

Molecular Geometry Optimisation

Using Evolutionary Computation

Dissertation Project

Developer: Sophie Turner

Supervisor: Dr. David Walker

May 2021

Contents

[Abstract 2](#_Toc68352781)

[Introduction 2](#_Toc68352782)

[Links to Work 2](#_Toc68352783)

[Background 2](#_Toc68352784)

[Aims 3](#_Toc68352785)

[Literature Review 3](#_Toc68352786)

[Project Plan 4](#_Toc68352787)

[Roadmap & Sprint plans 4](#_Toc68352788)

[Risk Assessment 7](#_Toc68352789)

[Legal, Social, Ethical and Professional Matters 8](#_Toc68352790)

[Implementation 9](#_Toc68352791)

[Design Patterns & Programming Conventions 9](#_Toc68352792)

[Technologies Used 9](#_Toc68352793)

[Energy Calculators 9](#_Toc68352794)

[User Interface 9](#_Toc68352795)

[Sprint Zero – Planning 9](#_Toc68352796)

[Sprint One – Building a Molecule 10](#_Toc68352797)

[Sprint Two – Predicting a Shape 12](#_Toc68352798)

[Sprint Three – Other Algorithms 13](#_Toc68352799)

[Sprint Four – Viewing Results 14](#_Toc68352800)

[Sprint Five – Testing 15](#_Toc68352801)

[Sprint Six – Interactivity 15](#_Toc68352802)

[Sprint Seven – Testing 15](#_Toc68352803)

[Sprint Eight – Final Touches 15](#_Toc68352804)

[Sprint Nine – Showcase Materials and Report 15](#_Toc68352805)

[Algorithms 16](#_Toc68352806)

[Many-Molecule Evolutionary Algorithm 16](#_Toc68352807)

[Per-Atom Exhaustive Test 16](#_Toc68352808)

[Other Algorithms 16](#_Toc68352809)

[User Feedback 16](#_Toc68352810)

[Discussion & Evaluation 16](#_Toc68352811)

[Conclusions 17](#_Toc68352812)

[Suggestions 17](#_Toc68352813)

[References 17](#_Toc68352814)

[Appendices 17](#_Toc68352815)

[Appendix x – Questionnaire Information Sheet 17](#_Toc68352816)

[Appendix x – Target User Questionnaire 19](#_Toc68352817)

[Appendix x – Participant A’s Responses to Target User Questionnaire 22](#_Toc68352818)

[Appendix x – Participant B’s Responses to Target User Questionnaire 24](#_Toc68352819)

[Appendix x – First Usability Test Questionnaire 26](#_Toc68352820)

[Appendix x – Participant C’s Responses to First Usability Test Questionnaire 28](#_Toc68352821)

[Appendix x – Participant D’s Responses to First Usability Test Questionnaire 29](#_Toc68352822)

[Appendix x – Participant E’s Responses to First Usability Test Questionnaire 30](#_Toc68352823)

[Appendix x – Second Usability Test Questionnaire 30](#_Toc68352824)

[Appendix x – Participants’ Responses to Second Usability Test Questionnaire 30](#_Toc68352825)

[Appendix x – Project Initiation Document 30](#_Toc68352826)

[Appendix x – Performance Comparison 32](#_Toc68352827)

# Abstract

The program is for chemists and physicists whowant to estimate and view the structures of molecules without spending hours doing calculations or using a supercomputer. The program’s name is Geopt and it uses an evolutionary algorithm to predict the shapes of theoretical molecules.

# Introduction

Find first references of acronyms!

## Links to Work

YouTube video of demonstration of the program at sprint five: <https://youtu.be/ylv4J85m95Y>

YouTube video of demonstration of the program at sprint seven: <https://youtu.be/llvHbYyEO6Q>

YouTube video of demonstration of the program later in sprint seven: <https://youtu.be/irX853Cr7zk>

YouTube video of demonstration of the final product:

GitHub repository: <https://github.com/Squidgeypea/SophieCOMP3000/tree/main>

Microsoft Planner:

Showcase:

## Background

The purpose of energy optimisation in molecular geometry is to find the most probable configuration of a molecule or other system as it occurs in the real world. This allows scientists to make more accurate calculations and design more efficient chemical processes for industrial uses. The process can be for intermolecular as well as intramolecular bonds or forces. This project focussed on intramolecular bonds, considering a single molecule.

By considering the wave-like properties of electrons, the Schrödinger equation could theoretically give accurate results for the energy of a system, but it is not possible to obtain a precise value for a system of more than one electron from the Schrödinger equation due to the movement and repulsion of electrons. Furthermore, the computational expense required to calculate all possible solutions using this method would be unreasonable as it is NP-hard and, therefore, it is believed that not even a super-computer could perform this task in a reasonable amount of time, as explained by Matthews et al. (2020). For this reason, computer programs must attempt to estimate the correct values by making assumptions about the system in order to simplify the calculation methods and by finding alternative formulae to approximate the values of energy, forces and other properties. The output of the algorithm can be compared to the known shape, structure and total potential energy of the molecule to determine its accuracy.

The unit of measurement for distances within molecules in this project is the angstrom, whose symbol is Å. It is equivalent to 10⁻¹⁰ metres. The report refers to distances between atoms rather than ‘bond lengths’ because the algorithms and calculation methods used in this project do not take into account any information about bonding and have no knowledge of the existence, types or locations of bonds present in the molecule. In real applications, a chemist might only be interested in one specific bond in a molecule because they may be considering the chemical reactions that could involve that particular bond.

## Aims

The aim of this project was to create an interactive evolutionary algorithm (EA) to predict the geometric structure of a molecule or system of atoms, based only on an estimate of the system’s total potential energy.

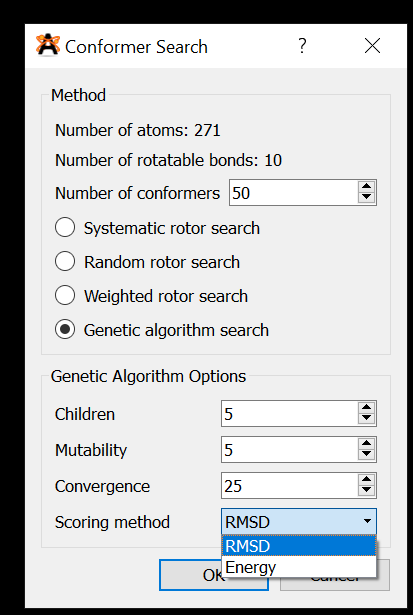
This is a one-objective problem. The objective is to find the arrangement of atoms, and therefore their bonds, where the net force on each atom is as close to zero as possible.

The aim of this project is to create an evolutionary algorithm to find energetically optimal configurations of atoms in space. The project will use Python because there are many scientific tools available in Python which will be useful.

# Literature Review

PubChem, Avogadro, Biovia and Chemdraw. One of them finds geometry by searching a database for matching structures. They don't have interactive EAs, but from watching YouTube videos about Gaussian & GaussView, I saw that this program does have an interactive EA for geometry optimisation. I can't use Gaussian because it is expensive.

Avogadro



Avogadro glitches all the time and doesn’t display any view of the molecule. Hard to use.

Pubchem is also hard to use and it’s difficult to join up atoms.

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1145/1068009.1068339>

These guys ^ used an interactive EA where the user (an experienced chemist) contributed to the fitness function, allowing the chemist to influence the EA. This gives the chemist the ability to use their knowledge to improve the outcome.

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1109/SC.2014.61>

some DFT stuff ^ does DFT already have an EA in it? A lot of this has been done on supercomputers…. :/

<https://www.nature.com/articles/s42004-020-0255-8>

Buckyball EA ^

<https://www.sciencedirect.com/science/article/abs/pii/000926149087064X>

another DFT algo ^

<http://kitchingroup.cheme.cmu.edu/dft-book/dft.html>

a function of the electron density, DFT. approximations

It uses the wave function. It assumes that the wave function is the sum of the waves. Coefficients of each plane wave (each electron) are altered until the energy is the lowest because the wave function is used to work out the energy in DFT. This is done using matrix algebra.

Sometimes theories are applied to atoms or systems which couldn’t actually exist and don’t follow the laws of physics.

The energy for each plane wave in the system is changed (like decision variable mutation) to find the lowest overall energy. The more combinations that are tried, the better the results, just as in an EA. In other words, a perfect result would use an infinite number of plane wave functions.

Implementation, Validation and Profiling of a Genetic Algorithm for Molecular Conformational Optimization

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1145/2987491.2987529>

we test our implementation across a range of data set sizes to characterize the performance of the algorithm as chain length increases: benchmarking that is necessary for future optimization and parallelization of the algorithm.

Determination of the three dimensional structure of an arbitrary protein from the sequence of its constituent amino acids has been identified as one of the ten most sought after solutions in protein bioinformatics

The most accurate atom models are quantum-mechanical. However, although the interactions between atoms are governed by quantum mechanics, currently it is not feasible to perform a precise quantum mechanical conformational optimization for large molecules such as proteins.

even with the simplest models, the conformational space available to a protein is enormous

an exact solution to the conformational optimization problem rapidly becomes infeasible as the length of the input protein chain increases

increased performance (as measured by fewer function evaluations)

Molecular parameter optimization gateway (ParamChem): workflow management through TeraGrid ASTA

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1145/2016741.2016779>

Parameter optimization for chemical systems requires generation of initial guesses. These parameters should be generated using systematic sampling of parameter space, minimizing differences between output data and the corresponding reference data

(old) A discrete-continuous algorithm for molecular energy minimization

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.5555/147877.148130>

minimum energies for clusters of up to n = 1000 atoms using a massively parallel processor, the Thinking Machines CM-5. (Back in 1992)

Developing a high-performance quantum chemistry program with a dynamic scripting language

<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1145/2830168.2830170>

Python to develop first-principles quantum chemistry programs for high-performance computing environments. Dynamic scripting programming languages, in general, have distinct advantages in terms of developer productivity over compiled languages such as C/C++ and Fortran, because of their ease of use and extensive libraries.

Using a Meta-GA for parametric optimization of simple gas in the computational chemistry domain

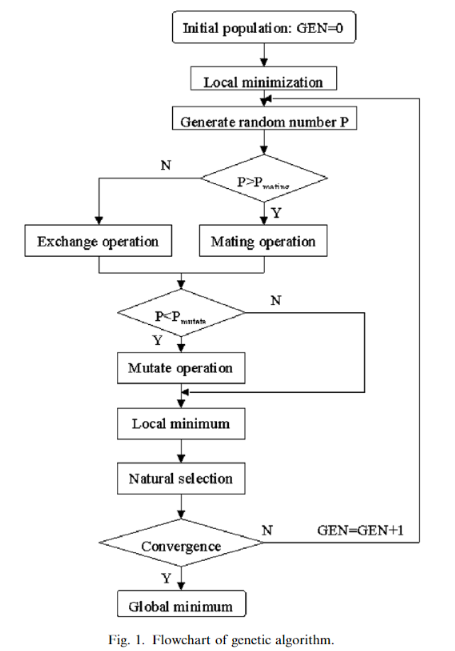
<https://dl-acm-org.plymouth.idm.oclc.org/doi/10.1145/1830483.1830630>

The determination of the lowest energy conformer for long-chain molecules by exhaustive search methods quickly becomes infeasible as the length increases. Typically, resources required are proportional to the number of possible conformers (shapes), O(3^n) where n is the length.

A genetic algorithm (GA) that calculates energies in a feasible time is described, using an open-source off-the shelf tool, PyEvolve.

Structures and properties of Si6N8 clusters: Genetic algorithm and density functional theory approach

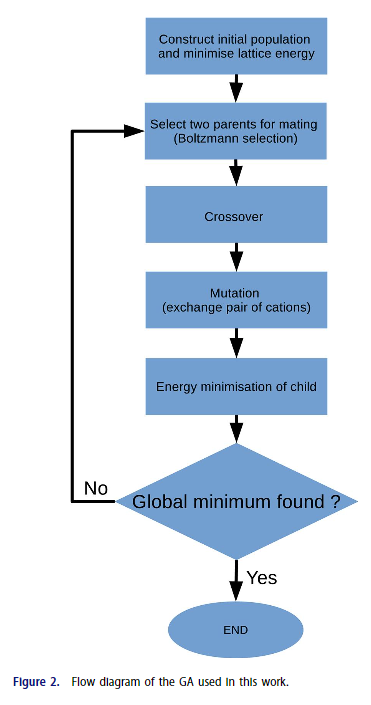
<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=2&SID=E13awNKKeScO16zvHhl&page=1&doc=2>



Predicting cation ordering in MgAl2O4 using genetic algorithms and density functional theory

<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=2&SID=E13awNKKeScO16zvHhl&page=1&doc=7>

energy minimizations are expensive at the DFT level



<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=9&SID=E13awNKKeScO16zvHhl&page=1&doc=1>

Lots of maths

Read this because it’s for Benzene, not a big molecule.

Geometry optimisation of aluminium clusters using a genetic algorithm

<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=9&SID=E13awNKKeScO16zvHhl&page=1&doc=4>

About unit cells and crystal structures.

face-centred cubic, hexagonal close packed, decahedral and icosahedralstructures



Mixing parameters for geometry optimization using the Hamiltonian algorithm

<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=9&SID=E13awNKKeScO16zvHhl&page=1&doc=9>

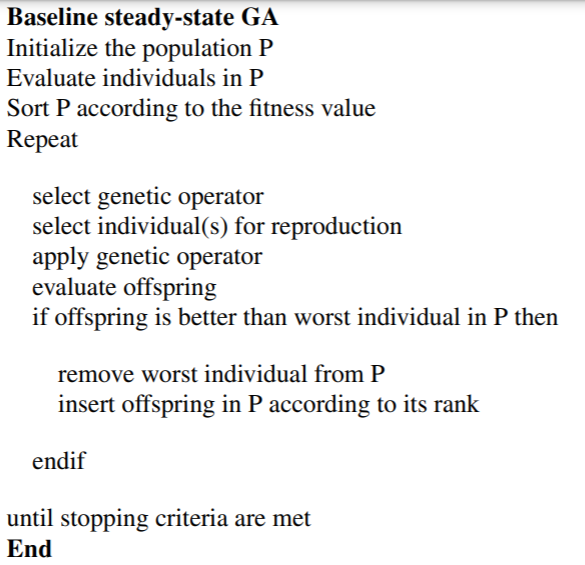
We choose the dihedral angle -C-C-C-C- of the butane molecule

The Monte Carlo (MC) method, as well as the simulated annealing method, is usually applied to escape from a local minimum. In the MC method, a large part of the computational time is wasted in calculating an unstable structure of higher energy. In the simulated annealing method, it is difficult to control the temperature of the critical point. Optimization using the above two methods combined withab initio molecular orbital calculations is considered difficult and has not been reported to the authors’ best knowledge.

HA enables the effective search of the potential energy surface.

A GA-simplex hybrid algorithm for global minimization of molecular potential energy <https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=19&SID=E13awNKKeScO16zvHhl&page=1&doc=7>

the global minimum of the potential energy of a molecule corresponds to its most stable conformation, which dictates its properties. The search for the global minimum of a potential energy function is very difficult since the number of local minima grows exponentially with molecule size.



Make a pseudo code like this ^ to show our algorithms.

Investigation of an effective medium theory for metallic periodic structures: a fitting-based approach

<https://apps-webofknowledge-com.plymouth.idm.oclc.org/full_record.do?product=WOS&search_mode=GeneralSearch&qid=42&SID=E13awNKKeScO16zvHhl&page=1&doc=7>

While the effective medium theory (EMT) has been useful to explain optical characteristics of a dielectric periodic structure analytically, it has failed to describe metallic structures correctly.

Searching for an effective medium has been performed through binary searches rather than a time-consuming simulated-annealing algorithm.

Include the original EMT literature!

# Project Plan

## Roadmap & Sprint plans

Tables x to x show the roadmap, which outlined the plan for the sprints. The plan was that each sprint would last a fortnight and would be based around a user story, apart from the sprints that were reserved for the planning and testing stages of the project. Sprint zero, shown in table x, was dedicated to planning the project. Sprint one, shown in table x, was to fulfil the user story: a user wishes to create a molecule. Sprint two, shown in table x, was to fulfil the user story: a user wishes to predict a shape. Sprint three, shown in table x, was to fulfil the user story: a user wishes to use a different EA. Sprint four, shown in table x, was to fulfil the user story: a user wishes to view a molecule and its analytical information, including a potential energy surface. Sprint five, shown in table x, was dedicated to usability testing and making changes to the program. Sprint six, shown in table x, was to fulfil the user story: a user wishes to interact with the algorithm. Sprint seven shown in table x, was dedicated to usability testing and making changes to the program. Sprint eight, shown in table x, was dedicated to making finishing touches and bring the programming part of the project to an end. Sprint nine, shown in table x, was reserved for creating showcase materials and making the transition from the coding stage to the report writing stage.

The time after sprint nine was planned to be for writing the report, although continued code development during this time was not explicitly ruled out. Talk about agile being flexible and get a reference.

## Risk Assessment

Table x shows the risk assessment for the project, which was created during sprint zero, the planning phase. The ‘exposure’ measure is the ‘likelihood’ rating multiplied by the ‘impact’ rating and can be used as an indication of the severity of the risk.

Eventually, all six of the identified risks were realised in minor forms. This suggested that the likelihoods of the risks had been underestimated, a mistake to bear in mind in future projects, along with the discovery that the one risk often led to another. Risks R1 (calculations too computationally expensive) and R2 (evolutionary algorithms take too long) were linked because they were affected by the same code, so they occurred simultaneously. The planned measures for these risks were to use optimisations such as multiprocessing and try different techniques. This was effective and was added as an interactive feature which the user may adjust.

Risk R3 (programming difficulties/lack of knowledge) occurred at the same time as R5 (unable to get desired results from algorithms), for the same reason. The algorithm often produced undesirable and inaccurate results because it used an energy calculation which was not suitable for the molecule chosen. More knowledge of the area could have enabled the programmer to improve the way in which the calculations were performed, leading to better results. The planned measures for these risks were to research the area and discuss any problems with the supervisor. The depth of research into the area was limited by the fact that the priority for the project was computer science, not chemistry, and the supervisor’s advice was that the algorithms’ success should be measured, primarily, from a computational point of view. It was important to bear this in mind to avoid risk R4 (too much to do/not finish on time), which had the maximum exposure rating (nine out of nine). Risks R4 and R6 (coronavirus or illness) were also linked because the developer was unwell for some time in March, which meant that the progression of the project was delayed by up to two weeks. The planned responses to these risks were to prioritise the workload and use Agile, as well as knowing the extenuating circumstances (EC) policy for deadline extension. An extension was not sought, but the prior planning and Agile methodology helped the project to get back on track after the set-back and it was possible to alter the plan in order to catch up.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ref | Risk event | Likelihood  1=low  3=high | Impact  1=low  3=high | Exposure  1=min  9=max | Plan |
| R1 | Calculations too computationally expensive | 2 | 2 | 4 | Use Python packages for calculations. Choose appropriate methods. Simplify calculations. Try multithreading/GPU options etc. |
| R2 | Evolutionary algorithms take too long | 3 | 2 | 6 | Use fewer iterations. Use different technique, selection criteria etc. Make code simpler. |
| R3 | Programming difficulties/lack of knowledge | 3 | 2 | 6 | Have regular meetings with David. Do lots of research and practice. |
| R4 | Too much to do/not finish on time | 3 | 3 | 9 | Keep working throughout the year. Adjust plans if necessary. Focus on most important things first. Use Agile. |
| R5 | Unable to get desired results from algorithms | 2 | 1 | 2 | Alter mutations etc. Analyse and test algorithms. Do plenty of research. |
| R6 | Coronavirus or illness | 2 | 3 | 6 | Know the University’s EC policies. Work from home where possible. |

*Table x – Risk assessment.*

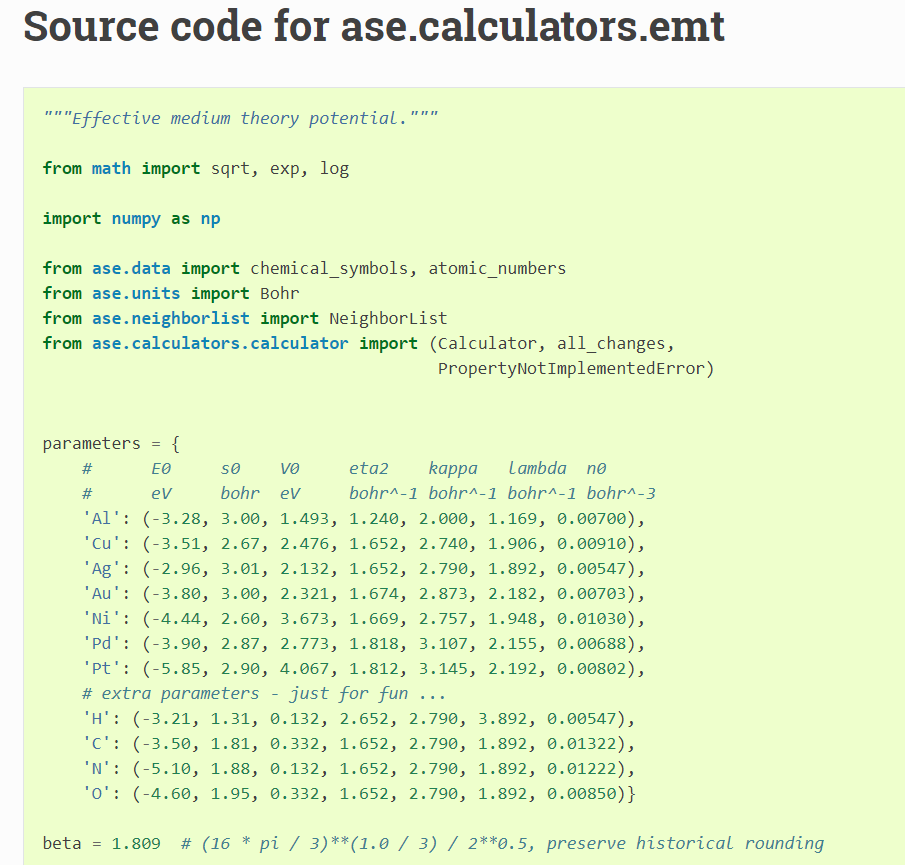
## Legal, Social, Ethical and Professional Matters

Ethics of AI

Using others’ code

All the images shown in the application, including the logo, were created by the developer. The background image of the poster was created by Filipe (2017).

For all three stages of user feedback, the generic ethical approval documentation supplied by Dr. Shirley Atkinson was used because all the participants were students of the University of Plymouth. As required by the University, consent forms and information sheets were written for the user feedback tests and questionnaires. This documentation can be viewed in appendices x to x. No identifying information about participants is given in this report or its appendices.



# Implementation

## Design Patterns & Programming Conventions

The project will use a cyclic approach rather than the waterfall method. This is because it will be important not to leave the analysis to the end of the project, as doing it incrementally throughout the course of the project will help the programmer to choose better designs and parameters for the algorithms.

## Technologies Used

Use Python because it’s easier to show graphs, plots and models in Python. Python is preferable for this project because it has more relevant functions available for the topic.

## Energy Calculators

## User Interface

## Sprint Zero – Planning

move some of this writing to other sections.

Once the project idea was approved and the supervisor was determined, some research was done into the subject area, as described in the literature review of this report. Software that performed similar tasks was identified and studied, and included PubChem, Avogadro, Biovia and Chemdraw. This software revealed some requirements, such as visualisation of the structure as an image and the option for the user to type the desired molecular formula. Additionally, some undesirable features were presented by the software; some of the user interfaces had a large number of unexplained options on the screen which could be overwhelming or confusing to the user. Geopt was planned to be unique by suggesting multiple structures which could potentially even be isomers in some circumstances, and by not restricting the molecule creation to realistic formulae so that the user could study theoretical systems that would be unlikely to occur in reality. why? find a source/example? Geopt would also use random initial molecule structures rather than estimating initial starting positions based on chemical properties. This became the main topic of research for this project; it was not known whether any geometric structure could successfully be estimated purely through randomisation, with no *a priori* knowledge of the system. It was not anticipated that all types of molecule could be modelled in this way, but perhaps it could work for some simple molecules. It was also expected that the number of atoms in the molecule would be restricted to a small value by the computational expense required.

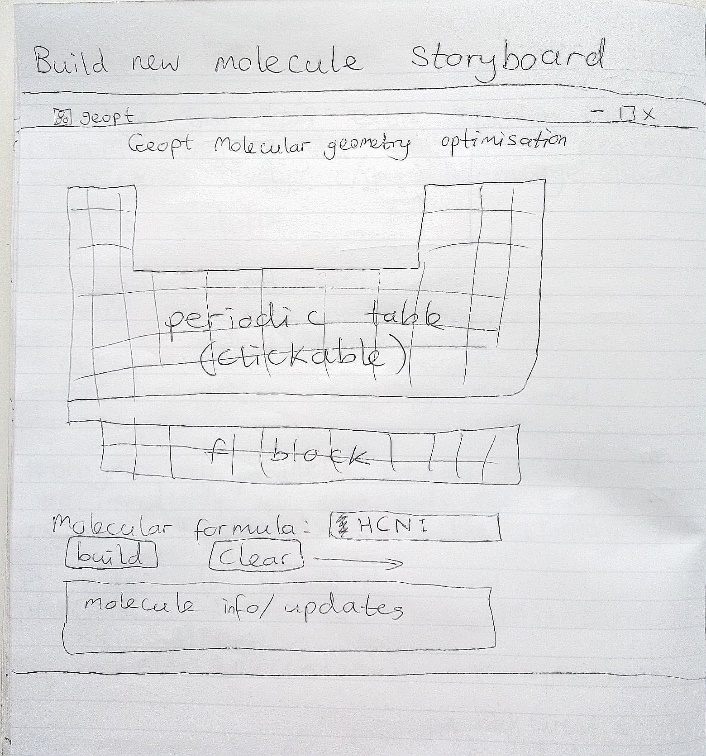
A questionnaire was created and sent to target users, science students, so that the project could be designed to meet their requirements. This questionnaire received two responses which can be seen in appendices x and x. The full, uncompleted version of the questionnaire can be found in appendix x.

After this, the backlog began and the roadmap was composed. The GitHub repository, Microsoft Planner and ReadMe file were created and the primary language was chosen as Python because…. The EMT calculator was discovered, tested and chosen.

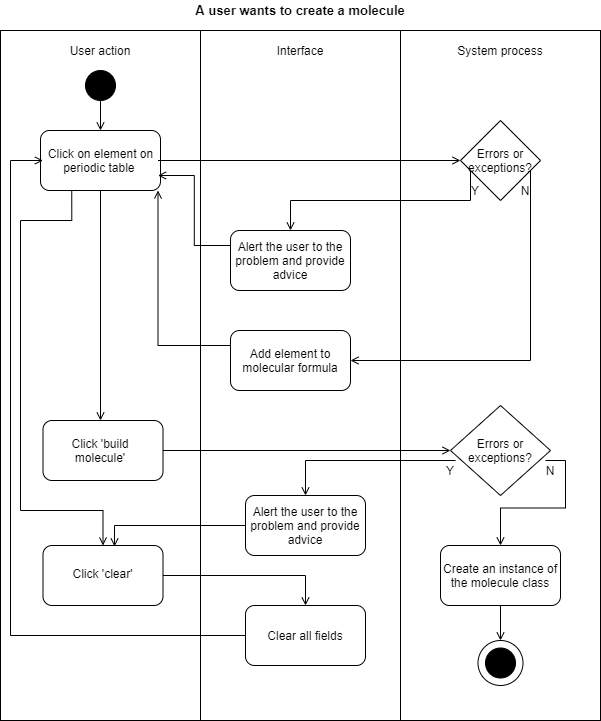
The project initiation document was compiled and submitted. This can be viewed in appendix x. The supervisor suggested that the project may have been at risk of focussing too much on chemistry and not enough on computer science. In response to this, it was planned that pre-existing Python modules would be used for energy calculations and other chemical properties so that the majority of time and effort could be directed towards computing-related tasks.

## Sprint One – Building a Molecule

After the main planning phase, sprint one was planned. It was based around the user story, ‘a user wishes to create a molecule’. Firstly, a page layout plan was drawn, and is shown in figure x. Secondly, an activity diagram was drawn and is shown in figure x. Explain the figs! It was planned that the user would be able to select elements from a periodic table or type a molecular formula to build a molecule by instantiating a ‘molecule’ class. Later in the project, this was changed, as discussed further in this report. where? The entry point of the application and user interface was created in Python. The user interface for this task was built using the Tkinter tool in Python. An XML file was constructed to hold information about each element, and this was used to populate a clickable periodic table. Explain this algorithm and show a code fig.



*Figure x – Layout design for sprint one.* Draw it up properly!



*Figure x – An activity diagram for sprint one.*

During the second week of the sprint, the idea for the user interface was changed to improve the layout. Explain this and show a fig. The string typed into the text entry widget, which was for the user to type the molecular formula, required several steps of processing to convert it to a list of atoms. Explain this algorithm and show a fig. The algorithm iterates through the string, picking out capital letters, lowercase letters and digits to find the constituent atoms.

No need to have a molecule class as we won’t be building more than one at once anyway.

## Sprint Two – Predicting a Shape

Sprint 2 is for the user story, 'a user wants to predict a geometric structure'. The calculations are to place the atoms in a grid so that they are evenly distributed within the grid. Atoms are different sizes, which meant that extra calculations needed to be done to work out the atom's size and adjust their position and the grid size accordingly. I used the XML for this, which I created last week. This was possible because the atomic radii follow a pattern on the periodic table, so their place on the table was used to estimate a relative size value. I did some unit tests. The cell scales to the largest atom.

Place the molecule into a unit cell with relative co-ordinates of the atoms. These co-ordinates can then be used in functions which approximate the overall energy of the molecule (because the Schrödinger equation cannot reasonably be used). There are many different functions for this purpose, available from various Python packages.

Create an evolutionary algorithm to adjust the positions of the atoms to find the lowest overall energy.

activity diagram, a UML diagram, a plan of the cell structure

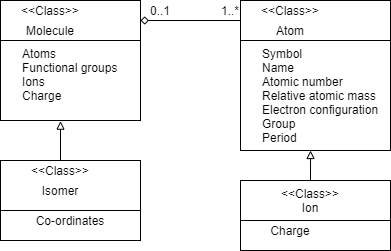
create atom and ion classes to match the UML diagram

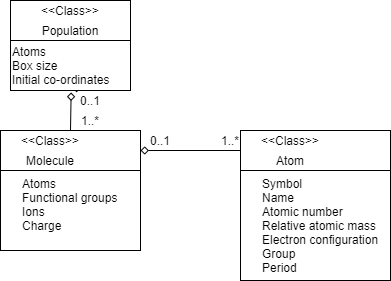
create an EA that could predict the shape of the molecules created in sprint 1. I made functions for selection, random generation, structured generation, permutation, mutation and crossover. I tried various different methods, some using all these functions and some only using a couple of them. I tried a large variety of parameters for these methods and did all this in the 'Test' directory of the application.

keep track of which methods and parameters worked best and worst. After days of tweaking, the results were disappointing and all my EAs were unable to predict geometries well, no matter what I tried, although I did manage to drastically improve the energy values, which was the main objective.

improved the energy predictions by three times of what I'd achieved when creating the first EA, which was good. I settled on an EA which created a population of many permutations of a molecule, chose the best permutation, created many mutated copies of it, introduced some completely random structures, selected the best of these, and continued like this until the best energy didn't change for a few iterations. The problem of getting trapped in local minima was much harder to overcome than I'd expected. I created a function to correct a tendency for Hydrogen atoms to 'fall off' the molecule, and pushed them towards carbon atoms, but this didn't seem to help and actually made the permutations less favourable in some cases, so I omitted it in the final EA.

Unit cell size and shape – The theoretical space around the molecule affects the distance and interactions between multiple molecules





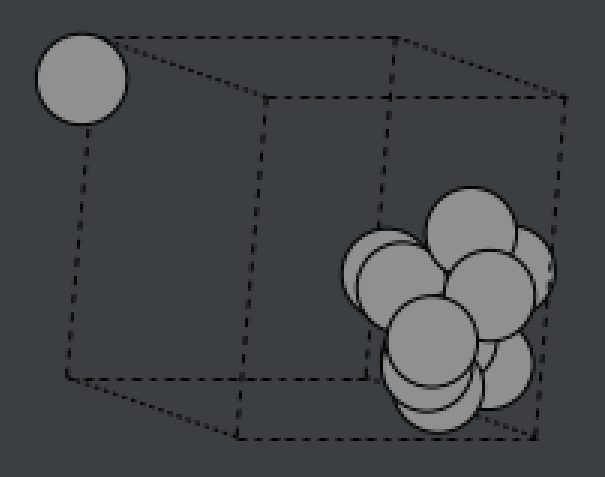
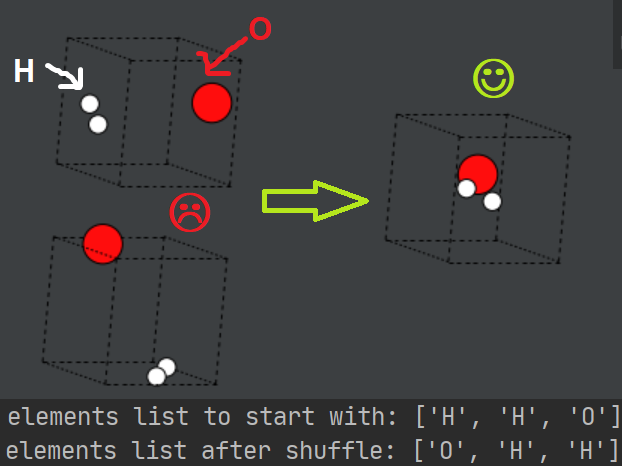
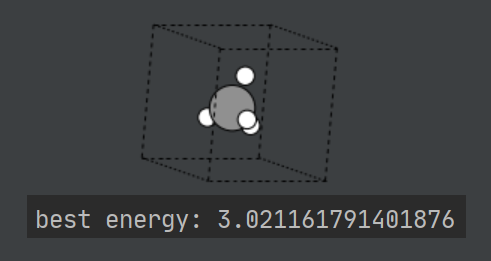
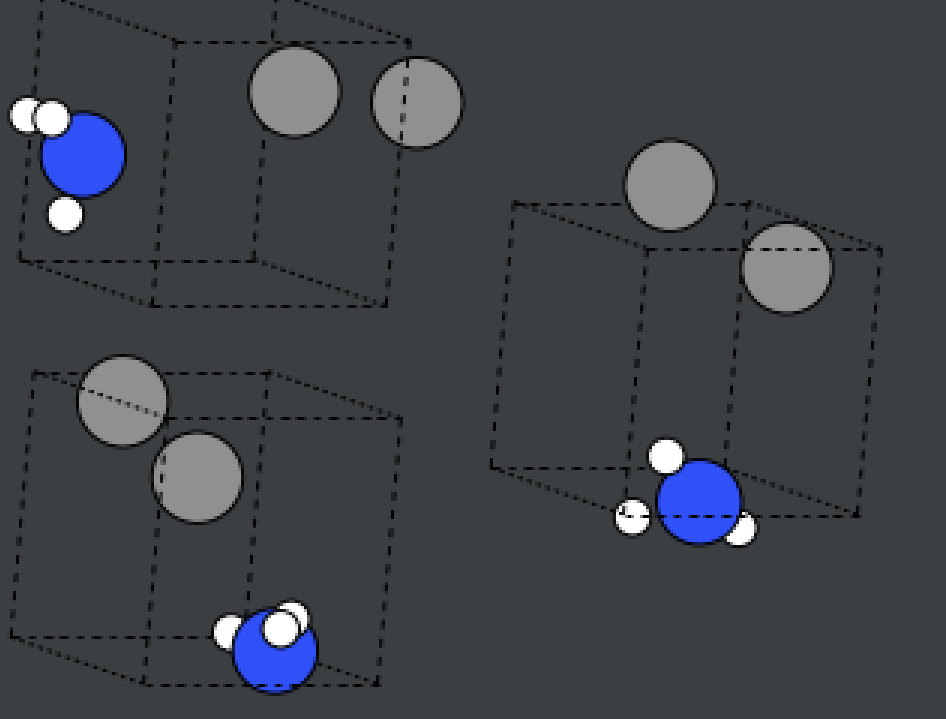
## Sprint Three – Other Algorithms

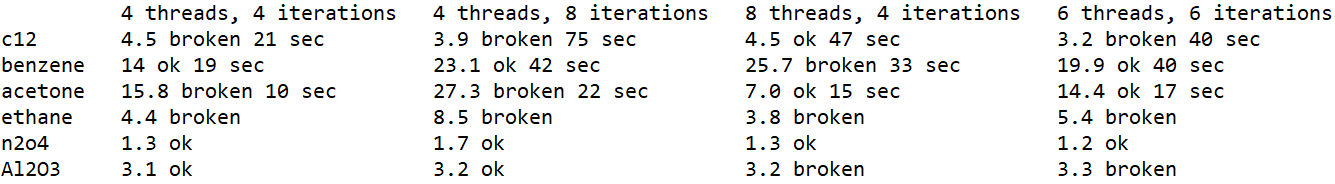
When I originally planned the project I had expected the EAs to perform quite poorly, so had set aside another sprint for creating new algorithms. I tried an EA which started with an empty cell and introduced one atom at a time, moving all other atoms systematically around one another on each atom addition, continuously testing for the best energy. The main problem with this was that since every atom moved around every other atom each time, the time taken to complete the algorithm increased exponentially with the input size. For this reason, I needed to use a small population size and a low number of iterations, which meant that not enough variations or comparisons were made, leading to inaccurate results. In some cases, this meant the best energy achieved by this algorithm was ten times worse than that of the first algorithm! The standard deviation of the Gaussian distribution within which each atom was moved around the other atoms was altered as the EA progressed, as it seemed to have a high impact on the best energy achieved. I would have liked to create a function to constantly adapt this range, but didn't, partly because I needed to work on my other modules, and partly because this EA already took a long time for larger molecules, with thousands of calculations, function calls and array changes per molecule.

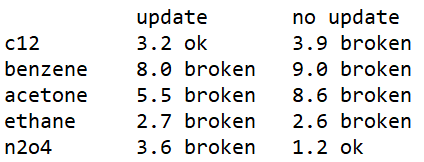
It was decided that the best ways to further improve these EAs were to use a more accurate calculator, such as VASP, as the EMT calculator used so far only works well for a few elements, and to use Python multi-processing to speed them up.

the per-atom EA was improved by scaling atom movements relative to their size and their neighbours' sizes and positions, and by using Python multiprocessing to speed it up. I've attached some pictures of common problems that have arisen with the EAs. These problems are caused by the fact that, if the EA runs for too many iterations, the molecule starts to break apart, as the energy calculator doesn't anticipate the energy required to break bonds and just analyses the ground state. This means that the energy is lowest overall when no atom is interacting with any other atom, and will cause atoms to move apart. Additionally, the most electronegative atom has a tendency to 'steal' all hydrogen atoms and push other atoms away. I implemented various corrections for this which will be discussed in the final report.

The goal was to find a set of parameters that generalises to all molecules, rather than working well for some but poorly for others.





## Sprint Four – Viewing Results

Sprint 4 was originally going to be testing, user feedback and amendments, and sprint 5 was planned to be to display visualisations and analytical info, but I decided to swap these two sprints around because without the visualisations, there isn't much for the users to comment on. I don't expect most users would understand or appreciate the evolutionary algorithms or the chemistry as much as the user interface.

designed the layout and features of the user interface and placed the best molecule on the screen.

Sprint four was started before the Christmas holiday, and continued after it, as per the roadmap plan. The analysis view was expanded to show the 3 best versions of the molecule so that the user can view the differences between them. Datasets were created to hold information needed for the plots. This was done to avoid needing to iterate back through all the previous solutions to create the plot, which caused a delay on loading the view. The plots were displayed using a grid layout but then I decided to include a potential energy surface (PES) and a plot of all tested co-ordinates as well, which meant that I couldn't fit all the plots on the screen at once without shrinking them too much to read easily. I redesigned the layout and began again from scratch, rolling back all the work I'd done on the analysis part of the GUI! Although this added quite a lot of extra time to the sprint, it was worth it, because the new layout is much better and everything fits on the page. I had a problem with the axis rotations on the PES plot not displaying properly. The solution I found was to allow the user to click on the version of the atom for more information, which brings up a new window with larger plots and more details.

The energies of molecules were scaled to fit within the range 0 to 1. This was done using my own implementation because it was faster than using Python's existing functions.

axis rotation problem

information page

diatomic because the PES would be 1D and the PES graph would be 2d as the only factor to consider (degrees of freedom) is bond length.

a triatomic molecule eg. water. The PES graph is 4d so can use colour + space. Bent molecules with 3 different atoms can theoretically have 2 different length bonds but most will have both bonds the same length.

Don’t go all the way down to bond length=0. Nuclear fusion isn’t part of this topic!

Made a plot of all tested positions of atoms in the molecule. This shows where the algorithm tried placing them during its evolution. Points were labelled according to atom symbol and scaled and coloured according to energy.

Tried standardisation, minmax scaler etc but came up with my own version because it was simpler. It is just energy / highest energy. Subtracts that number from green and add it to red for energy scale.

Build up datasets for plots and stats while the algorithms are running.

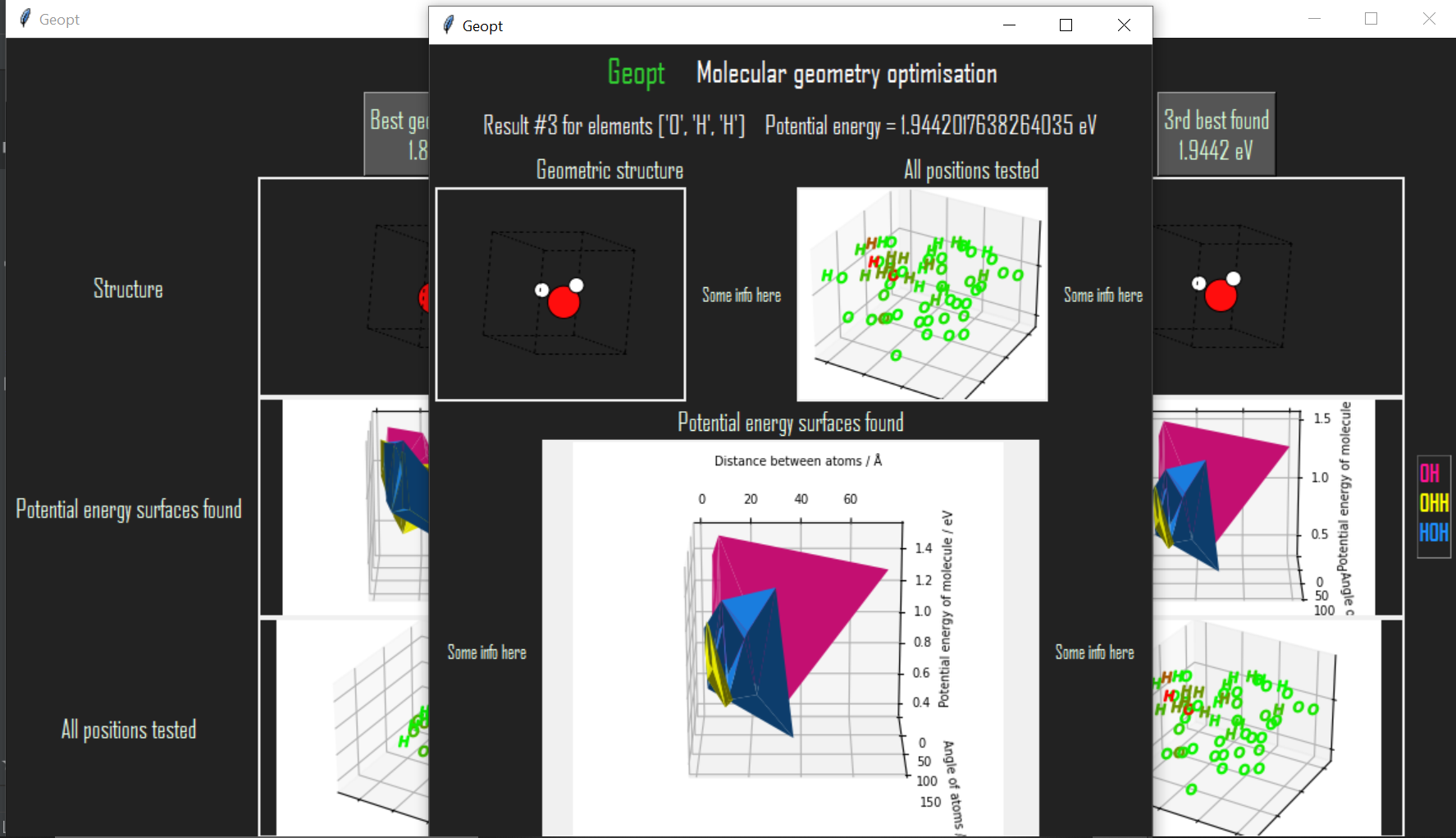
Many different things computed inside same loops.

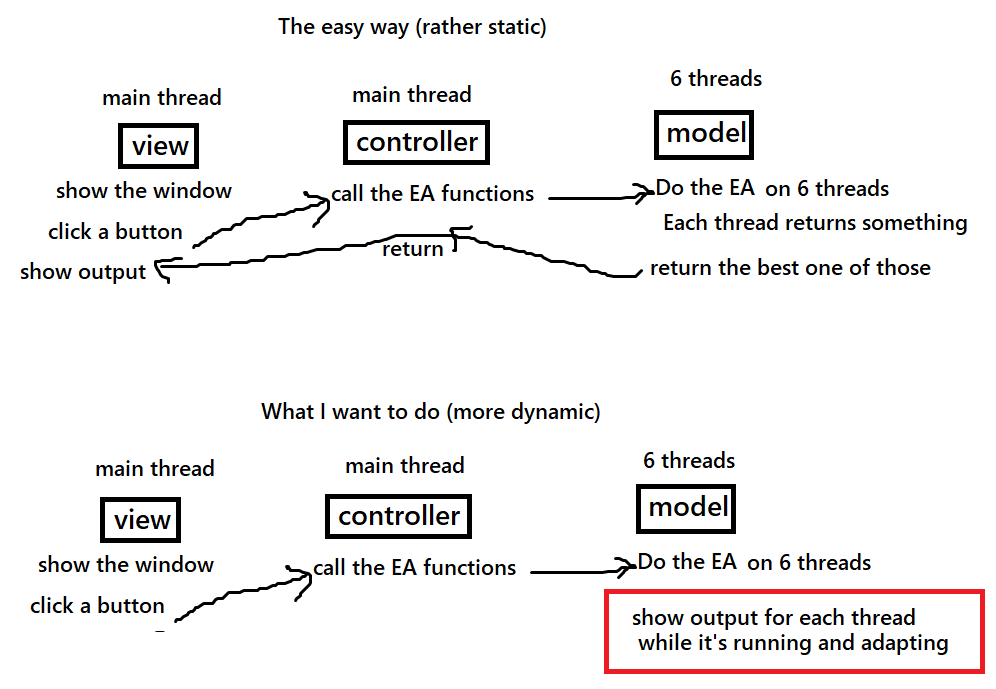
Code is not so separable (not as SOLID) but it is faster to execute.

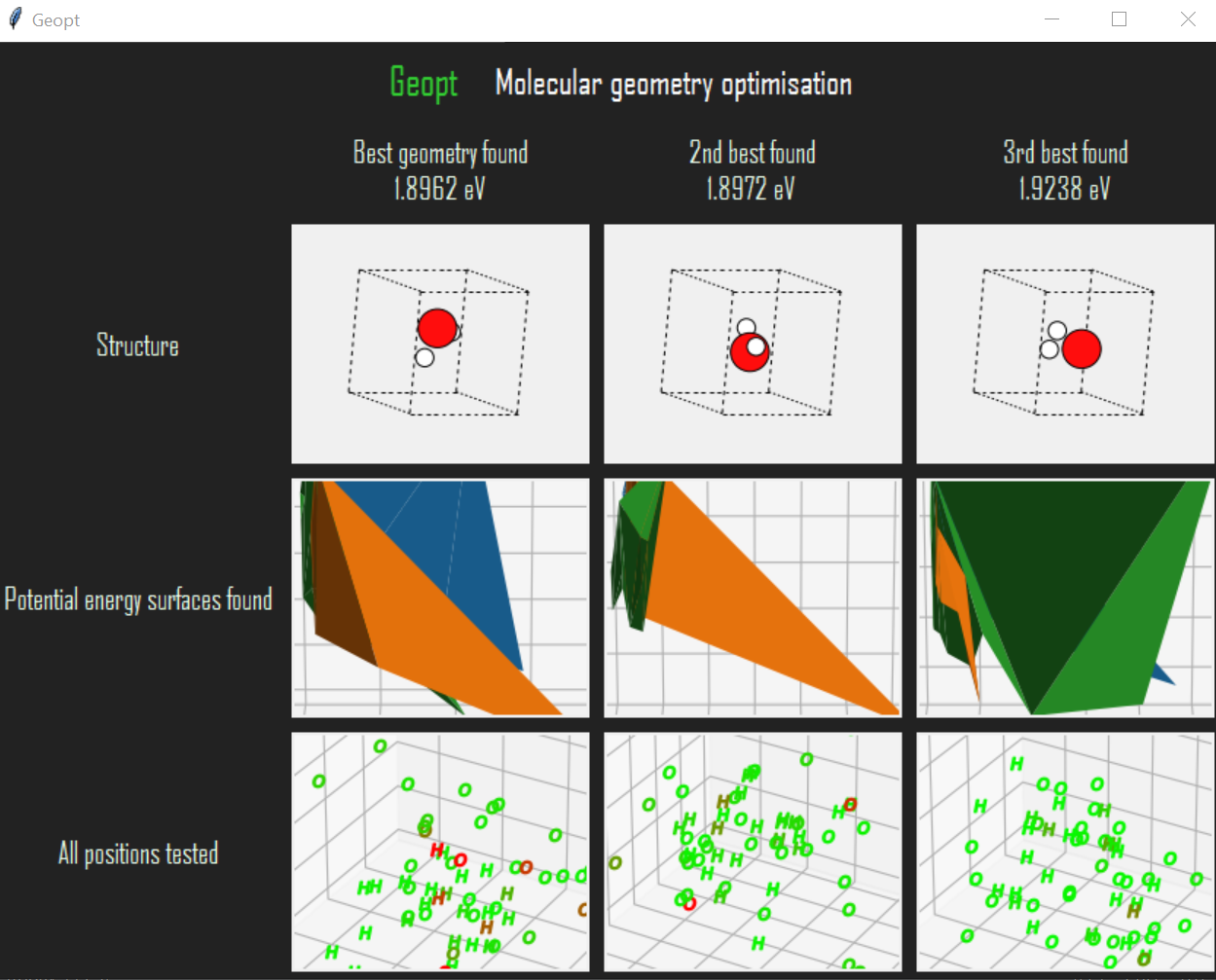
3d plot of angle over 3 atoms, distance between 2 of those 3 atoms and the total potential energy of the molecule. Difficult because each axis has a different number of variables.

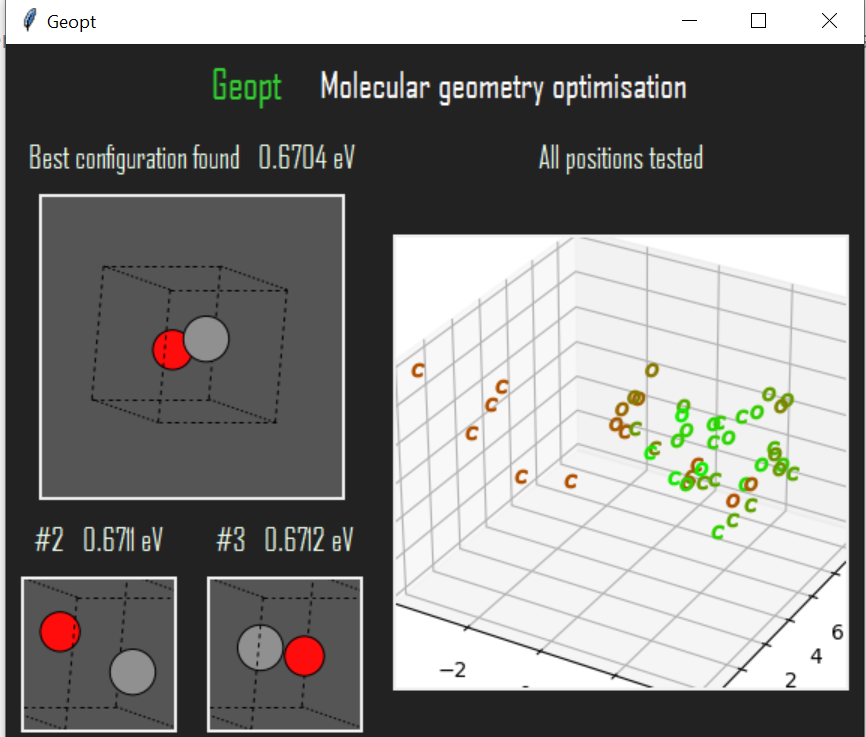
Tri-surface plots were found to be better than surface plots with mesh-grids or wire-frames. This is because of the vast variation in data points.

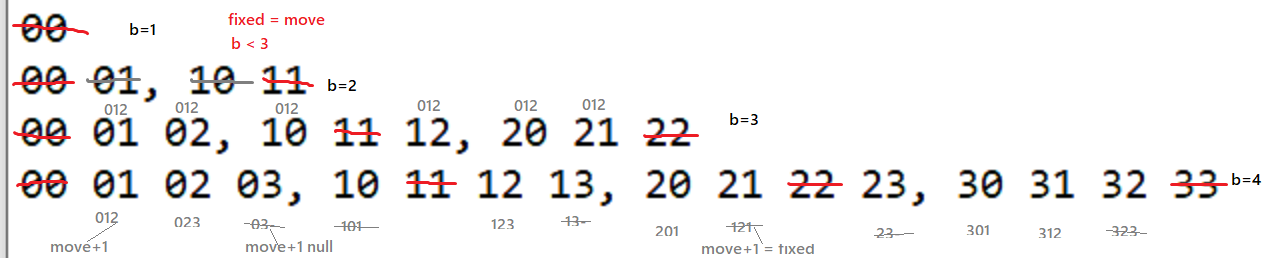
My original layout was cramped and not everything fitted on the page so I redesigned the whole thing and started again. The new design included an extra info window for each version on the molecule. Utilised the datasets created earlier to display lots of useful information to the user.

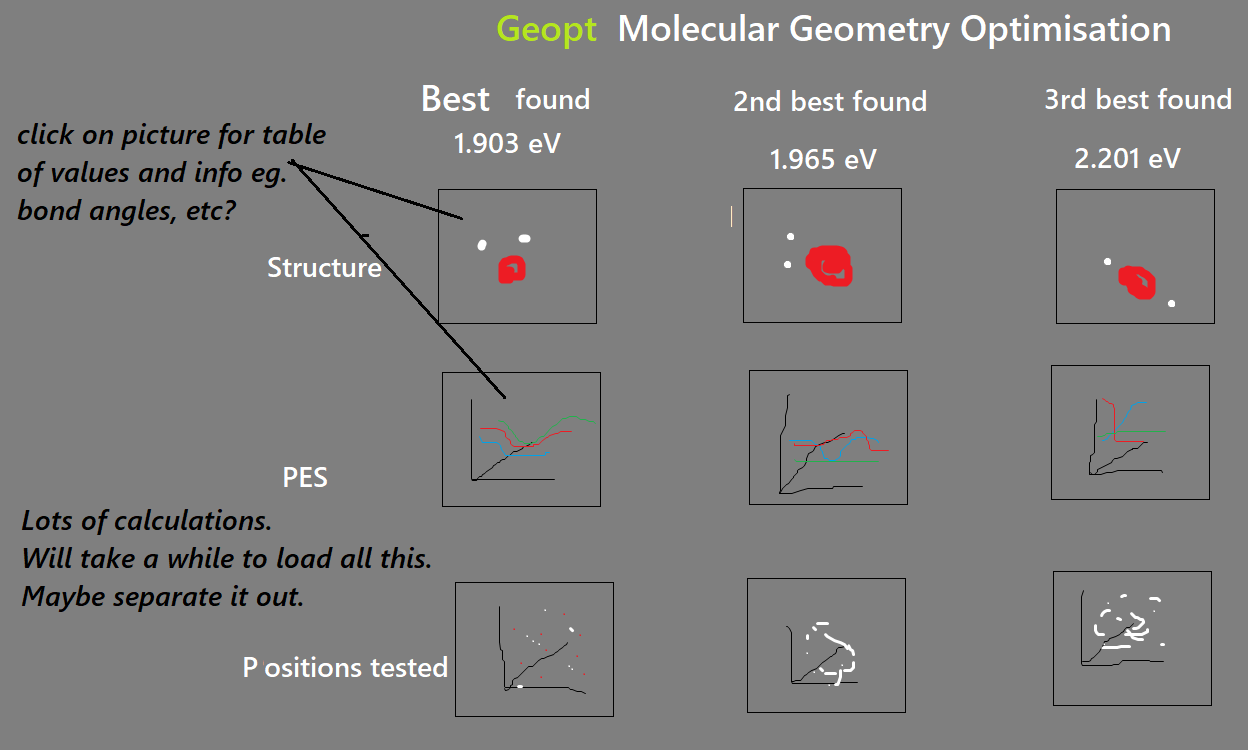


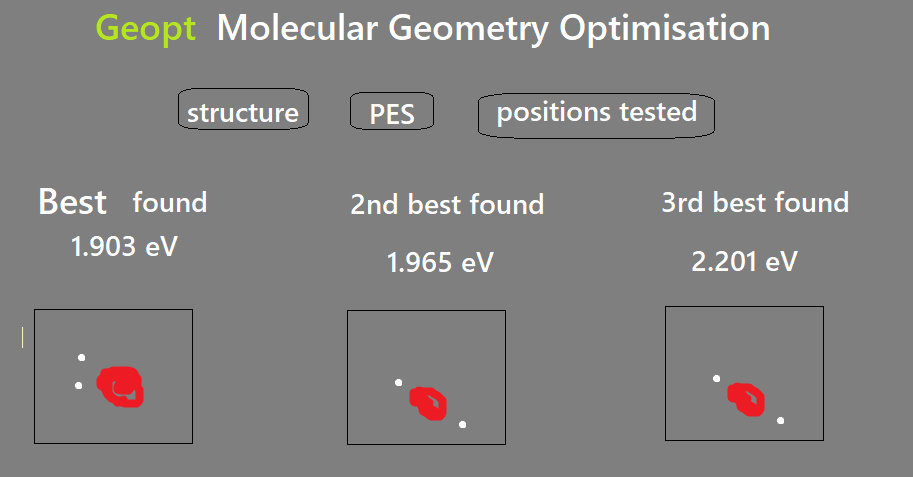


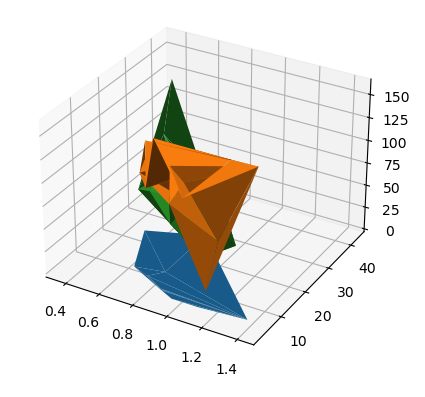


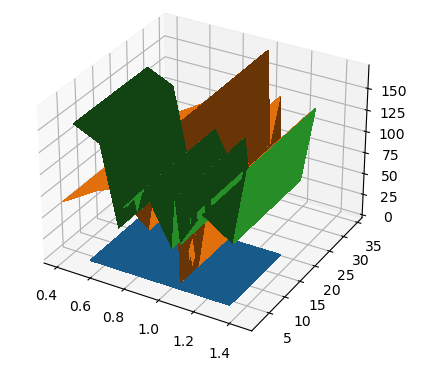
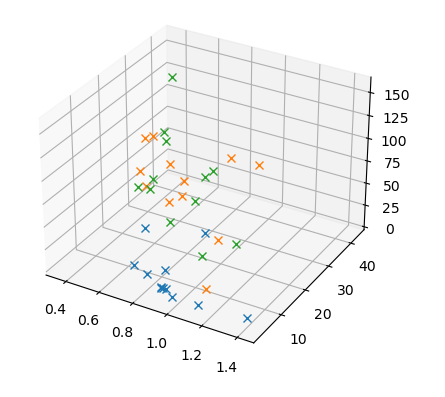


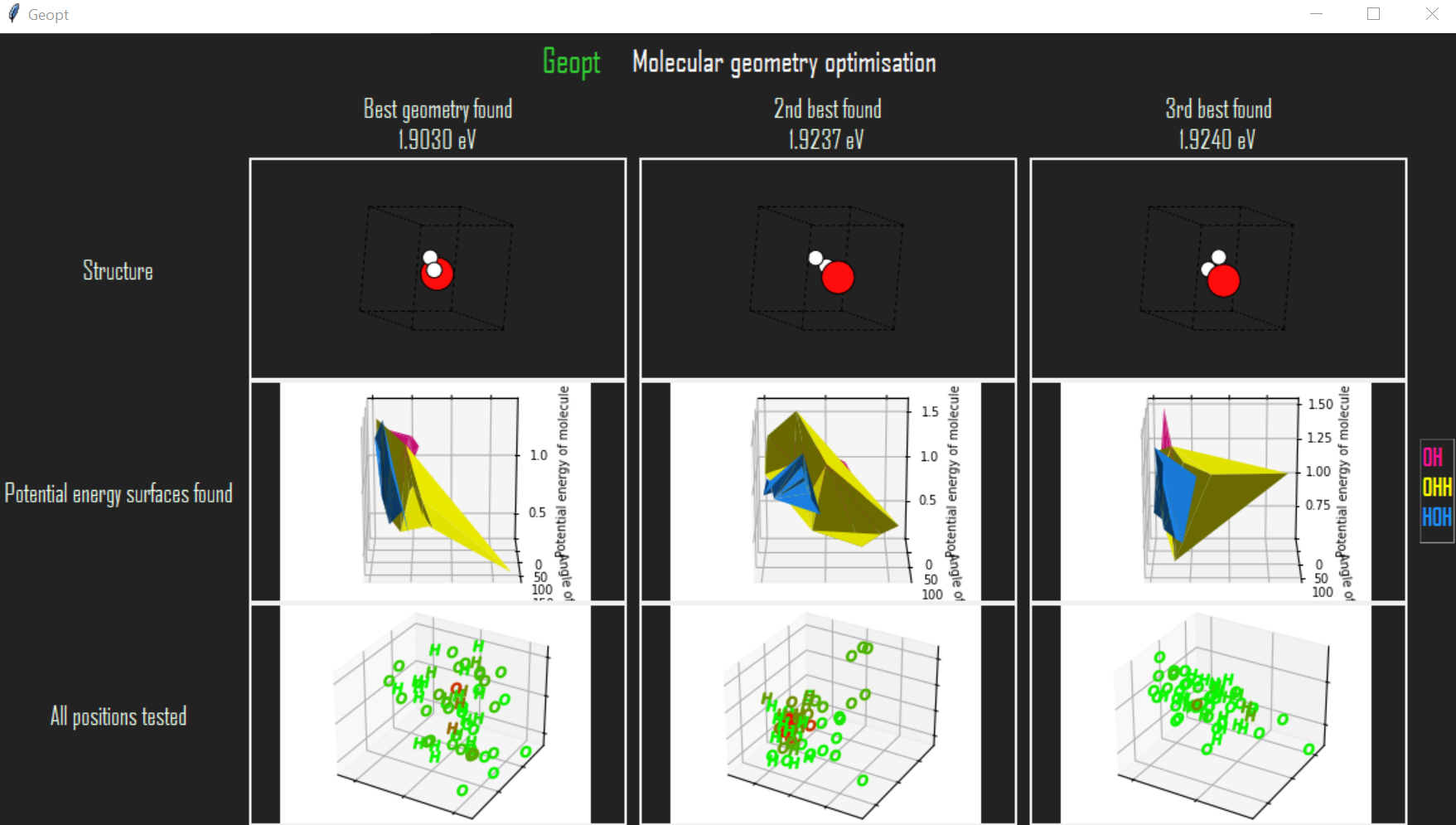












## Sprint Five – Testing

Sprint 5 was to do some halfway-point usability tests and act on user feedback (where deemed appropriate). Three usability tests were conducted; one with a chemist and two with computer scientists. The computer scientists struggled to understand the application's content but the chemist understood it.

I fixed all the errors and acted on recommendations from the user feedback. There were four recommendations:

\* Make the 'more info' buttons clearer.

\* Make it clearer what the test positions plot is showing.

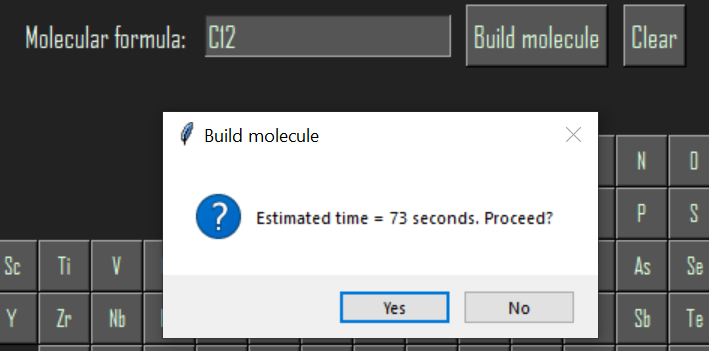
\* Show something to reassure the user that the program is still calculating and hasn't just crashed during long processes.

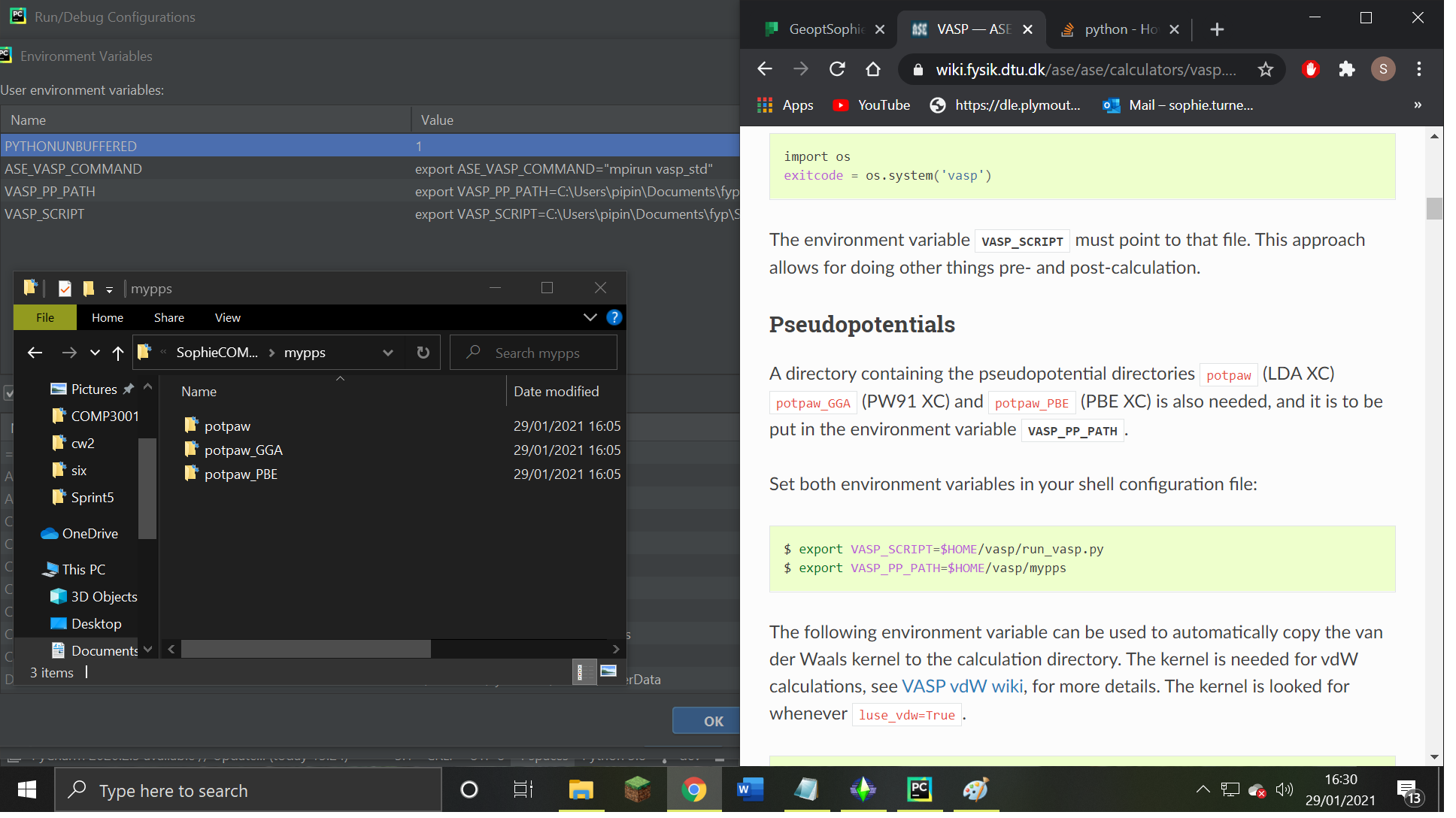
\* Make the number of iterations user-defined.

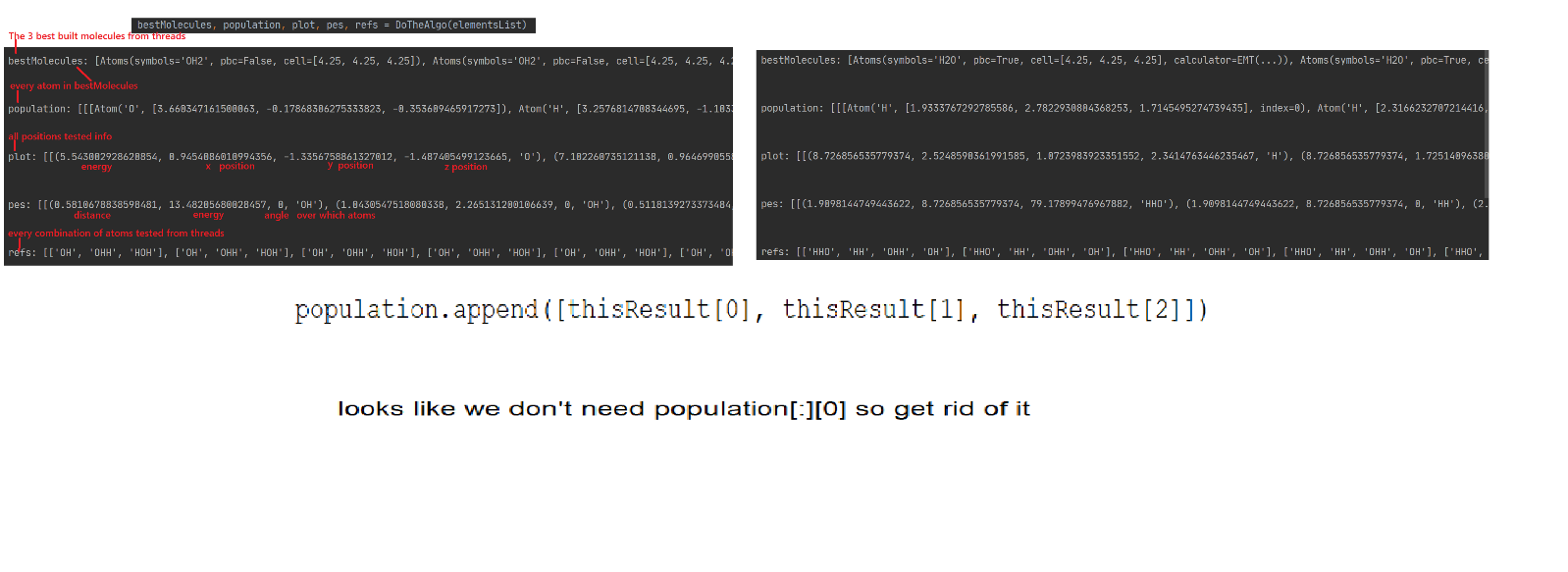
converting my original EA over to work with the new GUI format. I also adapted my PerAtom EA so much that it is no longer really an EA at all!

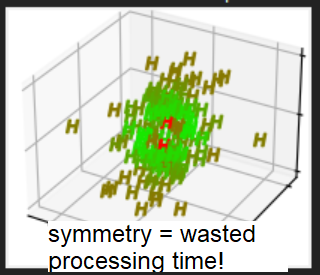
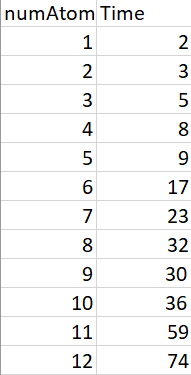
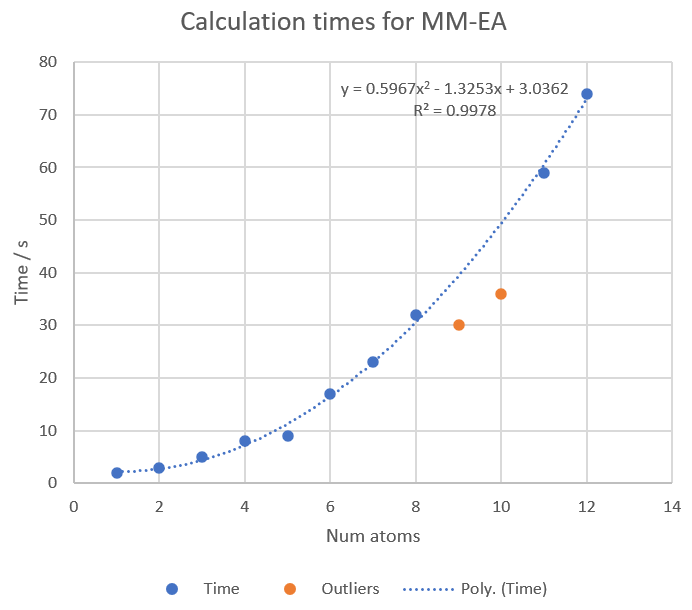
An improvement is found after implementing multiprocessing.

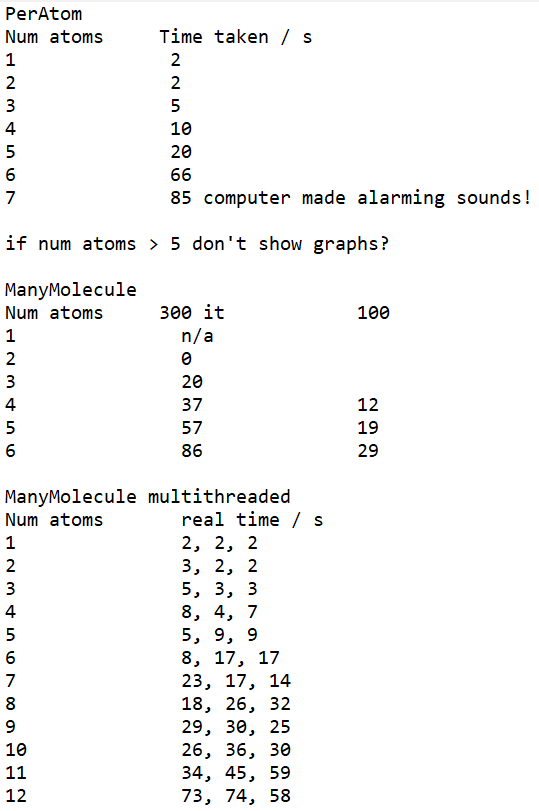
* Feedback – it takes too long!
* Actions – Implemented multiprocessing, limited number of iterations, limited plot dataset sizes, made some code optimisations.
* Feedback – I thought the program had frozen!
* Actions – Let the user know how long it will take.



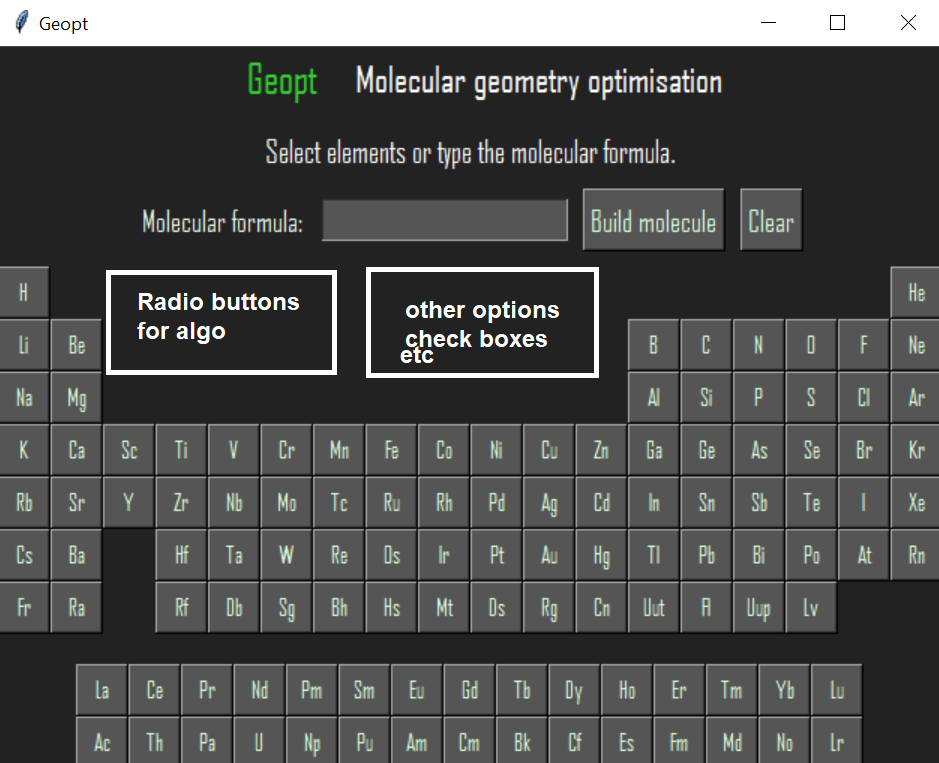


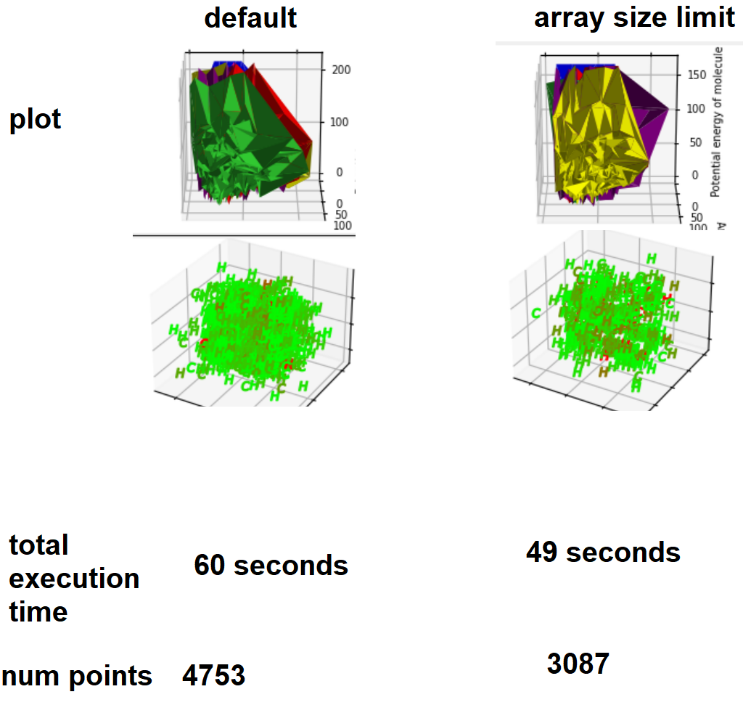
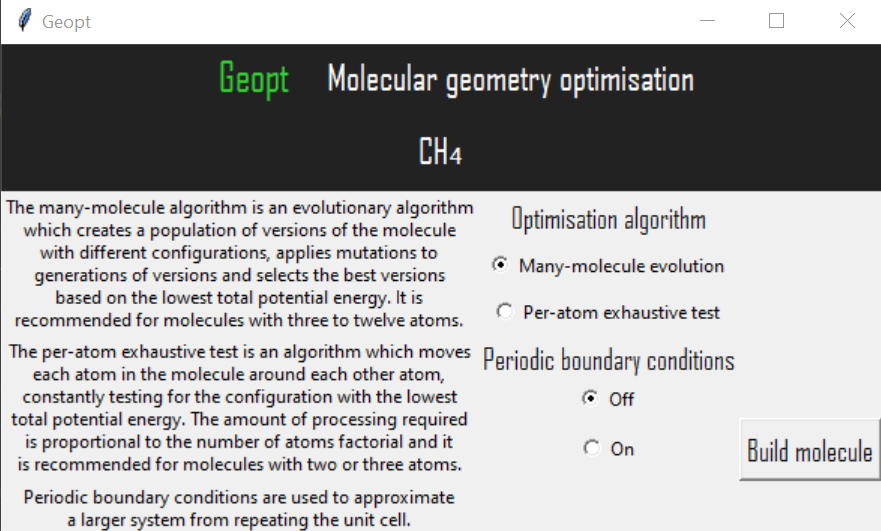




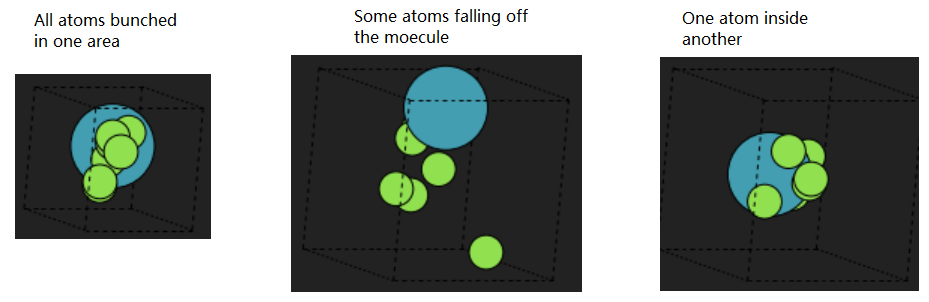
## Sprint Six – Interactivity

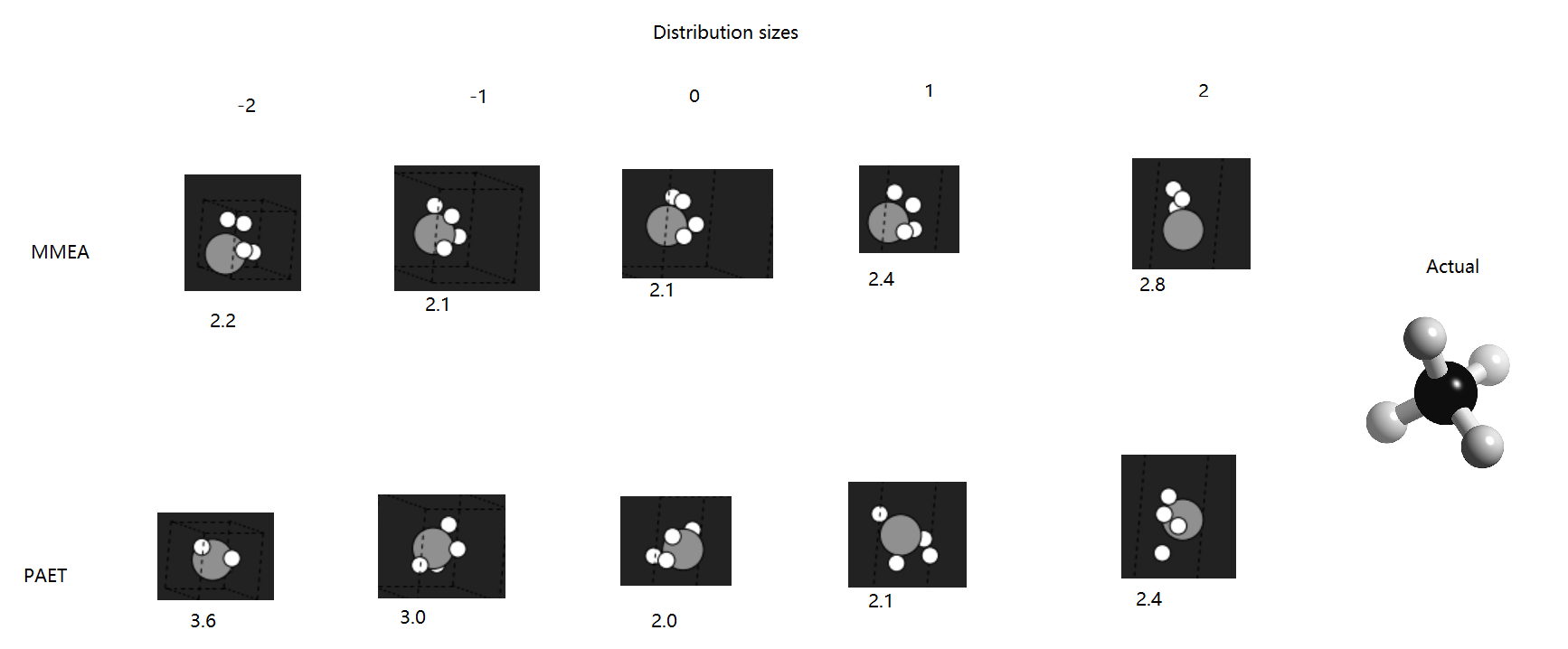


## Sprint Seven – Testing

The sprint was to act on user feedback and make final improvements to the project. This was done. Improvements included: changing help buttons into tooltips; showing a rotated image of the molecule; measuring distances; and fixing various errors.





## Sprint Eight – Final Touches

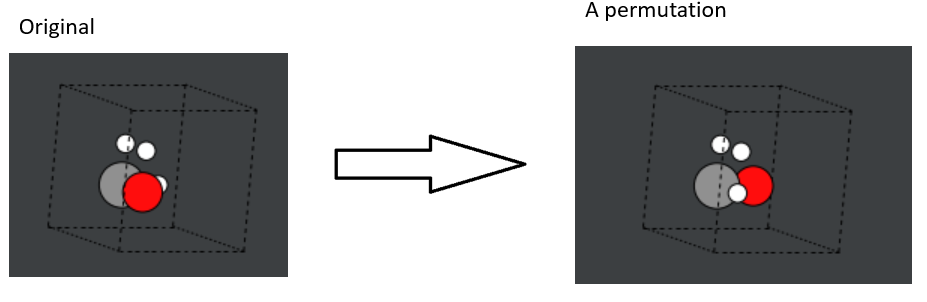
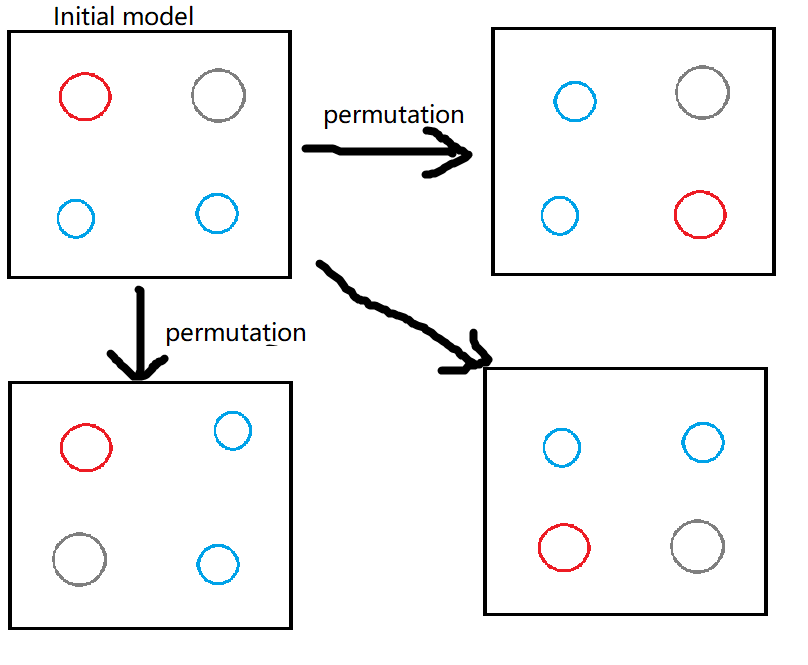
## Sprint Nine – Showcase Materials and Report

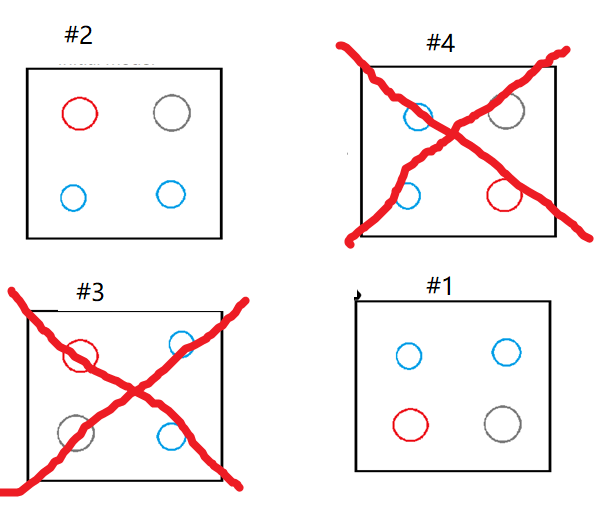
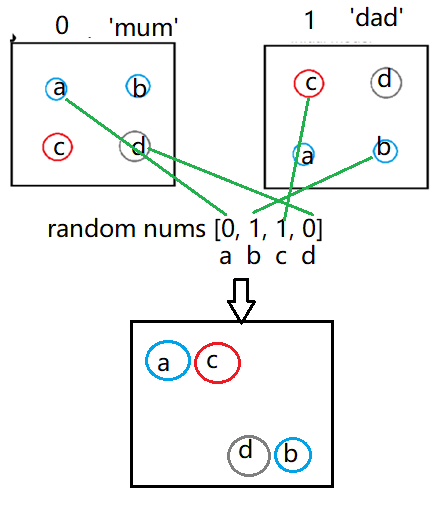
# Algorithms

## Many-Molecule Evolutionary Algorithm

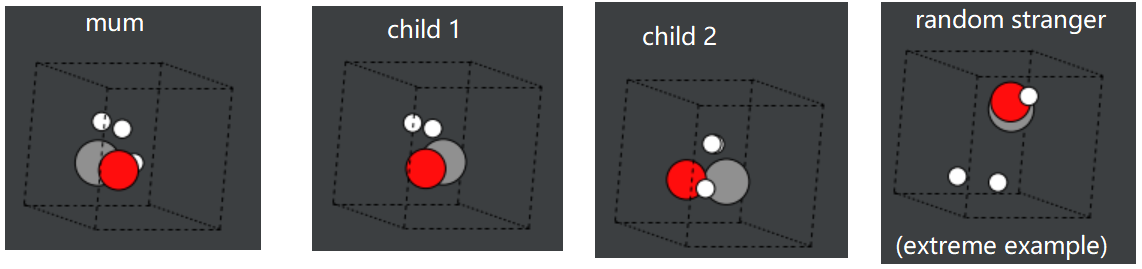
Diagram

Description automatically generated



  Small random movement for parents’ descendants.

Large random movement for new ‘stranger’ introduced to diversify the population and avoid getting stuck at a local minimum.



If the best energy value doesn’t change very much for several consecutive iterations, it is probably finished.

It performed badly. It immediately found a poor local minimum and stayed there, because:

* Population was too small.
* Mutations were too big or too small.
* Atom movements were almost always unfavourable, apart from permutations.
* Stopping criterion wasn’t thorough/specific enough.
* Effective Medium Theory is not an accurate calculation method.

Improvements:

1. Stopped using crossover. Only used mutation and permutation.
2. Increased population size.
3. Introduced more regular permutations, as these produced the best results.
4. Adjusted random mutation ranges and changed it from uniform to Gaussian.
5. Introduced an extra random structure at each iteration with less extreme alterations.
6. Introduced a correction which pushed hydrogen atoms towards carbon atoms.
7. Specified stricter stopping criteria and stopped extreme energies being added.
8. Decreased space between atoms in initial model.

## Per-Atom Exhaustive Test

* Previous problem: Atom movements were almost always unfavourable, apart from permutations.
* Previous problem: Atoms move too far apart.
* For the new EA, try moving 1 atom at a time.
* Start with 2 atoms in the cell, and add one at a time.
* Each time an atom is added, adjust all atoms in the cell.
* Move using a Gaussian distribution around the centre of each atom.

New algo problems

* Exponential increase in time taken (lots of nested for loops) as input size increases.
* Atom movements are still mostly unfavourable.
* Still getting stuck in local minima.
* EMT is still being used (instead of DFT).
* My poor, trusty computer is working really hard - thousands of calculations, array changes & function calls per iteration.

using each atom’s relative size as a way of defining the standard deviation of the Gaussian distribution for atom movements, to create more favourable movements.

Use Python’s Multiprocessing to reduce time spent in loops.

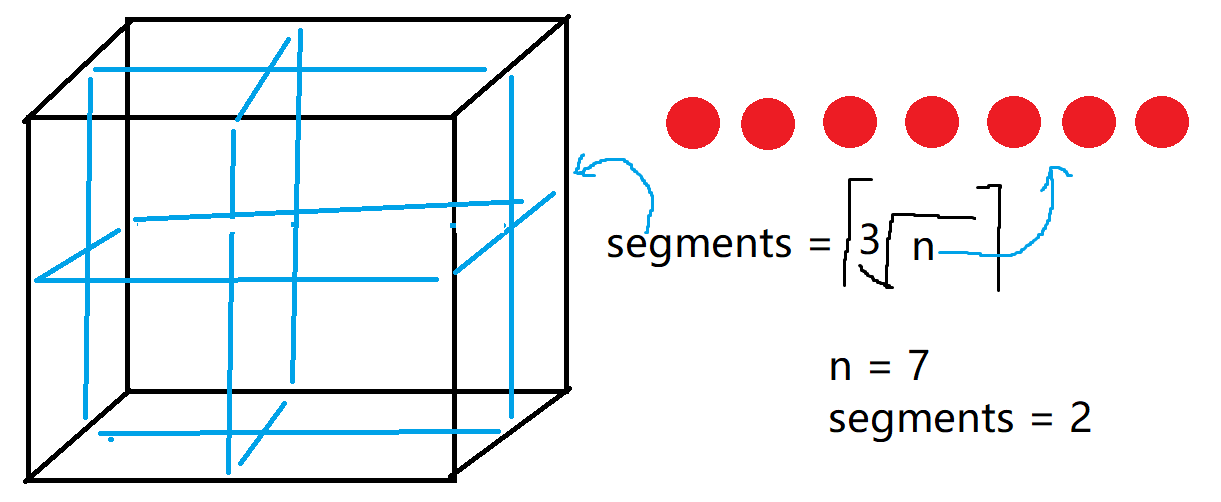
Specifically move atoms closer or further apart as the algorithm progresses rather than just moving them randomly.

Apply code optimisations like those learnt in the Vasilios module.

Use a radial distribution function (doughnut) instead of Gaussian

The PerAtom EA was found to perform better and faster without being an EA at all! The ManyMolecule EA is still an EA. Evolutionary functions were removed from the PerAtom algorithm but remain in the shared module of evolutionary functions.

## Other Algorithms



# User Feedback

User feedback was sought in three stages, at sprints zero, five and seven. Originally, it had been planned that the second stage of user feedback would be conducted at sprint four, but it was decided that this would be more fruitful if it was moved to sprint five to allow work to be done on the user interface first, as usability testing could be difficult and confusing if it was conducted without a substantial user interface.

# Discussion & Evaluation

Talk about how there weren’t enough users to test it. Cite someone saying how many you should have. Justify why we only had a few.

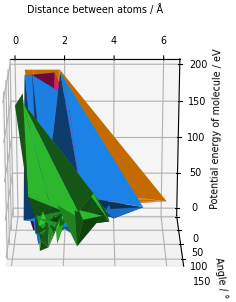
Talk about how the fitness function wasn’t good enough and that limited the whole project.

Talk about getting lost in reams of code and the importance of good practice.

Talk about going over the deadlines for sprints:

I redesigned the layout and began again from scratch, rolling back all the work I'd done on the analysis part of the GUI! Although this added quite a lot of extra time to the sprint, it was worth it, because the new layout is much better and everything fits on the page. I had a problem with the axis rotations on the PES plot not displaying properly. The solution I found was to allow the user to click on the version of the atom for more information, which brings up a new window with larger plots and more details. This is the first sprint that has gone slightly over its allocated time, so I'm not too worried about that as it's within reason to expect this now and then. I hope to finish it over this weekend. In the future I will be more careful about changing the plan at the last minute and think more about how much time it will add.

Make a new one of these ^ as timings may be better now.



# Conclusions

# Suggestions

How will we decide whether there are multiple isomers? We need to look at similar energy values.

# References

Include links to my own work? Not here. At the start.

Filipe, J. (2017). *Joel Filipe.* Available: https://unsplash.com/photos/uHJubAEZklE. Last accessed 30th March 2021.

Matthews, A., Pfau, D., Spencer, S. (2020). Ab initiosolution of the many-electron Schrödinger equation with deep neural networks. *Physical Review Research*. 2.

# Appendices

## Appendix x – Questionnaire Information Sheet

UNIVERSITY OF PLYMOUTH

**FACULTY OF SCIENCE AND ENGINEERING**

RESEARCH INFORMATION SHEET

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name of Principal Investigator

Sophie Turner

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Title of Research

Molecular geometry prediction software feedback from scientists

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Aim of research

To get the opinions of chemists and other relevant scientists about what they want from software which predicts the geometry of molecules.

Description of procedure

Answer some questions about their experiences of similar software.

Description of risks

None.

Benefits of proposed research

Help a computer scientist to design an appropriate application for natural scientists to use.

Right to withdraw

You can withdraw from the research at any time and can request that your data be destroyed be emailing sophie.turner@plymouth.ac.uk.

If you are dissatisfied with the way the research is conducted, please contact the principal investigator in the first instance: sophie.turner@plymouth.ac.uk

If you feel the problem has not been resolved please contact the secretary to the Faculty of Science and Engineering Research Ethics & Integrity Committee: Mrs Paula Simson 01752 584503.

## Appendix x – Target User Questionnaire

UNIVERSITY OF PLYMOUTH

**FACULTY OF SCIENCE AND ENGINEERING**

**Consent Form**

CONSENT TO PARICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name of Principal Investigator

**Sophie Turner**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Title of Research

**Molecular geometry prediction software feedback from scientists**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Brief statement of purpose of work

I am a final year student of computer science at the University of Plymouth and I am creating a chemistry application for my dissertation project. I am looking for feedback and recommendations from chemists and other relevant scientists regarding the design of this application. These forms are anonymous but will be discussed in, and appended to, my dissertation report. Your name will **not** be included in this. You can request that your answers be deleted and not included in the project by emailing me at [sophie.turner@plymouth.ac.uk](mailto:sophie.turner@plymouth.ac.uk)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The objectives of this research have been explained to me.

I understand that I am free to withdraw from the research at any stage, and ask for my data to be destroyed if I wish.

I understand that my anonymity is guaranteed, unless I expressly state otherwise.

I understand that the Principal Investigator of this work will have attempted, as far

as possible, to avoid any risks, and that safety and health risks will have been

separately assessed by appropriate authorities (e.g. under COSHH regulations)

Under these circumstances, I agree to participate in the research.

Name: ……………………………………….

Signature: .....................................…………….. Date: ……………………..

Molecular geometry prediction software feedback from scientists

1. Which science are you mostly involved in?
2. What is your role in this science (e.g. undergraduate, technician, etc)?
3. Please name all the software you have used for chemical structure/geometry modelling, e.g. GaussView, PubChem, Avogadro, etc. If you have never used software for this purpose, please skip to question 8.
4. Regarding the most recent time you used this software, please describe what you used the software for.
5. Why did you choose this software over others?
6. What do you consider to be the best things about this software?
7. What do you consider to be the worst things about this software?
8. In your opinion, what would be the most important features of software for predicting the geometric structure of a theoretical molecule? What would you want to be able to do, as a user of this software?

1. An evolutionary algorithm is a computer algorithm which can be used to find solutions to problems by following these steps:

* **Start with an initial estimate of a solution to a problem**, e.g. the positions of atoms in a molecule.
* **Alter variables which affect this solution**, e.g. the distance between the atoms.
* **Pass these variables to a function**, e.g. an energy calculation, **which returns an output – another possible solution**, e.g. the energy of the system.
* **Compare this solution to the previous solutions**.
* **Choose the best solution**, e.g. the lowest energy of the system.
* **Repeat these steps until reaching an optimal solution**, e.g. the bond lengths which create the lowest net force on each atom.

Algorithms like this can be used to predict geometric properties of molecules, such as bond lengths and bond angles. As a user of this software, do you think it would be useful to be able to view this process and adjust parts of the algorithm, such as which variables to change, or would you prefer it to be a ‘black box’ that worked behind the scenes and just showed you the output?

1. Was this questionnaire easy to understand and fill out? Is there anything that you think should be changed about it?
2. Is there anything else you would like to mention which could be useful for this project?

Thank you for your time. Please return your completed form to [sophie.turner@plymouth.ac.uk](mailto:sophie.turner@plymouth.ac.uk)

## Appendix x – Participant A’s Responses to Target User Questionnaire

1. Which science are you mostly involved in?

*Chemistry*

1. What is your role in this science (e.g. undergraduate, technician, etc)?

*BSc (Hons) Chemistry*

1. Please name all the software you have used for chemical structure/geometry modelling, e.g. GaussView, PubChem, Avogadro.cc, etc. If you have never used software for this purpose, please skip to question 8.

Chemdraw, GaussView

1. Regarding the most recent time you used this software, please describe in more detail what you used the software for.

GaussView – Approximate energies of molecular orbitals

Chemdraw – Model skeletal formula in organic chemistry and interactions between molecules

1. Why did you choose this software over others?

Chemdraw – free licence

GaussView – free licence

1. What do you consider to be the best things about this software?

Chemdraw – User-friendly, made by chemists so valencies and structures are sensible.

GaussView – Fast, user-friendly, 3D representation of orbitals on structure given.

1. What do you consider to be the worst things about this software?

Chemdraw – Sometimes incorrectly corrects valencies when working with unusual structures so it can be difficult when working with novel systems

GaussView – Only allows energies to be computed for single systems. Small structural size limits to reduce computational expense.

1. In your opinion, what would be the most important features of software for predicting the geometric structure of a theoretical molecule? What would you want to be able to do, as a user of this software?

* Easy to use
* Designed by chemists
* 360 degree view of finished molecule
* Able to model interactions between different systems of molecules

1. An evolutionary algorithm is a computer algorithm which can be used to find solutions to problems by following these steps:

* **Start with an initial estimate of a solution to a problem or function**, e.g. the positions of atoms in a molecule.
* **Alter variables which affect this solution**, e.g. the distance between the atoms.
* **Pass these variables to a function**, e.g. an energy calculation, **which returns an output – another possible solution**, e.g. the energy of the system.
* **Compare this solution to the previous solutions**.
* **Choose the best solution**, e.g. the lowest energy of the system.
* **Repeat these steps until reaching an optimal solution**, e.g. the bond lengths which create the lowest net force on each atom.

Algorithms like this can be used to predict geometric properties of molecules, such as bond lengths and bond angles. As a user of this software, do you think it would be useful to be able to view this process and adjust parts of the algorithm, such as which variables to change, or would you prefer it to be a ‘black box’ that worked behind the scenes and just showed you the output?

*View and adjust, with option to make output simpler when required*

1. Was this questionnaire easy to understand and fill out? Is there anything that you think should be changed about it?

*Good questionnaire.*

1. Is there anything else you would like to mention which could be useful for this project?

*An option to analyse the interactions of structures in different solvents would be good.*

## Appendix x – Participant B’s Responses to Target User Questionnaire

1. Which science are you mostly involved in?

Biology

1. What is your role in this science (e.g. undergraduate, technician, etc)?

Postgraduate

1. Please name all the software you have used for chemical structure/geometry modelling, e.g. GaussView, PubChem, Avogadro.cc, etc. If you have never used software for this purpose, please skip to question 8.

PubChem

1. Regarding the most recent time you used this software, please describe in more detail what you used the software for.

Visualising chemical structures.

1. Why did you choose this software over others?

I was instructed to as a learning exercise.

1. What do you consider to be the best things about this software?
2. What do you consider to be the worst things about this software?
3. In your opinion, what would be the most important features of software for predicting the geometric structure of a theoretical molecule? What would you want to be able to do, as a user of this software?

Visualise the structure and view reasoning behind the prediction.

1. An evolutionary algorithm is a computer algorithm which can be used to find solutions to problems by following these steps:

* **Start with an initial estimate of a solution to a problem**, e.g. the positions of atoms in a molecule.
* **Alter variables which affect this solution**, e.g. the distance between the atoms.
* **Pass these variables to a function**, e.g. an energy calculation, **which returns an output – another possible solution**, e.g. the energy of the system.
* **Compare this solution to the previous solutions**.
* **Choose the best solution**, e.g. the lowest energy of the system.
* **Repeat these steps until reaching an optimal solution**, e.g. the bond lengths which create the lowest net force on each atom.

Algorithms like this can be used to predict geometric properties of molecules, such as bond lengths and bond angles. As a user of this software, do you think it would be useful to be able to view this process and adjust parts of the algorithm, such as which variables to change, or would you prefer it to be a ‘black box’ that worked behind the scenes and just showed you the output?

I would like to be able to view this process if I were using the software.

1. Was this questionnaire easy to understand and fill out? Is there anything that you think should be changed about it?

Yes, however as someone who hasn’t used this kind of software much I found it difficult to answer the questions.

1. Is there anything else you would like to mention which could be useful for this project?

## Appendix x – First Usability Test Questionnaire

UNIVERSITY OF PLYMOUTH

**FACULTY OF SCIENCE AND ENGINEERING**

**Consent Form**

CONSENT TO PARICIPATE IN RESEARCH PROJECT / PRACTICAL STUDY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name of Principal Investigator

**Sophie Turner**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Title of Research

**Molecular geometry prediction software feedback from scientists**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Brief statement of purpose of work

I am a final year student of computer science at the University of Plymouth and I am creating a chemistry application for my dissertation project. I am looking for feedback regarding the design of this application. These forms are anonymous but will be discussed in, and appended to, my dissertation report. Your name will **not** be included in this. You can request that your answers be deleted and not included in the project by emailing me at [sophie.turner@plymouth.ac.uk](mailto:sophie.turner@plymouth.ac.uk)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

By participating in this usability test, you agree that:

* The objectives of this research have been explained to you.
* You understand that you are free to withdraw from the research at any stage, and ask for your data to be destroyed if you wish.
* You understand that your anonymity is guaranteed, unless you expressly state otherwise.
* You understand that the Principal Investigator of this work will have attempted, as far as possible, to avoid any risks, and that safety and health risks will have been separately assessed by appropriate authorities (e.g. under COSHH regulations)
* Under these circumstances, you agree to participate in the research.

For each task, please comment on whether the program worked as expected and how easy it was to use. Please also mention any bugs you find.

Date:

**Task 1.** Please select up to eight atoms from any combination of H, C, O, N and/or Al. Ions and other elements are not yet supported by this program. You can select more than eight atoms if you wish to do so, but the algorithm may take some minutes to perform its calculations as the time taken rises exponentially with the size of the molecule.

Comments:

**Task 2**. View information about the structures and potential energy surfaces.

Comments:

**Task 3**. Create another system of atoms.

Comments:

**Task 4.** View information about the structures and potential energy surfaces.

Comments:

Did you find this form comprehensive and easy to complete?

Additional comments and suggestions:

Thank you for your time.

## Appendix x – Participant C’s Responses to First Usability Test Questionnaire

Date:

***10/02/2021***

**Task 1.** Please select up to eight atoms from any combination of H, C, O, N and/or Al. Ions and other elements are not yet supported by this program. You can select more than eight atoms if you wish to do so, but the algorithm may take some minutes to perform its calculations as the time taken rises exponentially with the size of the molecule.

Comments:

Program worked? Y

Easy to use? Y

I used HCN as the example. HCN is a linear molecule with a triple bond between C and N. The results were unexpected; the structure with the H atom closer to both C and N was lower in energy than the more linear structure with 180 degree bond angles between H, C and N. The program appears to only be able to anticipate singly bonded molecules.

**Task 2**. View information about the structures and potential energy surfaces.

Comments:

Program worked? Y

Easy to use? Y

Structure information is excellent. It is easy to see the molecules in 3D. It might be nice to be able to rotate the structure diagrams to see more detail about the arrangement of atoms. The PES is good. Can’t understand the ‘All positions tested’ part. The colours all overlap.

**Task 3**. Create another system of atoms.

Comments:

This time I used CH2N. The structures were clear, and the lowest energy structure was again unexpected.

**Task 4.** View information about the structures and potential energy surfaces.

Comments:

It’s still difficult to interpret the positions tested and the 3D plot is difficult to read when there are more atoms.

Did you find this form comprehensive and easy to complete?

Yes

Additional comments and suggestions:

You could add a separate yes/no section to the feedback form to make it easier to interpret whether the form was user friendly.

## Appendix x – Participant D’s Responses to First Usability Test Questionnaire

Date: 25/1/21

**Task 1.** Please select up to eight atoms from any combination of H, C, O, N and/or Al. Ions and other elements are not yet supported by this program. You can select more than eight atoms if you wish to do so, but the algorithm may take some minutes to perform its calculations as the time taken rises exponentially with the size of the molecule.

Comments: Looks good. All worked fine and easy to understand the UI. User might think the program has crashed or isn’t responding because it can take a long time so you could tell the user to wait or have something saying ‘calculating…’.

**Task 2**. View information about the structures and potential energy surfaces.

Comments: I didn’t realise at first that I could click the button for more info. Maybe have a ‘more info’ hint.

**Task 3**. Create another system of atoms.

Comments: All good.

**Task 4.** View information about the structures and potential energy surfaces.

Comments: All good. Well done.

Did you find this form comprehensive and easy to complete? Yes

Additional comments and suggestions: Hard to understand if you are not a chemist. I can’t comment on that part of it but the UI is good.

## Appendix x – Participant E’s Responses to First Usability Test Questionnaire

Date: 25/1/21

**Task 1.** Please select up to eight atoms from any combination of H, C, O, N and/or Al. Ions and other elements are not yet supported by this program. You can select more than eight atoms if you wish to do so, but the algorithm may take some minutes to perform its calculations as the time taken rises exponentially with the size of the molecule.

Comments:

**It worked as expected.**

**Task 2**. View information about the structures and potential energy surfaces.

Comments:

**Very good UI and nice graphs. Don’t understand them though.**

**Task 3**. Create another system of atoms.

Comments:

**I chose N N O O and it said it was not a valid molecule. Didn’t work.**

**Task 4.** View information about the structures and potential energy surfaces.

Comments:

Did you find this form comprehensive and easy to complete?

**Yes.**

Additional comments and suggestions:

Perhaps you could let the user choose how long it takes, how many iterations etc.

## Appendix x – Second Usability Test Questionnaire

## Appendix x – Participants’ Responses to Second Usability Test Questionnaire

## Appendix x – Project Initiation Document

COMP3000

Computing Project

2020/2021

Project Title

Molecular Geometry Optimisation Using Evolutionary Computation

Links

Source code:https://github.com/Squidgeypea/SophieCOMP3000

Backlog :https://tasks.office.com/live.plymouth.ac.uk/en-GB/Home/Planner/#/plantaskboard?groupId=94cc8cbf-c90e-473e-a3f2-7d7d5dee52d9&planId=VmtMkciqc0GG6F--BwFrqpYAESf1

Project Vision

This program is for chemists and physicists whowant to estimate and view the structures of molecules without spending hours doing calculations or using a supercomputer. Geopt is aprogram which uses machine learning to predict the shapes of theoretical molecules.

Risk Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ref | Risk event | Likelihood  1=low  3=high | Impact  1=low  3=high | Exposure  1=min  9=max | Plan |
| R1 | Calculations too computationally expensive | 2 | 2 | 4 | Use Python packages for calculations. Choose appropriate methods. Simplify calculations. Try multithreading/GPU options etc. |
| R2 | Evolutionary algorithms take too long | 3 | 2 | 6 | Use fewer iterations. Use different technique, selection criteria etc. Make code simpler. |
| R3 | Programming difficulties/lack of knowledge | 3 | 2 | 6 | Have regular meetings with David. Do lots of research and practice. |
| R4 | Too much to do/not finish on time | 3 | 3 | 9 | Keep working throughout the year. Adjust plans if necessary. Focus on most important things first. Use Agile. |
| R5 | Unable to get desired results from algorithms | 2 | 1 | 2 | Alter mutations etc. Analyse and test algorithms. Do plenty of research. |
| R6 | Coronavirus or illness | 2 | 3 | 6 | Know the University’s EC policies. Work from home where possible. |

Keywords

Molecular, chemical, atoms, molecules, geometry, energy, optimisation, minimisation, Python, simulation, evolutionary, machine learning.

## Appendix x – Performance Comparison

Excellent results are shown in yellow. Overall, the Many Molecules EA performed better.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Molecule  (many of these can rotate at each bond so often won’t appear like their pictures) | EA Many  Molecules  Average best energy / eV | Per Atom Average best energy / eV | EA Many  Molecules  Average shape | Per Atom Average shape | EA Many  Molecules  Approx. average time taken / seconds | Per Atom Approx. average time taken / seconds |
| Water  H2O | Chemistry of Water | 1.900 | 2.531 | Excellent | Good | 2 | 2 |
| Carbon dioxide | 0.9230 | 0.9720 | Poor | Poor | < 1 | 1 |
| Methane  Ch4 Molecule Images, Stock Photos & Vectors | Shutterstock | 2.993 | 3.890 | Good | Poor | 1 | 3 |
| Dinitrogen tetroxide  Nitrogen tetroxide (dinitrogen tetroxide, N2O4) rocket propellant molecule.  3D rendering. Atoms are represented as spheres with conventional colour  coding: nitrogen (blue), oxygen (red Stock Photo - Alamy | 3.418 | 6.617 | Good | Poor | 10 | 4 |
| Acetonitrile  Acetonitrile | 1.827 | 4.204 | Poor | Poor | 1 | 4 |
| Acetone | 3.455 | 27.69 | Good | Good | 12 | 20 |
| Lattice of 12 carbons  cubic crystal system | NIH 3D Print Exchange | 3.020 | 11.32 | Good | Poor | 2 | 48 |
| Benzene  Illustrated Glossary of Organic Chemistry - Benzene | 2.416 | 69.32 | Excellent | Good | 39 | 50 |
| Aluminium oxide | 4.467 | 4.139 | Good | Good | 8 | 3 |

## Appendix x – Roadmap

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Week 1 | 09-Oct | Week 2 | 16-Oct | Week 3 | 23-Oct | Week 4 | 30-Oct |
| **Milestones** | |  | **Sprint** | **zero** |  |  | **Sprint** | **one** |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  | **Planning** |  |  | **User story: I wish to create a molecule** | | | |
| **Plan** |  |  | **Setup** |  |  |  |  |  |  |
|  |  |  | **Literature review** | |  |  |  |  |  |
| **Test** |  |  | **Target user questionnaires** | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | **Allow user to select elements** | | |
| **Application** | |  |  |  |  |  | **Create a logo** | |  |
|  |  |  |  |  |  |  | **Create UI** |  |  |
|  |  |  |  |  |  |  | **Database** |  |  |
|  |  |  |  |  |  |  | **Display the molecular formula** | | |

*Table x – Sprints zero and one.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Week 5 | 06-Nov | Week 6 | 13-Nov | Week 7 | 20-Nov | Week 8 | 27-Nov |
| **Milestones** | |  | **Sprint** | **two** |  |  | **Sprint** | **three** |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **User story: I wish to predict a shape** | | |  | **User story: I wish to use a different EA.** | | | |
| **Plan** |  |  | **Update risk assessment for COVID** | | | **Try an EA which alters one atom at a time.** | | | |
|  |  | **Try an EA which makes many molecules and compares them.** | | | |  |  |  |  |
| **Test** |  | **See if the EA works to decrease energy** | | | | **Try to get a better structure prediction than before.** | | | |
|  |  | **See if the molecule's shape is displayed in the cell properly** | | | | **Test num iterations, E calcs, array changes.** | | | |
|  |  | **Make the EA** | |  |  |  | **Make the EA.** | |  |
| **Application** | | **Improve exception handling** | | |  | **Try without crossover and remove all parents.** | | | |
|  |  | **Choose EA parameters** | | |  | **Move from test area to application.** | | | |
|  |  | **Create models of the molecule** | | |  |  |  |  |  |

*Table x – Sprints two and three.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Week 9 | 04-Dec | Week 10 | 11-Dec |
| **Milestones** | |  | **Sprint** | **four** |  |
|  |  |  |  |  |  |
|  |  | **User story: I wish to view molecule, info & PES** | | | |
| **Plan** |  |  |  |  |  |
|  |  |  |  |  |  |
| **Test** |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | **Potential energy surface plot** | | |  |
| **Application** | | **Display best molecules** | | |  |
|  |  | **Annotations** | |  |  |
|  |  | **Display information (angles, distances)** | | | |
|  |  | **Use of colours** | |  |  |

*Table x – Sprint four.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | Week 11 | 18-Dec | Christmas holiday | Week 12 | 15-Jan |
| **Milestones** | | **Sprint** | **five** | Christmas holiday | **Sprint** | **five** |
|  |  |  |  | Christmas holiday |  |  |
|  |  | **Testing** |  | Christmas holiday | **Testing** |  |
| **Plan** |  | **Feedback & changes** | | Christmas holiday | **Feedback & changes** | |
|  |  | **Fixes & improvements** | | Christmas holiday | **Fixes & improvements** | |
| **Test** |  | **Usability tests** | | Christmas holiday | **Usability tests** | |
|  |  |  |  | Christmas holiday |  |  |
|  |  |  |  | Christmas holiday | **Act on user feedback** | |
| **Application** | |  |  | Christmas holiday | **Fix errors & bugs** | |
|  |  |  |  | Christmas holiday |  |  |
|  |  |  |  | Christmas holiday |  |  |
|  |  |  |  | Christmas holiday |  |  |

*Table x – Sprint five and the Christmas holiday.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Week 13 | 22-Jan | Week 14 | 29-Jan |
| **Milestones** | |  | **Sprint** | **six** |  |
|  |  |  |  |  |  |
|  |  | **User story: I wish to interact with the algorithm** | | | |
| **Plan** |  |  |  |  |  |
|  |  |  |  |  |  |
| **Test** |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | **Implement plot limits in perAtom** | | |  |
| **Application** | | **Include buttons etc for interactions** | | | |
|  |  | **Make user interface for choosing settings** | | | |
|  |  | **Let user choose the algorithm** | | |  |
|  |  | **Let user set cell boundaries** | | |  |

*Table x – Sprint six.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Week 15 | 05-Feb | Week 16 | 12-Feb | Week 17 | 19-Feb | Week 18 | 26-Feb |
| **Milestones** | |  | **Sprint** | **seven** |  |  | **Sprint** | **eight** |  |
|  |  |  |  |  |  |  |  | Finish code | |
|  |  |  | **Testing** |  |  |  | Final fixes & clean up | | |
| **Plan** |  |  | **Feedback & changes** | |  |  | Start writing report | |  |
|  |  |  |  |  |  |  |  |  |  |
| **Test** |  |  | **Usability tests** |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  | **Make changes from user feedback** | | | | **Make changes from user feedback** | | |
| **Application** | |  | **Find and fix bugs** | |  |  | **Find and fix bugs** | |  |
|  |  |  | **Tidy up** |  |  |  | **Tidy up** |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

*Table x – Sprints seven and eight.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Week 19 | 05-Mar | Week 20 | 12-Mar |
| **Milestones** | |  | **Sprint** | **nine** |  |
|  |  |  |  | **Submit showcase materials!** | |
|  |  |  | **Showcase materials** |  |  |
| **Plan** |  |  | **Start writing report** |  |  |
|  |  |  |  |  |  |
| **Test** |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| **Application** | |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

*Table x – The final sprint of the code, leading to the showcase submission.*

Find first reference of EMT and define it. Reference it too.

Include some maths. See ‘sources’ document