UPPSALA
UNIVERSITET

UPTEC X 14 025
Examensarbete 30 hp
September 2014

Single-molecule X-ray free-electron laser imaging

Interconnecting sample orientation with explosion data

Christofer Östlin

UPPSALA
UNIVERSITET

Molecular Biotechnology Programme

Uppsala University School of Engineering

UPTEC X 14 025	**Date of issue 2014-08**

Author

Christofer Östlin

Title (English)

Single-molecule X-ray free-electron laser imaging – Interconnecting sample orientation with explosion data

Title (Swedish)

Abstract

Single-molecule serial femtosecond imaging is a relatively new, emerging discipline of X-ray crystallography eliminating the need of a sample crystal. In this work we used molecular dynamics simulations to determine if the explosion pattern of a molecule undergoing such analysis can provide information regarding its initial spatial orientation.

(Full abstract on page vii)

Keywords

X-ray, free-electron laser, XFEL, diffraction analysis, structure determination, nanocrystal, molecular dynamics, GROMACS, biomolecular imaging, ubiquitin, trajectory, explosion

Supervisors

Dr. Carl Caleman
Dr. Nicusor Timneanu
Department of Physics and Astronomy, Uppsala University

Scientific reviewer

Prof. David van der Spoel
Department of Cell and Molecular Biology, Uppsala University

Project name	Sponsors
Language **English**	Security
ISSN 1401-2138	Classification
Supplementary bibliographical information	Pages **52**

Biology Education Centre Biomedical Center Husargatan 3 Uppsala
Box 592 S-75124 Uppsala Tel +46 (0)18 4710000 Fax +46 (0)18 471 4687

SINGLE-MOLECULE X-RAY FREE-ELECTRON LASER IMAGING

Interconnecting sample orientation with explosion data

POPULÄRVETENSKAPLIG SAMMANFATTNING

CHRISTOFER ÖSTLIN

Att kunna bestämma ett proteins struktur är fundamentalt för att kunna förstå dess funktion. Idag används vanligen så kallad röntgenkristallografi för att ge en uppfattning om hur ett protein ser ut på atomär skala. Metoden går ut på att proteinet först dupliceras och arrangeras i en makroskopisk kristall och sedan belyses med röntgenstrålning medan provet roteras. Strålarna som reflekteras från kristallen avslöjar dess struktur och kan därför översättas till en tredimensionell bild av molekylen.

Tyvärr är många proteiner besvärliga att kristallisera och är därför mycket svåra att undersöka med konventionell röntgenkristallografi. I de fallen kan man istället tänka sig att skjuta röntgenljus på ett *enstaka* protein. Problemet är då att den kraftiga strålningen som behövs genast får proteinet att explodera och möjligheten att kontrollera och rotera det mikroskopiska provet förloras. Den resulterande bilden blir alltså bara tvådimensionell. Genom att repetera ett sådant experiment många gånger kan visserligen bilder ur olika vinklar erhållas, men deras relation sinsemellan förblir okänd. Om förhållandet var känt skulle bilderna kunna kombineras för att återspegla proteinets tredimensionella struktur.

I det här arbetet undersöktes möjligheten att avläsa varje enskild bildvinkel utifrån hur proteinet exploderar – och därmed direkt relatera bilderna till varandra. De datorsimuleringar som utfördes påvisar att detta, åtminstone till viss del, är teoretiskt möjligt och tekniken har därför potential att kunna användas i praktiken så småningom. Metoden skulle då kunna leda till förbättringar inom den redan befintliga röntgenkristallografin, men också underlätta strukturbestämning av livsviktiga, icke-kristalliserbara proteiner.

EXAMENSARBETE 30 HP
CIVILINGENJÖRSPROGRAMMET I MOLEKYLÄR BIOTEKNIK
UPPSALA UNIVERSITET, JUNI 2014

SINGLE-MOLECULE X-RAY FREE-ELECTRON LASER IMAGING

Interconnecting sample orientation with explosion data

CHRISTOFER ÖSTLIN

Abstract

X-ray crystallography has been around for 100 years and remains the preferred technique for solving molecular structures today. However, its reliance on the production of sufficiently large crystals is limiting, considering that crystallization cannot be achieved for a vast range of biomolecules. A promising way of circumventing this problem is the method of serial femtosecond imaging of single-molecules or nanocrystals utilizing an X-ray free-electron laser.[1]–[3]

In such an approach, X-ray pulses brief enough to outrun radiation damage and intense enough to provide usable diffraction signals are employed. This way accurate snapshots can be collected one at a time, despite the sample molecule exploding immediately following the pulse due to extreme ionization. But as opposed to in conventional crystallography, the spatial orientation of the molecule at the time of X-ray exposure is generally unknown. Consequentially, assembling the snapshots to form a three-dimensional representation of the structure of interest is cumbersome, and normally tackled using algorithms to analyze the diffraction patterns.[4]

Here we explore the idea that the explosion data can provide useful insights regarding the orientation of ubiquitin, a eukaryotic regulatory protein. Through two series of molecular dynamics simulations totaling 588 unique explosions, we found that a majority of the carbon atoms prevalent in ubiquitin are directionally limited in their respective escape paths. As such we conclude it to be theoretically possible to orient a sample with known structure based on its explosion pattern. Working with an unknown sample, we suggest these discoveries could be applicable in tandem with X-ray diffraction data to optimize image assembly.

Master thesis
Department of Physics and Astronomy, Uppsala University
Sweden, 2014

Table of Contents

Abbreviations

ESI	Electrospray Ionization
FWHM	Full Width at Half Maximum
GROMACS	Groningen Machine for Chemical Simulations
MD	Molecular Dynamics
PDB	Protein Data Bank
SDE	Standard Deviation Ellipse
XFEL	X-Ray Free-Electron Laser

INTRODUCTION

The seemingly inconsequential idea of bombarding a sample crystal with X-rays to determine the atomic structure of its building blocks was first conceived in the early 20th century. The technique, which came to be known as *X-ray crystallography*, proved to be a fundamental discovery to the fields of physics, chemistry, materials science and structural biology. After years of further development and refinement it became widely adapted and is still today the predominant method for structure determination of molecules.

When illuminating a crystal with X-rays, the photons are diffracted by the crystalline atoms (similar to when passing through a grating) and thus give rise to a pattern. This pattern contains detailed information regarding the atomic positions and can therefore be translated into a two-dimensional image of the structure of the sample. By rotating the sample during the analysis, a large number of such 2D snapshots from different angles can be collected. Once a sufficient number of images have been obtained, they can be used to trace back and establish an accurate model of the desired three-dimensional structure. All of this is made possible due to the properties of the crystal, which allows it to amplify the diffraction signals while withstanding the harmful X-rays.

Due to its immense versatility and accuracy, X-ray crystallography is highly unlikely to be replaced anytime soon. However, despite its affluent properties, this praised technique is not entirely drawback-free – one issue being its dependence on crystals. While most molecules indeed can be crystallized, there are those where this process has proven to be remarkably troublesome. One such group of exceptions is certain cellular membrane proteins whose structures naturally are of substantial interest, given their oftentimes life-essential functions.

Recent findings[2] indicate that a contingent solution to this problem lie in the application of X-ray free-electron lasers (XFELs). Because these X-ray sources can produce ultra-short, high-intensity pulses they enable new ways of diffraction-based structural studies, not previously possible using the conventional synchrotrons.[5] By blasting small samples with pulses short enough to pass through and retain the structural information before the molecule(s) deteriorates, the need for a crystal may be bypassed.[1]

In this thesis we aim to investigate the possibilities of circumventing one of the obstacles presented when implementing such approach, namely the inability to rotate the sample. With this new method a target sample will explode as a direct consequence of the extreme ionization immediately following the X-ray pulse, rendering any subsequent rotational study inutile. Instead one has to introduce another, identical sample of different orientation and, by repeating the process, gather the necessary diffraction patterns one by one. Supposing that the spatial orientation of each sample is random and unknown (some research has gone into methodically aligning molecules)[6] we have in this study conducted a series of molecular dynamics (MD) simulations to determine whether the explosion event itself can be used as a rough map, allowing for the initial orientation to be reversely reconstructed.

THEORY

When a molecule is subjected to an XFEL pulse, it rapidly becomes ionized. As a result, the charged particles exhibit strong, counteracting electrostatic forces that cause the explosion. With molecular dynamics, we can simulate this event and investigate whether the atoms are restricted in their respective escape paths during the explosion. This section seeks to explain the inner workings behind these methods and phenomena, before going into the details of the experimental methods and setup.

MOLECULAR DYNAMICS

GROMACS, *Groningen Machine for Chemical Simulations*, is the open-source molecular dynamics program package used in this study.[7] It implements the most common of MD approaches, namely the three-step process of:

1. Determining the starting conditions of the molecular system.
2. Computing the forces acting on all particles.
3. Updating the current system configuration.

Once done, the system is considered to have moved one step in time and the procedure is repeated. Together, these time steps constitute what we call a simulation and the procedure is sometimes referred to as the global MD algorithm.

The *starting conditions* are either set-up manually by the user or determined computationally by built-in functions. Typically data such as atomic positions can be obtained, for example by using a structurally known protein downloaded from the RCSB Protein Data Bank,[8] while atomic velocities often are generated stochastically from a Maxwellian distribution given a certain absolute temperature. Initial conditions also include a pre-defined mathematical function called *force field*, which describe potential interactions based on atomic species, intramolecular configurations and positions of different molecules. There are several generalized ways to construct a force field for MD, depending on the nature of the simulation.[9] Regardless of how the initial conditions are obtained, they are needed to proceed to the next step in the simulation process.

The *forces* are then computed classically. This means that quantum effects are neglected and it is assumed that classical mechanics are sufficiently accurate. This is important to keep in mind, especially when simulating small systems where quantum phenomena become important. In the classical approximation, the force $\boldsymbol{F}_i$ on any atom i is calculated as

$$\boldsymbol{F}_i = -\frac{\partial V}{\partial \boldsymbol{r}_i} \tag{1}$$

where V is potential function and $\boldsymbol{r}_i$ is the position of atom i. In GROMACS, the potential function in equation (1) contains different terms to consider multiple types of force contributions. Calculations of these contributions are based upon the force field and can therefore vary depending on which force field is chosen.

Lastly, the *system configuration* is updated by computing the motion of each particle in the given time frame. Again, a classical approach is employed utilizing Newton's second law of motion

$$\frac{\boldsymbol{F}_i}{m_i} = \frac{d^2\boldsymbol{r}_i}{dt^2} \tag{2}$$

where m_i is the mass of atom i. By solving the differential equation numerically with values of $\boldsymbol{F}_i$ found in the previous step, the updated individual position of each atom after a short time interval dt is obtained. These positions are then saved as the new starting conditions and the whole process is repeated a desired number of times.

Throughout each run, GROMACS produces documents containing relevant data output. These documents are highly customizable to allow for versatility without affecting the computational time and, in particular, the memory usage. This becomes particularly important whenever it is necessary to simulate a large molecular system multiple times. One must therefore plan the data analysis carefully before performing a larger number of excessive simulations and adjust the settings accordingly.

IONIZATION

The ionization of a protein when exposed to an X-ray pulse is a direct consequence of the phenomenon known as *the photoelectric effect*. It refers to an event of energy transfer from a photon to an electron, which then is emitted from the atom (see Figure 1). The energy needed to release the now-called photoelectron and its kinetic energy corresponds to the energy of the photon.

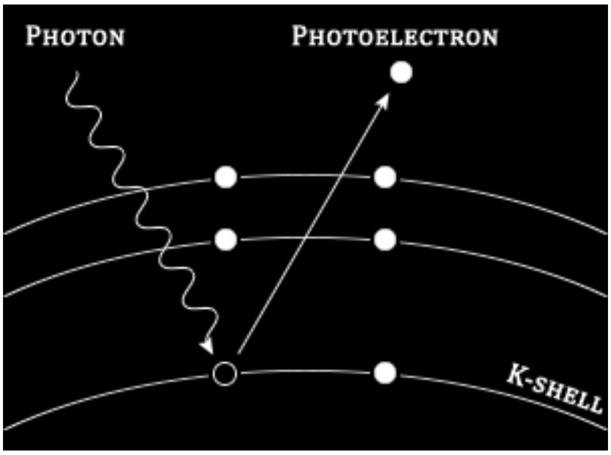

Figure 1: The photoelectric effect. A photon is absorbed by a K-shell electron, which in turn is ejected from the atom.

This leaves the atom in an unfavorable state, since there is an electron vacancy in the K-shell. Letting another electron of higher energy level fill the vacancy adjusts this and results in a release of excess energy. While the energy can be emitted as a photon, it can also be transferred to another electron, which then leaves the system. The latter is known as *the Auger effect*[10] (see Figure 2 below) and therefore the ejected electron is called an Auger electron.

The life span of a K-shell vacancy in biologically relevant atoms such as carbon, oxygen, nitrogen and sulfur, is along the time scale of 10 fs. This is similar to the durations of the XFEL pulses, making the vacancy important in the understanding of sample ionization during exposure.[11]

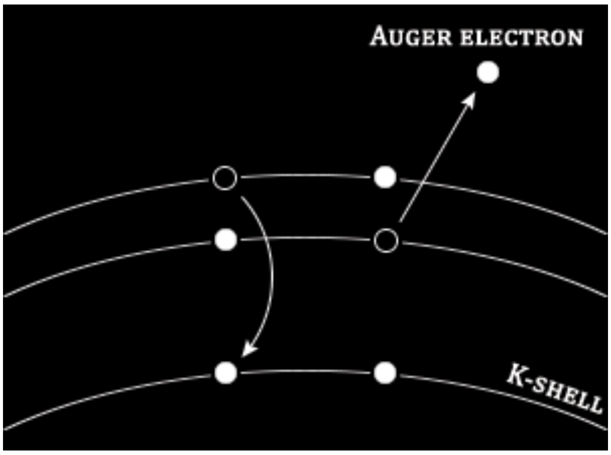

Figure 2: The Auger effect. A secondary electron is released as another electron fills the K-shell vacancy following the photoelectric effect; hence the atom becomes further ionized.

These are the primary means of ionization in molecules as they are irradiated with X-rays of energies achieved in free-electron lasers. However, the escaping electrons may then, in turn, cause further ionization by transferring kinetic energy to other electrons in the sample, liberating them from their respective shells. This inelastic scattering causes the number of free electrons to increase and ultimately give rise to a cascading effect.[12]

In GROMACS, the effects of X-ray exposure to a system are simulated as ionization of the involved atoms through the photoelectric effect and the Auger effect based on atomic cross-sections. Moreover, the sequence in which the atoms are ionized is based upon a single integer. As MD simulations are deterministic – meaning that the same ionization sequence always results in the same outcome – it is crucial that this integer is chosen randomly and uniquely in order to collect meaningful data.

EXPLOSION

As more and more atoms within the system become ionized, opposing Coulomb forces arise and eventually cause a molecular explosion. Figure 3 depicts ubiquitin undergoing such an event.

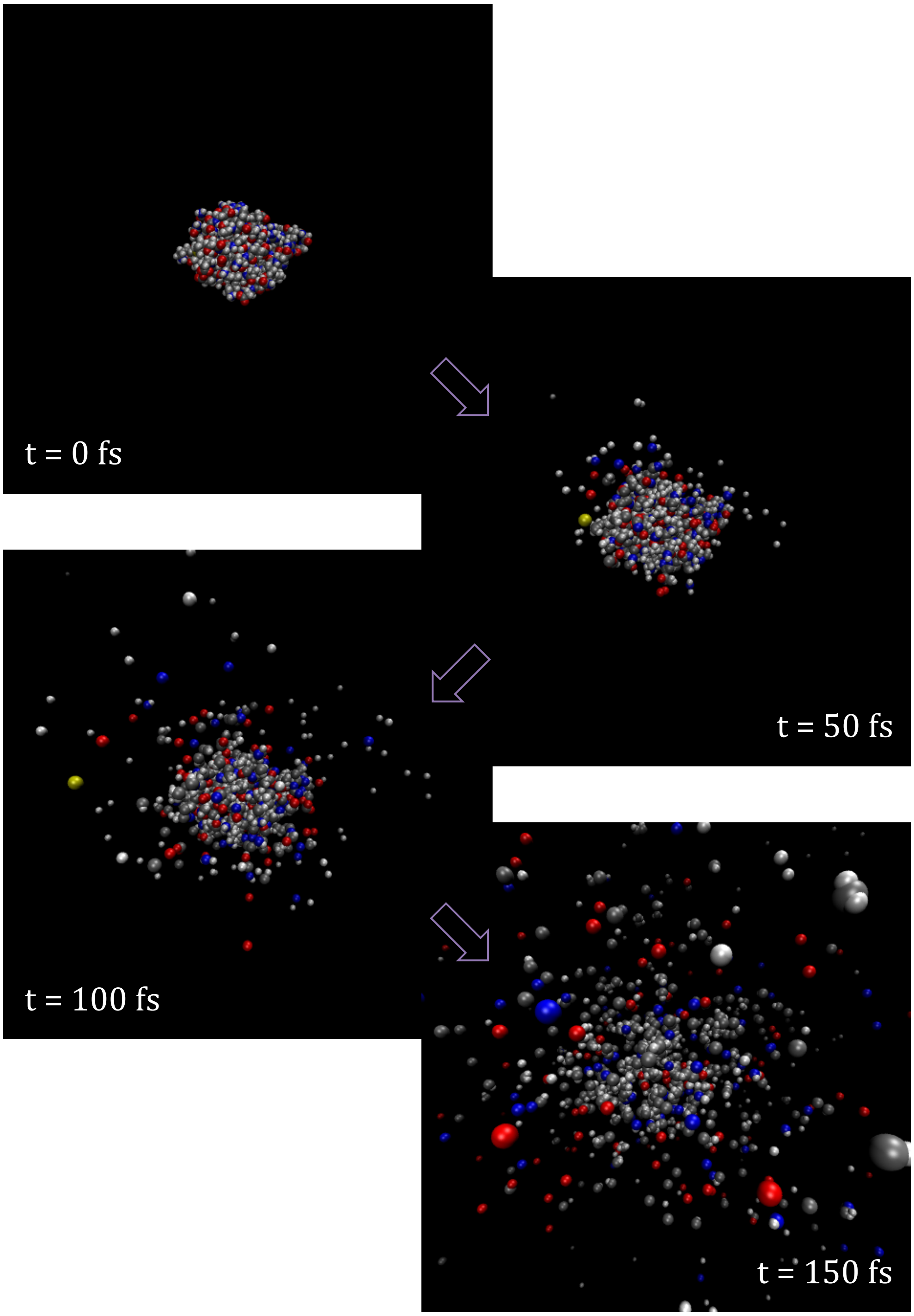

Figure 3: Molecular explosion. Four still frames showing the first 150 fs effects on ubiquitin following intense ionization from an X-ray pulse. The simulated pulse consisted of $2 \cdot 10^{12}$ photons at 8 keV, was Gaussian shaped with a full width at half maximum (FWHM) of 100 fs and was focused on a 100 nm in diameter spot.

This will inevitably happen to any single molecule exposed to an intense XFEL pulse. If the pulse is too long, the structural changes of the sample will be reflected in the diffraction pattern and result in a loss in resolution of the reconstructed image. By shortening the pulse to timescales of a few femtoseconds these effects can be outrun. It is in these settings we aim to investigate whether the explosion data can be helpful to determine the sample orientation in space.

METHODS

Data was collected through a series of computer simulations mimicking the conditions of an X-ray diffraction experiment. As the sample exploded, we tracked the trajectories of each of the atoms present in an attempt to map significant explosion patterns – independent of the initial orientation of the molecule. The idea being that such patterns could provide a valuable tool in determining the spatial orientation whenever it is not known, such as in electrospray single-molecule X-ray imaging.[1]

All simulations were performed using the OPLS-AA/L force field[13] and TIP4P water.[14] Noteworthy is also the fact that we used the older version 3.3.3 of GROMACS to gather all data presented. While newer iterations are computationally superior due to incorporation of multi-core parallelization and other enhancements, they lack the option to ionize systems as described above.

MODEL PROTEIN

For this study the human regulatory protein ubiquitin was chosen as model sample. Ubiquitin is found in eukaryotes where it acts as a post-translational modifier of other proteins, controlling mechanisms such as proteolysis and cellular location.[15] At a mere 76 amino acid residues it is fairly small, and combined with a highly conserved and well-known structure it seemingly is a suitable candidate for our purposes. Its structure (see Figure 4) was originally obtained from Vijay-Kumar *et al.*[16]

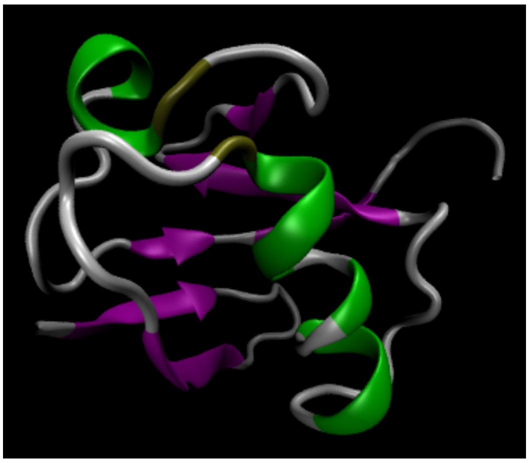

Figure 4: Ubiquitin. Representation of the proposed 3D structure of the studied protein. α-helices are shown in green and β-sheets in purple.

We wanted our model protein to be relatively small since our aim was to track each of the atoms following the explosion. A larger sample would necessitate an extended simulation time without generating data significant enough to justify such extension. It has also been suggested that the actual explosion event changes as the sample size increase, going from an explosion solely driven by Coulomb forces to one of a hydrodynamic nature.[17] Moreover, using a small molecule aids a potential transfer into a real setting where differentiating the atoms from each other becomes progressively more difficult as the number of atoms rise. For instance, ubiquitin has a single sulfur atom that could potentially reveal information about the initial orientation following explosion. Such analysis would be considerably harder in the case of biomolecules containing multiple sulfurs.

In a previous study by Marklund *et al.*, ubiquitin was also shown to be rather stable under vacuum conditions.[18] This is significant since it obviously affects the reproducibility of the explosion. In their investigation, MD-simulations were employed to process the protein by adding a thin layer (3 Å) of surrounding water and letting it settle around the protein as it would in vacuum. That way, they obtained output files describing ubiquitin along with 254 “well-placed” water molecules. Such configuration closely resembles the conditions during electrospray ionization (ESI), which likely would be used in single molecule diffractive imaging. For this reason, the very same structure files were adopted in this study.

Another aspect considered was the small structural fluctuations naturally rising in molecular systems. While these vibrations can be limited by lowering the temperature, it is possible that they would affect the explosion pattern nonetheless. A suggested way of investigating this effect was to let the ubiquitin molecule undergo a short simulative step before being ionized, randomly altering its initial structure slightly. This idea was dropped, however, since this would have added to the running time of each simulation and thereby reduced the total number of comparable explosions.

ATOM TRACKING

The sample input file to GROMACS is in typical protein data bank (PDB) format, meaning it contains x-, y- and z-coordinates of every single atom. During the simulation, these coordinates are continuously updated and saved. Naturally the changes will be more significant during and immediately following the pulse when atoms are ionized but still tightly packed, due to the strong repelling forces. But as the atoms are dislocated from the explosion, and the distance between them grows larger, the forces acting upon them weaken. Eventually, each atom will be affected by virtually no forces and will follow a linear trajectory from there on.

It is this state we want our system to reach. Converting the Cartesian coordinates to spherical ones yields three parameters – two angles and the distance to origin – and once the trajectory has stabilized the angles will no longer undergo major changes. In practicality this state is reached by letting the simulation run for a long period of time, determined empirically *in silico*. Note that this approach is an approximation of a real-life setting, where the atoms would be detected at a specific distance from the explosion centrum. Because some atoms travel at low speeds, and because the distance to the detector is large, a simulation would need to be ongoing for considerably longer to truly reflect the real experiment. This is beyond the scope of what we can afford computationally and thus the approximation is adopted instead.

Once converged, the values of the polar and azimuthal angles can be extracted. Plotting these angular values against each other then yields an image showing the approximate direction a certain atom will travel from an XFEL-induced explosion. By merging such plots of the same atom from multiple simulations of different ionization sequences, we obtain a plot where potential clustering tendencies can be visually detected (see Figure 5).

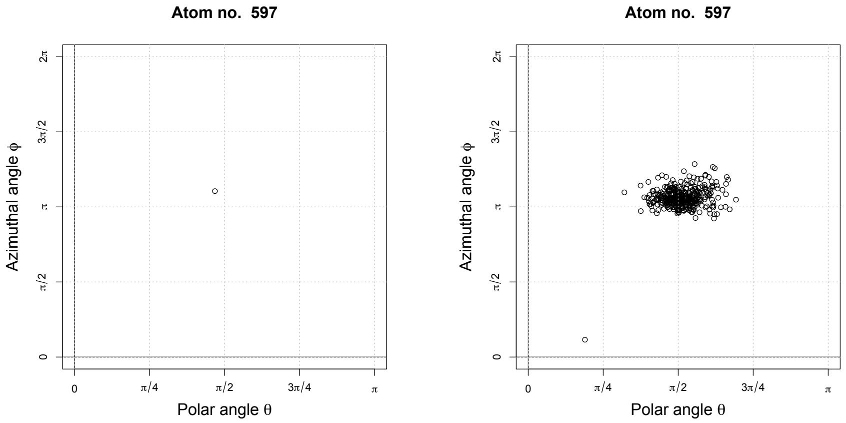

Figure 5: Plotting the simulation results. By the end of a simulation, the final position of an atom is converted into spherical coordinates and the resulting angles are plotted, yielding the image on the left. Potential clustering patterns can then be visually observed as more and more of such simulation results are combined (image on the right).

Next, we need a method to quantify the level of clustering in order to allow for comparability between atoms. Since we have a graphical representation we can with relative ease calculate the mean and standard deviation of all the points in the two-dimensional space. Furthermore this can be visualized in the plot as an ellipse centered at the mean point, see Figure 6. Because the ellipse will be smaller for more clustered points, the area of the ellipse was found to be a suitable measurement of "clusteredness".

To give this metric more meaning, consider the following: the polar angle ranges from 0 to π and the azimuthal angle spans an interval of length 2π (it is periodic, meaning that $\phi = \phi + 2\pi n$ for all integers n). This means the entire plot area spans a total of $2\pi \cdot \pi \approx 19.74$ area units. Furthermore, a uniformly distributed spread of points – that is, a complete lack of clustering behavior – would yield an SDE area of approximately 10.34 area units. It is helpful to keep this value in mind as it serves as a baseline for level of clustering; it is the theoretical maximum of the SDE area. However, this assumes that any atom can yield at most one single cluster. The reader may consult Appendix 1 on page 34 to see how the value is derived.

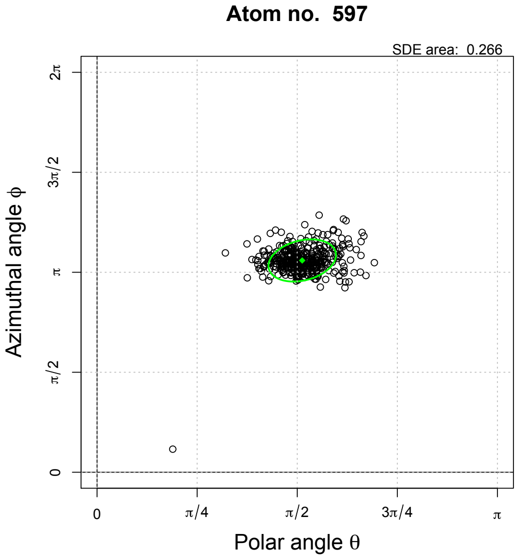

Figure 6: Standard deviation ellipse. The plot is complemented with a standard deviation ellipse calculated from the point spread along two orthogonal axes. The area of the ellipse is then used to quantify how clustered the points are.

The SDE area variable makes it possible to identify which atoms are more promising as pointers toward the initial molecular orientation. In theory this is useful, but experimentally we are unable to separate for example one carbon atom from another. In light of this fact, a more rigorous method would be one that considers all these pointers at once. We achieve this by plotting the standard deviation ellipse area against the atom ID number of all same-type atoms, resulting in an explosion footprint that makes it possible to compare how different experimental parameters affect the atomic trajectories. While this may not provide us with conclusive information regarding the orientation of the sample, it enables us to evaluate the feasibility of such approach. To make comparisons between such graphs fair we chose to order the atom IDs in the same way, namely in terms of increasing initial distance from the origin. Lastly, we also decided to mainly focus on carbon since it is commonly occurring and undergoes K-shell ionization as described previously. From here on, we call this plot the *carbon footprint*.

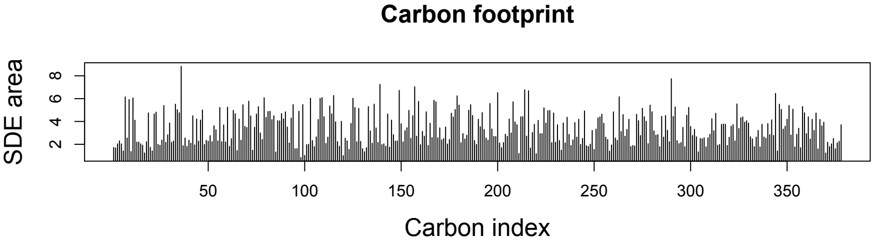

Figure 7: Carbon footprint. Conceptual plot demonstrating the carbon footprint principle. The SDE area for each carbon atom in ubiquitin calculated at a specific set of simulation parameters and plotted against atomic index. The indices are sorted by increasing initial Euclidian distance from the Cartesian origin – *i.e.* the mass centrum. Therefore the leftmost bar represents the SDE area of the carbon closest to the centrum of ubiquitin, while the rightmost indicates the same measure of the carbon most distant from the centrum. Actual distances increase non-linearly along the x-axis due to the equal width of the bars.

All of this was done by importing the set of initial and final atom coordinates of each simulation into the statistical computing environment R[19][18](Ihaka and Gentleman)(18), where it could be manipulated and visualized as desired.[19] To calculate and display the standard deviation ellipses the package *aspace* was utilized. The full script can be found in Appendix 4 on page 42.

Parameters

A few sets of trial simulations were performed during the course of a couple of weeks in order to tweak the simulation parameters. The goal was to make the main simulations as accurate as possible without becoming too demanding time-wise. Parameters considered, and their respective units, are shown in Table 1 below.

Table 1: Simulation parameters. The parameters and units considered when designing the main simulations.

Parameter	Description	Unit
Photon energy	Individual energy of each photon.	keV
Pulse length	Full width at half maximum of the Gaussian pulse.	ps
Pulse intensity	The total number of photons.	-
Spot diameter	Diameter of the laser spot.	nm
Simulation time	Total length of the simulation.	ps
Time step	Passing time between calculations.	ps

Some parameters were set to simply mimic the settings of a real experimental setup. For instance, the photon energy was set to 8 keV (corresponding to a wavelength of approximately 1.5 Å), which is in the common range for X-ray crystallography. The intensity was set to $2 \cdot 10^{12}$ photons, a number relevant when simulating XFEL experiments.[3] Lastly, the circular focal spot – over which the photons are uniformly distributed – had a fixed diameter of 100 nm, as can be achieved experimentally.

As for the simulation time and time step parameters, ideally we would want the former to be as great as possible and the latter to be as small as possible. By increasing the simulation time we let the atoms travel further away from the point of explosion, guaranteeing more stable values of the spherical angels. By decreasing the time step we ensure a greater accuracy in our calculations, giving data more representative of reality. However, evidently the angles need to be stable for the data to be useful – no matter the calculation accuracy. Proceeding from this fact, a number of trial simulations showed that 7 ps were (by far) enough to obtain virtually unchanging spherical angles – see one example in Figure 8 below.

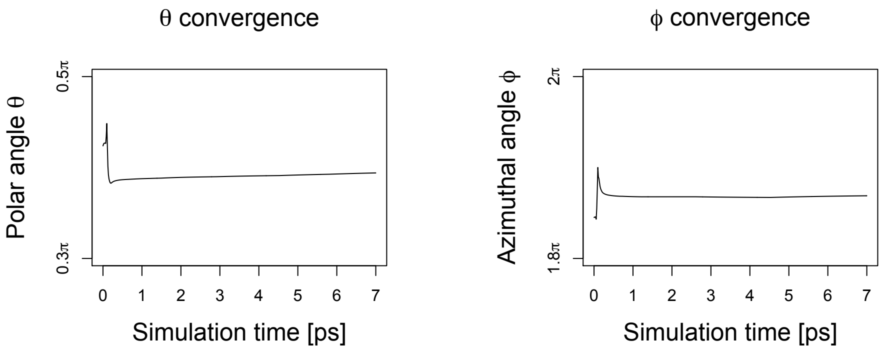

Figure 8: Angular convergence. The above graphs display the angular behavior of a randomly chosen carbon in one of the simulations. The changes in spherical angles of this particular carbon are negligible after simulating a system for one picosecond. A total simulation time of 7 ps was chosen to allow for any major deviations from this rule of thumb.

The initial aim was to reach a total of 300 distinct simulations per set of parameters, but this number was later adjusted to 296. The time step parameter was then set to 0.00001 ps (resulting in 700,000 steps per simulation and very good accuracy) giving a computational time of roughly 10 hours per 8-simulation run. Collecting one dataset thereby took just over two weeks, which was considered feasible.

Table 2: Main simulation settings. The parameter values of both simulation sets used for analysis.

Parameter	Simulation set 1	Simulation set 2
Photon energy	8 keV	8 keV
Pulse length	0.1 ps	0.05 ps
Pulse intensity	$2 \cdot 10^{12}$ photons	$2 \cdot 10^{12}$ photons
Spot diameter	100 nm	100 nm
Simulation time	7 ps	7 ps
Time step	0.00001 ps	0.00001 ps

For purposes of comparison, two datasets were collected in which only the pulse length variable was set differently. This would allow for implementation of the carbon footprint metric, and both hint toward the effects of varying settings overall and to how the pulse length in particular controls the directionality of the explosion. Table 2 above summarizes the parameter values used in both simulation sets.

SCRIPTS

A three-step pipeline was established to carry out all the different steps of the process described above. Two of these, called *explode.pl* and *analyze.pl*, were written in Perl and controlled the actual simulations and what data to extract from them. The last script, *plot.R*, was written in R and handled the statistics and visualization of the extracted data. All three scripts were called upon individually from a UNIX terminal in a Mac OSX environment. Figure 9 on the following page shows a schematic flowchart describing the internal work structure of each script, as well as how they are interlinked. Moreover, the actual code snippets of the three separate scripts are enclosed as appendices, see pages 36–42.

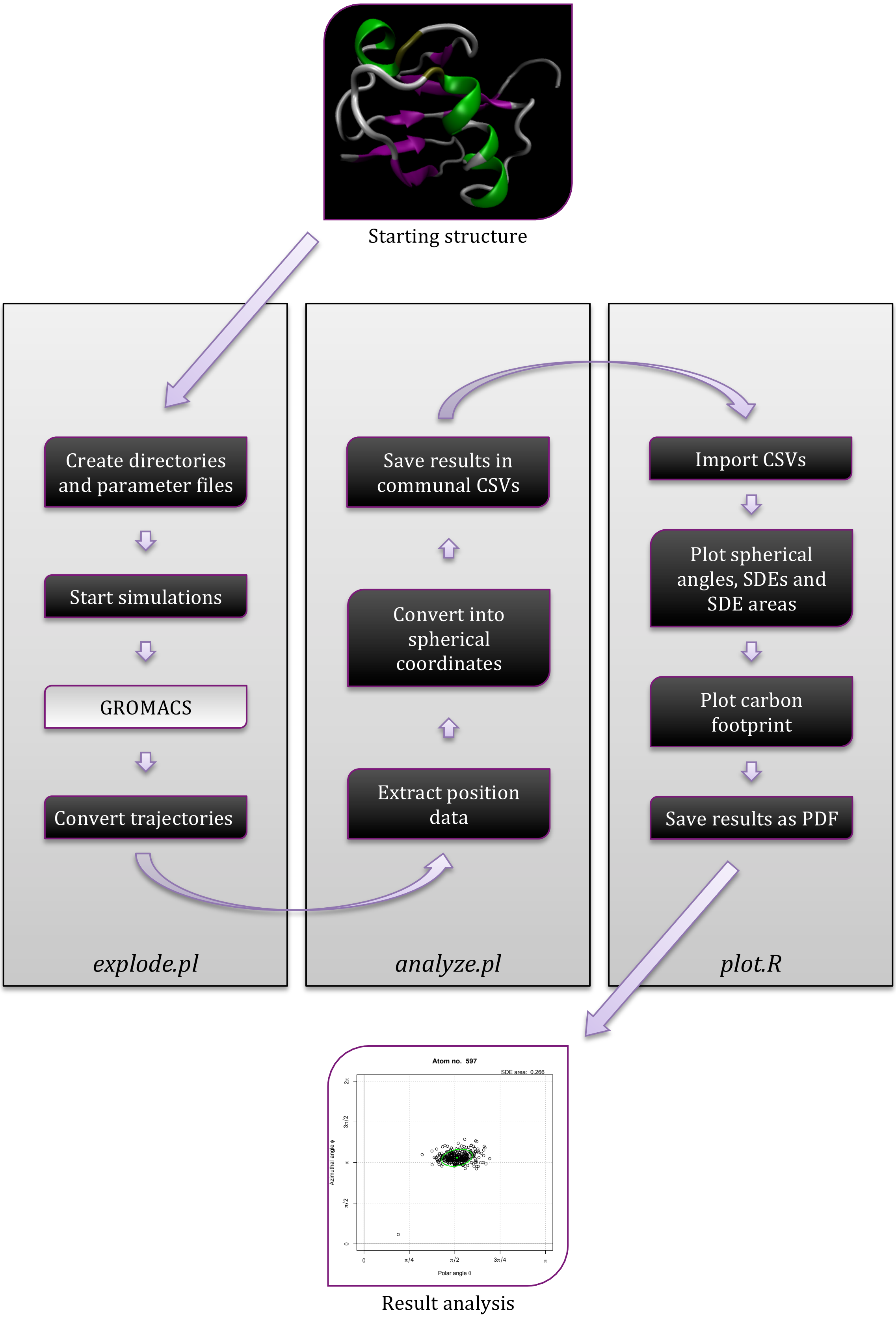

Figure 9: Work structure. Chart displaying the different steps of the data mining and managing process. The arrows indicate flow direction. Note that the last step also includes the carbon footprint analysis method.

EXPLODE.PL

The first script manages the actual simulations, including preprocessing and direct follow-up necessities. It is also designed to handle eight simultaneous runs, since the data collection was performed on an 8-core server. It assigns values to all the parameters by creating run files for all the individual runs carried out by GROMACS, and place these in well-defined directories. Obviously, these files will be identical for each simulation within the same set, apart from the random numerical value upon which the ionization sequence is determined. The script then calls for GROMACS and provides it with the three necessary input files – simulation parameters, molecular structure and sample topology. Once a simulation has finished, *explode.pl* calls for a trajectory file converter included in the GROMACS package. This way we obtain a human readable output file for each simulated explosion, ready to be processed by *analyze.pl*.

ANALYZE.PL

The second script then parse each of the trajectory files and extracts the initial and final coordinates of a specific atom type. The data is temporarily saved in a nested system of a Perl hash and multiple arrays to allow for further manipulation. For each trajectory parsed, the script then converts the data into spherical coordinates and writes the results to two separate CSV-files (comma-separated values), which facilitates later importation into R. By iterating the process, *analyze.pl* continually add spherical data to the CSV-files until all the explosion trajectories have been analyzed. Therefore we eventually get two different output files, one containing the initial positions and one containing the final positions of our atom type of interest – here chosen to be carbon.

PLOT.R

In the last section of the pipeline, the CSV-files are imported into R to be plotted. Each atom is managed separately so that it's concluding traveling direction in every explosion is represented in the same plot. Once done, the mean center and standard deviation ellipse (SDE) of the points is calculated and added to the visualization. The area of the ellipse is used as a measure of clustering and is therefore also calculated, saved and included in the top-right corner of each figure. All the plots are then saved to a common PDF-document, allowing for easy access. A separate PDF-document is also produced, containing the carbon footprint metric described earlier.

RESULTS

While the trial simulations' foremost purpose was to pave way for the true data gathering, the information obtained from them were considered interesting results nonetheless. Hence, they are presented here despite not providing sufficiently accurate data to base any conclusions on in themselves.

TRIAL SIMULATIONS

Once the computational pipeline had been established a total of three sets of trial simulations were conducted. All of them used the PDB entry of ubiquitin as starting molecule – whereas the main simulations used a modeled version of ubiquitin that had a 3 Å layer of water added to it.

FIRST TRIAL SET

The first trial set consisted of 692 simulations in total, defined by the following parameter values:

First trial set parameters

Intensity	10^{12} photons
Photon energy	12 keV
Pulse length (FWHM)	100 fs
Focal spot diameter	100 nm
Simulation time	10 ps
Number of steps	200,000
Number of simulations	692

The carbon atoms showed varied signs of clustering with SDE areas raging from just under 1 up to almost 9 area units. The most promising candidate, atom number 526, displayed an SDE area of 0.842 while the SDE area of atom number 9 – the least promising candidate – was evaluated to 8.826 (see Figure 10). The clustering behavior in the former is particularly striking, while the latter seems to be almost randomly dispersed. Interestingly, both of these carbons are found fairly close to the origin possibly suggesting that there might not be a direct correlation between how embedded an atom is and its level of directional restriction.

Despite relatively few obvious clustering tendencies, the mean SDE area calculated to 3.38 indicates that the data set is skewed toward a clustering behavior. The value is significantly lower than the 10.34 area unit-maximum derived on page 35 and manual inspection of the plots showed no apparent cases of multiple clusters which could affect the mean. It seems probable that most atoms are more or less directionally limited during the explosion.

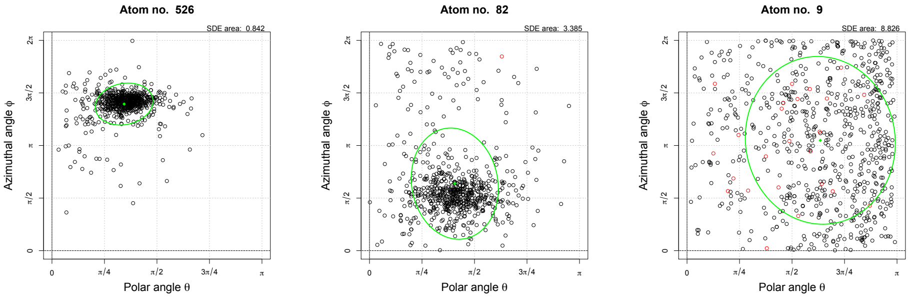

Figure 10: First trial simulation set results. Three plots representing the various results obtained in the first trial simulation. The leftmost image shows atom 526, the carbon that exhibited the highest clustering tendency with an SDE area of 0.842. In contrast, the rightmost image shows a similar plot of atom 9, which at the SDE area of 8.826 was the least clustered carbon. The middle image is that of atom 82 with an SDE area of 3.385 – closely resembling the arithmetic mean of all the SDE measurements calculated to 3.38. Points highlighted in red indicate simulations where the atom of interest was found at a distance of less than 10 nm from the initial molecular centrum.

SECOND TRIAL SET

In the second trial set most of the parameters were left unchanged from the previous set, except for the intensity, which was increased 10-fold, and the number of simulations, which was reduced. The reason behind the former was to see how the intensity affected the explosion in terms of atom clustering. One hypothesis is that a higher intensity would cause a more chaotic ionization event that, in turn, could make the explosions less predictable. Another suggests the complete opposite: a more rapid ionization of the molecule could give the individual atoms less leeway and thereby result in more concentrated clusters. The number of simulations was also reduced to 394, an amount judged to still provide sufficiently accurate data.

Below is a table summarizing all parameter values used in the second trial simulation.

Second trial set parameters

Intensity	10^{13} photons
Photon energy	12 keV
Pulse length (FWHM)	100 fs
Focal spot diameter	100 nm
Simulation time	10 ps
Number of steps	200,000
Number of simulations	394

As in the previous set, some carbons showed clear clustering tendencies, while others seemed far less predictable. Again, atom number 526 provided the nicest plot with a remarkable SDE area of 0.071 – significantly lower than in the first trial simulation set. Even the worst carbon (atom number 388) yielded an SDE area smaller than its previous simulation worst-candidate counterpart, although at a value of 8.05 it is hardly too informative.

The mean SDE area, however, was measured to approximately 2.93 and does strengthen the idea that a higher intensity may limit the atoms and thus make their escape paths more predictable. Figure 11 shows the plots of the best and worst atoms along with the plot of atom number 60, which happened to give an SDE of 2.93 area units and should therefore (due to its proximity to the mean value) give a feel for what a "typical" simulation plot in this set looks like.

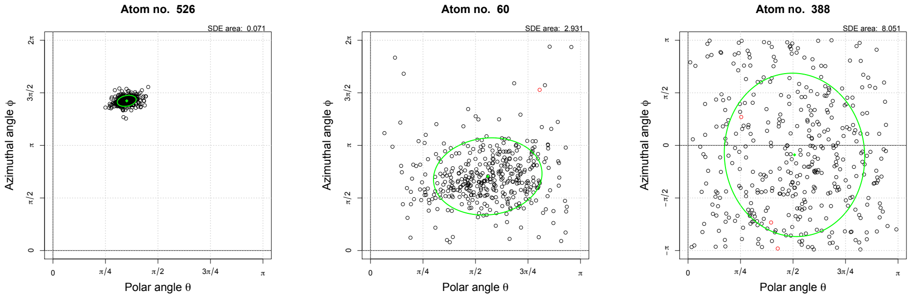

Figure 11: Second trial simulation set results. Three plots representing the various results obtained in the second trial simulation. On the left is that of atom number 526 and on the right is that of atom number 388, the most and least clustered atoms respectively in the set. SDE areas of the two measured to 0.071 and 8.05. The middle image is the plot of atom 60, whose SDE area best corresponded to the set mean. Again, red points represent atoms that were found less than 10 nm from the initial molecule centrum by the end of the simulation.

To give a clearer comparison between the two trial simulation sets, Figure 12 shows a plot of the difference in SDE area between the first and the second set for every carbon. Since there is a slight bias to positive values (the mean is approximately 0.45) we see that the second simulation set actually gave better results in terms of level of clustering.

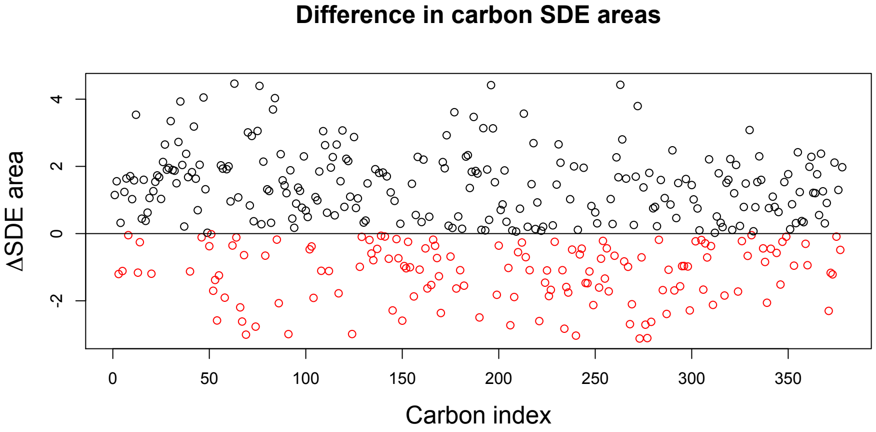

Figure 12: Comparative plot. The difference between the carbon SDE areas obtained in the two trial simulations. Atoms represented above the horizontal line (black) were found to be more clustered in the second set, while the opposite holds true for points below the line (red). The carbons are indexed by their increasing initial distance to the Cartesian origin, as given by the original ubiquitin PDB file.

However, when calculating the variance within the collection of SDE areas for each set we find that the second set yielded more varied values. With a variance more than twice as great in the second set (4.69 versus 2.05) it is possible that a higher intensity need not necessarily focus clusters further, but could also have the opposite effect.

Third trial set

Lastly, a small set of eight simulations was performed with significantly higher resolution – *i.e.* a larger number of steps were employed without changing the total simulation time – in order to address two problems that had arisen during the first two trial sets. For one, some atoms seemed to fail to travel particularly far from the origin of the explosion in some of the simulations, especially those that showed little or no clustering tendencies. Examples of this can be seen highlighted in red in some of the plots of Figure 10 and Figure 11. This poses a problem since in a real setting the particle detector would be placed at a large distance from the molecule, meaning that short-traveling atoms cannot be detected. For the other, a strange angular behavior was observed among some of the atoms. Some plots indicated that certain values of both the polar angle θ and the azimuthal angle ϕ were preferred in a peculiar way. While this could be an affect of clustering, it seemed highly improbable. Figure 13 below show examples of this suspected problem.

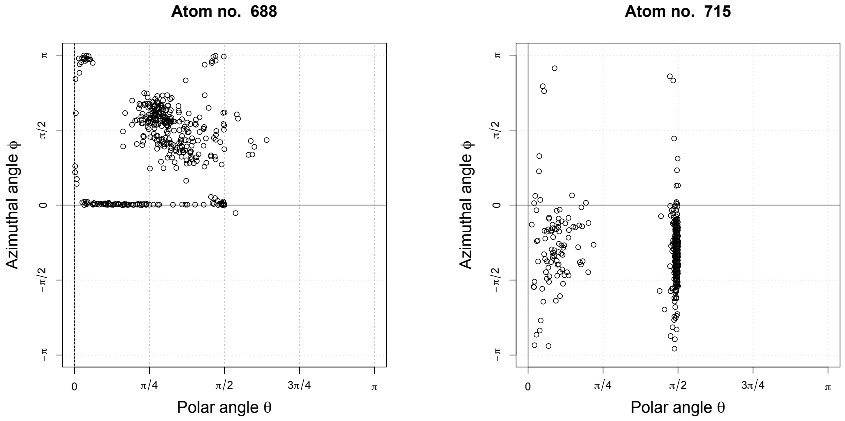

Figure 13: Angular preference. Selected plots from the second trial simulation set showing a strange behavior among the points.

Both these problems could be the effect of data loss due to a long time step in the simulations. To explore whether this was the case, and to simultaneously realize the time estimates of higher-resolution simulations, we constructed the third trial set with the following settings:

Third trial set parameters

Intensity	$2\cdot10^{12}$ photons
Photon energy	8 keV
Pulse length (FWHM)	100 fs
Focal spot diameter	100 nm
Simulation time	10 ps
Number of steps	1,000,000
Number of simulations	8

Due to the low number of samples, the SDE areas of this set are far too inaccurate to be compared to those of the other two trial sets. This set did however not seem to yield plots showing signs of the angular preference problem, and since the simulation time still was acceptable, the same time step was adopted in the main simulations with the compromise that the total simulation time was shortened to 7 ps.

The problem with the occasional short traveling distances of certain atoms, on the other hand, remained even in the high-resolution set. It could be an effect of the shortcomings of GROMACS, as to how forces are calculated and applied to the system *etc.* But it is also possible, and even likely, that some atoms either experience cancelling forces or do not get ionized in the first place. The result would be a slow trajectory in both these cases, and could very well result in a barely noticeable movement during the 10 ps time span.

Main simulations

As briefly mentioned earlier, a different starting structure was utilized in the main simulations. In this version of ubiquitin, a 3 Å water layer had been added and distributed along the surface of the protein *in silico*. Such starting point better reflect the circumstances of an ESI setup, which likely would be used when attempting single-molecule imaging. Because of this, the atom numbering differs between the trial and main simulations and can therefore not be directly compared. Between the main simulations this should pose no problems, however.

First main set

The first set of main simulations used laser settings found to be suitable based on the results of the trial simulations. Since no clear difference was observed from a 10-fold increase in photon intensity between the first two trial sets, the intermediate value $2 \cdot 10^{12}$ opted for in the last few trial simulations was left unchanged.

First main set parameters

Intensity	$2 \cdot 10^{12}$ photons
Photon energy	8 keV
Pulse length (FWHM)	100 fs
Focal spot diameter	100 nm
Simulation time	7 ps
Number of steps	700,000
Number of simulations	294

The spherical angle plots obtained from this major set of high-resolution simulations showed remarkably high levels of clustering among a vast majority of the carbon atoms. The mean SDE area was found to be 0.67 area units, with individual atoms giving values as low as 0.159. Furthermore, the highest SDE area value measured was 4.773, a reduction by almost fifty percent in comparison with the corresponding value in the second trial set. These data strongly suggests that there is an explosion pattern that theoretically could be mapped and aid the determination of the orientation of the molecule.

Figure 14 shows the best and worst plots of the set. Note how even the latter indicates a certain degree of clustering.

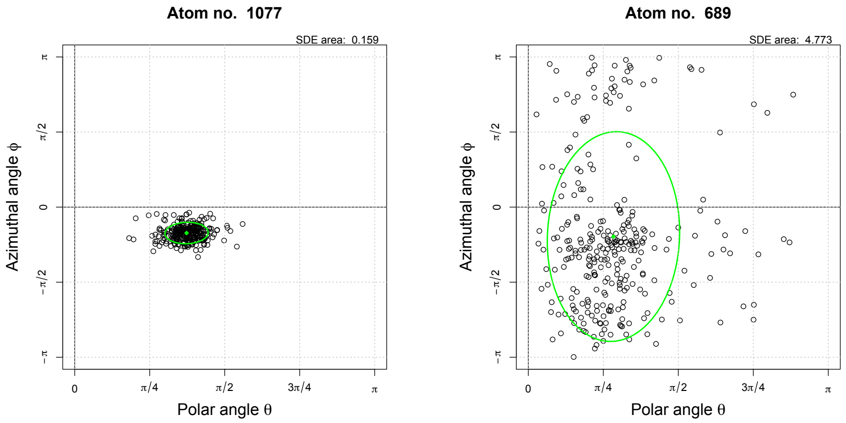

Figure 14: First main simulation set results. Plots of atoms number 1077 and 689, which expressed the most and least distinctive signs of clustering respectively.

In order to analyze the entire data set, we examine the previously discussed carbon footprint for the specific setup. The carbon footprint of the first main set can be seen in Figure 15 and in it we find a couple of interesting pieces of information. Firstly, there seems to be a slight propensity towards more accurate clustering at greater distances from the origin, which suggests that the trajectory of atoms embedded within the protein are more difficult to predict. Secondly, as already implied by the low mean value, there are surprisingly few high peaks with a majority of carbons showing SDE areas of less than 1 area unit. Upon closer investigation, it turns out, this accounts for 314 out of the 378 total carbons in ubiquitin. Evidently, over 83 % of the carbon atoms are particularly restricted when it comes to possible escape paths during the explosion.

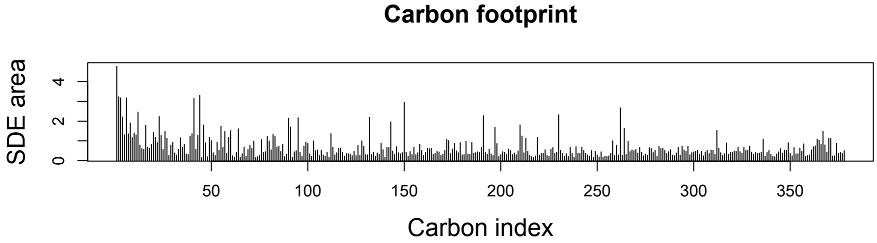

Figure 15: Carbon footprint of first main simulation set. Each bar represents a carbon and shows its resulting SDA area from the first set of main simulations. The carbons were indexed by their respective distance to the mass centrum of the starting structure – lower index means shorter distance.

SECOND MAIN SET

To put the first main set into perspective, another set of simulations was performed. Since this thesis focuses primarily on applications in single-molecule XFEL imaging – where short pulse durations are employed to minimize the effects of radiation damage – the pulse length was halved while the other parameters were kept identical. This would also reveal whether or not the pulse length affects the explosion accuracy at all.

Second main set parameters

Intensity	$2 \cdot 10^{12}$ photons
Photon energy	8 keV
Pulse length (FWHM)	50 fs
Focal spot diameter	100 nm
Simulation time	7 ps
Number of steps	700,000
Number of simulations	294

Explosions were found to be even more restricted in nature when ubiquitin was exposed to the same amount of X-ray photons in half the time. The mean SDE area of the carbons was calculated to 0.54, a 19 % improvement from the previous set. When comparing individual atoms we found that the most clustered carbon (atomic number 382) had an SDE area of 0.065, and the least clustered carbon (atomic number 689) had an SDE area of 4.22 – both surpassing their counterparts of the first main set.

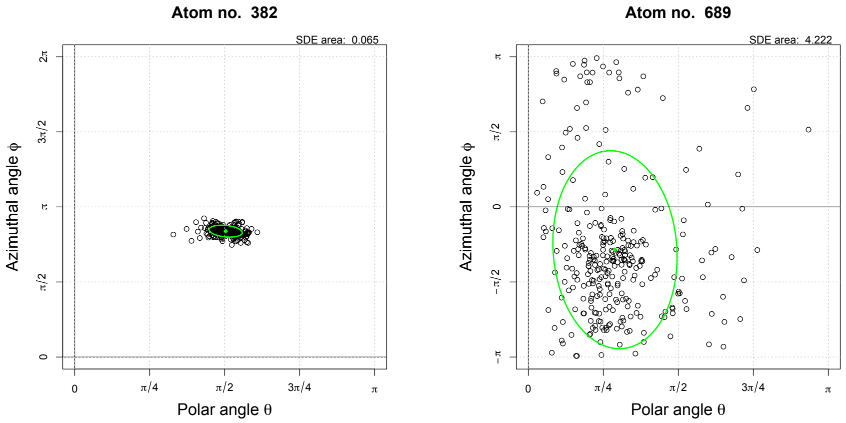

Figure 16: Second main simulation set results. Plots of carbons numbered 382 and 689. The former displayed the highest level of clustering within the set, while the latter was found to give the highest SDE area among all carbons.

Apparently, shortening the pulse length while retaining the total number of photons generally seems to benefit the predictability of carbon trajectories. It is however not unlikely that, while most carbons tend to become more directionally restricted, others might become less so. To see if this was the case – and if so, to what extent – we both compared the variance in SDE areas of each set, and plotted the difference in SDE area for each carbon.

The SDE area variances of the first and second sets were 0.345 and 0.328, respectively. This decrease in variance shows that the SDE area values of the second set are slightly more unanimous than in the first, which implies that it is unlikely that the second set contains a considerable amount of extreme outliers. Figure 17 below strengthens this by showing that only four carbons display a gain of 1 area unit or more from the first set to the second. It is noteworthy though that while a majority of SDE areas has improved, a non-negligible amount of carbons does seem to become slightly less predictable when shortening the pulse duration. In summary, most carbons benefit from the change, while a select few instead become less clustered. Unfortunately, we were unable to find definitive reasons behind this behavior, and no obvious traits seem to fully explain it.

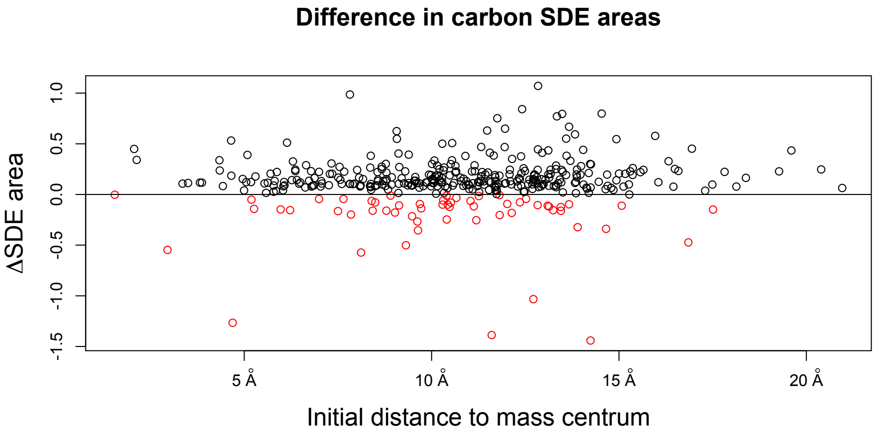

Figure 17: Pulse duration effects. A plot showing the change in SDE area for each carbon between the main sets of simulations. Points above the horizontal line (black) indicates an improved level of clustering in the second set with the 50 fs pulse, while those below (red) gave better results in the first set, which had the pulse length set to 100 fs.

Lastly, a complete picture of the second main set is visualized in Figure 18, which shows the carbon footprint. In it, many of the peaks are less than 1 area unit high – these carbons constitute approximately 86.5 % of the entire set. Extending this limit to 3 area units, we find almost all carbons (~99.5 %) represented, indicating that a propensity toward clustering is commonplace. There also seems to be a slight correlation between initial placement and level of clustering among atoms found close to the molecular centrum, a phenomenon likely to be more prevalent and important when studying larger proteins. Both of these observations were reflected in the first data set as well, further reinforcing their credibility.

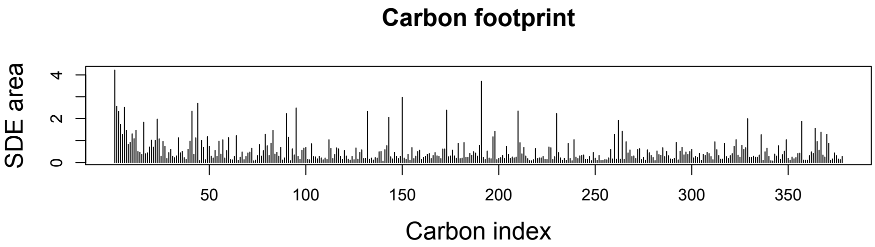

Figure 18: Carbon footprint of second main simulation set. Clustering behavior of all carbon atoms in water-covered ubiquitin once exposed to a 50 fs X-ray pulse. Again, carbon atoms are named such that lower index indicates a shorter initial distance to the mass centrum.

To clearly visualize the effects of shortening the pulse length, Figure 19 shows the change in carbon footprints between the simulation sets. Most peaks of the first set tend to become even lower in the second set, as can be seen by the majority of positively valued bars. There are however exceptions, in particular among higher-index carbons. This could possibly point toward a greater uncertainty among trajectories of atoms placed further from the molecular core. But while such conclusions remain speculative, the trajectory of an arbitrary carbon seems to generally become more predictable at a pulse length of 50 fs rather than 100 fs, despite their placement within the sample.

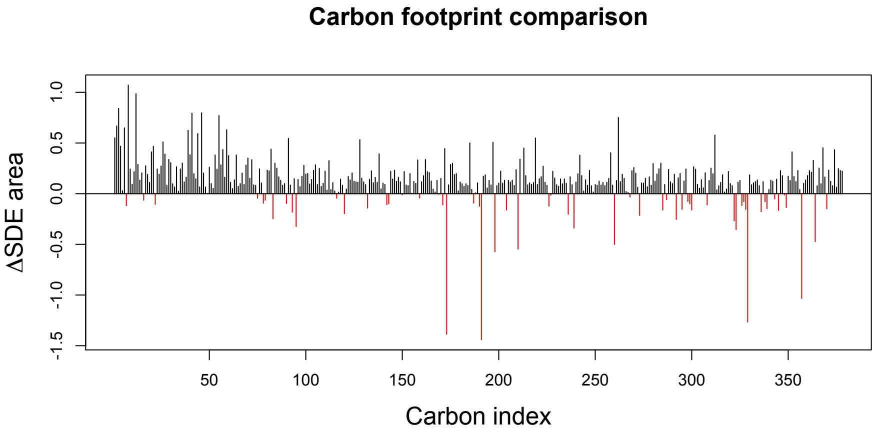

Figure 19: Comparison of the carbon footprints. The red, negative bars depict carbons that were more clustered when exposed to a 100 fs pulse. Conversely, black bars indicate which carbons became more restricted from the shorter 50 fs pulse. The height of each bar shows the change in SDE area associated with the atom between the two sets.

DISCUSSION

The findings presented here are based upon a limited number of data points. Considering how many potential variations of the ionization sequence there could exist for ubiquitin, we are working with a mere fraction of the entire landscape of possible outcomes. This does not, however, mean that the findings are any less true – but rather that there is an undeniable uncertainty to them. Ideally, one would perform a greater number of simulations within each set to somewhat rectify this – something easily done following the same workflow as presented here. On the other hand, MD-simulations are foundationally built upon approximations of the atomic reality and will therefore always yield imprecise data. Luckily, the approximations are still fairly reliable and despite lacking many of the possible data points a lot can be said about the results.

For example, judging by the carbon footprints found in Figure 15 (page 24) and Figure 18 (page 26) almost all carbons in ubiquitin show a certain level of clustering behavior. Considering the discussed inaccuracy some of them might not actually be as clustered as suggested, but the statistical odds of obtaining such convincing results at random – for that many atoms – are infinitesimally small. So while we should be careful about stating that a certain number of carbons have predictable trajectories or that a specific carbon is clustered, we can conclude that some carbons are directionally bound during an X-ray induced explosion. Which ones, and why, are questions that remains to be answered, however. In our brief study we could not find any obvious connections between clustered atoms.

Application of the results presented here is an entirely different matter. One needs to consider that in practicality we cannot distinguish between different atoms of the same type. Therefore, we can only know where a carbon ended up, not *which* carbon it was. This complicates things, especially considering we might not know all that much about our sample. In order to utilize explosion data for determining spatial orientation we would consequently need to take all carbons (or any other element prevalent in our sample) into account. Doing so, though, we could construct a map – let us call it the *carbon fingerprint* of the sample – of roughly where all the carbons are detected after an explosion. Making this map time-resolved by adding a third dimension (the first two being polar and azimuthal angles) representing atomic time-of-flight could help separating the carbons somewhat if necessary.

The carbon fingerprint would then be linked to the sample structure and could that way be used to fit explosion data and thereby reconstruct the orientation.

However, the sample structure would presumably be unknown considering the whole idea behind single-molecule X-ray imaging is usually to determine the structure. In these cases there would be no fingerprint available and the explosion data would be of limited use. Conversely, when working with a sample of known structure the diffraction pattern in itself would provide sufficient data to orientate the molecule. So does this mean that collecting explosion data would be fruitless?

Not necessarily. We suggest that by allowing the explosion and diffraction data to work in tandem both sample orientation and structure can be determined at a quicker pace, once a collaborative pipeline has been established. This way, while reconstructing the structure by fitting diffraction frames, one could also construct the carbon fingerprint piece by piece. The more frames collected, the more accurate the fingerprint and the more helpful it would be in connecting the frames.

CONCLUSION AND OUTLOOK

Through this work, we have created a platform for semi-automated gathering and analysis of *in silico* explosion data. The platform, consisting of three interconnected scripts, was used to assemble all of the information presented here, and can easily be applied to larger systems. As such, it is suitable for exploring the explosion dynamics of a biomolecule prior to real-life testing at an X-ray free-electron laser.

We have also shown that the directions in which the carbon atoms of ubiquitin travel from a molecular explosion caused by a short X-ray pulse are not random. This seems to be the case for most of the atoms, although some tend to be more restricted than others. Such restrictions are not clearly correlated to the placement within the sample, at least not in the case of ubiquitin, but additional testing is needed to truly verify this. Although it does seem plausible to assume that it might influence the explosion outcome, particularly in larger and more complex macromolecules.

Building an explosion fingerprint of the sample while performing diffraction studies could aid the assembly of frames, and once obtained serve as a guide in other experiments where spatial orientation is of interest. But until then, more studies of this type are needed to verify our results and build upon them. For instance, a larger number of simulations would provide data more statistically accurate than in this thesis. In doing such simulations one could preferably also consider variables left out here, such as minor structural changes, different force fields, size of water layer and so on.

Another aspect is to evaluate the intermolecular applicability of explosion patterns; does it apply to other proteins as well? If so, can we find common factors between the molecules that can help us predict the outcome of the explosion? MD-simulations could be used to answer these questions – answers that are tremendously important if this method were to be employed when studying molecules of unknown structure.

In this study we used a model of ubiquitin that was covered in a thin water layer. Another additional way to increase the authenticity of the simulations would be to slightly vary both the amount of water as well as the actual starting structure conformation. The former since in a real ESI experiment, there is an uncertainty as to how much water will surround the sample once ejected into the vacuum chamber. The latter since even at cryogenic temperatures, proteins vibrate and undergo slight structural changes. Both of these factors could affect the explosion pattern and should therefore be taken into account.

To summarize, the future of single-molecule X-ray imaging looks bright. Examining and utilizing explosion patterns may be of paramount importance in that future and should consequently be subject to further studies. We suggest conducting a large number of high-resolution explosion simulations of a wide range of starting structures. The simulations would preferably cover as many physical factors likely playing a role in the explosion outcome as possible. These include quantum mechanical phenomena, amount of water surrounding the sample and deviations in sample starting structure. Once such data has been obtained and analyzed, and a solid foundation of X-ray induced explosions has been set, we will be ready to confirm them through actual, real-life experiments.

Acknowledgements

The author would like to wholeheartedly extend his sincere gratitude toward the following people for their involvement with this project. Without their help, this work would never have seen the light of day.

Carl Caleman, **Nicusor Timneanu**, for their support and guidance as supervisors.

David van der Spoel, for his useful input as a scientific reviewer as well as for developing GROMACS.

Mattias Lundén, for accepting to act as opponent at the oral defense of this thesis, and indeed following through with it.

Olof Jönsson, for his consistently uplifting mood and surprisingly enlightening napkin drawings.

Björn Viklund, for his invaluable programming skills and borderline ridiculous enthusiasm.

And finally to my incredible **Caroline**.
(For her peak brilliance.)

REFERENCES

[1] R. Neutze, R. Wouts, D. van der Spoel, E. Weckert, and J. Hajdu, "Potential for biomolecular imaging with femtosecond X-ray pulses," *Nature*, vol. 406, no. 6797, pp. 752–757, Aug. 2000.

[2] H. N. Chapman, P. Fromme, A. Barty, T. A. White, R. A. Kirian, A. Aquila, M. S. Hunter, J. Schulz, D. P. DePonte, U. Weierstall, R. B. Doak, F. R. N. C. Maia, A. V. Martin, I. Schlichting, L. Lomb, N. Coppola, R. L. Shoeman, S. W. Epp, R. Hartmann, D. Rolles, A. Rudenko, L. Foucar, N. Kimmel, G. Weidenspointner, P. Holl, M. Liang, M. Barthelmess, C. Caleman, S. Boutet, M. J. Bogan, J. Krzywinski, C. Bostedt, S. Bajt, L. Gumprecht, B. Rudek, B. Erk, C. Schmidt, A. Hömke, C. Reich, D. Pietschner, L. Strüder, G. Hauser, H. Gorke, J. Ullrich, S. Herrmann, G. Schaller, F. Schopper, H. Soltau, K.-U. Kühnel, M. Messerschmidt, J. D. Bozek, S. P. Hau-Riege, M. Frank, C. Y. Hampton, R. G. Sierra, D. Starodub, G. J. Williams, J. Hajdu, N. Timneanu, M. M. Seibert, J. Andreasson, A. Rocker, O. Jönsson, M. Svenda, S. Stern, K. Nass, R. Andritschke, C.-D. Schröter, F. Krasniqi, M. Bott, K. E. Schmidt, X. Wang, I. Grotjohann, J. M. Holton, T. R. M. Barends, R. Neutze, S. Marchesini, R. Fromme, S. Schorb, D. Rupp, M. Adolph, T. Gorkhover, I. Andersson, H. Hirsemann, G. Potdevin, H. Graafsma, B. Nilsson, and J. C. H. Spence, "Femtosecond X-ray protein nanocrystallography," *Nature*, vol. 470, no. 7332, pp. 73–77, Feb. 2011.

[3] S. Boutet, L. Lomb, G. J. Williams, T. R. M. Barends, A. Aquila, R. B. Doak, U. Weierstall, D. P. DePonte, J. Steinbrener, R. L. Shoeman, M. Messerschmidt, A. Barty, T. A. White, S. Kassemeyer, R. A. Kirian, M. M. Seibert, P. A. Montanez, C. Kenney, R. Herbst, P. Hart, J. Pines, G. Haller, S. M. Gruner, H. T. Philipp, M. W. Tate, M. Hromalik, L. J. Koerner, N. van Bakel, J. Morse, W. Ghonsalves, D. Arnlund, M. J. Bogan, C. Caleman, R. Fromme, C. Y. Hampton, M. S. Hunter, L. C. Johansson, G. Katona, C. Kupitz, M. Liang, A. V. Martin, K. Nass, L. Redecke, F. Stellato, N. Timneanu, D. Wang, N. A. Zatsepin, D. Schafer, J. Defever, R. Neutze, P. Fromme, J. C. H. Spence, H. N. Chapman, and I. Schlichting, "High-Resolution Protein Structure Determination by Serial Femtosecond Crystallography," *Science*, vol. 337, no. 6092, pp. 362–364, Jul. 2012.

[4] R. A. Kirian, X. Wang, U. Weierstall, K. E. Schmidt, J. C. H. Spence, M. Hunter, P. Fromme, T. White, H. N. Chapman, and J. Holton, "Femtosecond protein nanocrystallography—data analysis methods," *Optics Express*, vol. 18, no. 6, p. 5713, Mar. 2010.

[5] J. Feldhaus, J. Arthur, and J. B. Hastings, "X-ray free-electron lasers," *Journal of Physics B: Atomic, Molecular and Optical Physics*, vol. 38, no. 9, pp. S799–S819, May 2005.

[6] K. F. Lee, D. M. Villeneuve, P. B. Corkum, A. Stolow, and J. G. Underwood, "Field-Free Three-Dimensional Alignment of Polyatomic Molecules," *Physical Review Letters*, vol. 97, no. 17, Oct. 2006.

[7] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen, “GROMACS: Fast, flexible, and free,” *Journal of Computational Chemistry*, vol. 26, no. 16, pp. 1701–1718, Dec. 2005.

[8] “RCSB Protein Data Bank - RCSB PDB.” [Online]. Available: http://www.rcsb.org/pdb/home/home.do. [Accessed: 01-Sep-2014].

[9] M. P. Allen and D. J. Tildesley, *Computer simulation of liquids*. Oxford [England]; New York: Clarendon Press ; Oxford University Press, 1989.

[10] P. Auger, P. Ehrenfest, R. Maze, J. Daudin, and R. Fréon, “Extensive Cosmic-Ray Showers,” *Reviews of Modern Physics*, vol. 11, no. 3–4, pp. 288–291, Jul. 1939.

[11] M. O. Krause, “Atomic radiative and radiationless yields for K and L shells,” *Journal of Physical and Chemical Reference Data*, vol. 8, no. 2, p. 307, 1979.

[12] B. Ziaja, D. van der Spoel, A. Szöke, and J. Hajdu, “Auger-electron cascades in diamond and amorphous carbon,” *Physical Review B*, vol. 64, no. 21, Nov. 2001.

[13] G. A. Kaminski, R. A. Friesner, J. Tirado-Rives, and W. L. Jorgensen, “Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides,” *The Journal of Physical Chemistry B*, vol. 105, no. 28, pp. 6474–6487, Jul. 2001.

[14] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, “Comparison of simple potential functions for simulating liquid water,” *The Journal of Chemical Physics*, vol. 79, no. 2, p. 926, 1983.

[15] C. M. Pickart and M. J. Eddins, “Ubiquitin: structures, functions, mechanisms,” *Biochimica et Biophysica Acta (BBA) - Molecular Cell Research*, vol. 1695, no. 1–3, pp. 55–72, Nov. 2004.

[16] S. Vijay-kumar, C. E. Bugg, and W. J. Cook, “Structure of ubiquitin refined at 1.8 Å resolution,” *Journal of Molecular Biology*, vol. 194, no. 3, pp. 531–544, Apr. 1987.

[17] C. Caleman, G. Huldt, F. R. N. C. Maia, C. Ortiz, F. G. Parak, J. Hajdu, D. van der Spoel, H. N. Chapman, and N. Timneanu, “On the Feasibility of Nanocrystal Imaging Using Intense and Ultrashort X-ray Pulses,” *ACS Nano*, vol. 5, no. 1, pp. 139–146, Jan. 2011.

[18] E. G. Marklund, D. S. D. Larsson, D. van der Spoel, A. Patriksson, and C. Caleman, “Structural stability of electrosprayed proteins: temperature and hydration effects,” *Physical Chemistry Chemical Physics*, vol. 11, no. 36, p. 8069, 2009.

[19] R. Ihaka and R. Gentleman, “R: A Language for Data Analysis and Graphics,” *Journal of Computational and Graphical Statistics*, vol. 5, no. 3, pp. 299–314, Sep. 1996.

1. Derivation of maximal SDE area

We start with two known formula valid for a uniform distribution on the interval (a, b), namely the expected value

$$\mu = \frac{(a+b)}{2} \tag{A1}$$

and the standard deviation

$$\sigma = \frac{b-a}{\sqrt{12}} \tag{A2}$$

Denoting the polar angle by θ and the azimuthal angle by ϕ we get that the SDE in this case would be centered at

$$(\mu_\theta, \mu_\phi) = \left(\frac{0+\pi}{2}, \frac{0+2\pi}{2}\right) = \left(\frac{\pi}{2}, \pi\right)$$

and the standard deviation in each direction would be

$$(\sigma_\theta, \sigma_\phi) = \frac{\pi - 0}{\sqrt{12}}, \frac{2\pi - 0}{\sqrt{12}} = \left(\frac{\pi}{\sqrt{12}}, \frac{2\pi}{\sqrt{12}}\right)$$

From these two facts we can draw a standard deviation rectangle that will be inscribed in the needed SDE. To make calculations simpler, we translate our reference system such that the origin coincides with the common center point of the ellipse and the rectangle, and denote the sides of the rectangle by r_1 and r_2 (where $r_1 > r_2$). The four corners of the rectangle then lie at the points

$$\left(\pm\frac{r_1}{2}, \pm\frac{r_2}{2}\right)$$

Moreover, because the data points are uniformly distributed along the θ- and ϕ-axes it is clear that the major axis of the ellipse will be parallel to the longer side of the rectangle (and consequently, the minor axis will be parallel to the shorter side). In our new reference system the axes will therefore not only be parallel, but coinciding, hence we can use the general equation of a centered, non-rotated ellipse.

$$\left(\frac{x}{e_1}\right)^2 + \left(\frac{y}{e_2}\right)^2 = 1 \tag{A3}$$

Such an ellipse is not unique, however, but by further demanding that the proportions of the ellipse are equal to those of our rectangle – as must be the case here due to the random point dispersion – we limit the number of possibilities to exactly one. That is, by needing the ratio between the major and minor axes of the ellipse to be equal to the ratio of the sides of the rectangle we can determine the ellipse uniquely. Let e_1 and e_2 denote the major and minor axes of the ellipse respectively, and we obtain the following relationship:

$$\frac{e_1}{e_2} = \frac{r_1}{r_2} \quad \Leftrightarrow \quad e_1 = e_2 \frac{r_1}{r_2} \tag{A4}$$

Combining (A3) with (A4) and inserting an arbitrary point on the ellipse, *i.e.* any of the corners in the rectangle, we get

$$1 = \left(\frac{r_1/2}{e_1}\right)^2 + \left(\frac{r_2/2}{e_2}\right)^2 = \left(\frac{r_1 r_2}{2e_2 r_1}\right)^2 + \left(\frac{r_2}{2e_2}\right)^2 = \left(\frac{r_2}{2e_2}\right)^2 + \left(\frac{r_2}{2e_2}\right)^2 = 2\left(\frac{r_2}{2e_2}\right)^2 = \frac{r_2^2}{2e_2^2}$$

Rearranging this equation, we find that

$$e_2 = \frac{r_2}{\sqrt{2}} \tag{A5}$$

and from (A4) it follows that

$$e_1 = \frac{r_2}{\sqrt{2}} \frac{r_1}{r_2} = \frac{r_1}{\sqrt{2}} \tag{A6}$$

Because the sides of the rectangle are determined by the standard deviations along each axis, with r_1 corresponding to the direction of greater dispersion, we have that $r_1 = 2\sigma_\phi$ and $r_2 = 2\sigma_\theta$ (since $\sigma_\phi > \sigma_\theta$). We calculate the axes of the SDE with (A5) and (A6) respectively:

$$e_1 = \frac{r_1}{\sqrt{2}} = \frac{2\sigma_\phi}{\sqrt{2}} = \frac{2\left(\frac{2\pi}{\sqrt{12}}\right)}{\sqrt{2}} = \frac{4\pi}{\sqrt{24}}$$
$$e_2 = \frac{r_2}{\sqrt{2}} = \frac{2\sigma_\theta}{\sqrt{2}} = \frac{2\left(\frac{\pi}{\sqrt{12}}\right)}{\sqrt{2}} = \frac{2\pi}{\sqrt{24}}$$

Finally, the area of the ellipse is calculated as

$$A = \pi \cdot e_1 \cdot e_2 = \pi \cdot \frac{4\pi}{\sqrt{24}} \cdot \frac{2\pi}{\sqrt{24}} = \frac{8\pi^3}{24} = \frac{\pi^3}{3} \approx \mathbf{10.34}$$

Q.E.D.

2. EXPLODE.PL

```
#!/usr/bin/perl -w
use warnings;
use strict;
use diagnostics;

my $energy = 8;
my $parallel = 8; # number of simulations to be run simultaneously
my $bin = $ENV{GMXBIN};
my $cwd = `pwd`; chop $cwd;
my $tmp = "$cwd/SIMS";
my $grodir = "$cwd/GRO";
my @tprarray;
my @trrarray;
my @groarray;

mkdir($tmp);
mkdir($grodir);

my @imax = ( '2e12' );
my @pulse = ('0.1','0.05' );

my $fwhm_to_sigma = 2*sqrt(2*log(2));

for (my $k = 0; $k < $ARGV[0]; $k++) {
   foreach $imax ( @imax ) {
      foreach $p ( @pulse ) {
         my $gropath = "$grodir/$imax-$p-$energy";
         if ( -d $gropath ) { system("rm -rf $gropath"); }
         mkdir $gropath,0777;

         for (my $i = 0; $i <= $parallel-1; $i++) {
            my $myrand = int(rand(10000));
            my $dir = "$tmp/$imax-$p-$energy-$myrand";
            if ( -d $dir ) { system("rm -rf $dir"); }
            mkdir $dir,0777;
            chdir $dir;

            my $pulse = $p/$fwhm_to_sigma;
            my $dt = 0.00001; #0.00005;
            my $nsteps = 7/$dt; # simulation length / step length
            my $tinit = -$pulse*3;
            my $nstxout = int($nsteps/500);
            my $nstlog = int(0.0005/$dt);

            print ('*' x 80)."\n".('*' x 80)."\n";
            print "* Starting simulation with the following parameters\n";
            print "* imax = $imax\n";
            print "* pulse = $pulse\n";
            print "* energy = $energy\n";
            print "* nsteps = $nsteps\n";
            print "* nstxout = $nstxout\n";
            print "* tinit = $tinit\n";
            print "* DT = $dt\n";
            print ('*' x 80)."\n".('*' x 80)."\n";

            open FP, ">grompp.mdp";
            print FP <<"EOF";

; VARIOUS PREPROCESSING OPTIONS =
title = Testing imax = $imax, pulse = $pulse, energy = $energy, start = 0
cpp = /lib/cpp
define = -DFLEX_SPC
```

```
; RUN CONTROL PARAMETERS =
integrator = md
; start time and timestep in ps =
tinit = $tinit
dt = $dt
nsteps = $nsteps
; number of steps for center of mass motion removal =
nstcomm = 0

; LANGEVIN DYNAMICS OPTIONS =
; Temparature, friction coefficient (amu/ps) and random seed =
ld_temp = 300
ld_fric = 0
ld_seed = 1993

; ENERGY MINIMIZATION OPTIONS =
emtol = 0.001
emstep = 0.1
nstcgsteep = 1000

; OUTPUT CONTROL OPTIONS =
; Output frequency for coords (x), velocities (v) and forces (f) =
nstxout = $nstxout
nstvout = 0
nstfout = 0
; Output frequency for group stuff and for energies (nstprint) =
nstlog = $nstlog
nstenergy = $nstlog
; Output frequency for xtc files, and associated precision =
nstxtcout = 0
xtc_precision = 1000
; This selects the subset of atoms for the XTC file. =
; Only the first group gets written out, it does not make sense =
; to have multiple groups. By default all atoms will be written =
xtc_grps = Protein
; Selection of energy groups =
energygrps = system

; NEIGHBORSEARCHING PARAMETERS =
; nblist update frequency =
nstlist = 0
; ns algorithm (simple or grid) =
ns_type = simple
deltagrid = 2
; Box type, rectangular, triclinic, none =
pbc = No

; OPTIONS FOR ELECTROSTATICS =
; Method for doing electrostatics =
coulombtype = cut-off
vdwtype = cut-off
; cut-off lengths =
rlist = 0
rcoulomb = 0
rvdw = 0
rvdw_switch = 0
; Dielectric constant (DC) for twin-range or DC of reaction field =
epsilon_r = 1
; Apply long range dispersion corrections for Energy and Pressure =
bdispcorr = no

; OPTIONS FOR WEAK COUPLING ALGORITHMS =
; Temperature coupling =
Tcoupl = no
tc_grps = system
```

```
tau_t = 0.1
ref_t = 300
; Pressure coupling =
Pcoupl = no

; SIMULATED ANNEALING CONTROL =
annealing = no
; Time at which temperature should be zero (ps) =
zero_temp_time = 0

; GENERATE VELOCITIES FOR STARTUP RUN =
gen_vel = yes
gen_temp = 20.0
gen_seed = 173529

; OPTIMIZATIONS FOR SOLVENT MODELS =
; Solvent molecule name (blank: no optimization) =
solvent_optimization = SOL

; Solvent molecule name (blank: no optimization) =
solvent_optimization = SOL
; Number of atoms in solvent model. =
; (Not implemented for non-three atom models) =
nsatoms = 3

; OPTIONS FOR BONDS =
constraints = none;all-bonds
morse = yes

; NMR refinement stuff =
; Distance restraints type: None, Simple or Ensemble =
disre = No

; Free energy control stuff =
free_energy = no

; User defined thingies =
userint1 = $myrand
userint2 = $energy
userint3 = 0
userint4 = 0
userreal1 = 0.0
userreal2 = $imax
userreal3 = $pulse
userreal4 = 100
EOF

            close FP;

            my $tpr = "expprep.tpr";
            push (@tprarray, "$dir/expprep.tpr");

            system("grompp -v -f $dir/grompp.mdp -c $cwd/minimized.gro -p
$cwd/topology.top -o $tpr");

            my $trr = "traj-$imax-$p-$energy-$myrand.trr";
            push (@trrarray, "$dir/$trr");
            my $gro = "$gropath/$imax-$p-$energy-$myrand.gro";
            push (@groarray, $gro);

            my $ion = '';
            if ( $imax != 0 ) { $ion = '-ionize' };
            if ( -f $tpr ) {
               if ( $i != $parallel-1 ) {
                  system("mdrun -v $ion -s $tpr -o $trr &");
               }
               else {
                  system("mdrun -v $ion -s $tpr -o $trr");
               }
            }
            else {
               warn "GROMPP FAILED\n";
            }
            chdir $cwd;
         } #Close for:i (Russian parallelization)
      } #Close foreach:pulse
   } #Close foreach:imax
} #Close for:k

#Convert the trajectory files to human-readable format
for (my $j = 0; $j < scalar @trrarray; $j++) {
   system("echo 1 | trjconv -f $trrarray[$j] -s $tprarray[$j] -o
$groarray[$j]");
}
```

3. ANALYZE.PL

```
#!/usr/bin/perl
use warnings;
use strict;
use diagnostics;
use Math::Trig ':radial';

my $atom = 'C';
my $cwd = `pwd`; chop $cwd;
my $csvdir = "$cwd/CSV";

$ARGV[0] =~ m/GRO\/(.*)-\d+.gro$/;
my $filename = "$atom-$1";

unless ( -d $csvdir ) { mkdir $csvdir,0777; }

system("rm $csvdir/ia_$filename.csv");
open (IA, ">$csvdir/ia_$filename.csv") or die $!;
print IA "Atom\tRandom\tr\ttheta\tphi\n";
close IA;

system("rm $csvdir/fa_$filename.csv");
open (FA, ">$csvdir/fa_$filename.csv") or die $!;
print FA "Atom\tRandom\tr\ttheta\tphi\n";
close FA;

print "Reading the following files:\n@ARGV\n";
my %hash;

foreach (@ARGV) {
   open (TEMP, "<$_") or die $!;
   $_ = m/.*-(\d+).gro$/;
   my $random = $1;
   # Parse each line of the trajecory file and extract spatial
   # data from those lines corresponding to atoms specified in $atom.
   while (<TEMP>) {
      if($_ =~ /\s*\d+\w*\s+$atom[A-z0-9]*\s+(\d+)\s+(-{0,1}\d+.\d+)\s*(-
{0,1}\d+.\d+)\s*(-{0,1}\d+.\d+)\s*.*$/) {
         push(@{$hash{$1}[0]},$2);
         push(@{$hash{$1}[1]},$3);
         push(@{$hash{$1}[2]},$4);
      }
   }

   # Retreive spatial coordinates for each sulfur atom and
   # convert them to spherical coordinates.
   while (my ($key,$value) = each %hash) {
      my $currentAtom = $key;
      my @x = @{@{$value}[0]};
      my @y = @{@{$value}[1]};
      my @z = @{@{$value}[2]};

      my @r;
      my @phi;
      my @theta;

      for (my $i = 0; $i < scalar @x; $i++) {
         (my $r, my $phi, my $theta) = cartesian_to_spherical($x[$i],
$y[$i], $z[$i]);
         $r[$i] = $r;
         $phi[$i] = $phi;
         $theta[$i] = $theta;
      }
      open (IA, ">>$csvdir/ia_$filename.csv") or die $!;
      print IA $currentAtom . "\t" . $random . "\t" . $r[0] . "\t" .
$theta[0] . "\t" . $phi[0] . "\n";
      close IA;

      open (FA, ">>$csvdir/fa_$filename.csv") or die $!;
      print FA $currentAtom . "\t" . $random . "\t" . $r[-1] . "\t" .
$theta[-1] . "\t" . $phi[-1] . "\n";
      close FA;

      #    print $key . "\n";
      #    print "x\ty\tz\t\n";
      #    print $x[0] . "\t" . $y[0] . "\t" . $z[0] . "\n";
      print $x[-1] . "\t" . $y[-1] . "\t" . $z[-1] . "\n";

      #    (my $r_init, my $phi_init, my $theta_init) =
cartesian_to_spherical($x[0], $y[0], $z[0]);
      #    print "Initial coords: \n";
      #    print "r\ttheta\tphi\n";
      #    print $r_init . "\t" . $theta_init . "\t" . $phi_init . "\n";
      #
         # (my $r_final, my $phi_final, my $theta_final) =
cartesian_to_spherical($x[-1], $y[-1], $z[-1]);
         # print "Final coords: \n";
         # print "r\ttheta\tphi\n";
         # print $r_final . "\t" . $theta_final . "\t" . $phi_final . "\n";

   }
   undef %hash;
   close TEMP;
}
```

4. PLOT.R

```
library(aspace)
args <- commandArgs(TRUE)
initial <- read.csv(paste0("CSV/ia_",args[1],".csv"),head=TRUE,sep="\t")
final <- read.csv(paste0("CSV/fa_",args[1],".csv"),head=TRUE,sep="\t")
order <- final[order(final[,1]), ]
atomlist <- order$Atom[!duplicated(order$Atom)]
sdes <- vector()
rinit <- vector()
pdf(paste0(args[1],".pdf"), width=10, height=10)

for (i in atomlist) {
   rinit = append(rinit, mean(initial$r[which(initial$Atom == i)]))
        theta = order$theta[which(order$Atom == i)]
        phi = order$phi[which(order$Atom == i)]
        if (sum(phi < -pi/2 | phi > pi/2) > sum(phi > -pi/2 & phi < pi/2)) {
         phi = ifelse(phi < 0, phi + 2*pi, phi)
         h1 = 0
         h2 = 2*pi
        }
        else {
         h1 = -pi
         h2 = pi
        }
        calc_sde(id=i, calccentre=TRUE, points=cbind(theta,phi))
        sdearea = round(r.SDE$Area.sde, 3)
        sdes = append(sdes,sdearea)
        plot(theta, phi, col = ifelse(order$r[which(order$Atom == i)] < 10,
"red", "black"), main=paste("Atom no. ", i, sep = " "),
xlab=expression(paste("Polar angle ", theta)),
ylab=expression(paste("Azimuthal angle ", phi)), xlim=c(0,pi),
ylim=c(h1,h2), xaxt='n', yaxt='n')
        plot_sde(plotnew=FALSE, plotpoints=FALSE, centre.pch=18,
centre.col="green", titletxt="", sde.col="green")
        abline(v=0)
        abline(h=0)
        abline(v=(seq(0,pi,pi/4)), col="gray", lty="dotted")
        abline(h=(seq(h1,h2,pi/2)), col="gray", lty="dotted")
        axis(1, at = c(0, pi/4, pi/2, 3*pi/4, pi), labels = expression(0,
pi/4, pi/2, 3*pi/4, pi))
        ifelse(h1 == 0, axis(2, at = c(h1, h1+pi/2, h1+pi, h2-pi/2, h2),
labels = expression(0, pi/2, pi, 3*pi/2, 2*pi)), axis(2, at = c(h1, h1+pi/2,
h1+pi, h2-pi/2, h2), labels = expression(-pi, -pi/2, 0, pi/2, pi)))
        par(xpd=TRUE)
        text(2.75, ifelse(h1 == 0, 3.5+pi, 3.5), paste("SDE area: ",
sdearea))
        par(xpd=FALSE)
}
dev.off()
foot <- cbind(rinit,atomlist,sdes)
print <- foot[order(foot[,1]), ]
pdf(paste0("fp_",args[1],".pdf"), width=10, height=10)
plot.new()
frame()
plot(print[,3], type="l")
dev.off()
```