancy between the measured distributions and the simulation at lower angles is suggestive of stepwise processes, however it isn't clear that this is not simply due to the effect of bending in both channels. In order to investigate this further the relationship between χ and $θ_v$ is plotted in figure 6. This distribution has been previously used to observe the sequential breakup of triatomic molecules [13]. Figure 6 shows that χ and $θ_v$ change in concert as $θ_v$ reduces from 180 degrees, χ too reduces. In the stepwise case, however, we expect that a distribution of χ and $θ_v$ parameters should extend to a large range. For CS_2, a chain was observed extending from 140 to 180 degrees in χ and 0 to 180 degrees in $θ_v$[ref]. In the case of (1,1,1) there is significant data throughout the χ range but for (2,2,2) there is little data below 90 degrees indicating that this channel does not undergo stepwise fragmentation as would be expected from a higher ionization where the metastable ion produced would have to be a quadrupley charged . To further clarify the possible part played by stepwise processes a simulation of χ vs $θ_v$ for the concerted channel was performed and superimposed as a solid curve in Figure 40. In both cases the data follows the trend indicated by the concerted process with uncertainty about the line attributable to non-equilibrium bond lengths and asymmetric stretching (asynchronous processes). This indicates that "near" concerted processes are dominant for both channels. Nevertheless the data below χ = 100 degrees is inconclusive and so in

order to categorically confirm the extent of any stepwise processes, a new visualization is needed – one which is less ambiguous than the χ vs θ_v plot is for OCS.
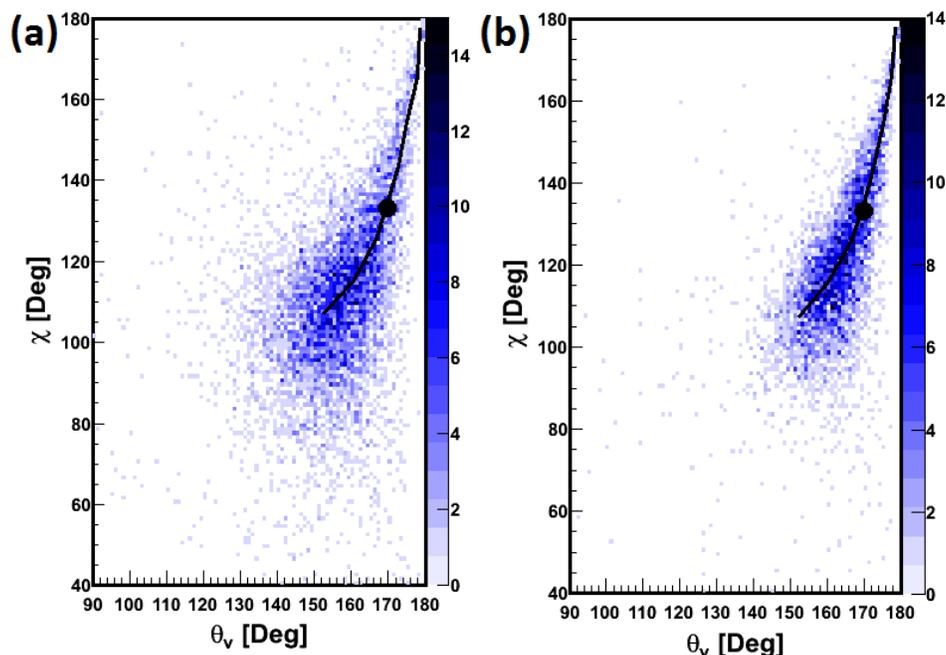
Figure 40 – χ vs θ_v distributions for (a) Ar[4+] generated (1,1,1) channel and (b) Ar[8+] generated (2,2,2) channel Filled circle indicates expected value for equilibrium geometry while solid curve is the expected range resulting from bending.

Two well established methods which have recently been used to investigate the existence of stepwise processes during highly charged ion impact are the Dalitz and Newton plots [1]. Firstly the Newton plot relies on the fact that the fragment ion momenta are measured in the lab frame resulting in three 3-dimensional vectors. Since triatomic molecules are confined to a plane, the collection of 9 measurable parameters can easily be visualized by plotting the momentum vectors in the molecular frame. The convention used here is to rotate the molecule such that the sulfur ion momentum points along the positive Y-axis; the oxygen momentum is confined to the positive half of the X-axis; and the carbon momentum is confined to the negative half of the X-axis. The Newton Plot of a concerted process creates one point (for the S ion) and two island distributions representing the momentum of the C and O ions. A stepwise process also has a clear signature if the metastable molecular ion created has a long enough lifetime for the angular momentum, generated by the break up, to cause it to rotate one or more times before it dissociates. This results in two half ring structures in the Newton plot. Figure 41 displays plots for (1,1,1) and (2,2,2) where the momentum from each triple coincidence has been scaled to the momentum of the sulfur ion. For reference, half rings are plotted where the sequential process is expected to appear in the (1,1,1) channel with their radius set by the momentum from a coulomb explosion of CO^{2+} from equilibrium. In addition to the clear concerted signal there is a diffuse background within the half rings which is attributable to stepwise processes in the case of the (1,1,1) from $CO^{2+} + S^+$. An upper bound of 5% can be set on the amount of data associated with this stepwise

process for (1,1,1) from the region within the half circles. For the (2,2,2) channel the diffuse background is minimal in Figure 41(b) as expected from the χ vs θ_v plot, also the size of the momentum islands are noticeably smaller indicating the lower amount of induced asymmetric bond stretching (asynchronous processes) in transient OCS^{6+}. Sequential breakup with a metastable CS^{2+} ion can only appear clearly when the molecular frame convention is switched such that the oxygen ion points along the Y-axis, carbon is to the right, and sulfur is to the left. Figure 42 displays the same channels with this new convention and the associated sequential rings. The data in the stepwise region is limited to 2% of the total counts. For completeness Figure 42(b) shows the plot for the 6+ channel.
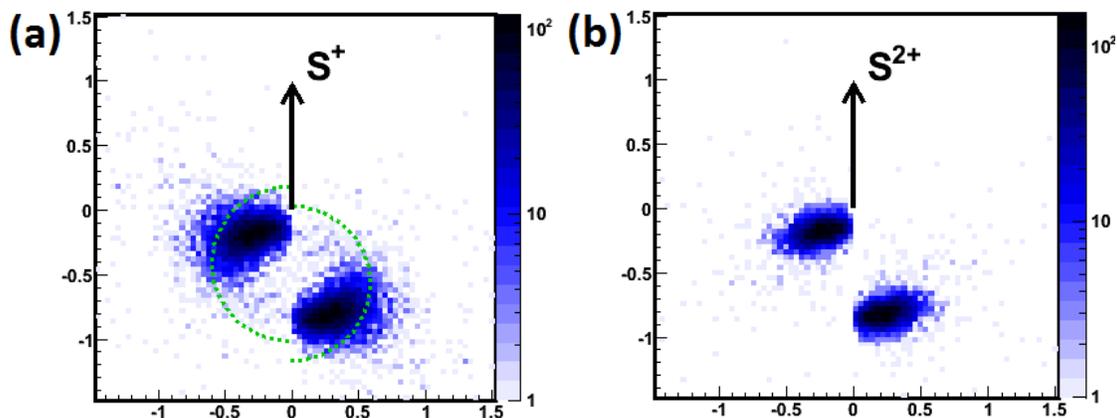
**Figure 41 – Normalized Newton plots for (a) Ar^{4+} generated (1,1,1) and (b) Ar^{8+} generated (2,2,2)
The arrow indicates the momentum vector of the sulfur ion. Data on the left and right halves are for carbon and oxygen ions respectively. The dotted rings indicate regions of CO^{2+} metastable sequential fragmentation.**
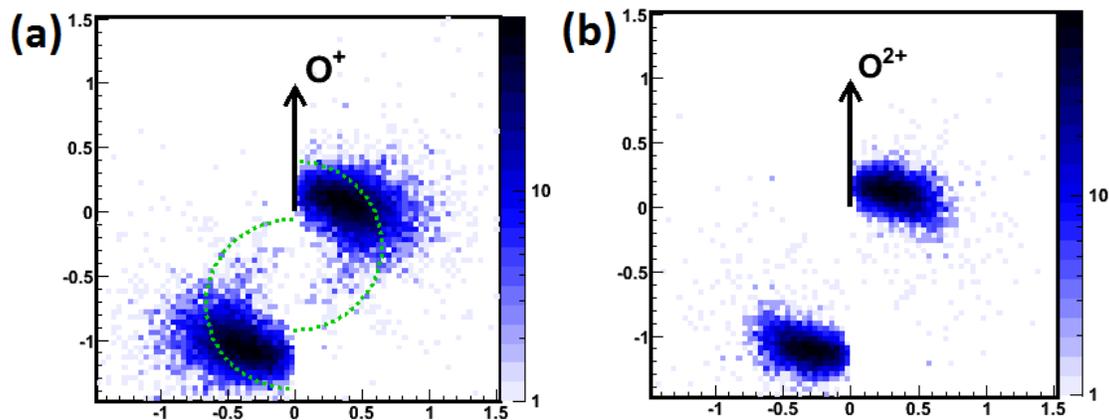
**Figure 42 – Normalized Newton plots for (a) Ar^{4+} generated (1,1,1) and (b) Ar^{8+} generated (2,2,2)
The arrow indicates the momentum vector of the oxygen ion. Data on the left and right halves are for sulfur and carbon ions respectively. The dotted rings indicate regions of CS^{2+} metastable sequential fragmentation.**

A clearer picture has emerged from the Newton plots but one drawback which is associated with them is the inhomogeneity in phase space, which means that one cannot clearly see the trend for both stepwise channels at the same time. A more robust view of molecular geometry is available with the Dalitz plot.

In this two dimensional histogram, the fraction of energy carried by the carbon ion is plotted along the y-axis (ε_{c+} / KER – 1/3) while the difference in the fraction of energy between the sulfur and oxygen ions is plotted along the x-axis ((ε_{O+} - ε_{S+})/ $\sqrt{3}$ KER). The resulting space is a well define oval where each point represents specific arrangements of momentum vectors (Figure 43(a)). Though the geometry of the molecule and the momentum vectors are related in a complicated way, a simple simulation shows that the equilibrium geometry should appear on the Dalitz plot at (0.11, -0.29) for both the (1,1,1) and (2,2,2) channels. Bending of the molecule results roughly in a decrease of θ_v and is associated with vertical changes from the equilibrium point on the Dalitz plot. Asymmetric stretching of the molecule results in signal to the left or right of the vertical and sequential fragmentation of OCS into a metastable CO^{2+} ion modifies all the momentum vectors and should appear as data along a diagonal across the plot from bottom to top left [1]. In the case of a metastable CS^{2+} ion, a diagonal in the opposite direction should appear. For the (1,1,1) channel as with the Newton plots there is clearly data in the region for both stepwise channels, the overall picture is of a broad spread in geometry incorporating a high degree of asymmetry. The smaller much tighter distribution of the 6+ data well illustrates its domination by concerted processes and Coulomb explosion from bond angles close to the equilibrium distribution.
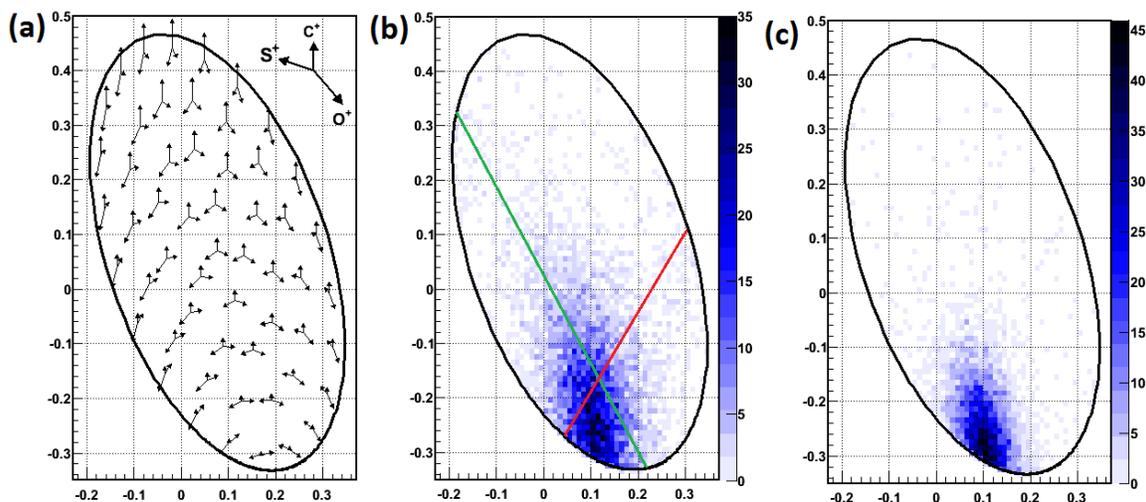
Figure 43 – Momentum vectors for various points on the Dalitz plot (a)
Dalitz plot for Ar^{4+} generated (1,1,1) (b) and Ar^{8+} generated (2,2,2) (c). The fractional energies are given by ε_x = ($k_x^2/2m_x$) / KER. The leftmost green line indicates the region for CO^{2+}+S$^+$ the rightmost red line represents CS^{2+}+O$^+$.

6.5 Conclusions

By investigating the complete ionic breakup of OCS in collisions with Ar^{4+} and Ar^{8+} at 15keV/q the effect of substituting one sulfur atom for an oxygen atom in CO_2 has been assessed. The biggest global effect

has been to reduce the kinetic energy released across the range of channels for OCS in comparison with CO_2, although discrepancies between previous work on the 3+ ion in CO_2 make a clear comparison for the (1,1,1) channel difficult. The overall kinetic energy released for each fragment has a considerably wider distribution than calculated but similar to the case of CO_2. Several coincidence methods have been used to reveal dynamics of the two channels which can most (2,2,2) and least (1,1,1) be described by a simple Coulomb explosion of point like particles. Although the χ parameter offers limited insight into the degree of stepwise processes for the 3+ state, because of the molecular asymmetry, Newton and Dalitz plots reveal the small but measurable population of stepwise breakup events, which favor the generation of a CO^{2+} over a CS^{2+} metastable by a ratio 5/2. For the 6+ state the Dalitz plot in particular reveals a behavior close to equilibrium in terms of bond angle and from the left right symmetry shows there is little preference for either the CO or the CS bond to be preferentially modified in asynchronous stretching, despite the initial asymmetry in bond length and electronic structure.

7 Conclusion and Future Work

The previous standards for data acquisition and analysis have been revamped through the implementation of sophisticated computer algorithms in waveform analysis, PSD calibration, and coincidence detection. These improvements have been demonstrated with several experiments studying a variety of molecules and finally provided new measurements of OCS fragmentation via Ar^{4+} and Ar^{8+} initiated ionization. The successful overhaul of the CEI framework serves as the ground work on which polyatomic molecular imaging and pump-probe dynamic imaging will be built. Direct improvements on this work include integrating geometry reconstruction within momentum imaging to give geometry information on the fly. Such an implementation would reduce the limitations of imaging molecules with degenerate fragments. Furthermore, building the coincidence algorithm with a recursive base and the scope to analyze all fragmentation channels simultaneously would provide fluidity to the developing technique.

The CEI apparatus at UW is under constant augmentation. A newly acquired data acquisition machine with high performance ADCs will allow for more precise measurements of ion momentum. A polished version of the waveform analysis algorithm will be used to calibrate the detector. A supersonic molecular beam source is being installed to cool the target molecules and improve the imaging resolution. Lastly UW has recently acquired a new short pulse laser system. The combination of these changes will allow the development of pump-probe experiments with which the first galloping molecule is expected to be observed.

References

[1]Kanter EP, Cooney PJ, Gemmell DS, Groeneveld KO, Pietsch WJ, Ratkowski AJ, et al. Role of Excited Electronic States in the Interactions of Fast (Mev) Molecular-Ions with Solids and Gases. Phys Rev A 1979;20(3):834-854.

[2]Vager Z. Coulomb explosion imaging of molecules. Advances in Atomic, Molecular, and Optical Physics, Vol 45 2001;45:203-239.

[3]Neumann N, Hant D, Schmidt LPH, Titze J, Jahnke T, Czasch A, et al. Fragmentation Dynamics of $CO_2(3+)$ Investigated by Multiple Electron Capture in Collisions with Slow Highly Charged Ions. Phys Rev Lett 2010 MAR 12;104(10):103201.

[4]Sanderson JH, Nishide T, Shiromaru H, Achiba Y, Kobayashi N. Near-Coulombic behavior in the dissociative ionization of CO_2 due to impact by Ar8+. Physical Review a 1999 JUN 1999;59(6):4817-4820.

[5]Siegmann B, Werner U, Lutz HO, Mann R. Complete Coulomb fragmentation of CO_2 in collisions with 5.9 MeV u(-1) Xe18+ and Xe43+. Journal of Physics B-Atomic Molecular and Optical Physics 2002 SEP 2002;35(17):3755-3765.

[6]Bocharova I, Karimi R, Penka EF, Brichta J, Lassonde P, Fu X, et al. Charge Resonance Enhanced Ionization of CO_2 Probed by Laser Coulomb Explosion Imaging. Phys Rev Lett 2011 AUG 2 2011;107(6):063201.

[7]Jean-Paul Brichta. Laser-initiated Coulomb explosion imaging of small moleculesUniversity of Waterloo, Waterloo, Ontario, Canada; 2008.

[8]Legare F, Lee KF, Litvinyuk IV, Dooley PW, Bandrauk AD, Villeneuve DM, et al. Imaging the time-dependent structure of a molecule as it undergoes dynamics. Phys Rev A 2005 NOV;72(5):052717.

[9]Legare F, Litvinyuk I, Dooley P, Quere F, Bandrauk A, Villeneuve D, et al. Time-resolved double ionization with few cycle laser pulses RID A-5739-2009. Phys Rev Lett 2003 AUG 29;91(9):093002.

[10]Légaré F, Bocharova IA, Lassonde P, Karimi R, Sanderson JH, Johnston T, et al. Inelastic rescattering processes in molecules measured with few-cycle laser pulses. J Phys B: At Mol Opt Phys 2009 14 December 2009;42(23):1-5.

[11]Sanderson JH, Goodworth TRJ, El-Zein A, Bryan WA, Newell WR, Langley AJ, et al. Coulombic and pre-Coulombic geometry evolution of carbonyl sulfide in an intense femtosecond laser pulse, determined by momentum imaging. Phys Rev A 2002 APR;65(4):043403.

[12]Brichta JP, Walker SJ, Helsten R, Sanderson JH. Ultrafast imaging of multielectronic dissociative ionization of CO_2 in an intense laser field. J Phys B-At Mol Opt Phys 2007 JAN 14;40(1):117-129.

[13]Hasegawa H, Doi K, Hishikawa A, Yamanouchi K. Coincidence Imaging of coulomb explosion of CS2 in intense laser fields. Science of Superstrong Field Interactions 2002;634:197-198.

[14]Hishikawa A, Hasegawa H, Yamanouchi K. Sequential three-body Coulomb explosion of CS2 in intense laser fields appearing in momentum correlation map. Chemical Physics Letters 2002 JUL 30 2002;361(3-4):245-250.

[15]Hasegawa H, Hishikawa A, Yamanouchi K. Coincidence imaging of Coulomb explosion of CS2 in intense laser fields. Chem Phys Lett 2001 NOV 23;349(1-2):57-63.

[16]Dooley PW, Litvinyuk IV, Lee KF, Rayner DM, Spanner M, Villeneuve DM, et al. Direct imaging of rotational wave-packet dynamics of diatomic molecules. Phys Rev A 2003 AUG;68(2):023406.

[17]Zhao K, Zhang G, Hill WT. Strong-field dissociative ionization of a linear triatomic molecule: Relationship between Coulomb-explosion energies and bond angle. Phys Rev A 2003 DEC;68(6):063408.

[18]Legare F, Lee KF, Litvinyuk IV, Dooley PW, Wesolowski SS, Bunker PR, et al. Laser Coulomb-explosion imaging of small molecules. Phys Rev A 2005 JAN;71(1):013415.

[19]Matsuda A, Takahashi EJ, Hishikawa A. Dalitz plot analysis of Coulomb exploding O3 in ultrashort intense laser fields. J Chem Phys 2007 2007 Sep 21;127(11):114318.

[20]Brichta J, Seaman AN, Sanderson JH. Ultrafast imaging of polyatomic molecules with simplex algorithm. Comput Phys Commun 2009 FEB;180(2):197-200.

[21]Legare F, Litvinyuk IV, Dooley PW, Quere F, Bandrauk AD, Villeneuve DM, et al. Time-resolved double ionization with few cycle laser pulses. Phys Rev Lett 2003 AUG 29 2003;91(9):093002.

[22]Wales B, Bisson E, Karimi R, Kieffer J, Légaré F, Sanderson J. A coincidence detection algorithm for improving detection rates in Coulomb Explosion Imaging. Nucl Instrum Methods Phys Res A 2011 Accepted for Publication.

[23]Patrick Dooley. Molecular imaging using femtosecond laser pulsesMcMaster University, Hamilton, Ontario, Canada; 2003.

[24]Mizogawa T, Shiromaru H, Sato M, Ito Y. A two-dimensional position-sensitive ion detector based on modified backgammon method with weighted-coupling capacitors. International Journal of Mass Spectrometry 2002 4/1;215(1-3):141-149.

[25]Bevington PR, Robinson DK. Data Reduction and Error Analysis. 3rd ed. 1221 Avenue of the Americas, New York, NY 10020: McGraw-Hill; 2003.

[26]Xu H, Okino T, Nakai K, Yamanouchi K, Roither S, Xie X, et al. Two-proton migration in 1,3-butadiene in intense laser fields. Phys Chem Chem Phys 2010;12(40):12939-12942.

[27]Gagnon J, Lee KF, Rayner DM, Corkum PB, Bhardwaj VR. Coincidence imaging of polyatomic molecules via laser-induced Coulomb explosion. J Phys B-At Mol Opt Phys 2008 NOV 14;41(21):215104.

[28]Mizogawa T, Awaya Y, Isozumi Y, Katano R, Ito S, Maeda N. New Readout Technique for 2-Dimensional Position-Sensitive Detectors. Nuclear Instruments & Methods in Physics Research Section A-Accelerators Spectrometers Detectors and Associated Equipment 1992 FEB 15 1992;312(3):547-552.

[29]Crutzen PJ, Heidt LE, Krasnec JP, Pollock WH, Seiler W. Biomass Burning as a Source of Atmospheric Gases Co, H-2, N2o, No, Ch3cl and Cos. Nature 1979 1979;282(5736):253-256.

[30]Jana MR, Ray B, Ghosh PN, Safvan CP. Dissociation of carbonyl sulfide by 150 keV Ar(+) ion impact. Journal of Physics B-Atomic Molecular and Optical Physics 2010 NOV 14 2010;43(21):215207.

[31]Mizogawa T, Sato M, Awaya Y. Application of the Mbwc 2-Dimensional Position Readout Technique to a Multiwire Proportional Counter. Nuclear Instruments & Methods in Physics Research Section A-Accelerators Spectrometers Detectors and Associated Equipment 1995 NOV 21 1995;366(1):129-136.

[32]Rajgara FA, Mathur D, Nishide T, Kitamura T, Shiromaru H, Achiba Y, et al. Multi-hit, position-sensitive, time-of-flight spectrometry using a modified-backgammon-weighed-capacitor anode. International Journal of Mass Spectrometry 2002 APR 1 2002;215(1-3):151-162.

Appendix

A   Waveform Analysis

The following C++ code implements the waveform algorithm to calculate the position and time of the events appearing in a given waveform. To improve efficiency, some of the discrimination techniques described in section 2.5 have moved from the function *processPeaks* to *findPeaks*. The class *TSIGDataModule* contains the private variables *rawData[4][shotSize], sumData[shotSize],* and *modData[shotSize].* The first is a 2 dimensional array containing the 4 waveforms output from the PSD and digitized with the ADCs. The second is the sum of waveform of the four channels. The last is the modified waveform after a triangle filter has smoothed the sum waveform. The functions are implemented in the following order: *smoothData, findPeaks,* then *processPeaks*. The internal variables *naHitT[naNHitBlPeak], naHitXY[naNHitBlPeak][2], naHitQT[naNHitBlPeak],* and *naNHitBlPeak* store the event time, position, and number of events respectively.

A-I   smoothData

```cpp
void TSIGDataModule::smoothData(){
        // Performs triangular smoothing on the summed waveform
        int i;

        modData[0] = sumData[0];
        modData[1] = sumData[1];
        for(i=2; i<shotSize-2; i++){
                modData[i] = (sumData[i-2] + 2 * sumData[i-1] +  3 * sumData[i] + 2 * sumData[i+1] +
sumData[i+2]) / 9;
        }
        modData[shotSize - 2] = sumData[shotSize - 2];
        modData[shotSize - 1] = sumData[shotSize - 1];
}
```

A-II   findPeaks

```cpp
int TSIGDataModule::findPeaks(int DivThreshold, int RisingThreshold, int NBaselineAvg, int BaselineCutoff,
int RingDiffThreshold){
        // Returns the number of peaks found
        // Mountain peaks are described as A peaks, valley peaks are described as V peaks


        // New algorithm variables
        int diff, i, j;
        int nawavei;
        int nabl1i, nabl2i, nabl3i;
        int nablitemp;
        int nadropstreak;
        float nabl12, nabl23;
        float napeakcur, napeakprev;
        float namht1, namht2, namht3, namht4, namht5;

        nabl1i = 0;
        nabl2i = 0;
        nabl3i = 0;
```

```
        nadropstreak = 0;
        DivThreshold = -1 * DivThreshold; // The DivThreshold parameter is given as a positive value, but
the steps are negative
        nawavei = 0;
        naNPeaks = 1;
        naPeakList[0] = 0;
        naBliList[0][0] = 0;
        naBliList[0][1] = 0;
        naBliList[1][0] = 0;
        naNBli = 1;
        nawavei = -1;
        napeakcur = 0;
        naNBL = 1;
        naBLList[0][0] = modData[0] + modData[1];
        naBLList[0][1] = modData[0] + modData[1];
        naNBLList[0][0] = 2;
        naNBLList[0][1] = 2;
        naBLList[1][0] = modData[0] + modData[1];
        naNBLList[1][0] = 2;
        while (nawavei < (shotSize-NBaselineAvg)){
                // Calculate baselines until a rising streak is found
                nabl12 = 0;
                nabl23 = 0;
                nadropstreak = 0;
                while ((nadropstreak < RisingThreshold) && (nawavei < (shotSize-NBaselineAvg))){
                        nadropstreak = 0;
                        nawavei++;
                        diff = modData[nawavei+1] - modData[nawavei];
                        if (diff < DivThreshold){
                                // negative derivative more extreme than the DivThreshold
                                do {
                                        nadropstreak++;
                                        nawavei++;
                                        diff = modData[nawavei+1] - modData[nawavei];
                                } while ((diff < DivThreshold) && (nadropstreak < RisingThreshold) &&
(nawavei < (shotSize-NBaselineAvg)));
                        }
                }
                if (nawavei < (shotSize-NBaselineAvg)) {
                        // Not the end of the waveform, continue processing

                        nabl3i = nawavei - RisingThreshold;

                        // Load the baselines into the array
                        naBliList[naNBli][1] = nabl3i;
                        naNBli++;

                        // Calculate baseline value for bl1 - beginning of plateau
                        // Only use points up to the NBaselineAvg value
                        naBLList[naNBL][0] = 0;
                        if ((nabl3i - nabl1i + 1) > NBaselineAvg)
                                nablitemp = nabl1i + NBaselineAvg - 1;
                        else
                                nablitemp = nabl3i;
                        for (i=nabl1i; i<=nablitemp; i++)
                                naBLList[naNBL][0] += modData[i];
                        naNBLList[naNBL][0] = nablitemp - nabl1i + 1;
                        naNBL++;

                        // Find the V peak following the rise. It will be at the index nawavei
                        while ((modData[nawavei+1] < modData[nawavei]) && (nawavei < (shotSize-
NBaselineAvg))) nawavei++;

                        // Store the peak location in a peak array
                        naPeakList[naNPeaks] = nawavei;
                        naNPeaks++;
                        nabl1i = nawavei;

                        // Find the A peak following the V peak. It will be at the index nawavei
```

```
                        nawavei++;
                        while ((modData[nawavei+1] >= modData[nawavei]) && (nawavei < (shotSize-
NBaselineAvg))) nawavei++;
                        naPeakList[naNPeaks] = nawavei;

                        // Set the first baseline halfway between the last two peaks
                        nabl1i = (nawavei + nabl1i) / 2;

                        // Do multihit / multipeak check
                        // If this peak and the previous peak are far enough apart, assume the steps are
well defined.
                        // Otherwise, check if they come from the same step or two steps together
                        napeakprev = napeakcur;
                        napeakcur = modData[naPeakList[naNPeaks-1]];
                        if ((naPeakList[naNPeaks-1] - naPeakList[naNPeaks-2]) < NBaselineAvg){
                                namht1 = naBLList[naNBL-2][1] / naNBLList[naNBL-2][1];
                                namht2 = 0.85 * (napeakprev - namht1);
                                namht3 = napeakcur - namht1;
                                namht4 = exp((float)(naPeakList[naNPeaks-2] - naPeakList[naNPeaks-1]) /
BaselineCutoff) * (0.35 * (napeakprev - namht1));
                                namht5 = napeakcur - (0.65 * (napeakprev - namht1) + namht1);
                                if (fabs(namht5 - namht4) < RingDiffThreshold){
                                        // peaks come from ringing in one step
                                        // delete current peak
                                        naNPeaks--;
                                        // delete current baseline
                                        naNBli--;
                                        naNBL--;
                                        nabl1i = naBliList[naNBli][0];
                                        napeakcur = napeakprev;
                                } else {
                                        // peak is from new step - calculate baseline for bl2
                                        // Calculate baseline value for bl2 - end of plateau
                                        // Only use points up to the NBaselineAvg value
                                        naBLList[naNBL-1][1] = 0;
                                        if ((nabl3i - naBliList[naNBL-1][0] + 1) > NBaselineAvg)
                                                nablitemp = nabl3i - NBaselineAvg + 1;
                                        else
                                                nablitemp = naBliList[naNBL-1][0];
                                        for (i=nablitemp; i<=nabl3i; i++)
                                                naBLList[naNBL-1][1] += modData[i];
                                        naNBLList[naNBL-1][1] = nabl3i - nablitemp + 1;
                                }
                        } else {
                                // peak is from new step - calculate baseline for bl2
                                // Calculate baseline value for bl2 - end of plateau
                                // Only use points up to the NBaselineAvg value
                                naBLList[naNBL-1][1] = 0;
                                if ((nabl3i - naBliList[naNBL-1][0] + 1) > NBaselineAvg)
                                        nablitemp = nabl3i - NBaselineAvg + 1;
                                else
                                        nablitemp = naBliList[naNBL-1][0];
                                for (i=nablitemp; i<=nabl3i; i++)
                                        naBLList[naNBL-1][1] += modData[i];
                                naNBLList[naNBL-1][1] = nabl3i - nablitemp + 1;
                        }
                        naBliList[naNBli][0] = nabl1i;
                } else {
                        // Calculate final plateau values by setting nabl3i to the last waveform point
                        nabl3i = shotSize - 1;

                        naBliList[naNBli][1] = nabl3i;
                        naNBli++;

                        // Calculate baseline value for bl1 - beginning of plateau
                        // Only use points up to the NBaselineAvg value
                        naBLList[naNBL][0] = 0;
                        if ((nabl3i - nabl1i + 1) > NBaselineAvg)
                                nablitemp = nabl1i + NBaselineAvg - 1;
```

```
                else
                        nabltemp = nabl3i;
                for (i=nabl1i; i<=nabltemp; i++)
                        naBLList[naNBL][0] += modData[i];
                naNBLList[naNBL][0] = nabltemp - nabl1i + 1;
                naNBL++;

                // Calculate baseline value for bl2 - end of plateau
                // Only use points up to the NBaselineAvg value
                naBLList[naNBL-1][1] = 0;

                if ((nabl3i - naBliList[naNBL-1][0] + 1) > NBaselineAvg)
                        nabltemp = nabl3i - NBaselineAvg + 1;
                else
                        nabltemp = naBliList[naNBL-1][0];
                for (i=nabltemp; i<=nabl3i; i++)
                        naBLList[naNBL-1][1] += modData[i];
                naNBLList[naNBL-1][1] = nabl3i - nabltemp + 1;
            }
        }

        return naNPeaks;
}
```

A-III   processPeaks

```
int TSIGDataModule::processPeaks(int BaselineCutoff, int RingDiffThreshold, int DeadtimeThreshold){
        // This function runs through the peaks and baselines produced from the findPeaks function
        // and produces a list of peaks and baselines that pass certain criteria.
        // This function cuts out those which fail the following tests:
        // 1] a step where the beginning and end baselines are not far enough apart (BaselineCutoff)
        // 2] two steps occur too close together to allow for correct position calculation
(DeadtimeThreshold)
        // 3] a step where the peak and the difference in baselines don't match the ring factor
(RingDiffThreshold)
        // The positions and times of the event are then calculated

        int i;
        bool failedHit;
        // Variables for averaging the baselines for each channel
        int chi, chj;
        float bl;
        int starti, endi;

        float QTotal; // Total charge used to calculate positions

        // Start at 1 as the 0th baseline is a dud artifact of the peak finding algorithm
        float stepPrev, stepCur, stepNext;
        int peakDiffBack, peakDiffForward;
        stepCur = (naBLList[0][1] / naNBLList[0][1] - naBLList[1][0] / naNBLList[1][0]);
        stepNext = (naBLList[1][1] / naNBLList[1][1] - naBLList[2][0] / naNBLList[2][0]);
        peakDiffForward = (naPeakList[1] - naPeakList[0]);
        naNHitBlPeak = 0;
        // Pad an extra entry on the end
        naBLList[naNBL][0] = naBLList[naNBL-1][1] / naNBLList[naNBL-1][1];
        naBLList[naNBL][1] = naBLList[naNBL][0] / naNBLList[naNBL][0];
        naNBLList[naNBL][0] = 1;
        naNBLList[naNBL][1] = 1;
        naPeakList[naNPeaks] = shotSize + 100; // Arbitrary length - Just needs to be far enough as to not
trigger anything
        for (i=1; i<naNBL-1; i++){
                stepPrev = stepCur;
                stepCur = stepNext;
                stepNext = (naBLList[i+1][1] / naNBLList[i+1][1] - naBLList[i+2][0] / naNBLList[i+2][0]);
                peakDiffBack = peakDiffForward;
                peakDiffForward = (naPeakList[i+1] - naPeakList[i]);
```

```
                    // Check Req 1
                    if (stepCur < BaselineCutoff){
                            // Failed
                            // Continue to next step
                            continue;
                    } else {
                            // Succeeded
                    }

                    // Check Req 2 - peak behind
                    if (stepPrev < BaselineCutoff) {
                            // Ignore previous peak as its step is not substantial
                    } else if (peakDiffBack < DeadtimeThreshold) {
                            // previous peak is substantial and too close to the current peak
                            continue;
                    } else {
                            // Success w.r.t. previous peak
                    }

                    // Check Req 2 - peak ahead
                    if (stepNext < BaselineCutoff) {
                            // Ignore next peak as its step is not substantial
                    } else if (peakDiffForward < DeadtimeThreshold) {
                            // next peak is substantial and too close to the current peak
                            // Skip the next peak as it will fail the previous peak test
                            // Update the forward and backward steps and peaks
                            i++;
                            peakDiffBack = peakDiffForward;
                            peakDiffForward = (naPeakList[i+1] - naPeakList[i]);
                            stepPrev = stepCur;
                            stepCur = stepNext;
                            stepNext = (naBLList[i+1][1] / naNBLList[i+1][1] - naBLList[i+2][0] /
naNBLList[i+2][0]);
                            continue;
                    } else {
                            // Success w.r.t. next peak
                    }

                    // Check Req 3
                    if (((modData[naPeakList[i]] - naBLList[i][1] / naNBLList[i][1]) * 0.65 -
(naBLList[i+1][0] / naNBLList[i+1][0] - naBLList[i][1] / naNBLList[i][1])) > RingDiffThreshold){
                            // Failed
                            // Skip this step
                            continue;
                    } else {
                            // Succeeded
                    }

                    // Passes all test, load the step
                    naHitBl[naNHitBlPeak][0] = naBLList[i][1] / naNBLList[i][1];
                    naHitBl[naNHitBlPeak][1] = naBLList[i+1][0] / naNBLList[i+1][0];
                    naHitBli[naNHitBlPeak][0] = naBliList[i][1];
                    naHitBli[naNHitBlPeak][1] = naBliList[i+1][0];
                    naHitPeak[naNHitBlPeak] = naPeakList[i];

                    // Get average baselines for each channel
                    // Calculate indexes for averaging a baseline
                    starti = naBliList[i][1];
                    endi = naBliList[i][1] - naNBLList[i][1] + 1;
                    // Fill the left of step baseline
                    for (chi = 0; chi < NMaxFiles; chi++){
                            bl = 0;
                            for (chj = starti; chj >= endi; chj--)
                                    bl += rawData[chi][chj];
                            bl = bl / (starti - endi + 1);
                            naHitChBl[chi][naNHitBlPeak][0] = bl;
                    }

                    // Calculate indexes for averaging a baseline
```

```
                    starti = naBliList[i+1][0];
                    endi = naBliList[i+1][0] + naNBLList[i+1][0] - 1;
                    // Fill the right of step baseline
                    for (chi = 0; chi < NMaxFiles; chi++){
                            bl = 0;
                            for (chj = starti; chj <= endi; chj++)
                                    bl += rawData[chi][chj];
                            bl = bl / (endi - starti + 1);
                            //printf("bl2=%f\n", 256-bl);
                            naHitChBl[chi][naNHitBlPeak][1] = bl;
                    }

                    // Calculate positions and event time
                    // Event time is the average difference between the baseline location before the step and
the peak location
                    naHitT[naNHitBlPeak] = (naHitPeak[naNHitBlPeak] + naHitBli[naNHitBlPeak][0] + 0.5) / 2;

                    // Event position is determined by the ratios
                    QTotal = 0;
                    failedHit = false;
                    for (chi=0; chi < NMaxFiles; chi++) {
                            QTotal += (naHitChBl[chi][naNHitBlPeak][0] - naHitChBl[chi][naNHitBlPeak][1]);
                            // If one of the channels registers a negative voltage drop, the signal must be
really messy, so
                            // simply don't record this hit
                            if (naHitChBl[chi][naNHitBlPeak][0] < naHitChBl[chi][naNHitBlPeak][1]){
                                    failedHit = true;
                            }
                    }
                    if (!failedHit){
                            naHitXY[naNHitBlPeak][0] = 2.0 * ((naHitChBl[0][naNHitBlPeak][0] -
naHitChBl[0][naNHitBlPeak][1]) + (naHitChBl[1][naNHitBlPeak][0] - naHitChBl[1][naNHitBlPeak][1])) / QTotal
- 1.0;
                            naHitXY[naNHitBlPeak][1] = 2.0 * ((naHitChBl[1][naNHitBlPeak][0] -
naHitChBl[1][naNHitBlPeak][1]) + (naHitChBl[3][naNHitBlPeak][0] - naHitChBl[3][naNHitBlPeak][1])) / QTotal
- 1.0;
                            naHitQT[naNHitBlPeak] = QTotal;
                            naNHitBlPeak++;
                    }
            }
        return naNHitBlPeak;
}
```

B   Coincidence Analysis

The following C++ code uses the ROOT framework (v 5.28) to implement the coincidence algorithm described above. Several data structures are used to maintain the large collection of parameters. *APFileType_t* contains the CEI apparatus calibration information, data file names, TOF windows, ion mass and charges, and momentum thresholds. *gCEIShot_t* contains the event information for a single laser shot: position and time both calibrated and uncalibrated, and the number of events in the laser shot. *gCEIMolecule_t* contains the event position and time (calibrated and uncalibrated) as well as mass, charge, and momentum for the 2 or 3 coincident ions making up a single molecule. The algorithm is capable of analyzing 2 and 3 ion coincidences depending on whether the global parameter *NIONS* is set to 2 or 3.

The entirety of this algorithm is contained within the C++ class *TCoincidenceAnalyzer*. The function *FindMoleculeWithMomentumCut* is called and given all the necessary parameters to analyze a single shot of data. Within this function the function *ProducePIM* is called to generate the matrix of hypothetical molecules. The parameters given to the first function are largely pointers to the data structures used to store the momentum information of the coincident ions as well as the zero TOF of each ion *t0[NIONS].*

B-I   Data Types

```
struct APFileType_t{
        Int_t AnalysisType;
        Double_t ChamberVoltage, ChamberLength;
        Double_t TimeShift, TimeMultiplier;
        Int_t Mass[NIONS], Charge[NIONS];
        Double_t TOFRanges[NIONS][2];
        Char_t RootDataFileName[200]; // Max file name length is 200 characters

        // For Momentum Cut Method
        Double_t FocusX, FocusY;
        Double_t BeamVelocity;
        Double_t PZLow, PZHigh, PXLow, PXHigh, PYLow, PYHigh;

        // For Molecule Center Method
        Double_t FocusRad;
        Double_t FocusTWidth;
};


// General structure containing the list of events for a single laser shot
struct gCEIShot_t {
        Int_t nshots; // number of shots entirely in the dataset
        Int_t wavenum; // Wave number in original dataset (a.k.a laser shot number)
        Int_t nevents; // number of events in current laser shot
        Float_t xu[MAXEVENTS], yu[MAXEVENTS], tu[MAXEVENTS]; // uncalibrated x,y,t information
        Float_t qt[MAXEVENTS]; // total charge output - used for calculating uncertainty
        Float_t x[MAXEVENTS], y[MAXEVENTS], t[MAXEVENTS]; // calibrated x,y,t [mm][mm][ns]
};


// General structure containing the list of ions in a molecule
struct gCEIMolecule_t {
        Int_t nshots; // number of shots entirely in the dataset
        Int_t wavenum; // Wave number in original dataset (a.k.a laser shot number)
        Int_t nevents; // number of events in current laser shot
```

```
        Float_t xu[NIONS], yu[NIONS], tu[NIONS]; // uncalibrated x,y,t information
        Float_t qt[NIONS]; // total charge output - used for calculating uncertainty
        Float_t x[NIONS], y[NIONS], t[NIONS]; // calibrated x,y,t [mm][mm][ns]
        Int_t m[NIONS], q[NIONS]; // mass and charge [amu]
        Float_t px[NIONS], py[NIONS], pz[NIONS]; // momentum [10e-23 kg m/s]
};
```

B-II   FindMoleculeWithMomentumCut

```
Int_t TCoincidenceAnalyzer::FindMoleculeWithMomentumCut(gCEIShot_t *shotevents, APFileType_t *anpa,
Double_t *t0, Double_t *y0l, Int_t nmions, Int_t lpions[][NIONS], Int_t mions[][NIONS][2], gCEIMolecule_t
*moldata, Int_t *degen){
        // This function uses the data in shotevents to generate molecular data
        // which is stored in moldata. It uses the momentum cut method to perform this action.
        // Returns 1 on success and 0 on failure

        // degen is a pointer for the degenerate ions (see ProducePIM for description)
        // If it is unused, then regular analysis is performed

        // Potential Ion List
        Int_t nmionspcut; // number of molecules to make the cut
        Int_t i, j;

        // Molecule position information
        Double_t xcpar1, xcpar2, ycpar1, ycpar2;
        Double_t tcpar1, tcpar2, tcpar3;
        Double_t tcres1, tcres2, tcres3;
        Double_t mct[NIONS], mct0[NIONS], mcq[NIONS];
        Double_t molx, moly, molt; // Molecule center coordinates [mm][mm][ns]

        Double_t pz, tz, t0z, pzsum;
        Double_t pzfac; // Defined using parts of the AP file = 1.6022e-19 * 2000.0 / 0.23 / 2.0 * 1e-9 /
1e-22; // gives pz in units of 10^-22 kgm/s
        Double_t px, pxsum, x;
        Double_t pxfac; // Defined using parts of the AP file = 0.0002545454 / 1e-9 / 1e-22 * 1.6605e-27;
// mm and ns multipliers, amu and correct unit
        Double_t py, pysum, y;
        Double_t pyfac; // Defined using parts of the AP file = 0.00026718 / 1e-9 / 1e-22 * 1.6605e-27; //
mm and ns multipliers. amu and correct unit

        // Ion momenta
        Int_t ionID;
        TVector3 vIonP[NIONS], vIonC[NIONS]; // Ion momentum values, ion coordinates [x,y,t] for the one
good molecule
        TVector3 vIonTP[NIONS], vIonTC[NIONS]; // Ion momentum and coordinates for current molecule
        TVector3 vMolC, vMolTC; // Molecule coordinates
        Double_t ionM[NIONS], ionQ[NIONS]; // Ion mass and charge

        pzfac = 1.6022e-19 * anpa->ChamberVoltage / anpa->ChamberLength / 2.0 * 1e-9 / 1e-22; // gives pz
in units of 10^-22 kgm/s
        pxfac = 0.001 / 1e-9 / 1e-22 * 1.6605e-27; // (mm into m) and (ns into s) multipliers, amu and
correct unit
        pyfac = 0.001 / 1e-9 / 1e-22 * 1.6605e-27; // (mm into m) and (ns into s) multipliers. amu and
correct unit

        if (degen){
                if(!ProducePIM(shotevents, anpa, &nmions, lpions, mions, degen))
                        return 0;
        } else {
                if(!ProducePIM(shotevents, anpa, &nmions, lpions, mions))
                        return 0;
        }

        nmionspcut = 0;
        // This loop runs through every combination of ions that could
```

```
        // reproduce the desired triatomic
        for (i=0; i<nmions; i++){
                pzsum = 0;
                pxsum = 0;
                pysum = 0;
                for (j=0; j<NIONS; j++){
                        ionID = lpions[mions[i][j][0]][mions[i][j][1]]; // 0, 1, 2 for first, second,
third - (O,C,S)
                        tz = shotevents->t[mions[i][j][0]];
                        t0z = t0[ionID];
                        pz = pzfac * anpa->Charge[ionID] * (pow((t0z - anpa->TimeShift),2) - pow((tz -
anpa->TimeShift),2)) / (tz - anpa->TimeShift);
                        pzsum += pz;

                        x = shotevents->x[mions[i][j][0]];
                        px = anpa->Mass[ionID] * pxfac * (x - anpa->FocusX) / (tz - anpa->TimeShift);
                        pxsum += px;

                        y = shotevents->y[mions[i][j][0]];
                        py = anpa->Mass[ionID] * pyfac * (y - y0l[ionID]) / (tz - anpa->TimeShift);
                        pysum += py;

                        vIonTP[ionID].SetXYZ(px, py, pz);
                        ionM[ionID] = anpa->Mass[ionID];
                        ionQ[ionID] = anpa->Charge[ionID];
                }

                if ((pzsum >= anpa->PZLow && pzsum < anpa->PZHigh) &&  (pxsum >= anpa->PXLow && pxsum <
anpa->PXHigh) && (pysum >= anpa->PYLow && pysum < anpa->PYHigh)){
                        nmionspcut++;
                        if (nmionspcut > 1)
                                return 0;
                        moldata->wavenum = shotevents->wavenum;
                        moldata->nevents = NIONS;
                        moldata->nshots = shotevents->nshots;
                        for (j=0; j<NIONS; j++){
                                ionID = lpions[mions[i][j][0]][mions[i][j][1]]; // 0, 1, 2 for first,
second, third - (O,C,S)
                                moldata->qt[ionID] = shotevents->qt[mions[i][j][0]];
                                moldata->t[ionID] = shotevents->t[mions[i][j][0]];
                                moldata->tu[ionID] = shotevents->tu[mions[i][j][0]];
                                moldata->x[ionID] = shotevents->x[mions[i][j][0]];
                                moldata->xu[ionID] = shotevents->xu[mions[i][j][0]];
                                moldata->y[ionID] = shotevents->y[mions[i][j][0]];
                                moldata->yu[ionID] = shotevents->yu[mions[i][j][0]];
                                moldata->px[j] = vIonTP[j].X();
                                moldata->py[j] = vIonTP[j].Y();
                                moldata->pz[j] = vIonTP[j].Z();
                                moldata->m[j] = ionM[j];
                                moldata->q[j] = ionQ[j];
                        }
                }
        }
        return nmionspcut;
}
```

B-III ProducePIM

```
Int_t TCoincidenceAnalyzer::ProducePIM(gCEIShot_t *shotevents, APFileType_t *anpa, Int_t *nmions, Int_t
lpions[][NIONS], Int_t mions[][NIONS][2], Int_t *degen){
        // Produces the Potential Ion Matrix used to create hypothetical molecules
        // Some care needs to be taken here in the case of ion degeneracy
        // In the case of N2O, analysis of the (a,a,b) channel will produce
        // twice the number of hypothetical molecules by matching the two N+ events
        // as the first and second ion twice when they are otherwise indistinguishable.
```

```cpp
        // degen vector is a two element vector containing the identities of the ions that have
degeneracies
        // I.e. if we are looking at the (1,1,1) of N2O, the first two ions are degenerate so the vector
would be [0 1]
        // For (1,1,1) of CS2 or NO2, we would use [0 2]
        // Note: the ions mustn't necessarily have exact TOF overlap for this to work. The degeneracy will
only
        // be effective in the overlapping region where an event CAN be identified as two of the same
ions.

        // Degeneracy counters
        Int_t di, dii, diii;
        Bool_t hasDIon1, hasDIon2;
        Bool_t ldeg[NMPions];
        Bool_t hasPIon[NIONS];

        // Potential Ion List
        Int_t npions[NMPions];
        Int_t nlpions;

        // Match List
        Int_t i, j, ii, jj, iii, jjj;
        Int_t nmionspcut;
        Int_t nmaxedevents;

        nmaxedevents = 0;
        nlpions = 0;
        npions[0] = 0;
        for (i=0; i < shotevents->nevents; i++){
                // Reset degenerate list
                hasPIon[degen[0]] = kFALSE;
                hasPIon[degen[1]] = kFALSE;
                // Search mqtof ranges for match
                for (j=0; j<NIONS; j++){
                        if (shotevents->t[i] >= anpa->TOFRanges[j][0] && shotevents->t[i] < anpa-
>TOFRanges[j][1]){
                                // Ion matches with mqtof range at index j
                                lpions[nlpions][npions[nlpions]] = j;
                                npions[nlpions]++;
                                hasPIon[j] = kTRUE;
                        }
                }
                // Flag degeneracy
                if (hasPIon[degen[0]] && hasPIon[degen[1]])
                        ldeg[nlpions] = kTRUE;
                else
                        ldeg[nlpions] = kFALSE;

                nlpions++;
                npions[nlpions] = 0;
        }
        // THIS NEXT SECTION should be done with a recursive function
        // for generality
        // Generate potential matches by searching for combinations
        // that make O+, C+, S+
        *nmions = 0;
        for (i = 0; i < nlpions; i++){
                for (j = 0; j < npions[i]; j++){
                        // Get First ion
                        for (ii = i+1; ii < nlpions; ii++){
                                for (jj = 0; jj < npions[ii]; jj++){
                                        // Check if second ion is different from first ion
                                        if (lpions[ii][jj] != lpions[i][j]){
#if NIONS==3
                                                for (iii = ii+1; iii < nlpions; iii++){
                                                        for (jjj = 0; jjj < npions[iii]; jjj++){
                                                                // Check if third ion is different from
first and second ion
```

```c
                                    if ((lpions[iii][jjj] != lpions[i][j]) &&

                                    (lpions[iii][jjj] != lpions[ii][jj])){
                                            // Check for degeneracies
                                            if ((ldeg[i] && ldeg[ii] &&

                                                    (ldeg[i] && ldeg[iii] &&

                                                    (ldeg[ii] && ldeg[iii]

(degen[1] == lpions[i][j]) && (degen[0] == lpions[ii][jj])) ||
                                                            // degenercy
(degen[1] == lpions[i][j]) && (degen[0] == lpions[iii][jjj])) ||
                                                            continue;
&& (degen[1] == lpions[ii][jj]) && (degen[0] == lpions[iii][jjj]))) {
                                            }
detected
                                            // Add all ions at once
                                            mions[*nmions][0][0] = i;
                                            mions[*nmions][0][1] = j;
                                            mions[*nmions][1][0] = ii;
                                            mions[*nmions][1][1] = jj;
                                            mions[*nmions][2][0] = iii;
                                            mions[*nmions][2][1] = jjj;
                                            *nmions = *nmions + 1;

                                            // Maximum reached, return fail
                                            if (*nmions == NMPions)
                                                    return 0;
                                    }
                            }
                    }
#elif NIONS==2

                    // Add all ions at once
                    mions[*nmions][0][0] = i;
                    mions[*nmions][0][1] = j;
                    mions[*nmions][1][0] = ii;
                    mions[*nmions][1][1] = jj;
                    *nmions = *nmions + 1;

                    // Maximum reached, return fail
                    if (*nmions == NMPions)
                            return 0;

#endif
                                }
                            }
                        }
                    }
            }
        return 1;
}
```

72

C   Coincidence Algorithm Addendums

The following sections are addendums to the coincidence algorithm and explain the method by which degenerate mass to charge ratios are handled and the improvement from using dimension specific momentum discrimination.

C-I   Degeneracies

As mentioned above, identifying events as specific ions is complicated by the mass to charge ratio degeneracy common to molecules of interest (OCS, for example). The developed algorithm circumvents this issue by considering all ion identities and testing the net momentum for all hypothetical molecules. For an event identified as both O^{n+} and S^{2n+} produced from the (1,1,2) channel of OCS, a different momentum is calculated for either ion identity, producing a different net momentum for each hypothetical molecule which in turn discriminates between the false ion identity. For a molecule with identical atoms like CO_2, it is clearly not possible to distinguish which terminal atom is which. Due to the symmetry of the molecule, such identification is superfluous, and one would define forwards and backwards TOF ranges to separate the ion identities. However, in the case of N_2O, the TOF overlap is not as clear cut. As a result, it is necessary to have two overlapping TOF windows for the central and terminal nitrogen ions. When such TOF windows are used in the described algorithm, two hypothetical molecules would be produced for every one coincident set of ions since the real terminal and central nitrogen ions could be falsely identified as central instead of terminal and vice versa. To circumvent this, ion identities in the PIM are first flagged as degenerate if two identities of the same mass and charge are registered for a single event. Then, if two hypothetical molecules are generated using two degenerate ion identities, only the first one is considered.

      An example of this system in use can be seen in the TOF spectra of the selected (1,1,2) channel of an N_2O CEI experiment. Figure 44 shows the TOF for the three ions when the same full range is used for the central and terminal nitrogen ions while Figure 45 shows the TOF when a narrow range is used for the central ion and a broad range for the terminal ion. The latter technique is more appropriate as the terminal nitrogen is expected to arrive far from the zero time of flight while the opposite is true for the central nitrogen. Since this technique assumes a strict TOF window for each ion, events arriving outside the ranges (central nitrogen arriving in the terminal range and vice versa) will pass the coincidence test with undetectable false ion identification. A zero count zone is observed between the peak regions for the central and terminal ions in Figure 44 and Figure 45, indicating that very few central ions are passing through the central TOF window into the terminal TOF window and vice versa.

      This issue cannot be resolved through further calculations of the measured momenta since extra information about ion position (within the molecule geometry) is needed. The geometry reconstruction step in the CEI technique does, however, provide such necessary ion position information. It is proposed that coincidence momenta could be passed through the geometry reconstruction algorithm to test for false ion identities. If the algorithm fails to reconstruct the molecular geometry, it is possible the central and terminal nitrogen ions in the case of N_2O were identified as central and terminal and merely

switching the identities could yield a successful reconstruction. Such a scheme would filter out false ion identities produced from degeneracies in current CEI technology.
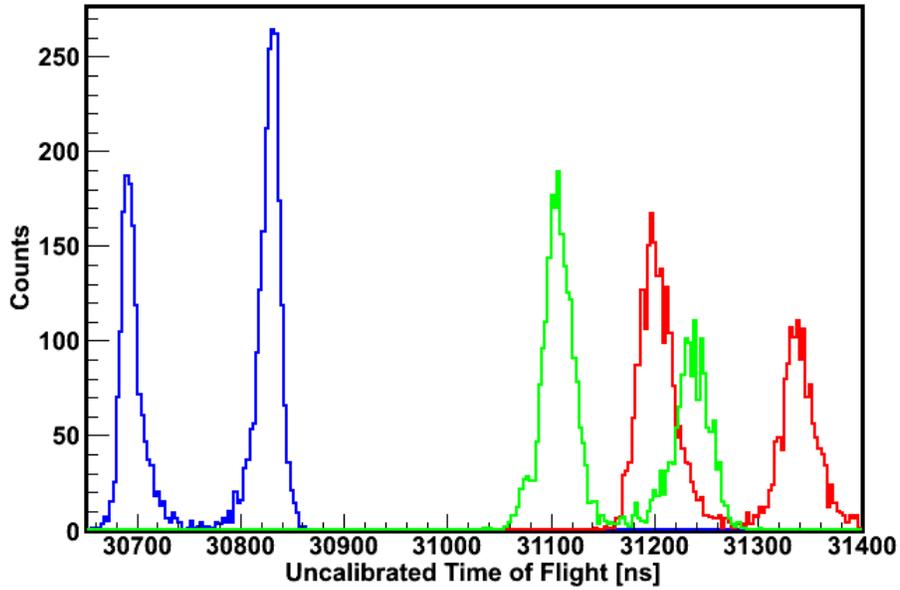

Figure 44 – Time of flight distribution for O^{2+} (blue) and N^+ ions (green and red) where the same TOF window is defined for the central and terminal N^+ ions.
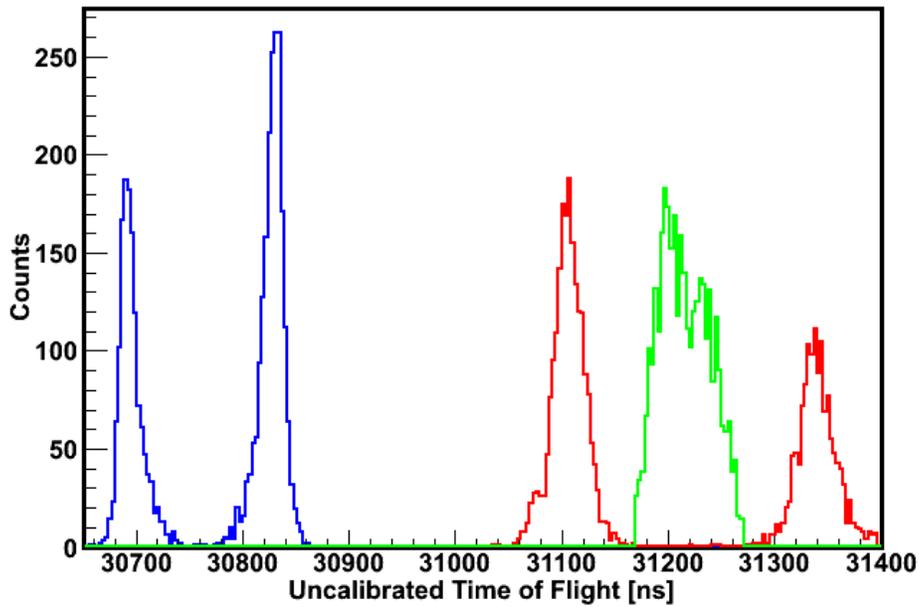

Figure 45 – Time of flight distribution for O^{2+} (blue) and N^+ ions (green and red) where a narrow TOF window (green) is used for the central N^+ ion and a wide TOF window (red) is used for the terminal N^+ ion.

C-II   Dimension Specific Momentum Discrimination

Performing momentum discrimination with custom thresholds for each dimension was used to retrieve true coincidence events in the above described OCS experiments. The previously employed technique of discriminating against the magnitude of the momentum sum is contrasted with this method in the following analysis. Additionally, the event numbers available to the coincidence method are highlighted.

During the analysis of a CEI dataset, the coincidence algorithm produces several hypothetical molecules for each set of events (or laser shot in the case of laser initiated CEI). To adjust the calibration parameters and ensure the momentum discrimination is accurately selecting true coincidences, five histograms are used as diagnostics: the momentum sum of hypothetical molecules' coincident fragment ions in the z direction; that in the x direction; that in the y direction; the magnitude of the vector momentum sum; and the total kinetic energy released. These histograms are first populated with the momenta from every hypothetical molecule the algorithm generates for a specific channel. A second set is populated using those molecules that pass the momentum sum threshold in the z-direction. A third set is populated using those molecules that pass the momentum sum threshold in the z-direction and x-direction. A final set is populated for those molecules passing momentum discrimination in all three directions. This set of 20 histograms is shown below for a one hour acquisition data set of OCS in the ALLS' CEI apparatus with operation parameters set such that 5 ions were detected on average for each laser shot and TOF windows set for the (1,1,2) channel.
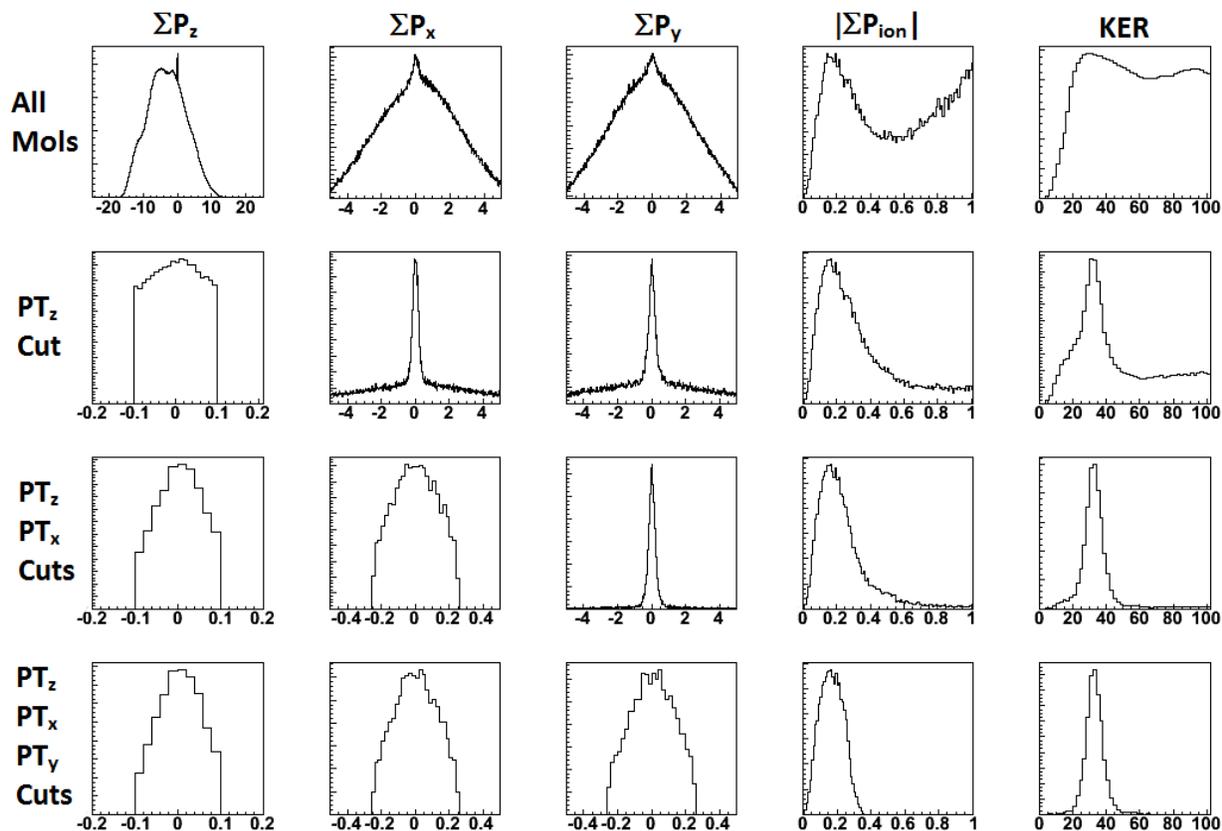
Figure 46 – Momentum discrimination diagnostic histograms for 5 ions per laser shot OCS dataset
From left to right: Z momentum sum; X momentum sum; Y momentum sum; magnitude of momentum sum;
KER. From top to bottom: all hypothetical molecules; post Z discrimination; post Z and X discrimination; post Z, X, and Y discrimination. Momentum units of [10^{-22}kg m/s], KER in [eV].

The three figures in the top left show small but distinguishable peaks of true coincidences set atop a large background of false coincidences. Successive discrimination of momentum sum eliminates the false coincidence background in all 5 histograms. The X-Y-Z momentum histograms ensure proper calibration in all three directions – a benefit not available if only the net momentum is considered. The first net momentum histogram (fourth from the left in the top row) reveals a large hump near zero corresponding to true coincidences. The rising edge results from false coincidences and continues to grow well beyond the edge value of 1×10^{-22}kg m/s. From this histogram alone it is difficult to choose an appropriate threshold for the momentum sum and no estimate of true to false coincidences can be made. The visible successive reduction of background in each histogram as a result of dimensional momentum discrimination does offer such a perspective, ensuring appropriate thresholds are being set. The last distribution of energy is valuable when molecules with similar masses are studied (such as N_2O) and when several ions with zero momentum strike within a TOF window. In the former case, if the TOF is slightly miscalibrated, it is possible that the net momentum of coincident ions in the case of N_2O is zero when terminal nitrogen events are identified as oxygen events and vice versa. This results in normal distributions of momentum but extraordinary measures of KER. In the latter case, zero momentum ions

always net zero momentum whether or not they are true coincidences. In the KER spectra, this results in peaks near zero KER, indicating false coincidences have passed the momentum discrimination step.

For higher count rate data, the need for the above method is further highlighted. Below is the same set of diagnostic figures from a dataset with on average 10 ions per laser shot. Identifying true coincidence peaks in the X,Y,Z momentum sum histograms is more difficult than with the previous case. The true coincidence peak in the momentum sum histogram is nearly level with the false coincidence slope. To confidently discriminate against false coincidences using just this histogram, one would need to set a threshold below 0.2×10^{-22}kg m/s. Using dimensional momentum discrimination leaves momentum sum values up to 0.3x10-22kg m/s, indicating that the former technique reduces the overall count rate.
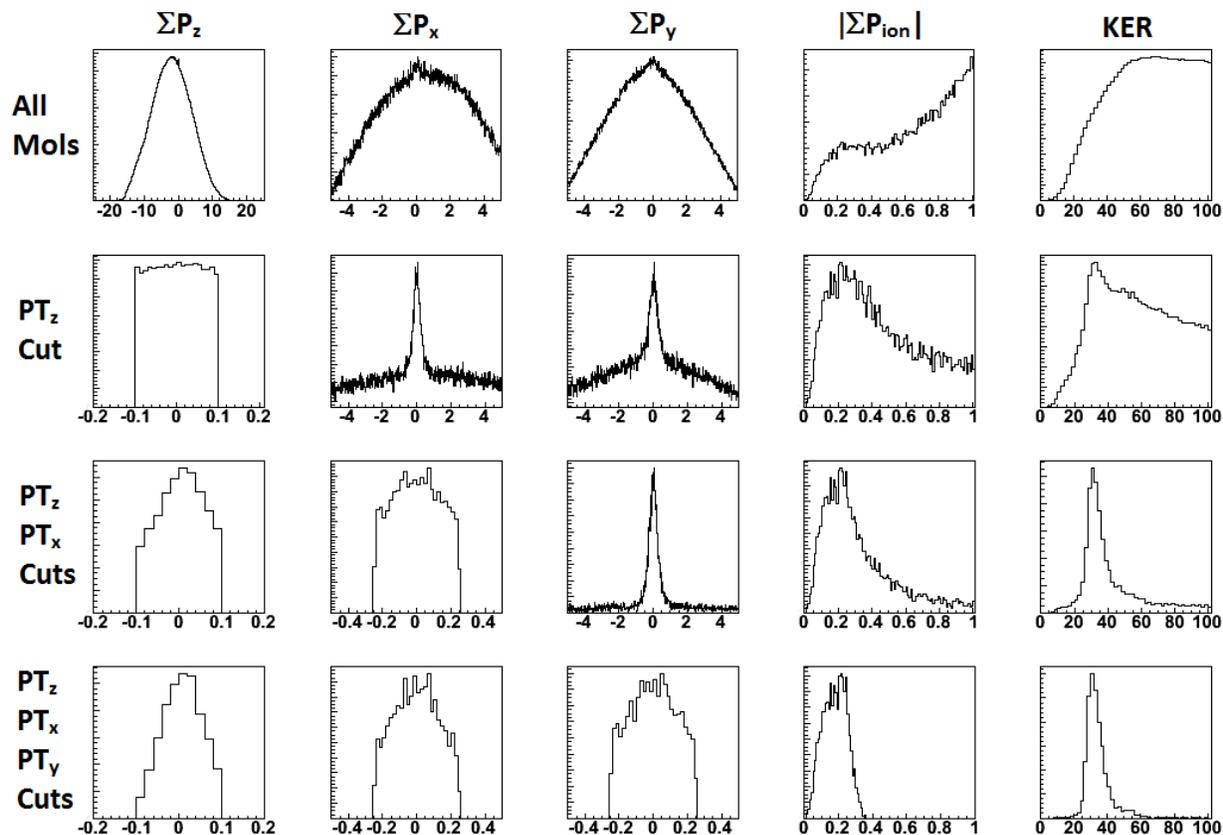

Figure 47 – Momentum discrimination diagnostic histograms for 10 ions per laser shot OCS dataset

To quantify the effect of considering a number of hypothetical molecules in the above experiment, a distribution of the number of hypothetical molecules was generated (**Error! Reference ource not found.**). The most probable value in this spectrum is 2 hypothetical molecules. The previously described scheme would, on average, only consider half of these molecules. For three, four, and five molecules, the simple scheme would only consider one third, one fourth, and one fifth of the molecules. With this data set, upwards of 200 hypothetical molecules were generated for a single laser shot, suggesting multiple molecules should pass the momentum discrimination. However, of the 3408 true

coincidences, only 2 laser shots produced more than one molecule that passed the 3-dimensional momentum discrimination.
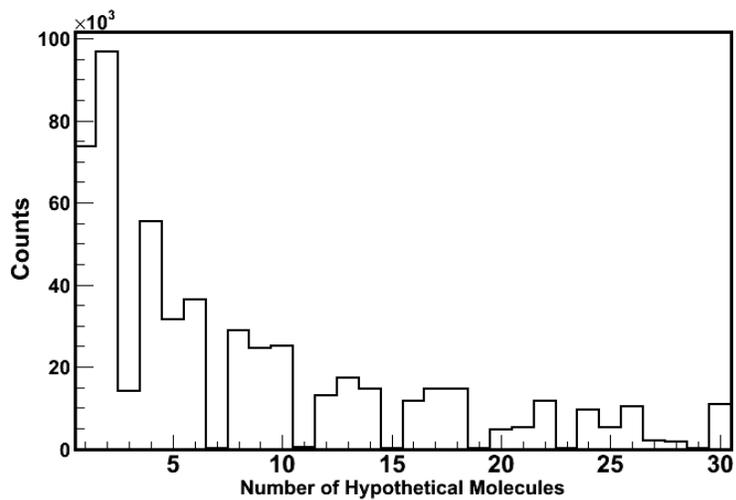

Figure 48 – Number of hypothetical molecules generated with 10 ions per laser shot OCS data set.