

UNDERSTANDING OF NOVEL CHEMISTRIES USING MASS SPECTROMETRY,
DIFFERENTIAL ION MOBILITY SPECTROMETRY, AND THEORETICAL
CALCULATIONS

David M. Crizer

A dissertation submitted to the faculty at the University of North Carolina at Chapel Hill in
partial fulfillment of the requirements for the degree of Doctor of Philosophy in the
Department of Chemistry.

Chapel Hill
2016

Approved by:

Gary Glish

Mark Wightman

Mark Schoenfisch

Cynthia Schauer

Frank Leibfarth

© 2016
David M. Crizer
ALL RIGHTS RESERVED

ABSTRACT

David M. Crizer: Understanding of Novel Chemistries Using Mass Spectrometry,
Differential Ion Mobility Spectrometry, and Theoretical Calculations
(Under the direction of Gary Glish)

Mass spectrometry is a very powerful analytical method that can be used to study a variety of interesting chemistries. The work in this dissertation is generally focused on the understanding of novel chemistries through the use of mass spectrometry, electron capture dissociation (ECD), electron transfer dissociation (ETD), differential ion mobility spectrometry (DIMS), electrospray ionization mass spectrometry, electrochemistry, and computational methods. The effects of cation recombination energy/Coulomb repulsion in electron capture dissociation and electron transfer dissociation processes were investigated using a set of peptides of the form $X(AG)_nX$, where X is one of the three basic amino acid residues and n is 1, 2, or 4. DIMS has become a very useful post ionization separation technique in recent years and herein its applicability in the use of separating vitamin K₂ protomers is discussed in detail. Unique ion types were observed when a series of halobenzoquinones were investigated using electrospray ionization mass spectrometry. These unique ion types were determined to be formed via electrochemical reduction. These electrochemical processes were studied using cyclic voltammetry and theoretical calculations to aid in the understanding of the mass spectrometry data. DIMS was also shown to be useful for separating different halobenzoquinone isomers. The energetics involved when ions move through the DIMS device was used to understand why mass spectra of different halobenzoquinone isomers had differing amounts of the previously mentioned unique ion

types. Finally, a program is discussed that calculates theoretical properties of interest. The program was written using the Perl programming language and requires very little user input. It completely automates a variety of density functional theory calculations and manipulation of the output from these calculations into ionization energies, proton affinities, and heats of formation.

In memory of Amy Vess Crizer

To Katey, thank you for all that you do.
I couldn't have done this without all of your support, much love to you.

“But I’m glad you finally finished it”
-Phish

ACKNOWLEDGEMENTS

First, I would like to thank Gary for being supportive throughout my entire career in grad school, which has had quite a few trials and tribulations. I would also like to thank him for the freedom he gives us as students and researchers to do science that we really enjoy.

Thanks to all the Glish Lab members, past and present. You all have managed to keep things interesting and fun for many years. There have been many enlightening science discussions throughout the years and just as many enjoyable times outside of the lab. I wish you all the best.

I would like to thank all of my family members who have stood behind me all throughout my life. They have all been loving and supportive, which has always been appreciated, especially during some very difficult moments during my time as a grad student.

Many thanks go to Nick and Julia who I met while at UNC. They have always been there for me and were always willing to talk with me during times I needed it most. I look forward to our lifelong friendship and all that our futures hold.

Finally, I owe much thanks to Katey who has made the last couple of years as a grad student a wonderful time for me. She has been very supportive to me and I owe her much in helping me finally reach the end of this path that I started on what seems like forever ago. I am eternally grateful for her love and support.

TABLE OF CONTENTS

LIST OF TABLES.....	xii
LIST OF FIGURES.....	xiv
LIST OF ABBREVIATIONS AND SYMBOLS.....	xx
CHAPTER 1: INTODCUCTION TO TECHNIQUES USED TO STUDY NOVEL CHEMISTRIES.....	1
1.1 Investigation of Novel Chemistries.....	1
1.2 Electron Transfer and Electron Capture Dissociation.....	2
1.3 Differential Ion Mobility Spectrometry for Separations.....	7
1.4 Energy Deposition in Differential Ion Mobility Spectrometry.....	9
1.5 The Electrochemical Nature of the Electrospray Ionization Process.....	10
1.6 Theoretical Calculations and Mass Spectrometry.....	12
1.7 Summary.....	14
1.8 References.....	16
CHAPTER 2: CATION RECOMBINATION ENERGY/COULOMB REPULSION EFFECTS IN ELECTRON CAPTURE DISSOCIATION AND ELECTRON TRANSFER DISSOCIATION.....	22
2.1 Electron Capture Dissociation and Electron Transfer Dissociation.....	22
2.2 Experimental.....	25
2.2.1 Samples and Methods.....	25
2.2.2 Data Analysis.....	27
2.3 Results and Discussion.....	29

2.3.1 Cation Charge and Recombination Energy.....	29
2.3.2 Product Ion Partitioning.....	31
2.3.3 H [•] -atom Loss.....	39
2.3.4 c/z [•] -Ion Channels in ECD/ETD.....	43
2.4 Conclusions.....	49
2.5 References.....	51
CHAPTER 3: SEPARATING VITAMIN K ₂ PROTOMERS USING DIFFERENTIAL ION MOBILITY SPECTROMETRY.....	57
3.1 Introduction.....	57
3.2 Experimental.....	59
3.2.1 Samples.....	59
3.2.2 Differential Ion Mobility Spectrometry.....	59
3.2.3 Computational Methods.....	62
3.3 Results and Discussion.....	62
3.3.1 Solution Conditions.....	62
3.3.2 Separation of Protonated and Sodiated Structures of Vitamin K ₂	64
3.3.3 Decomposition of Vitamin K ₂	68
3.4 Conclusions.....	72
3.5 References.....	74
CHAPTER 4: ANALYSIS OF HALOBENZOQUINONES – MASS SPECTROMETRY, DIFFERENTIAL ION MOBILITY SPECTROMETRY, ELECTROCHEMISRY, AND THEORETICAL CALCULATIONS.....	77
4.1 Introduction.....	77
4.2 Experimental.....	78
4.2.1 Samples.....	78

4.2.2 Low-Temperature Plasma Ionization.....	79
4.2.3 Capillary-Skimmer Voltage Difference Studies.....	80
4.2.4 Differential Ion Mobility Spectrometry.....	81
4.2.5 Cyclic Voltammetry.....	82
4.2.6 Computational Methods.....	82
4.3 Results and Discussion.....	83
4.3.1 Unique Ion Types in Mass Spectrometry Formed via Electrochemical Reduction.....	83
4.3.2 Separation of Halobenzoquinone Isomers using Differential Ion Mobility Spectrometry.....	87
4.3.3 Effect of Increased Energy on Ion Types Observed using Differential Ion Mobility Spectrometry.....	89
4.3.4 Electrochemistry and Cyclic Voltammetry.....	92
4.3.5 Calculations.....	99
4.4 Conclusions.....	102
4.5 References.....	104
CHAPTER 5: A PROGRAM TO CALCULATE THEORETICAL PROPERTIES OF MOLECULES AND IONS.....	107
5.1 Introduction.....	107
5.2 Experimental.....	110
5.3 Results and Discussion.....	112
5.3.1 Graphical User Interface.....	112
5.3.2 Extraction of Important Information from NIST Webbook.....	113
5.3.3 Conversion of NIST 3D Structure Files to Gaussian Input Files.....	118
5.3.4 Communicating with KillDevil: Transferring Files, Starting Calculations, and Monitoring Progress of Calculations.....	120

5.3.5 Extracting Structure Geometries from Gaussian Output Files.....	124
5.3.6 Changing Neutral Molecules into Protonated Ions.....	125
5.3.7 Changing Neutral Molecules into Molecular Ions.....	128
5.3.8 Extraction of energies from Gaussian Output Files and Calculation of Important Properties.....	129
5.3.9 Calculation of Heats of Formation.....	135
5.4 Conclusions.....	141
5.5 References.....	143
CHAPTER 6: SUMMARY AND FUTURE DIRECTIONS.....	145
6.1 General Summary.....	145
6.2 Cation Recombination Energy/Coulomb Repulsion Effects in Electron Capture Dissociation and Electron Transfer Dissociation.....	145
6.2.1 Summary.....	145
6.2.2 Future Directions.....	147
6.3 Separating Vitamin K ₂ Protomers Using Differential Ion Mobility Spectrometry.....	147
6.3.1 Summary.....	147
6.3.2 Future Directions.....	148
6.4 Analysis of Halobenzoquinones – Mass Spectrometry, Differential Ion Mobility Spectrometry, Electrochemistry, and Theoretical Calculations.....	148
6.4.1 Summary.....	148
6.4.2 Future Directions.....	149
6.5 A Program to Calculate Theoretical Properties of Ions and Molecules.....	149
6.5.1 Summary.....	149
6.5.2 Future Directions.....	150

APPENDIX 1: AN INTRODUCTION TO ELECTRONIC STRUCTURE THEORY.....	152
APPENDIX 2: MOLECULAR FORMULAS DETERMINED BY HIGH MASS ACCURACY MASS SPECTROMETRY OF PYROLIZED CELLULOSE.....	158
APPENDIX 3: FULL CODE OF EXAMPLE.PL.....	170
APPENDIX 4: FULL CODE OF GUI.PL.....	200
APPENDIX 5: FULL CODE OF GEOMETRY_EXTRACT.PLX.....	201
APPENDIX 6: FULL CODE OF WORKUP.PL.....	206
APPENDIX 7: FULL CODE OF FIND_ENERGY.PLX.....	214
APPENDIX 8: FULL CODE OF FIND_ZPE2.PLX.....	215
APPENDIX 9: FULL CODE OF FIND_THERMAL.PLX.....	216
APPENDIX 10: THEORETICAL RESULTS FROM CALCULATION PROGRAM.....	217
APPENDIX 11: FULL CODE OF EMAIL.PL.....	228
APPENDIX 12: FULL CODE OF MERGE_FILES.PL.....	229
APPENDIX 13: FULL CODE OF RUN.SH.....	230
APPENDIX 14: GAUSSVIEW TUTORIAL.....	231
APPENDIX 15: OVERVIEW OF CALCULATION TYPES IN GAUSSIAN.....	252
APPENDIX 16: SIMULATED ANNEALING IN AMBER.....	263
APPENDIX 17: THEORETICAL COLLISON CROSS SECTION CALCULATIONS....	271
APPENDIX 18: TORSIONAL SAMPLING TUTORIAL.....	275

LIST OF TABLES

Table 2.1 - Product Ion Partitioning for the Ion/Ion Reactions of Doubly Protonated Peptides with the Radical Anion of Azobenzene.....	34
Table 2.2 - Product Ion Partitioning for the Ion/Ion Reactions of Doubly Protonated Peptides with the Radical Anion of 1,3-Dinitrobenzene.....	34
Table 2.3 - Product Ion Partitioning from the Ion/Electron Reactions of Doubly Protonated Peptides with Near-Thermal Electrons.....	35
Table 3.1 - Ratio of protonated species to sodiated species using various spray solution conditions.....	63
Table 4.1 - Ratio of intensities $M^+/[M+H]^+$ as the voltage difference between the capillary exit and skimmer is increased.....	87
Table 4.2 - Compensation field at peak intensity for various dispersion fields for TetraC-1,2-BQ and TetraC-1,4-BQ for the $[M+H]^+$ species.....	88
Table 4.3 - Compensation field at peak intensity for various dispersion fields for TetraC-1,2-BQ and TetraC-1,4-BQ for the M^+ species.....	88
Table 4.4 - Cyclic voltammetry Reduction potentials (V) summary for all three halobenzoquinones studied where all values are referenced versus the Ferrocene couple.....	99
Table 4.5 - Energetics summary from the theoretical calculations on TetraC-1,2-BQ and TetraC-1,4-BQ. All values are reported in eV. Energy values are reported as positive values for the M^+ species as that is the convention commonly used for reporting electron affinity values.....	100
Table A2 - Molecular Formulas Determined by High Mass Accuracy Mass Spectrometry of Pyrolized Cellulose.....	158
Table A10.1 - Theoretical Energy values for $C_{11}H_{12}O_5$	217
Table A10.2 - Theoretical Energy values for $C_{12}H_{14}O_5$	217
Table A10.3 - Theoretical Energy values for $C_{10}H_{10}O_4$	217
Table A10.4 - Theoretical Energy values for $C_{10}H_{12}O_4$	218
Table A10.5 - Theoretical Energy values for $C_{11}H_{12}O_4$	219
Table A10.6 - Theoretical Energy values for $C_{11}H_{14}O_4$	220

Table A10.7 - Theoretical Energy values for $C_{12}H_{12}O_4$	220
Table A10.8 - Theoretical Energy values for $C_{12}H_{14}O_4$	220
Table A10.9 - Theoretical Energy values for $C_{12}H_{16}O_4$	221
Table A10.10 - Theoretical Energy values for $C_{13}H_{14}O_4$	221
Table A10.11 - Theoretical Energy values for $C_{13}H_{16}O_4$	221
Table A10.12 - Theoretical Energy values for $C_{14}H_{18}O_4$	222
Table A10.13 - Theoretical Energy values for $C_{10}H_{18}O_3$	222
Table A10.14 - Theoretical Energy values for $C_{12}H_{14}O_3$	222
Table A10.15 - Theoretical Energy values for $C_{12}H_{16}O_3$	223
Table A10.16 - Theoretical Energy values for $C_{13}H_{16}O_3$	223
Table A10.17 - Theoretical Energy values for $C_{13}H_{18}O_3$	224
Table A10.18 - Theoretical Energy values for $C_{14}H_{18}O_3$	224
Table A10.19 - Theoretical Energy values for $C_9H_{10}O_2$	225
Table A10.20 - Theoretical Energy values for $C_9H_{12}O_2$	226

LIST OF FIGURES

Figure 1.1 - Peptide product ion nomenclature: For a, b, and c ions the charge is retained on the N-terminal fragment, while x, y, and z ions retain the charge on the C-terminal fragment.....	3
Figure 1.2 - Overview of the Cornell mechanism for N-C _α bond cleavage from ETD and ECD experiments of proteins and peptides. C-terminal donor amine group leads to formation of a charged c-type ion and a neutral z-type fragment.....	4
Figure 1.3 - Overview of the Cornell mechanism for N-C _α bond cleavage from ETD and ECD experiments of proteins and peptides. N-terminal donor amine group leads to formation of a neutral c-type fragment and a charged z-type ion.....	5
Figure 1.4 - Overview of the Utah-Washington mechanism for N-C _α bond cleavage from ETD and ECD experiments of proteins and peptides.....	6
Figure 1.5 - Simple representation of ion mobility as a function of electric field strength for three ions. DIMS separates ion based on the differential ion mobility between high-field (red bar) and low-field (green bar).....	8
Figure 1.6 - Schematic representation of three ions being separated by a DIMS device with an applied dc compensation voltage which corrects for the ion trajectory of the ion represented in purple. At this specific dc compensation voltage applied the red and green ions would be neutralized on the electrodes. Alternatively, this dc compensation voltage can be scanned to sequentially pass the different ions through the device.....	9
Figure 1.7 - Schematic of the electrospray ionization process.....	11
Figure 2.1 - Schematic of the NanoFrontier LIT-TOF provided by the Hitachi Corporation (Hitachi High Technologies).....	26
Figure 2.2 - Schematic of the triple quadrupole/linear ion trap mass spectrometer (QTRAP, AB Sciex).....	27
Figure 2.3 - Schematic representation of product partitioning in (a) ECD and (b) ETD.....	33
Figure 2.4 - Plots of P _{ET} for the formation of ground state product cations as a function of cation recombination energy for azobenzene (left-most curve) and 1,3-dinitrobenzene (right-most curve) radical anions generated using the model summarized above. The gray area indicates the range of cation recombination energies likely to apply to the cations in Tables 2.1, 2.2, and 2.3.....	38
Figure 2.5 - Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of [KAGK + 2H ⁺] ²⁺ (b) ECD (top) and ETD (bottom)	

of $[KAGAGK + 2H^+]^{2+}$	44
Figure 2.6 - Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[KAGAGAGAGK + 2H^+]^{2+}$, (b) ECD (top) and ETD (bottom) of $[KAGAGKAGAG + 2H^+]^{2+}$, (c) ECD (top) and ETD (bottom) of $[KAGKAGAGAG + 2H^+]^{2+}$	45
Figure 2.7 - Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[RAGR + 2H^+]^{2+}$, (b) ECD (top) and ETD (bottom) of $[RAGAGR + 2H^+]^{2+}$, (c) ECD (top) and ETD (bottom) of $[RAGAGAGAGR + 2H^+]^{2+}$	47
Figure 2.8 - Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[HAGH + 2H^+]^{2+}$, (b) ECD (top) and ETD (bottom) of $[HAGAGH + 2H^+]^{2+}$, (c) ECD (top) and ETD (bottom) of $[HAGAGAGAGH + 2H^+]^{2+}$	48
Figure 3.1 - Structures of vitamin K ₁ and vitamin K ₂	58
Figure 3.2 - DIMS assemblies a) AutoCAD drawing of G3 b) Esquire 3000 source with spray shield c) picture of source with removed spray shield and DIMS assembly in its place d) G4 DIMS device without housing and e) AutoCAD drawing of G4 design.....	60
Figure 3.3 - Summation of two sinusoidal waveforms to form a bisinusoidal waveform to approximate a rectangular wave.....	61
Figure 3.4 - DIMS-MS scan with G3 device of 5 μ M vitamin K ₂ at dispersion field strength of 30 kV/cm. Two peaks are observed for protonated vitamin K ₂ , while only one peak is observed for sodiated vitamin K ₂	64
Figure 3.5 - DIMS-MS scan with G4 device of 5 μ M vitamin K ₂ at dispersion field strength of 36.7 kV/cm. Two peaks were observed for protonated vitamin K ₂ , while only one peak was observed for sodiated vitamin K ₂	65
Figure 3.6 - MS/MS of two vitamin K ₂ protonated peaks. (a) MS/MS of protonated vitamin K ₂ that passes at a lower CV shows less fragmentation along the side-chain (b) MS/MS of protonated vitamin K ₂ that passes at a higher CV shows more fragmentation along the side-chain which corresponds to the lowest energy structure that has the extended side chain. (c) Fragmentation scheme of vitamin K ₂	66
Figure 3.7 - Proposed vitamin K ₂ structures from DFT calculations fully optimized at B3LYP/6-31G(d,p) level of theory. (a) vitamin K ₂ protonated near the methyl group (b) vitamin K ₂ protonated near side-chain.....	67
Figure 3.8 - Proposed vitamin K ₂ structures from DFT calculations fully optimized at B3LYP/6-31G(d,p) level of theory. (a) vitamin K ₂ sodiated near the methyl	

group (b) vitamin K ₂ sodiated near side-chain.....	68
Figure 3.9 - DIMS-MS scan with G4 device of 5 μM Vitamin K ₂ taken 5 days apart to show the decomposition of vitamin K ₂ at dispersion field strength of 36.7 kV/cm. Fresh spray sample (left) and same sample used 5 days later (right).....	69
Figure 3.10 - High mass accuracy MS scan of decomposed vitamin K ₂ sample on LTQ-FT Ultra.....	70
Figure 3.11 - MS scan of decomposed vitamin K ₂ sample with Bruker HCT's supply gas changed from nitrogen to air.....	71
Figure 3.12 - CID of products from the decomposition of vitamin K ₂ taken at same activation voltage. (a) [vitamin K ₂ - H [•] + H ⁺] ⁺ and (b) [vitamin K ₂ - 2H [•] + H ⁺] ⁺	72
Figure 4.1 - Halobenzoquinone molecules investigated in this study.....	78
Figure 4.2 - Low temperature plasma ionization (LTPI) probe schematic.....	80
Figure 4.3 - Bruker HCT ion trap schematic showing Capillary-Skimmer Voltage Difference.....	80
Figure 4.4 - ESI of HBQs using settings of capillary exit at -150 V and skimmer at -60 V (a) TetraC-1,4-BQ and (b) TetraC-1,2-BQ (the behavior of 2,6-DCBQ is identical to that of tetraC-1,4-BQ and therefore is not shown).....	85
Figure 4.5 - DIMS-MS scans for LTPI ionized 2,6-DCBQ at a dispersion field of 36.7 kV/cm in various solvents (a) ethanol (b) acetonitrile and (c) solid, no solvent.....	89
Figure 4.6 - DIMS-MS scans for ESI ionized 2,6-DCBQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min.....	90
Figure 4.7 - DIMS-MS scans for ESI ionized TetraC-1,4-BQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min.....	91
Figure 4.8 - DIMS-MS scans for ESI ionized TetraC-1,2-BQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min.....	92
Figure 4.9 - Cyclic voltammetry of (a) 2,6-DCBQ, (b) TetraC-1,4-BQ, and (C) TetraC-1,2-BQ in phosphate buffer at pH 7 with a scan rate of 25 mV/sec using a Ag/AgCl reference electrode. Background corrected.....	94
Figure 4.10 - Cyclic voltammetry of (a) 2,6-DCBQ, (b) TetraC-1,4-BQ, and	

(C) TetraC-1,2-BQ in 0.1 M Bu ₄ NPF ₆ in acetonitrile with a scan rate of 1 V/sec using a silver/silver nitrate reference electrode. Background uncorrected.....	97
Figure 4.11 - Cyclic voltammetry of 0.1 M Bu ₄ NPF ₆ in acetonitrile with a scan rate of 1 V/sec using a silver/silver nitrate reference electrode.....	98
Figure 4.12 - Schematic showing the important species that are formed via the electrochemical reduction of halobenzoquinones. These structures served as a starting points for theoretical calculations.....	100
Figure 4.13 - Optimized structures for the [M+H] ⁺ ions of TetraC-1,2-BQ (left) and TetraC-1,4-BQ (right).....	102
Figure 5.1 - Example mass spectrum for pyrolyzed cellulose.....	110
Figure 5.2 - Graphical User Interface (GUI) for user input.....	113
Figure 5.3 - Using Perl module WWW::Mechanize to access the NIST Webbook.....	114
Figure 5.4 - Example of NIST Webbook for compounds with formula C ₁₂ H ₁₄ O ₄	115
Figure 5.5 - Web scrapping url and compound name from NIST Webbook.....	116
Figure 5.6 - Accessing url for each compound and saving 2D and 3D chemical structures.....	117
Figure 5.7 - Using Open babel to convert 3D chemical structures from NIST Webbook into Gaussian input file format.....	118
Figure 5.8 - Creating header information for Gaussian input file.....	119
Figure 5.9 - Using Perl program <i>merge_files.pl</i> to combiner header and geometry information to create a complete Gaussian input file.....	119
Figure 5.10 - Example of complete Gaussian input file. Blank lines around the title of the calculation and the blank line at the end of the file are necessary for correct submission to Gaussian.....	120
Figure 5.11 - Creating a tarball (archive of files, similarly to a zipped folder of files).....	121
Figure 5.12 - Secure copy of files from user's computer to the KillDevil supercomputer...	121
Figure 5.13 - Commands to move Gaussian input files to appropriate directory on KillDevil.....	122
Figure 5.14 - Shell script to start Gaussian jobs on KillDevil.....	123

Figure 5.15 - Communicating with KillDevil to determine when the calculations have finished running.....	124
Figure 5.16 - Extracting optimized geometries from Gaussian output file into new file to be used for Gaussian calculation at next level of theory.....	124
Figure 5.17 - Finding all available oxygens in the neutral structure.....	126
Figure 5.18 - Converting coordinates for oxygens atoms into coordinates for hydrogen atom.....	127
Figure 5.19 - Determining the number of oxygens in the structure.....	128
Figure 5.20 - Command to start program to workup data from calculations.....	129
Figure 5.21 - Creating and formatting Excel workbook for data entry and processing.....	129
Figure 5.22 - Extracting compound ID numbers and writing them into the Excel workbook.....	130
Figure 5.23 - Extracting total energy values from Gaussian output files and writing the values into Excel workbook.....	131
Figure 5.24 - Inserting formulas into Excel to calculate ionization energy.....	132
Figure 5.25 - Determining number of protonated structures for each compound.....	133
Figure 5.26 - Writing formulas into Excel to calculate proton affinity.....	134
Figure 5.27 - Example output of calculation data for the compound C ₁₂ H ₁₄ O ₄	135
Figure 5.28 - Creating and setting up new worksheet in Excel workbook to calculate heats of formation.....	137
Figure 5.29 - Importing literature values needed to calculate heats of formation into the Excel worksheet.....	139
Figure 5.30 - Inserting formulas into excel to calculated heats of formation at 0 K and 298 K.....	140
Figure 5.31 - Example Excel worksheet output for the heats of formation calculations for formula C ₁₂ H ₁₄ O ₄	141
Figure A1.1 - Schematic of the Hartree-Fock iterative process.....	154
Figure A15.1 - Gaussian thermochemistry output.....	257

Figure A16.1 - Schematic of a potential energy surface.....	263
Figure A16.2 - Simulated annealing process.....	264
Figure A16.3 - AMBER Minimization script.....	268
Figure A16.4 - AMBER Simulated annealing script.....	268
Figure A16.5 - LSF run script.....	269

LIST OF ABBREVIATIONS AND SYMBOLS

A	alanine
Å	angstrom
α	polarizability of neutral
Ac	N-terminal acetylation
ACN	acetonitrile
AcOH	acetic acid
Ag ⁺	silver ion
a _n	a-type peptide fragment ions
APCI	atmospheric pressure chemical ionization
b	impact parameter
B3LYP	density functional theory method: Becke's three parameter exchange potential and the correlation functional of Lee, Yang, and Parr
b _n	b-type peptide fragment ions
Bu ₄ NPF ₆	tetra-n-butyl-ammonium hexafluoro phosphate
c	neutral c-type peptide fragment species
¹³ C	carbon-13 isotope
°C	degrees Celsius
c ⁺	positively charged c-type peptide fragment ion
cc-pVXZ	form of correlation consistent basis sets
cc	indicates correlation consistent basis set
CE	capillary electrophoresis
CID	collision induced dissociation
cm	centimeter
cm ² /Vs	centimeters squared per volt second

c_n	c-type peptide fragment ions
CV	compensation voltage
c/z^+	c-type or z-type peptide fragment ions
$\chi_{1v'} \chi_{2v''}$	Franck-Condon overlap between reactant and product vibrational wave functions
DBPs	disinfection byproducts
dc	direct current
$\Delta_f H^\circ(M, 0K)$	heat of formation of molecular at 0 K
$\Delta_f H^\circ(X, 0K)$	heat of formation of atom at 0 K
DFT	density functional theory
ΔH_{ET}	reaction enthalpy of the electron transfer reaction
DIMS	differential ion mobility spectrometry
DIMS-MS	differential ion mobility spectrometry mass spectrometry scan
DNA	deoxyribonucleic acid
dr/dt	radial velocity at the crossing point
DTIMS	drift-tube ion mobility spectrometry
DV	dispersion voltage
E	electric field
E	energy
e^-	electron
$\epsilon_0(X)$	total energy of atom
$\epsilon_0(M)$	total energy of molecule
EA	electron affinity
EA_R	electron affinity of the reagent
E_C	compensation field

ECD	electron capture dissociation
EC, noD	electron capture without dissociation
E_D	dispersion field
E/N	ratio of electric field strength over number density of the gas
ESI	electrospray ionization
ET	electron transfer
ETD	electron transfer dissociation
ET, noD	electron transfer without dissociation
eV	electron volt
$\epsilon_{ZPE}(M)$	zero-point energy of molecule
F-C	Franck-Condon factor
ϕ_μ	basis function
FTICR	Fourier transform ion cyclotron resonance
G	glycine
G	indicates that Gaussian type orbitals are used
G3	generation three differential ion mobility spectrometry device
G4	generation four differential ion mobility spectrometry device
GTO	Gaussian type orbital
GUI	graphical user interface
H	histidine
\hat{H}	Hamiltonian of the system
H^+	proton
H_{12}	coupling matrix element at the point of closest approach
H_2O	water

HCT	high capacity trap
HF	Hartree-Fock
H [•] loss	hydrogen atom loss
HASTE	high amplitude short time excitation
HBQs	halobenzoquinones
HPLC	high performance liquid chromatography
H [°] _M	enthalpy of molecule
H [°] _x	enthalpy of atom
<i>i</i>	current
i.d.	inner diameter
IE	ionization energy
IMS	ion mobility spectrometry
IRMPD	infrared multiphoton dissociation
J	angular momentum of the collision complex
K	mobility
K	lysine
K	Kelvin
K	number of sp-type inner shell Gaussian type orbitals
K ⁺	potassium ion
kcal/mol	kilocalorie per mole
kJ/mol	kilojoule per mole
K-LMG	notation of Pople basis set
kV	kilovolt
L	liter

L	angular momentum
L	number of inner valence s-type and p-type Gaussian type orbitals
Li ⁺	lithium ion
LIT-TOF	linear ion trap/time-of-flight mass spectrometer
LIT _{ECD}	linear ion trap ECD cell
LIT _{therm}	linear ion trap thermalize cell
LTPI	low temperature plasma ionization
LTQ-FT	linear ion trap/Fourier transform mass spectrometer
M	molar
M	number of outer valence s-type and p-type Gaussian type orbitals
μ	reduced mass of the collision partners
M ⁺	positive molecular ion
M [•]	negative molecular ion
[M+H] ⁺	protonated molecule
[M+Na] ⁺	sodiated molecule
[M+2H]	neutral species that is molecule with two added protons and two added electrons
[M+H] ⁻	negatively charged species with one added proton and one added electron
[M-nH] ⁿ⁻	deprotonated ion
μA	microamp
mg	milligram
MHz	megahertz
min	minute
MK-n	menaquinone molecule with n number of isoprenoid groups in side-chain
mm	millimeter

mM	millimolar
μM	micromolar
MS	mass spectrometry
MSAE	mass selective axial ejection
m/z	mass-to-charge ratio
ms	millisecond
mT	millitesla
mTorr	milliTorr
n	integer number
n	number of protons
N	number density of the gas
Na^+	sodium ion
N-C α	nitrogen to alpha-carbon bond in peptide
nESI	nano electrospray ionization
ng/L	nanogram per liter
NH_4OOCH	ammonium formate
v_{rel}	relative velocity of the ions
ω	radial frequency ($2\pi f$)
o.d.	outer diameter
PA	proton affinity
P_{ET}	probability of electron transfer reaction
P_{LZ}	non-adiabatic probability between states at avoided crossing, based on Ladau-Zener theory
PT	proton transfer

PTMs	post-translational modifications
pV	indicates a polarized valence basis
Q1	quadrupole mass spectrometer
Q2	quadrupole ion trap collision cell
Q3	quadrupole mass spectrometer
QTRAP	triple quadrupole/linear ion trap mass spectrometer
R	arginine
RE _{cat}	recombination energy of the cation
r _{ET}	distance at which electron transfer occurs
rf	radio frequency
R _n	amino acid side-chain residue
s	second
sc loss	side-chain loss
sec	second
σ _{ET}	cross section for the two-body interaction between analyte ion and reagent ion
sin	sine wave
t	time
\hat{T}_e	kinetic energy of electrons
tetraC-1,2-BQ	tetrachloro-1,2-benzoquinone
tetraC-1,4-BQ	tetrachloro-1,4-benzoquinone
TOF	time-of-flight
TOFMS	time-of-flight mass spectrometer
\hat{T}_n	kinetic energy of nuclei
UVPD	ultraviolet photo dissociation

V	volt
V/cm	volt per centimeter (electric field strength)
$V_{\text{eff,rot}}$	magnitude of the rotational barrier
\hat{V}_{ee}	potential energy of electron-electron interactions
$V_{0\text{-p}}$	voltage zero to peak
\hat{V}_{ne}	potential energy of electron-nuclei interactions
\hat{V}_{nn}	potential energy of nuclei-nuclei interactions
X	any basic amino acid
X	zeta number: D for double, T for triple, or Q for quadruple
x	number of atoms
x_n	x-type peptide fragment ions
x_1	x Cartesian coordinate
x_2	x Cartesian coordinate
XZ	indicates the zeta number for correlation consistent basis set
Ψ	many particle wave function
Ψ_0	wave function
ψ_i	one electron wave function
y_n	y-type peptide fragment ions
y_1	y Cartesian coordinate
y_2	y Cartesian coordinate
z_n	z-type peptide fragment ions
z_1	z Cartesian coordinate
z_2	z Cartesian coordinate

z^{\bullet}	neutral radical z-type peptide fragment species
$z^{+\bullet}$	positively charged radical z-type fragment ion
Z_{an}	unit charge of anionic reactant
Z_{cat}	unit charge of cationic reactant
2,6-DCBQ	2,6-dichlorobenzoquinone
3-21G	Pople basis set
6-31G	Pople basis set
6-31G(d,p)	Pople basis set with added polarized functions
6-311G+(d,p)	Pople basis set with added polarized and diffuse functions

Chapter 1

Introduction to Techniques Used to Study Novel Chemistries

1.1 Investigation of Novel Chemistries

This work is focused on the understanding of novel chemistries using a variety of techniques. In this work, novel chemistries are probed on the basis of ion energetics. The analysis of proteins using mass spectrometry is of great interest to the scientific community. Fragmentation of proteins and peptides gives information about the sequence and structure of the precursor ions. Investigating the underlying fundamental aspects of peptide fragmentation allows for a more thorough understanding of the processes involved which often leads to improvement of these techniques. Fragmentation of peptide ions can be studied using electron transfer dissociation (ETD) and electron capture dissociation (ECD). Which are useful alternatives to other dissociation techniques such as collision induced dissociation (CID) and ultraviolet photo dissociations (UVPD). The ion energetic factors that affect ETD and ECD are studied in detail in Chapter 2.

The use of differential ion mobility spectrometry (DIMS) in combination with mass spectrometry aids in the understanding of other novel chemical systems. DIMS has been well studied as a means for the gas-phase separation of isomeric species. Differences in ion energetics of isomers plays a role in which isomers (in this case protomers) can be formed. Gas-phase separations are used to distinguish different protomers (isomers that differ in protonate site location) of vitamin K₂ which is discussed in Chapter 3.

Furthermore, the use of DIMS allows some control on the amount of energy that is imparted to ions as they enter into a mass spectrometer. Subjecting ions to differing electric field strengths via DIMS allows for the investigation of unique ion types that are observed when ionizing halobenzoquinones. The electrochemical nature of electrospray ionization leads to the formation of electrochemically reduced species which results in unique ion types. The ion energetics associated with different isomeric halobenzoquinones was studied using the ability to control the energy imparted to these ions via DIMS.

The final chapter in this work discusses a computer program that was written to run theoretical calculations to obtain theoretical energetic data to aid in the determination of species observed in mass spectrometry experiments. The program minimizes the need for user control to determine a variety of theoretical energy values. This program is designed to be used to aid in the determination of products formed via the pyrolysis of cellulose. This is of major importance as cellulose is a major component of cigarettes and leads to a fundamental basis to allow for further research into the products formed during the smoking process.

1.2 Electron Transfer and Electron Capture Dissociation

Electron transfer dissociation (ETD) and electron capture dissociation (ECD) are radical driven reactions that are used for tandem mass spectrometry [1, 2]. Both of these methods are most commonly used in the analysis of proteins and peptides. However, ETD and ECD have also been used to study other species. The significance of ETD and ECD in protein identification has led to a variety of studies, both experimental and theoretical, that provide insights into the factors that affect ETD and ECD processes. This discussion will focus on the analysis of peptides. ETD and ECD involves the capture of an electron from an anionic

reagent or from a free electron in vacuo. Upon capture of the low energy electron fragmentation occurs at primarily the N-C $_{\alpha}$ bond, but can also occur at amino acid side-chains and disulfide bonds, while preserving post-translational modifications (PTMs) [3, 4]. Peptide fragmentation gives rise to a variety of fragment ion types (Figure 1.1). Fragmentation via ETD and ECD breaks the N-C $_{\alpha}$ bond which results in c-type ions (charge on N-terminus) and z-type ions (charge on C-terminus).

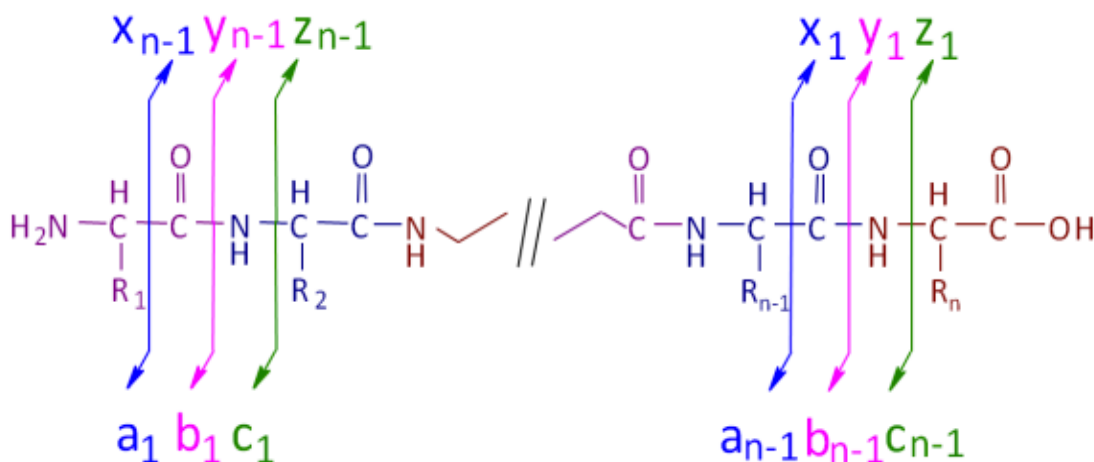


Figure 1.1. Peptide product ion nomenclature: For a, b, and c ions the charge is retained on the N-terminal fragment, while x, y, and z ions retain the charge on the C-terminal fragment

The mechanism of the fragmentation of peptides/proteins via ECD and ETD has been of much discussion. Most of this discussion has been focused on two suggested mechanisms. The first mechanism published is referred to as the Cornell mechanism and was proposed by Fred McLafferty and coworkers at Cornell University (Figures 1.2 and 1.3) [1]. The difference in these two schemes is the location of the amine donor group, which determines where the charge ends up in the fragment species. Electron capture happens at positively charged sites, which are located at protonated amine groups of N-terminus or a basic amino acid (lysine, arginine, or histidine) side chain. This captured electron can then reach the ground state via relaxation processes which results in a hydrogen atom being transferred to

an amide oxygen thus forming an aminoketyl radical intermediate species. Fragmentation then occurs at the N-C $_{\alpha}$ bond to the C-terminal side of the aminoketyl radical. This fragmentation results in the formation of c-type ions and radical z-types ions. The charge of these products ions depends upon the location of charges in the precursor peptide/protein.

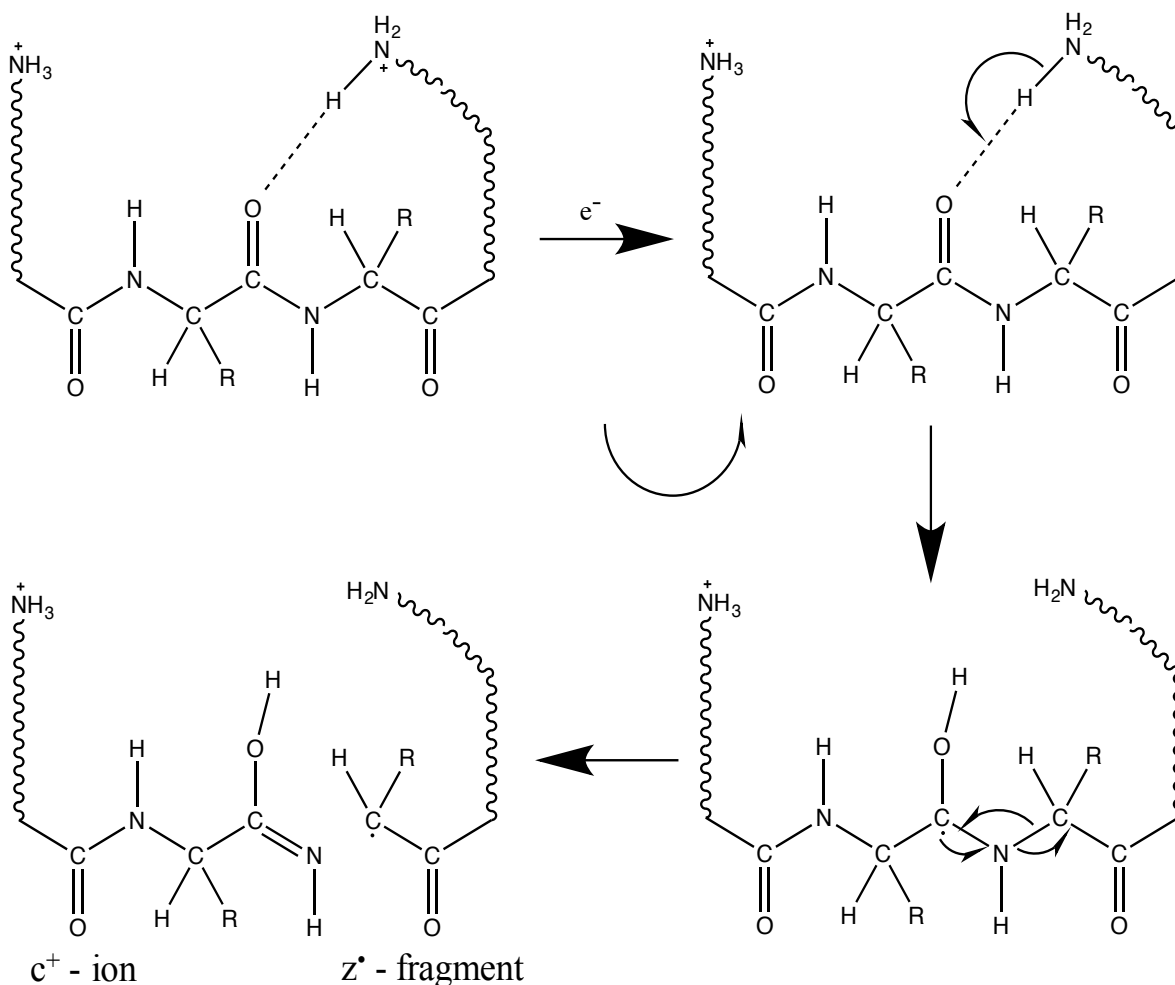


Figure 1.2. Overview of the Cornell mechanism for N-C $_{\alpha}$ bond cleavage from ETD and ECD experiments of proteins and peptides. C-terminal donor amine group leads to formation of a charged c-type ion and a neutral z-type fragment

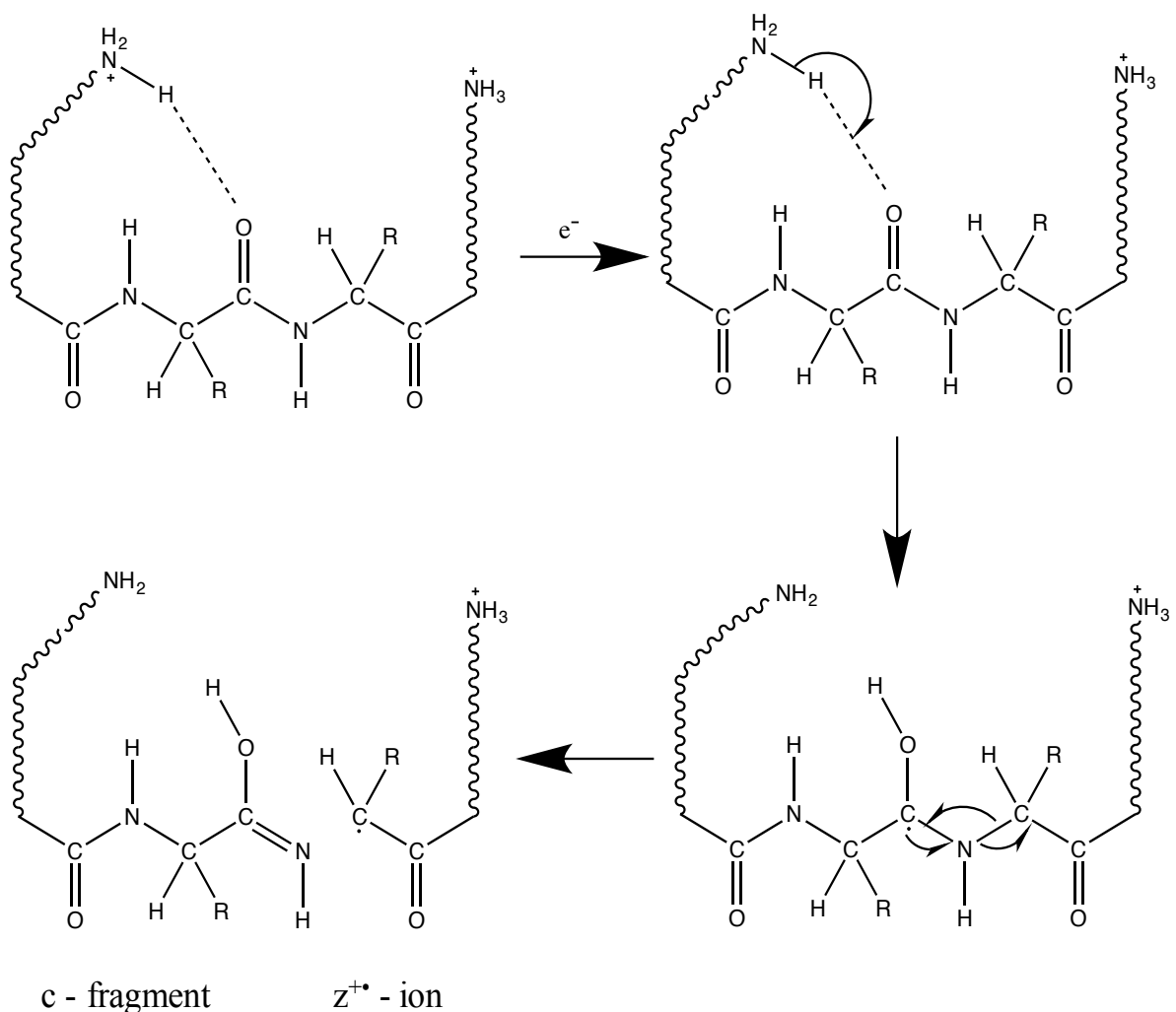


Figure 1.3. Overview of the Cornell mechanism for N-C α bond cleavage from ETD and ECD experiments of proteins and peptides. N-terminal donor amine group leads to formation of a neutral c-type fragment and a charged z-type ion

An alternative to the Cornell mechanism was first proposed by Jack Simons and coworkers at the University of Utah and by Frank Tureček and coworkers at the University of Washington [5-7]. Thus this mechanism has become to be known as the Utah-Washington mechanism (Figure 1.4). The basis of this mechanism is the direct capture of an electron at an amide group which results in the formation of the aminoketyl radical. Anion neutralization then occurs via proton transfer from a charged site resulting in the cleavage of the N-C α bond. The only difference between the two proposed mechanisms (Utah versus

Washington) is the order in which the previously discussed process occurs. The Washington mechanism suggests proton transfer followed by cleavage of the N-C $_{\alpha}$ bond, while the Utah mechanism proposes the reverse order.

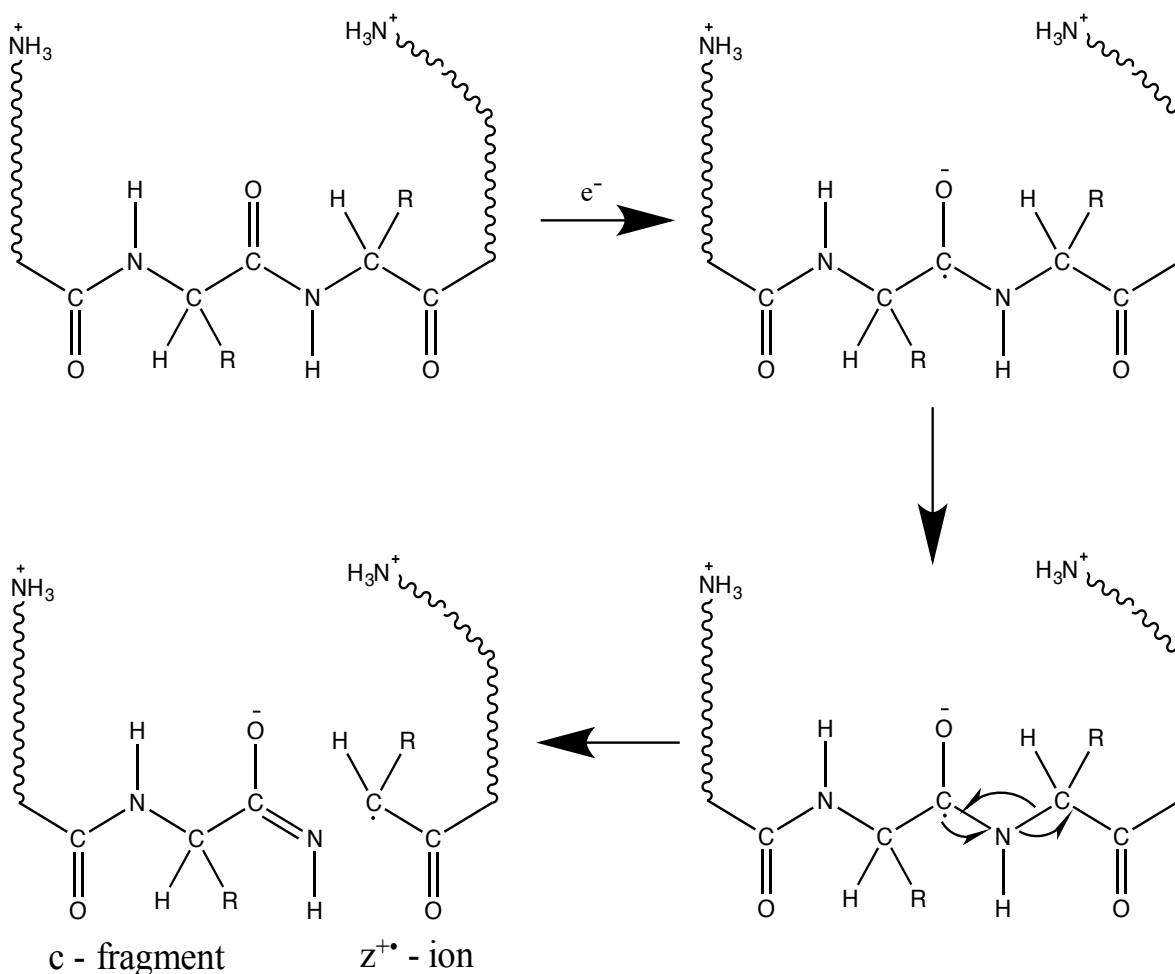


Figure 1.4. Overview of the Utah-Washington mechanism for N-C $_{\alpha}$ bond cleavage from ETD and ECD experiments of proteins and peptides.

The importance of different factors that affect ETD and ECD have been studied in detail. The role that different protonation sites have within a given charge state has been studied for both ETD and ECD [8]. The characteristics of reagent anions for ETD has been investigated with a major focus on the competition that takes place between electron transfer and proton transfer[9]. The role of cation charge state and the possible role of the amide

hydrogen in the mechanism of ETD have also been studied [10]. The work reported on ETD and ECD in Chapter 2 of this work follows these studies as to the different factors that affect ETD and ECD.

1.3 Differential Ion Mobility Spectrometry for Separations

The importance of mass spectrometry as an analytical technique can be attributed to the fact that it is highly sensitive and rapid; however, difficulties in analyzing complex mixtures can arise from low signal-to-background ratios and isomeric/isobaric interferences.

Separation techniques such as liquid chromatography and capillary electrophoresis are often used prior to mass spectrometry to help aid in reducing sample complexity, but these methods can often take a significant amount of time [11]. Separation techniques that occur post ionization such as ion mobility spectrometry (IMS) are significantly faster and can be used with or without prior separation techniques such as liquid chromatography [12-15]. Techniques based on the low-field mobility of ions as they move through low electric field strengths have been used extensively.

Another mobility based separation, which is used in this work, is differential ion mobility spectrometry (DIMS). DIMS differs from the previously mentioned low field mobility techniques in that as the electric field strength (E) is increased ion mobility becomes dependent on E/N , where N is the number density of the gas [16-19]. The dependence of ion mobility on E/N is used in DIMS to separate ions based upon differences in ion mobility in high and low electric fields. Thus a distinct advantage of DIMS is that it can separate ions that have equal low field mobilities, which could not be separated with low field techniques (Figure 1.5). At low electric field strengths only ions represented by the red and light blue lines could be separated. The ions represented by the red and blue lines have equal low field

mobilities and therefore can not be separated with low field techniques. However, the ion mobilities of the ion represented by the orange and blue lines differ at high field strengths and therefore can be separated using DIMS.

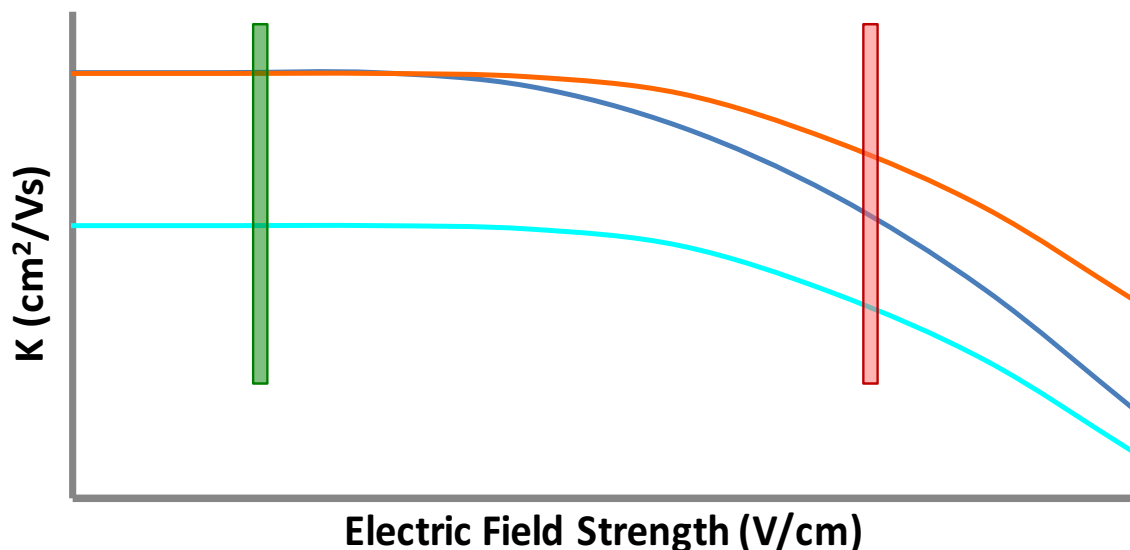


Figure 1.5. Simple representation of ion mobility as a function of electric field strength for three ions. DIMS separates ion based on the differential ion mobility between high-field (red bar) and low-field (green bar).

Differential ion mobility spectrometers consist of two parallel electrodes that are separated by a gap. A simplified schematic of the DIMS device that is used in this work is shown in Figure 1.6. The DIMS electrodes are shown in gray and the three colored circles represent ions that have various differential ion mobilities. When a DIMS device is coupled to a mass spectrometer, as is done in this work, the gas that flows into the mass spectrometer serves to pull ions through the device. An asymmetric rf waveform that oscillates between high and low electric field strengths is applied to the electrodes. The ions move between the electrodes dependent on their mobilities in high and low electric fields. For all experiments in this work a dc compensation voltage is applied to one of the electrodes to control displacement of ions to direct ions of a specific differential ion mobility through the

device[20]. Experiments discussed herein scan the dc compensation voltage to sequentially pass ions with various differential ion mobilities.

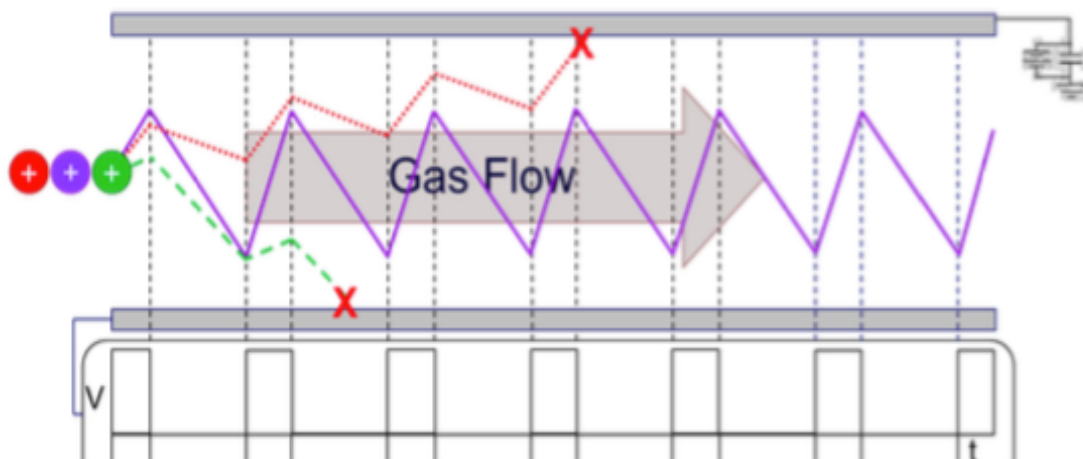


Figure 1.6. Schematic representation of three ions being separated by a DIMS device with an applied dc compensation voltage which corrects for the ion trajectory of the ion represented in purple. At this specific dc compensation voltage applied the red and green ions would be neutralized on the electrodes. Alternatively, this dc compensation voltage can be scanned to sequentially pass the different ions through the device.

DIMS has been studied extensively to separate isomeric and isobaric species. This technique has been shown capable of separating different conformations of model peptides such as ubiquitin and cytochrome C [21, 22]. DIMS has also been shown to be useful in separation of isobaric peptides [23]. Other applications include the separation of isomeric saccharides [24]. DIMS is used as a separation technique in Chapter 3 of this work to separate different protomers (isomers differing in protonation site location) of vitamin K₂.

1.4 Energy Deposition in Differential Ion Mobility Spectrometry

As ions move through the dynamic electric field of the DIMS device they are accelerated and decelerated which causes increases and decreases in the kinetic energy of ions. These ions can then undergo collisions with the carrier gas that can cause them to gain or lose internal energy. Many studies have been done to determine the extent to which ions are able to retain their structural attributes as they are subjected to ion mobility separations.

Most of these studies have been focused around the minimization of the internal energy that is deposited into ions and the changes that occur in the effective temperature of the ions [25-29]. The impact of the energy imparted to ions in DIMS is due to the fact that separations take place at atmospheric pressure, thus causing the ions to be subjected to repeated collisions with the DIMS carrier gas which is at a different kinetic temperature [30-32]. Increases in the DIMS dispersion field causes ions to accelerate which causes these repeated collisions to increase the internal energy deposited into the ions. The amount of energy that is deposited is dependent on ion velocity, gas number density, carrier gas used, and the temperature of the carrier gas [33-35]. The extent of these collisions, causing collisional heating, has been studied using a variety of approaches [26, 27, 36, 37]. Collisions in the DIMS device have been shown previously to cause enough of an increase in internal energy to cause multiply charged protein ions to isomerize as well as the fragmentation of proton bound dimers and small molecules [33, 38, 39].

Chapter 4 in this work discusses the effect that this energy that ions are subjected to has on the ion type that is observed in the mass spectrum. Increased energy imparted to ions is shown to cause loss of hydrogen atoms to change the ion type observed in mass spectrometry experiments of electrospray ionized halobenzoquinones.

1.5 The Electrochemical Nature of the Electrospray Ionization Process

Electrospray ionization (ESI) has become widely used in analytical analyses [40, 41]. It is well suited for use in the analysis of biomolecules due to the fact that ESI can produce multiply charged ions, which can move the mass-to-charge ratio of the ions into the detectable range for mass spectrometers that have a limited mass range (i.e. quadrupole ion traps). Additionally, since ionization occurs from the liquid phase, it makes ESI amenable to

the coupling with separation techniques such as high performance liquid chromatography (HPLC) and capillary electrophoresis (CE) [42-44]. ESI mass spectrometry is also more rapid, sensitive, and accurate to other methods for the analysis of proteins; thus becoming an essential tool for the analysis of proteins and other biological species [45-48].

ESI occurs when a dilute solution of the analyte of interest is pushed through a capillary and a voltage is applied to the emitter or the entrance of the mass spectrometer (Figure 1.7). The electric field that is formed via the application of this voltage causes charge separation at the liquid surface which pushes the solution through the capillary tip. This produces a Taylor cone at the tip and when the Rayleigh limit is reached droplets are formed. The droplets move through the atmosphere towards the mass spectrometer interface and form ions via one of the several proposed mechanisms.

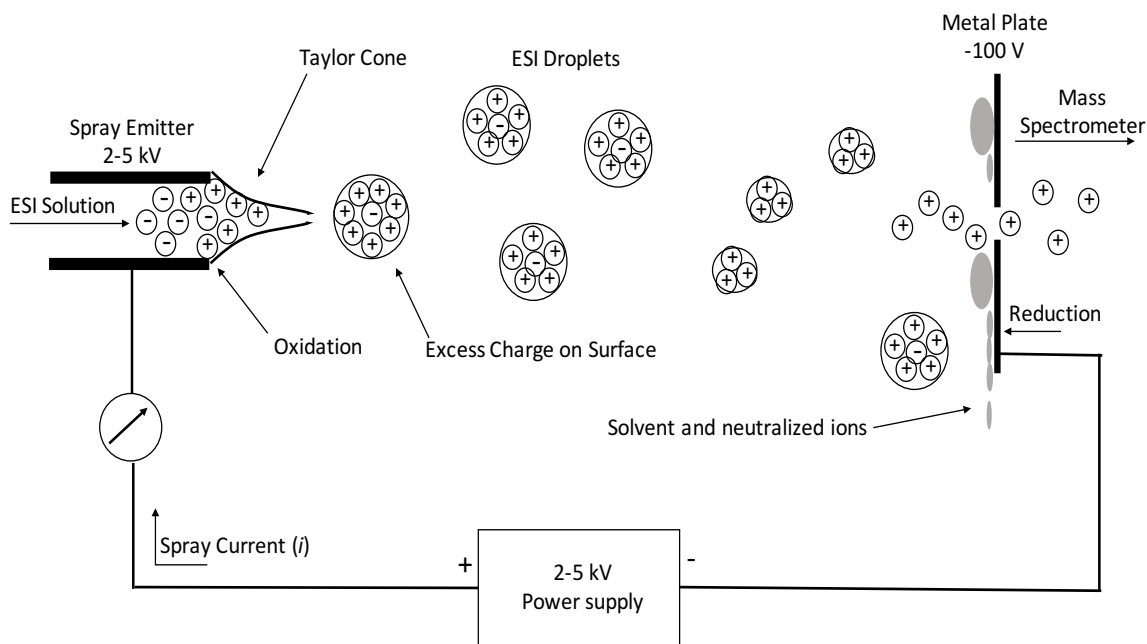


Figure 1.7. Schematic of the electrospray ionization process. Adapted from [49].

These underlying processes that occur during ESI are still under debate today, over thirty years after its introduction. While the mechanism of formation of ions during the ESI

process is still up for debate, the importance of electrochemistry to the process is not [50-52]. The inventor of ESI, John Fenn, when asked about the significance of electrochemistry in ESI responded that “the idea that electrochemical reactions might be taking place in an ESI source was too obvious to mention” [50]. While the electrochemistry involved in the ESI process might be obvious to John Fenn, that is unlikely the case for most users of ESI due to the fact that the electrochemical processes inherent in ESI are not of importance to the majority of mass spectrometry experiments that use ESI. For this discussion it is best to think of an ESI source as a simple two electrode system. Where the ESI emitter serves as one electrode, while the interface to the mass spectrometer serves as the other.

The ESI source used in this work consists of a grounded ESI emitter with a high voltage applied to the interface of the mass spectrometer. A thorough examination of this electrochemical system has been previously published [53]. To summarize, when operated in positive ion mode electrochemical oxidation can occur at the ESI emitter and conversely when operated in negative ion mode electrochemical reduction can occur at the emitter.

This is of great importance to the discussion in Chapter 4 of this work on the analysis of halobenzoquinones using mass spectrometry. The ions observed when halobenzoquinone compounds are ionized via ESI are not species that are normally observed during ESI mass spectrometry experiments. Essentially during the ESI process the halobenzoquinone compounds are electrochemically reduced to form the $[M+2H]^+$ species which can then lose a proton to form the unique ion $[M+H]^+$.

1.6 Theoretical Calculations and Mass Spectrometry

Theoretical calculations have long been used to support and help interpret experimental data. The study of theoretical chemistry is about the use of fundamental physical and

chemical laws to aid in understanding and predicting chemical phenomena. The availability of computer programs used for theoretical calculations and fast computer systems has made the use of theoretical calculations even more widespread. Mass spectrometry is an ideal situation in which theoretical calculations can be of use due to it being a gas-phase technique [54]. This is because of the fact that experiments are done in low pressure systems which is very closely related to isolated systems which is the focus of many theoretical methods.

Theoretical studies have aided in the analysis of mass spectrometry data in many ways. Calculations have been used to help determine ion structures observed in the gas-phase. This has become of greater importance in recent years with the increase of research into ion mobility and ion spectroscopy experiments in combination with mass spectrometry. Low field ion mobility experiments can result in the determination of collisional cross-sections which can be compared to theoretically calculated structures and cross-sections [55-59]. Comparison of experimental and theoretical data can aid in identifying the gas-phase structures that are observed experimentally. Likewise, theoretically calculated vibrational spectra can be compared to experimental ion spectroscopy data to determine the ion structures observed experimentally. These experiments have been found to be particularly useful in the understanding of gas-phase dissociation mechanisms [60-62].

Furthermore, theoretical methods can be used in understanding reaction mechanisms by doing a thorough analysis of the potential energy surface. The calculation of theoretical energetic values can be used to determine the relative stability of different ions observed in mass spectrometry experiments. Calculating basic properties such as ionization energy and proton affinity can also be helpful in determining species that may be preferentially ionized

in mass spectrometry experiments, thus aiding in determining what species are observed in the mass spectrometer.

Theoretical studies in this work are used in a variety of ways to aid in understanding the experimental data obtained. Chapter 3 and Chapter 4 include the use of theoretical work to determine why different behavior is observed in the differential ion mobility spectrometry and mass spectrometry experiments. The work discussed in chapter 5 is entirely theoretical. A computer program was written to automate the calculation of fundamental energy values to aid in the determination of species that are presented in an extremely complex sample.

1.7 Summary

This chapter serves to introduce the reader to the basic fundamentals associated with the techniques that are used in this work to investigate and understand novel chemistries. It serves as a foundation for understanding the more in depth discussions of these techniques used in each chapter. This also gives the reader context in which to place the information in this work in the experimental and theoretical aspects of investigating novel chemistries.

The work in **Chapter 2** is focused on fundamental aspects of electron based dissociation methods (ETD and ECD). It focuses on the dissociation of doubly charged peptides of the form $X(AG)_nX$ as a function of different experimental values. These include the identity of the basic amino acids present in the peptide ($X = K, R, \text{ or } H$), the length of the peptide ($n=1, 2, \text{ or } 4$), and the reagent (near-thermal electrons, anions of azobenzene, or anions of 1,3-dinitrobenzene). A full accounting of the various competitive product ion channels is discussed. A major emphasis is placed on the variation of cation recombination energy without changing net cation charge via variation of peptide size as well as variation of reaction exothermicity by changing the identity of the reagent.

Chapter 3 presents work focused on the ability of DIMS to separate isomeric species. Specifically, the separation of protomers (isomers that differ in protonation site location) of vitamin K₂. Differences in the collision induced dissociation of these protomers are discussed. DIMS data and DFT calculations are utilized to propose structures for protonated vitamin K₂. Finally, a discussion on the decomposition of vitamin K₂ over time is included.

The investigation of unique ion types observed when halobenzoquinones are studied using electrospray ionization mass spectrometry is discussed in **Chapter 4**. Included in this work are studies about the role that electrochemical reduction during the ionization process plays in the formation of these unique ion types. The ability of DIMS to impart energy to ions and manipulate the ion type observed is examined. A discussion on the results of theoretical calculations is used to better understand the behavior of different halobenzoquinone isomers. Finally, the ability of DIMS to separate halobenzoquinone isomers is studied.

A program designed to calculate fundamental energetic values is discussed in **Chapter 5**. This program is designed in such a way as to automate all steps in the process, thus allowing for very little needed input from users of the program. All steps involved in the process to calculate these values are covered. Example results from this program are included. Finally, **Chapter 6** serves as a summary of all results presented in each chapter and includes potential future directions that can extend from this work.

1.8 REFERENCES

1. Zubarev, R.A., Kelleher, N.L., McLafferty, F.W.; Electron Capture Dissociation of Multiply Charged Protein Cations. A Nonergodic Process. *Journal of the American Chemical Society*. **1998**, *120*, 3265-3266.
2. Syka, J.E.P., Coon, J.J., Schroeder, M.J., Shabanowitz, J., Hunt, D.F.; Peptide and Protein Sequence Analysis by Electron Transfer Dissociation Mass Spectrometry. *Proceedings of the National Academy of Sciences of the United States of America*. **2004**, *101*, 9528-9533.
3. Zubarev, R.A.; Reactions of Polypeptide Ions with Electrons in the Gas Phase. *Mass Spectrometry Reviews*. **2003**, *22*, 57-77.
4. Chi, A., Huttenhower, C., Geer, L.Y., Coon, J.J., Syka, J.E.P., Bai, D.L., et al.; Analysis of Phosphorylation Sites on Proteins from *Saccharomyces Cerevisiae* by Electron Transfer Dissociation (ETD) Mass Spectrometry. *Proceedings of the National Academy of Sciences*. **2007**, *104*, 2193-2198.
5. Sawicka, A., Skurski, P., Hudgins, R.R., Simons, J.; Model Calculations Relevant to Disulfide Bond Cleavage via Electron Capture Influenced by Positively Charged Groups. *The Journal of Physical Chemistry B*. **2003**, *107*, 13505-13511.
6. Syrstad, E.A., Tureček, F.; Toward a General Mechanism of Electron Capture Dissociation. *Journal of the American Society for Mass Spectrometry*. **2005**, *16*, 208-224.
7. Tureček, F., Chen, X., Hao, C.; Where Does the Electron Go? Electron Distribution and Reactivity of Peptide Cation Radicals Formed by Electron Transfer in the Gas phase. *Journal of the American Chemical Society*. **2008**, *130*, 8818-8833.
8. Xia, Y., Gunawardena, H.P., Erickson, D.E., McLuckey, S.A.; Effects of Cation Charge-Site Identity and Position on Electron-Transfer Dissociation of Polypeptide Cations. *Journal of the American Chemical Society*. **2007**, *129*, 12232-12243.
9. Gunawardena, H.P., He, M., Chrisman, P.A., Pitteri, S.J., Hogan, J.M., Hodges, B.D.M., et al.; Electron Transfer versus Proton Transfer in Gas-Phase Ion/Ion Reactions of Polyprotonated Peptides. *Journal of the American Chemical Society*. **2005**, *127*, 12627-12639.
10. Crizer, D.M., McLuckey, S.A.; Electron transfer dissociation of amide nitrogen methylated polypeptide cations. *Journal of the American Society for Mass Spectrometry*. **2009**, *20*, 1349-1354.
11. Schneider, B.B., Covey, T.R., Nazarov, E.G.; DMS-MS Separations with Different Transport Gas Modifiers. *International Journal for Ion Mobility Spectrometry*. **2013**, *16*, 207-216.

12. Kolakowski, B.M., Mester, Z.; Review of Applications of High-Field Asymmetric Waveform Ion Mobility Spectrometry (FAIMS) and Differential Mobility Spectrometry (DMS). *Analyst*. **2007**, *132*, 842-864.
13. Hatsis, P., Kapron, J.T.; A Review on the Application of High-Field Asymmetric Waveform Ion Mobility Spectrometry (FAIMS) in Drug Discovery. *Rapid Communications in Mass Spectrometry*. **2008**, *22*, 735-738.
14. Xia, Y.-Q., Wu, S.T., Jemal, M.; LC-FAIMS-MS/MS for Quantification of a Peptide in Plasma and Evaluation of FAIMS Global Selectivity from Plasma Components. *Analytical Chemistry*. **2008**, *80*, 7137-7143.
15. Kanu, A.B., Dwivedi, P., Tam, M., Matz, L., Hill, H.H.; Ion Mobility–Mass Spectrometry. *Journal of Mass Spectrometry*. **2008**, *43*, 1-22.
16. Shvartsburg, A.A. CRC Press, Boca Raton, FL **2009**.
17. Eiceman, G.A.K., Z. CRC Press, **2005**.
18. Mason, E.A.M., E. W. Wiley, New York, NY **1988**.
19. Purves, R.W., Guevremont, R., Day, S., Pipich, C.W., Matyjaszczyk, M.S.; Mass Spectrometric Characterization of a High-Field Asymmetric Waveform Ion Mobility Spectrometer. *Review of Scientific Instruments*. **1998**, *69*, 4094-4105.
20. Dharmasiri, U., Isenberg, S.L., Glish, G.L., Armistead, P.M.; Differential Ion Mobility Spectrometry Coupled to Tandem Mass Spectrometry Enables Targeted Leukemia Antigen Detection. *Journal of Proteome Research*. **2014**, *13*, 4356-4362.
21. Purves, R.W., Barnett, D.A., Ells, B., Guevremont, R.; Elongated Conformers of Charge States +11 to +15 of Bovine Ubiquitin Studied using ESI-FAIMS-MS. *Journal of the American Society for Mass Spectrometry*. **2001**, *12*, 894-901.
22. Purves, R.W., Guevremont, R.; Electrospray Ionization High-Field Asymmetric Waveform Ion Mobility Spectrometry–Mass Spectrometry. *Analytical Chemistry*. **1999**, *71*, 2346-2357.
23. Isenberg, S.L., Armistead, P.M., Glish, G.L.; Optimization of Peptide Separations by Differential Ion Mobility Spectrometry. *Journal of the American Society for Mass Spectrometry*. **2014**, *25*, 1592-1599.
24. Hernandez, O., Isenberg, S., Steinmetz, V., Glish, G.L., Maitre, P.; Probing Mobility-Selected Saccharide Isomers: Selective Ion–Molecule Reactions and Wavelength-Specific IR Activation. *The Journal of Physical Chemistry A*. **2015**, *119*, 6057-6064.

25. Merenbloom, S.I., Flick, T.G., Williams, E.R.; How Hot are Your Ions in TWAVE Ion Mobility Spectrometry? *Journal of the American Society for Mass Spectrometry*. **2012**, *23*, 553-562.
26. Shvartsburg, A.A., Prior, D.C., Tang, K., Smith, R.D.; High-Resolution Differential Ion Mobility Separations Using Planar Analyzers at Elevated Dispersion Fields. *Analytical Chemistry*. **2010**, *82*, 7649-7655.
27. Kendler, S., Lambertus, G.R., Dunietz, B.D., Coy, S.L., Nazarov, E.G., Miller, R.A., et al.; Fragmentation Pathways and Mechanisms of Aromatic Compounds in Atmospheric Pressure Studied by GC–DMS and DMS–MS. *International Journal of Mass Spectrometry*. **2007**, *263*, 137-147.
28. Shvartsburg, A.A., Danielson, W.F., Smith, R.D.; High-Resolution Differential Ion Mobility Separations Using Helium-Rich Gases. *Analytical Chemistry*. **2010**, *82*, 2456-2462.
29. Chen, S.-H., Russell, D.H.; How Closely Related Are Conformations of Protein Ions Sampled by IM-MS to Native Solution Structures? *Journal of the American Society for Mass Spectrometry*. **2015**, *26*, 1433-1443.
30. Krylov, E., Nazarov, E.G., Miller, R.A., Tadjikov, B., Eiceman, G.A.; Field Dependence of Mobilities for Gas-Phase-Protonated Monomers and Proton-Bound Dimers of Ketones by Planar Field Asymmetric Waveform Ion Mobility Spectrometer (PFAIMS). *The Journal of Physical Chemistry A*. **2002**, *106*, 5437-5444.
31. Shvartsburg, A.A., Creese, A.J., Smith, R.D., Cooper, H.J.; Separation of a Set of Peptide Sequence Isomers Using Differential Ion Mobility Spectrometry. *Analytical Chemistry*. **2011**, *83*, 6918-6923.
32. Levin, D.S., Vouros, P., Miller, R.A., Nazarov, E.G., Morris, J.C.; Characterization of Gas-Phase Molecular Interactions on Differential Mobility Ion Behavior Utilizing an Electrospray Ionization-Differential Mobility-Mass Spectrometer System. *Analytical Chemistry*. **2006**, *78*, 96-106.
33. Nazarov, E.G., Coy, S.L., Krylov, E.V., Miller, R.A., Eiceman, G.A.; Pressure Effects in Differential Mobility Spectrometry. *Analytical Chemistry*. **2006**, *78*, 7697-7706.
34. An, X., Eiceman, G.A., Rodriguez, J.E., Stone, J.A.; Gas Phase Fragmentation of Protonated Esters in Air at Ambient Pressure Through Ion Heating by Electric Field in Differential Mobility Spectrometry. *International Journal of Mass Spectrometry*. **2011**, *303*, 181-190.

35. An, X., Eiceman, G.A., Räsänen, R.-M., Rodriguez, J.E., Stone, J.A.; Dissociation of Proton Bound Ketone Dimers in Asymmetric Electric Fields with Differential Mobility Spectrometry and in Uniform Electric Fields with Linear Ion Mobility Spectrometry. *The Journal of Physical Chemistry A*. **2013**, *117*, 6389-6401.
36. Shvartsburg, A.A., Li, F., Tang, K., Smith, R.D.; Distortion of Ion Structures by Field Asymmetric Waveform Ion Mobility Spectrometry. *Analytical Chemistry*. **2007**, *79*, 1523-1528.
37. Robinson, E.W., Shvartsburg, A.A., Tang, K., Smith, R.D.; Control of Ion Distortion in Field Asymmetric Waveform Ion Mobility Spectrometry via Variation of Dispersion Field and Gas Temperature. *Analytical Chemistry*. **2008**, *80*, 7508-7515.
38. Menlyadiev, M.R., Tarassov, A., Kielnecker, A.M., Eiceman, G.A.; Tandem Differential Mobility Spectrometry with Ion Dissociation in Air at Ambient Pressure and Temperature. *Analyst*. **2015**, *140*, 2995-3002.
39. Veasey, C.A., Thomas, C.L.P.; Fast Quantitative Characterisation of Differential Mobility Responses. *Analyst*. **2004**, *129*, 198-204.
40. Cole, R.B., Ed. . Wiley, New York, NY **1997**.
41. Pramanik, B.N., Ganguly, A. K., Gross, M. L., Eds. Marcel Dekker, New York, NY **2002**.
42. Gelpí, E.; Biomedical and Biochemical Applications of Liquid Chromatography-Mass Spectrometry. *Journal of Chromatography A*. **1995**, *703*, 59-80.
43. Niessen, W.M.A., Tinke, A.P.; Liquid Chromatography-Mass Spectrometry General Principles and Instrumentation. *Journal of Chromatography A*. **1995**, *703*, 37-57.
44. Slobodnik, J., van Baar, B.L.M., Brinkman, U.A.T.; Column Liquid Chromatography-Mass Spectrometry: Selected Techniques in Environmental Applications for Polar Pesticides and Related Compounds. *Journal of Chromatography A*. **1995**, *703*, 81-121.
45. Fenn, J.B., Mann, M., Meng, C.K., Wong, S.F., Whitehouse, C.M.; Electrospray Ionization for Mass Spectrometry of Large Biomolecules. *Science*. **1989**, *246*, 64.
46. Cao, P., Moini, M.; Capillary Electrophoresis/Electrospray Ionization High Mass Accuracy Time-of-Flight Mass Spectrometry for Protein Identification using Peptide Mapping. *Rapid Communications in Mass Spectrometry*. **1998**, *12*, 864-870.
47. Mehlis, B., Kertscher, U.; Liquid Chromatography/Mass Spectrometry of Peptides of Biological Samples. *Analytica Chimica Acta*. **1997**, *352*, 71-83.

48. Shen, X., Perreault, H.; Characterization of Carbohydrates Using a Combination of Derivatization, High-Performance Liquid Chromatography and Mass Spectrometry. *Journal of Chromatography A*. **1998**, *811*, 47-59.
49. Cech, N.B., Enke, C.G.; Practical Implications of Some Recent Studies in Electrospray Ionization Fundamentals. *Mass Spectrometry Reviews*. **2001**, *20*, 362-387.
50. Mora, J.F.d.l., Van Berkel, G.J., Enke, C.G., Cole, R.B., Martinez-Sanchez, M., Fenn, J.B.; Electrochemical Processes in Electrospray Ionization Mass Spectrometry. *Journal of Mass Spectrometry*. **2000**, *35*, 939-952.
51. Blades, A.T., Ikononou, M.G., Kebarle, P.; Mechanism of Electrospray Mass Spectrometry. Electrospray as an Electrolysis Cell. *Analytical Chemistry*. **1991**, *63*, 2109-2114.
52. Kertesz, G.J.V.B., Vilmos; Using the Electrochemistry of the Electrospray Ion Source. *Analytical Chemistry*. **2007**, *79*, 5510-5520.
53. Jackson, G.S., Enke, C.G.; Electrical Equivalence of Electrospray Ionization with Conducting and Nonconducting Needles. *Analytical Chemistry*. **1999**, *71*, 3777-3784.
54. Alcamí, M., Mó, O., Yáñez, M.; Computational Chemistry: A Useful (Sometimes Mandatory) Tool in Mass Spectrometry Studies. *Mass Spectrometry Reviews*. **2001**, *20*, 195-245.
55. von Helden, G., Hsu, M.T., Kemper, P.R., Bowers, M.T.; Structures of Carbon Cluster Ions from 3 to 60 Atoms: Linears to Rings to Fullerenes. *The Journal of Chemical Physics*. **1991**, *95*, 3835-3837.
56. von Helden, G., Hsu, M.T., Gotts, N., Bowers, M.T.; Carbon Cluster Cations with Up to 84 Atoms: Structures, Formation Mechanism, and Reactivity. *The Journal of Physical Chemistry*. **1993**, *97*, 8182-8192.
57. Wyttenbach, T., Helden, G.v., Batka, J.J., Carlat, D., Bowers, M.T.; Effect of the Long-Range Potential on Ion Mobility Measurements. *Journal of the American Society for Mass Spectrometry*. **1997**, *8*, 275-282.
58. Wyttenbach, T., Witt, M., Bowers, M.T.; On the Stability of Amino Acid Zwitterions in the Gas Phase: The Influence of Derivatization, Proton Affinity, and Alkali Ion Addition. *Journal of the American Chemical Society*. **2000**, *122*, 3458-3464.
59. Mesleh, M.F., Hunter, J.M., Shvartsburg, A.A., Schatz, G.C., Jarrold, M.F.; Structural Information from Ion Mobility Measurements: Effects of the Long-Range Potential. *The Journal of Physical Chemistry*. **1996**, *100*, 16082-16086.

60. Bakker, J.M., Aleese, L.M., Meijer, G., von Helden, G.; Fingerprint IR Spectroscopy to Probe Amino Acid Conformations in the Gas Phase. *Physical Review Letters*. **2003**, *91*, 203003.
61. Kapota, C., Lemaire, J., Maître, P., Ohanessian, G.; Vibrational Signature of Charge Solvation vs Salt Bridge Isomers of Sodiated Amino Acids in the Gas Phase. *Journal of the American Chemical Society*. **2004**, *126*, 1836-1842.
62. Simons, J.; Bio-active molecules in the gas phase. *Physical Chemistry Chemical Physics-Cambridge-Royal Society of Chemistry*. **2004**, *7*, E7-E7.

Chapter 2

Cation Recombination Energy/Coulomb Repulsion Effects in Electron Capture Dissociation and Electron Transfer Dissociation

2.1 Electron Capture Dissociation and Electron Transfer Dissociation

Mass spectrometry has become increasingly useful for the identification and characterization of proteins and peptides with collision induced dissociation (CID) being the most frequently used technique for investigating these types of ions [1]. Infrared multiphoton dissociation (IRMPD) produces similar results to that of CID as they both depend on vibrational excitation of the precursor ion [2]. Electron capture dissociation (ECD) is one of several methods that have been developed to avoid the need to use slow heating vibrational excitation methods. ECD relies on a multiply charged precursor ion capturing a low energy thermal electron, which results in dissociation [3]. The ion/ion analog of ECD is electron transfer dissociation (ETD) [4,5]. During an ETD ion/ion reaction an electron is transferred from a reagent ion, such as the radical anions of fluoranthene or azobenzene, to a multiply charged cation [6,7]. Upon transfer of the electron, the charge-reduced ion undergoes fragmentation similar to ECD.

ECD and ETD both result in dissociation of backbone N-C $_{\alpha}$ bonds as well as cleavage of disulfide bonds. Electron based dissociation techniques are complementary to CID, as ECD/ETD is less dependent on amino acid sequence and labile post-translational modifications (PTMs) are preserved. The preservation of these labile PTMs has contributed heavily to the prevalence of using ECD/ETD for the analysis of

biomolecules. Preservation of CID-labile PTMs has been given as support for the claim that ECD and ETD are non-ergodic processes [8]. It is still unclear as to what extent intramolecular energy transfer occurs [9].

The mechanism by which ECD and ETD occurs is a subject of much debate. The initial proposed mechanism was developed by McLafferty and co-workers, referred to as the Cornell mechanism, suggested that electron localization occurred at a protonated arginine (R) or lysine (K) side chain resulting in neutralization and hydrogen atom loss [10]. The hydrogen atom is then transferred to either a carbonyl oxygen of an amide group leading to cleavage of the N-C α bond or to a disulfide bond which results in cleavage of that bond [11,12]. A coulomb stabilized dissociation, known as the Utah-Washington model, was developed independently by Tureček and co-workers and Simons and co-workers [13-15]. This mechanism considers electron capture at a π^* orbital delocalized over an amide group which gives rise to an electronic excited state of the charge-reduced radical cation. Then a proton can be abstracted to form a fragile aminoketyl radical, thus causing backbone dissociation. Alternatively, the charge stabilized amide π^* excited state can undergo N-C α bond cleavage to form an enole-imidate anion which can be stabilized by abstracting a proton intramolecularly to form product ions.

ECD and ETD mechanisms have much in common, especially when just considering the dynamics that take place after the electron has been captured by the ion. The dissociation that occurs during ECD and ETD result in N-C α cleavage, preservation of labile PTMs, and preferential cleavage of disulfide bonds. However, there are still both fundamental differences and distinctions in how the processes occur. One such fundamental difference is that electron capture is intrinsically more energetic than electron transfer because electron

transfer requires that the electron affinity of the reagent species be overcome in order to transfer the electron. Another major difference is that only electron capture occurs during the ion/electron process, while proton transfer always competes with electron transfer during the ion/ion reaction. Most reagents tested for ETD only remove protons from multiply charged biomolecules and show no propensity for electron transfer [16-23]. Only a few reagents have been shown to have electron transfer that competes with proton transfer, which has been described using a model based on Landau-Zener theory [24,25]. For electron transfer to compete with proton transfer, the reagent should have a low electron affinity and favorable Franck-Condon factors for vertical electron detachment [26,27].

Complications in comparing ECD and ETD data arise due to differences in how ECD and ETD experiments are implemented. ETD is typically done in electrodynamic ion traps with bath gas pressures of 1-10 mTorr, whereas ECD is done in ion cyclotron resonance cells with pressure in the 10^{-8} Torr range or lower. There are a few examples of ECD being implemented in ion traps with higher bath gas pressures [28-31]. These instruments employ reaction conditions that are much closer to those used for ETD.

In this work ECD and ETD experiments were both performed in ion traps operated in the presence of a bath gas in the 1-10 mTorr pressure range allowing for more direct comparisons to be made. The focus of this work is on systematically studying the product partitioning of ECD and ETD reactions. Emphasis was placed in the variation of cation recombination energy without changing net cation charge via variation of peptide size as well as variation of reaction exothermicity by changing the identity of the reagent (i.e., thermal electrons, radical anions of azobenzene and radical anions of 1,3-dinitrobenzene).

2.2 Experimental

2.2.1 Samples and Methods

The peptides XAGX, XAGAGX, and XAGAGAGAGX where both X's are lysine (K), arginine (R), or histidine (H) residues, and the peptides KAGKAGAGAG and KAGAGKAGAG were custom synthesized by CPC Scientific (San Jose, CA, USA). Methanol and glacial acetic acid were purchased from Fisher Scientific (Waltham, MA, USA). Azobenzene and 1,3-dinitrobenzene were obtained from Sigma Aldrich (St. Louis, MO, USA). Acetylation of the selected peptide was performed using a previously published protocol [32]. Peptide solutions were prepared to a concentration of 50 μ M for ETD studies and 10 μ M for ECD experiments in methanol/water/acetic acid (49.5/49.5/1).

The electron capture dissociation experiments were performed using a modified hybrid linear ion trap/time-of-flight mass spectrometer (LIT-TOF nano-Frontier, Hitachi High Technologies, Tokyo, Japan) [30]. A schematic of the LIT-TOF is shown in Figure 2.1. Peptide cations were generated using a nanospray emitter while electrons were generated using a tungsten filament. Ions of interest were isolated in the LIT_{CID} and transferred to the LIT_{ECD} where interactions with electrons occur. Peptide cations were reacted with the thermal electrons for 20 ms. Products from the ion/electron reaction were then released from the LIT_{ECD} and transmitted through the LIT_{therm} and into the acceleration region of the orthogonal TOF for mass analysis.

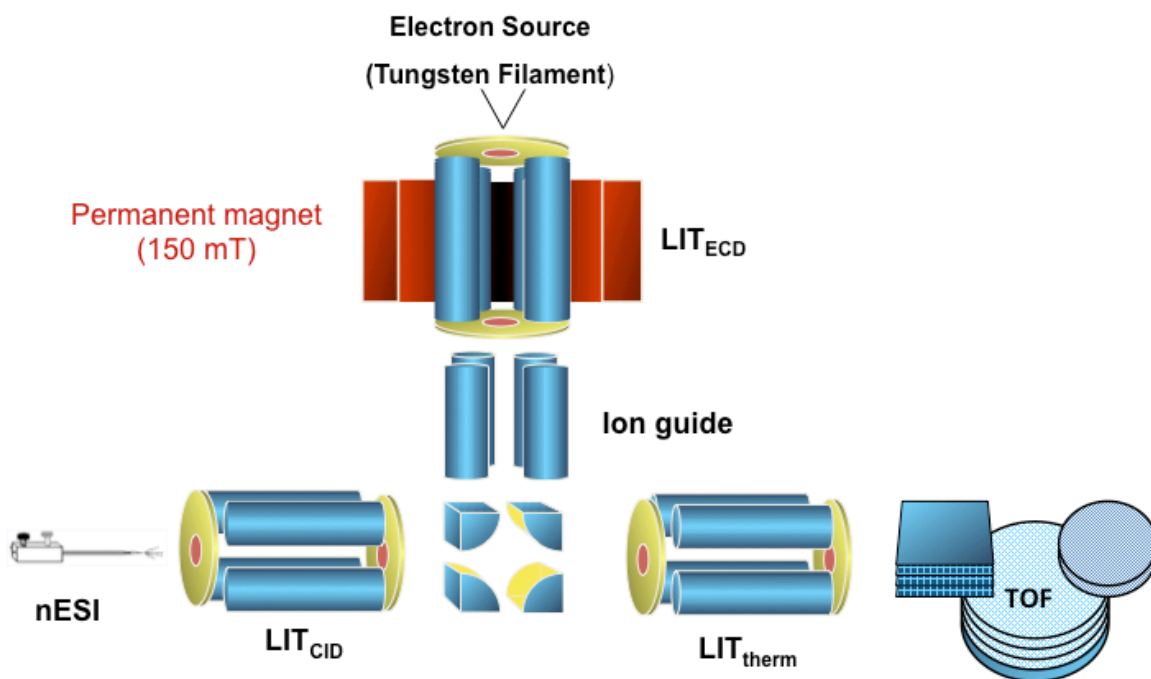


Figure 2.1. Schematic of the NanoFrontier LIT-TOF provided by the Hitachi Corporation (Hitachi High Technologies).

All electron transfer ion/ion reaction experiments were performed using a prototype version of a triple quadrupole/linear ion trap mass spectrometer (QTRAP, AB Sciex, Concord, ON, Canada).[33] A schematic of the QTRAP is shown in Figure 2.2. A home-built dual ionization source with a nanospray emitter for the formation of peptide cations and an atmospheric pressure chemical ionization (APCI) needle for the formation of reagent radical anions was coupled directly to the interface of the QTRAP [34]. Peptide cations and reagent radical anions are independently isolated in Q1 and then injected into the Q2 collision cell. The analyte cations and reagent anions were stored in Q2 for various reaction times and allowed to react in mutual storage mode [35,36]. The electron transfer ion/ion reaction products are then transferred from Q2 to Q3 for mass analysis via mass selective axial ejection (MSAE) [37]. Experiments were controlled using Daetalyt

3.14 research software provided by AB SCIEX. All of the ETD and ECD experiments were performed a minimum of three times.

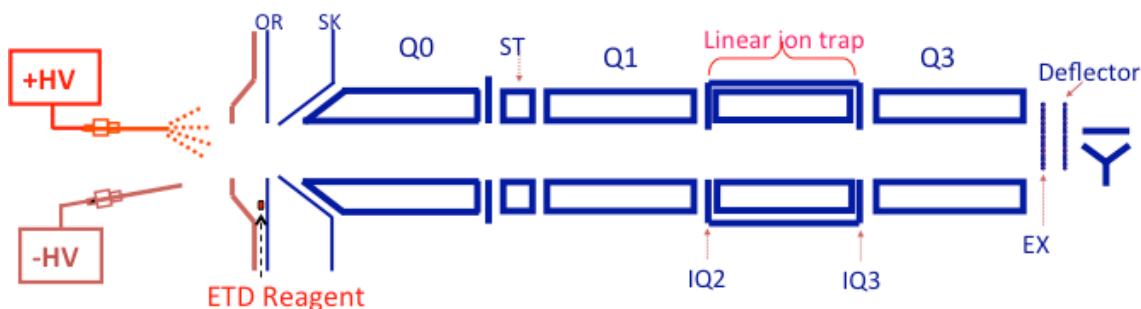


Figure 2.2. Schematic of the triple quadrupole/linear ion trap mass spectrometer (QTRAP, AB Sciex).

2.2.2 Data Analysis

The focus of this work is to study how products from ECD/ETD are partitioned.

Product partitioning is categorized here for ETD as:

1. proton transfer/ H^{\bullet} -loss (PT/ H^{\bullet} -loss)
2. c/z^{\bullet} -ion formation (c/z^{\bullet})
3. side-chain loss (sc loss)
4. electron transfer without dissociation (ET, noD).

Analogous categories for ECD are:

1. H^{\bullet} -loss
2. c/z^{\bullet} -ion formation
3. side-chain loss
4. electron capture without dissociation (EC, noD).

There are two levels of partitioning that are of interest. The first focuses on electron capture/transfer events that do not lead to a measurable change in analyte ion mass

(i.e., ET, noD and EC, noD) versus those channels that do lead to a measurable change in mass (i.e., ECD, ETD and PT channels). ECD and ETD channels are combinations of products of c/z[•]-ion formation (c/z[•]) and side-chain loss (sc loss)

For ETD partitioning definitions are:

$$\%ET, noD + \%ETD/PT/H^{\bullet} loss = 100\%$$

$$\%ETD/PT/H^{\bullet} loss = \frac{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss)}{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss + ET, noD)} \times 100\%$$

$$\%ET, noD = \frac{\sum ET, noD}{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss + ET, noD)} \times 100\%$$

The analogous relationships for ECD are:

$$\%EC, noD + \%ECD = 100\%$$

$$\%ECD = \frac{\sum (H^{\bullet} loss + c/z^{\bullet} + sc loss)}{\sum (H^{\bullet} loss + c/z^{\bullet} + sc loss + EC, noD)} \times 100\%$$

$$\%EC, noD = \frac{\sum EC, noD}{\sum (H^{\bullet} loss + c/z^{\bullet} + sc loss + EC, noD)} \times 100\%$$

The other level of partitioning focuses on the specific channels that lead to a change in analyte mass (i.e., side-chain loss, c/z[•]-ion formation, and PT/ H[•]-loss).

For ETD partitioning definitions are:

$$\%PT/H^{\bullet} loss + c/z^{\bullet} + sc loss = 100\%$$

$$\%PT/H^{\bullet} loss = \frac{\sum PT/H^{\bullet} loss}{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss)} \times 100\%$$

$$\%c/z^{\bullet} = \frac{\sum c/z^{\bullet}}{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss)} \times 100\%$$

$$\%sc loss = \frac{\sum sc loss}{\sum (PT/H^{\bullet} loss + c/z^{\bullet} + sc loss)} \times 100\%$$

The analogous relationships for ECD are:

$$\%H^{\cdot} loss + c/z^{\cdot} + sc\ loss = 100\%$$

$$\%H^{\cdot} loss = \frac{\sum H^{\cdot} loss}{\sum (H^{\cdot} loss + c/z^{\cdot} + sc\ loss)} \times 100\%$$

$$\%c/z^{\cdot} = \frac{\sum c/z^{\cdot}}{\sum (H^{\cdot} loss + c/z^{\cdot} + sc\ loss)} \times 100\%$$

$$\%sc\ loss = \frac{\sum sc\ loss}{\sum (H^{\cdot} loss + c/z^{\cdot} + sc\ loss)} \times 100\%$$

2.3 Results and Discussion

2.3.1 Cation Charge and Recombination Energy

Many factors are involved in the curve-crossing model for rationalizing the probability for electron transfer. This model is described briefly to provide context for the choice of peptides and reagents used in this work. Electron transfer (ET) is believed to occur at avoided crossings in the interaction between the reactant cation and anion. The cross section (σ_{ET}) for the two-body interaction is given by

$$\sigma_{ET} = P_{ET} \pi b_{ET}^2$$

where P_{ET} is the probability of reaction at classical impact parameter b_{ET} .

The square of the impact parameter (in atomic units) takes the form [38]:

$$b_{ET}^2 = r_{ET}^2 \left[1 + \frac{2|Z_{cat}Z_{an}|}{r_{ET}\mu v_{rel}^2} \right]$$

where Z_{cat} and Z_{an} are the unit charges of the cationic and anionic reactants; v_{rel} is the relative velocity of the ions; μ is the reduced mass of the collision partners; and r_{ET} is the distance at

which the electron transfer occurs. In this work, $Z_{cat} = +2$ and $Z_{an} = -1$ for all experiments.

An estimate of r_{ET} can be taken as the distance at which the ground state curves of the entrance and exit channels cross, which is given by (in atomic units):

$$r_{ET} \approx \frac{(Z_{cat}Z_{an})}{\Delta H_{ET}}$$

when at least one of the reactants is singly charged. The reaction enthalpy, ΔH_{ET} is given by

$$\Delta H_{ET} = EA_R - RE_{cat}$$

where EA_R represents electron affinity of the reagent used to generate the anion and RE_{cat} represents the recombination energy of the cation.

The probability for electron transfer at the point at which the entrance and exit channels cross, P_{ET} , is modeled based on Landau-Zener theory [24,25]. P_{ET} is determined by the non-adiabatic transition probability between states at the avoided crossing, P_{LZ} , by the relation:

$$P_{ET} = 2P_{LZ}(1 - P_{LZ})$$

The expression for P_{LZ} is:

$$P_{LZ} = \exp\left(\frac{2\pi H_{12}^2 \langle \chi_{1v'} | \chi_{2v''} \rangle^2}{\frac{dr}{dt} \left(\frac{Z_{cat}Z_{an}}{r_{ET}^2} \right)}\right)$$

where H_{12} is the coupling matrix element at the point of closest approach, $\langle \chi_{1v'} | \chi_{2v''} \rangle^2$ is the Franck-Condon (F-C) overlap between reactant and product vibrational wave functions associated with the transition from $1v'$ to $2v''$ for the reagent anion and its neutral counterpart, dr/dt is the radial velocity of the reactants at the crossing point and $Z_{cat}Z_{an}/r_{ET}^2$ gives the difference in the slopes of the entrance and exit channels at the crossing point. The radial velocity at the crossing point can be estimated by:

$$\frac{dr}{dt} \approx \left[\frac{2|Z_{cat}Z_{an}|}{r_{ET}\mu} \right]^{0.5}$$

The coupling matrix elements, which indicate the strength of the electronic coupling between states, are extremely difficult to determine for the large polyatomic systems of relevance to this work. A parameterized expression for H_{12} described by Olson, which is based on data collected for the reactions of a number of relatively small singly charged ions, has been used in this work [39,40]. This expression allows for an estimation of the coupling matrix based upon E_{AR} and RE_{cat} values:

$$H_{12} = (1.044\sqrt{EA_R}\sqrt{RE_{cat}}x) * \exp(-0.875x)$$

$$\text{where } x = \frac{r_{ET}}{2} (\sqrt{2EA_R}\sqrt{2RE_{cat}})$$

2.3.2 Product Ion Partitioning

The major focus of this work is to study how products are partitioned during ECD/ETD when cation recombination energy is changed. An important factor for determining RE_{cat} , as well as ΔH_{ET} , for a multiply charged ion is Z_{cat} [41]. As shown in 2.3.1 Z_{cat} can affect σ_{ET} in many ways. These include the roles that Z_{cat} plays in determining b_{ET} , r_{ET} , and dr/dt . Thus the interpretation of the effect of changes in Z_{cat} in ETD is complicated by various ways in which it affects σ_{ET} . To separate the role of RE_{cat} from that of total charge in this study, we chose to examine peptide ions in a single charge state (viz., doubly protonated peptides) of various sizes. To minimize protonation site effects peptides of the form XAGX, XAGAGX, and XAGAGAGAGX were compared with X= K, H, or R. RE_{cat} is expected to follow the order

$[XAGX + 2H^+]^{2+} > [XAGAGX + 2H^+]^{2+} > [XAGAGAGAGX + 2H^+]^{2+}$ due to decreasing electrostatic repulsion and possible increasing charge solvation as the size of the peptide increases.

Figure 2.3 summarizes the product ion partitioning in ECD (Figure 2.3a) and in ETD (Figure 2.3b). For ECD, electron capture is generally accepted as proceeding via initial capture into high lying Rydberg orbitals, as depicted by the electron orbit around a multiply protonated peptide. The electron is presumed to evolve to lower lying Rydberg orbitals and eventually occupy electronic states that give rise to ECD products. These electronic states are represented in Figure 2.3a as being present in the charge reduced intermediate. If this excited intermediate can be stabilized via emission or collisions a long-lived charge reduced species can be observed, which is represented by the EC, noD channel. Fragmentation channels can be classified as side-chain loss, c/z'-ions, and hydrogen atom loss (H•-loss). In the case of ETD, studies on reaction rates have suggested that the rate-determining step in gas-phase ion/ion reactions involving large polyatomic ions is the formation of a stable orbiting pair, as depicted by the orbit in Figure 2.3b [20,21]. This orbit can be considered the ion/ion analog to a high lying Rydberg state associated with ECD. Collapse of the orbiting complex due to collisions or tidal effects brings reactants in close enough proximity to each other to allow for an electron to transfer from the anion to the cation or for a proton to transfer from the cation to the anion [42,43]. At least some of these cation products from electron transfer will have sufficient enough energy to lead to dissociation. However, stabilization of this intermediate via emission or collision can result in a long-lived charge reduced species, as represented by the ET, noD channel. A complication results due to the fact that proton transfer and electron transfer followed by H•-loss are indistinguishable. The

ECD results are particularly helpful in interpreting the ETD partitioning due to the fact that proton transfer can not occur during the ECD process.

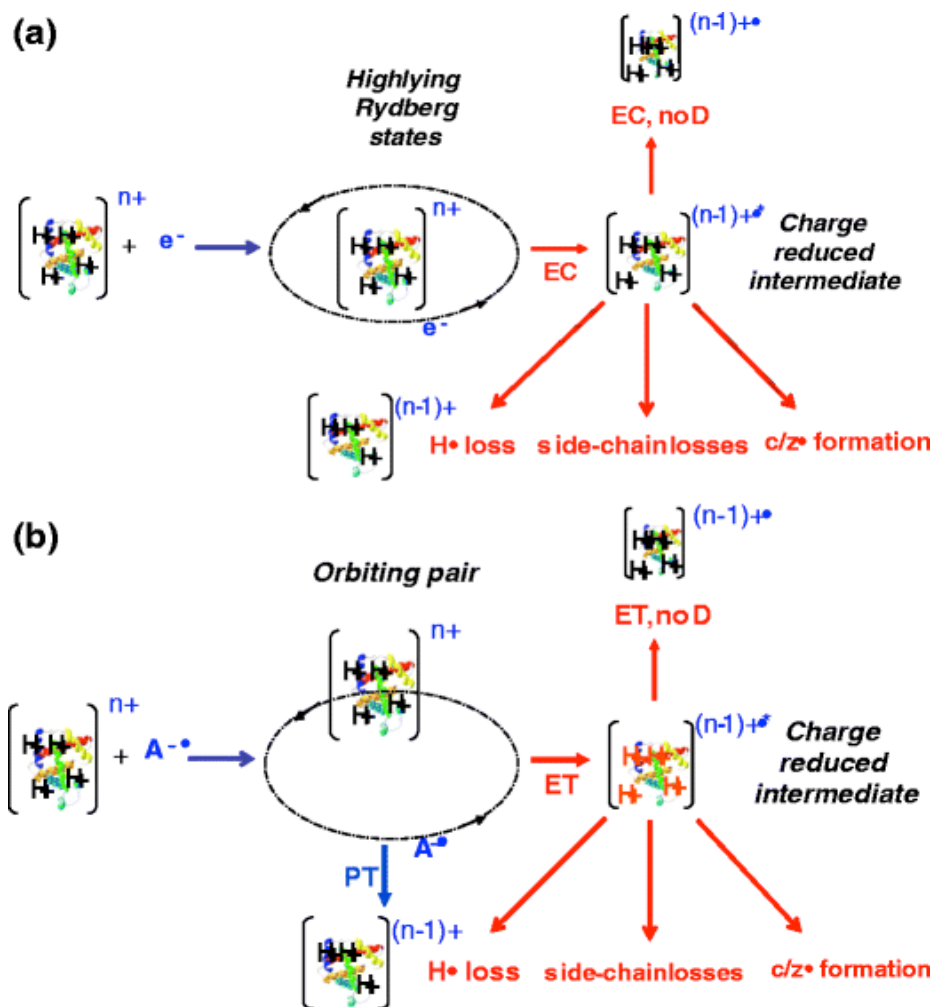


Figure 2.3. Schematic representation of product partitioning in (a) ECD and (b) ETD.

Tables 2.1 and 2.2 list the product ion partitioning in ETD of the doubly protonated $X(AG)_nX$ peptides using azobenzene (EA=0.6 eV) and 1,3-dinitrobenzene (EA=1.7 eV), respectively, and Table 2.3 summarizes the analogous results for ECD. For all three cases, doubly protonated peptides with $n=1, 2$, and 4 for $X = K, R$, and H were examined. Doubly protonated isomeric peptides KAGAGAGAGK, KAGAGKAGAG, and KAGKAGAGAG were subjected to ETD and ECD to determine if the relative positions of the basic residues are a key factor in these experiments. To determine the possible role of the N-terminus in the

X = R and H peptides, the N-terminally acetylated doubly protonated Ac-RAGAGR and Ac-HAGAGH were subjected to ETD.

Table 2.1. Product Ion Partitioning for the Ion/Ion Reactions of Doubly Protonated Peptides with the Radical Anion of Azobenzene

Peptide	%ETD/PT/H [•] loss	% ET, no D	% PT/H [•] loss	%c/z [•]	% side-chain
KAGK	100	0	26	69	5
KAGAGK	99	1	10	86	4
KAGAGAGAGK	98	2	20	78	2
KAGAGKAGAG	99	1	21	76	4
KAGKAGAGAG	99	1	17	76	7
RAGR	100	0	17	62	21
RAGAGR	99	1	23	55	22
Ac-RAGAGR	95	5	34	39	27
RAGAGAGAGR	98	2	48	28	24
HAGH	84	16	21	69	10
HAGAGH	80	20	27	66	7
Ac-HAGAGH	65	35	53	44	3
HAGAGAGAGH	83	17	32	62	6

Table 2.2. Product Ion Partitioning for the Ion/Ion Reactions of Doubly Protonated Peptides with the Radical Anion of 1,3-Dinitrobenzene

Peptide	%ETD/PT/H [•] loss	% ET, no D	% PT/H [•] loss	%c/z [•]	% side-chain
KAGK	100	0	64	35	1
KAGAGK	98	2	43	56	1
KAGAGAGAGK	97	3	79	21	0
KAGAGKAGAG	99	1	76	23	1
KAGKAGAGAG	99	1	77	22	1
RAGR	100	0	26	62	12
RAGAGR	97	3	71	21	8
Ac-RAGAGR	100	0	85	10	5
RAGAGAGAGR	97	3	91	2	7
HAGH	92	8	65	32	4
HAGAGH	89	11	85	12	3
Ac-HAGAGH	86	14	96	4	0
HAGAGAGAGH	93	7	92	7	1

Table 2.3. Product Ion Partitioning from the Ion/Electron Reactions of Doubly Protonated Peptides with Near-Thermal Electrons

Peptide	% ECD/H [•] loss	% EC, no D	% H [•] loss	% c/z [•]	% side-chain
---------	---------------------------	------------	-----------------------	--------------------	--------------

KAGK	100	0	36	57	7
KAGAGK	100	0	17	81	2
KAGAGAGAGK	100	0	4	93	3
KAGAGKAGAG	100	0	4	92	4
KAGKAGAGAG	99.9	0.1	3	93	4
RAGR	100	0	21	58	21
RAGAGR	100	0	12	73	15
RAGAGAGAGR	100	0	7	79	14
HAGH	100	0	51	35	14
HAGAGH	100	0	22	69	9
HAGAGAGAGH	86	14	9	79	12

A number of observations can be drawn from examination of the results in Tables 2.1, 2.2, and 2.3. First, the results are fully consistent with prior observations that distinguished the ETD behaviors of the three common basic residues. For example, the peptides with X = H showed the greatest contribution of ET, noD. HAGAGAGAGH was the only model peptide dication to show significant EC, noD. This observation is consistent with the “histidine effect” described by Tureček et al., which arises from an isomerization process that takes place within the radical species generated by electron capture by the protonated imidazole side chain [44-47]. Peptide cations with X = R showed the greatest degree of side-chain cleavage, which may be related to the reportedly poor H[•]-donating capability of the hypervalent arginine side-chain radical [48]. The greatest relative contribution from cleavages to yield complementary c/z[•]-ions was noted for X = K. Experiments with N-terminally acetylated X = H and X = R peptides, which precluded any contributions that might arise due to proton solvation by a primary amine, lead to increased contributions from ET, noD and side-chain loss, respectively. The effect of acetylation was most significant for the X = H peptides, which showed the unmodified ion to yield 20 % ET, noD while the N-terminally acetylated version of the ion yield 35 % ET, noD.

A second set of observations addressed the importance of the relative positions of the basic residues versus peptide size. The issue revolves around whether the isomeric KAGAGAGAGK, KAGAGKAGAG, and KAGKAGAGAG ions would show product ion partitioning similar to one another, product ion partitioning similar to the smaller peptides with the same number of AG residues between lysines (e.g, KAGKAGAGAG versus KAGK), or some intermediate partitioning. As shown in Tables 2.1, 2.2, and 2.3, the three isomeric 10-mer polypeptide ions behaved much more similarly to one another than to any of the smaller peptides for each of the respective experiments. While the three isomeric peptide ions showed quite similar overall partitioning into c/z^+ -ions with a given “reagent” (vis., anion or electron), they each showed somewhat different partitioning among the various individual c/z^+ -channels (see Figure 2.4, which is discussed later). The relative abundances of c/z^+ -ions is a subject of a current study within the ECD/ETD community but is not a primary focus of this work [49-52]. For example, the ECD experiment for doubly protonated KAGKAGAGAG yielded a small relative proportion of H^+ -loss, a 3 % - 4 % value similar to the other isomeric 10-mer cations, while KAGK cation shows H^+ -loss to be roughly 36 % of the total ECD products. This set of observations clearly shows that the peptide size for a given charge is more important in determining product ion partitioning than is the relative location of the basic residues in the peptide sequence.

A third set of observations indicates that for all direct comparisons in the ETD experiments summarized in Tables 2.1 and 2.2, the %PT/ H^+ -loss channel was significantly lower for azobenzene than for 1,3-dinitrobenzene. For example, a range of 17 % - 21 % for the %PT/ H^+ -loss channel was noted for the three isomeric 10-mer cations in reactions with the azobenzene radical anion whereas a range of 76 % - 79 % for the %PT/ H^+ -loss channel

was observed in reactions with the 1,3-dinitrobenzene anion. Given the experimental error associated with these measurements, the %PT/ H^\bullet -loss values for the three isomeric cations with a given reagent are statistically the same whereas the values for the two different reagents are quite different. This observation is consistent with previous studies on the role of the reagent anion in the competition between electron transfer and proton transfer [27]. A change in electron affinity of the reagent in the model summarized above has a major impact in determining the probability for electron transfer, P_{ET} , via its role in determining ΔH_{ET} , r_{ET} , and H_{12} . The F-C factors associated with each reagent also affect H_{12} . Figure 2.4 compares plots of P_{ET} for azobenzene and 1,3-dinitrobenzene as a function of cation RE using the model summarized, the Olson parameterization for H_{12} , and assuming $EA=0.6$ eV, $F-C=0.18$ for azobenzene and $EA=1.7$ eV, $F-C=0.27$ for 1,3-dinitrobenzene. Also indicated in the figure is the range of cation recombination energies likely to be relevant to this study [53].

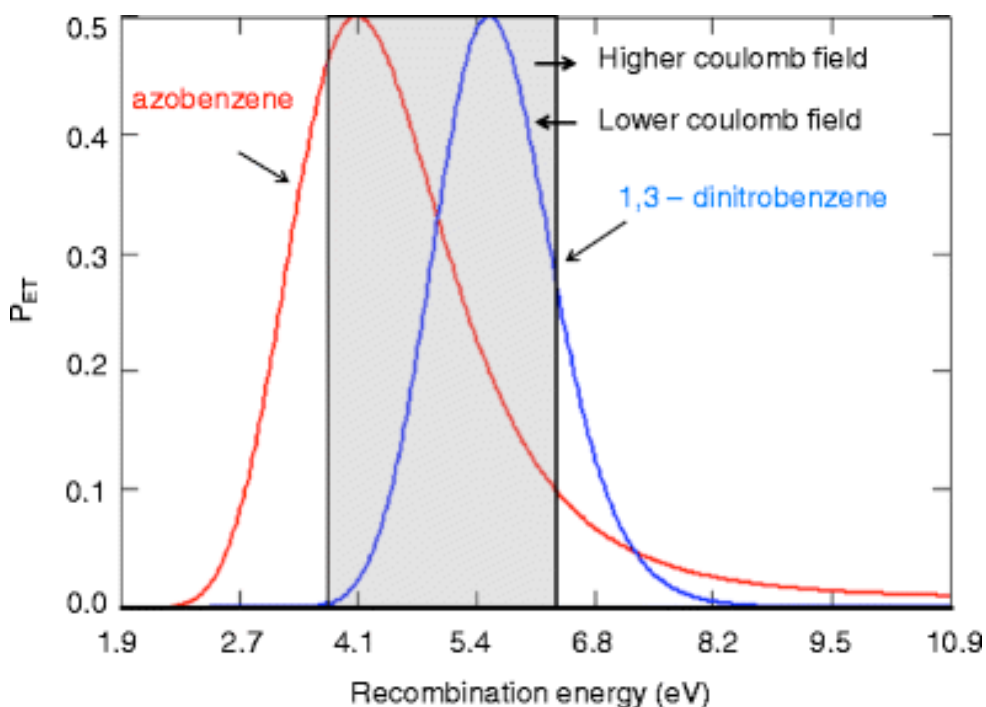


Figure 2.4. Plots of P_{ET} for the formation of ground state product cations as a function of cation recombination energy for azobenzene (left-most curve) and 1,3-dinitrobenzene (right-

most curve) radical anions generated using the model summarized above. The gray area indicates the range of cation recombination energies likely to apply to the cations in Tables 2.1, 2.2, and 2.3.

Given the assumptions and uncertainties in the values used in this model, the positions of the curves are expected to be semiquantitative at best. However, they are useful in making qualitative comparisons of different reagents. The curves in Figure 2.4 represent those for the generation of products in ground electronic states and therefore suggest that there is an optimal cation recombination energy. However, the cations are generally believed to be formed in excited electronic states, the curves for which are shifted to the right of the respective ground state curves. The significance of crossings that lead to the population of excited electronic states in the context of Figure 2.4 is that the decrease in P_{ET} at high RE values may not be observed as the probabilities for crossing into excited electronic states are expected to be increasingly likely as RE increases. However, if the cation RE is on the low side of the maximum, there are no lower energy states to populate. Hence, while the cation RE may not become too high to observe electron transfer, it can be too low. Figure 2.4 is consistent with the greater extent of electron transfer with the azobenzene anions than with 1,3-dinitrobenzene anions provided the cation recombination energies are not all fully to the right of the 1,3-dinitrobenzene curve.

2.3.3 H^{\bullet} -atom Loss

While the comparison of Tables 2.1 and 2.2 is consistent with previous findings regarding the relative efficiencies of azobenzene and 1,3-dinitrobenzene anions as ETD reagents, the fact that proton transfer and electron transfer followed by H^{\bullet} -loss give the same products introduces some ambiguity in interpretation. For example, a curious trend in %PT/ H^{\bullet} -loss is noted for the $K(AG)_nK$ ions for both reagents. The %PT/ H^{\bullet} -loss observed with azobenzene anions was 26 % for doubly protonated KAGK, 10 % for doubly protonated

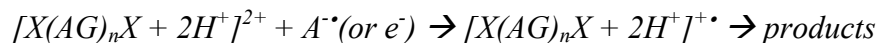
KAGAGK, and 17 % - 21 % for the isomeric 10-mer dications. The analogous %PT/ H[•]-loss values for the dinitrobenzene anions were 64 % for KAGK, 43 % for KAGAGK, and 76 %-79 % for the three isomeric 10-mer dications. The fact that the %PT/ H[•]-loss values for the KAGK dications are higher than those for KAGAGK ion with both reagent anions is inconsistent with the expected trend in RE values for the respective doubly protonated peptides (i.e., KAGK > KAGAGK > KAGAGAGAGK) if proton transfer is the dominant process. Higher cation RE values are expected to increase the probability for electron transfer (see Figure 2.4). This observation, however, may not be inconsistent with expectation if electron transfer followed by H[•]-loss is relatively more important for the small peptide than for the larger peptides. The ECD data of Table 2.3 are particularly informative in this regard. The contribution of H[•]-loss channel is clearly inversely related to the charge per residue in the ECD data. This observation suggests that the relatively high %PT/ H[•]-loss values for the KAGK dications arises from higher relative contributions from H[•]-loss than occurs with the larger dications. We speculate that the reason for greater H[•]-loss from high charge per residue peptides may be due to a structural effect from high electrostatic repulsion. Solvation of excess charge via interaction with polar groups, such as the carbonyl oxygen of the backbone, may be minimized when the excess charges are in close proximity. This is supported by the fact that for the ten residue peptide isomers with lysine as the basic site there is essentially no difference in the product ion partitioning. Such an intramolecular solvation interaction facilitates H[•]-transfer to an amide linkage and is a key element in the so-called “Cornell” mechanism for ECD to produce complementary c/z[•]-ions [10]. In the absence of a preexisting interaction between the excess charge and the peptide backbone, the

likelihood for capture/transfer of the hydrogen radical is diminished, thereby giving rise to greater H[•]-loss.

It is noteworthy that the relative contribution from H[•]-loss in the ECD of doubly protonated XAGX ions exceeds significantly the combined PT/ H[•]-loss contribution for the same reaction with azobenzene anions (compare Tables 2.1 and 2.3). This indicates that H[•]-loss takes place more readily from the ions of highest charge per residue under ECD conditions than under ETD conditions, at least for the azobenzene anions. The underlying reason why H[•]-loss might be more likely under ECD conditions than ETD conditions for the high charge per residue ions is not readily apparent from these data. Several differences between the electron capture and electron transfer processes might play a role. For example, electron capture is always a more exothermic process than electron transfer as the electron affinity of the reagent must be overcome in the latter process. The relevant studies to date suggest that H[•]-loss is increasingly competitive at higher internal energies [54]. If it is assumed that both ECD and ETD proceed through an evolution in electronic states from the initially formed Rydberg states to lower-lying states (i.e., a process of intramolecular electron transfer), and that H[•]-loss, c/z[•]-ion formation, and side-chain loss can take place via different electronic states, the initial population of electronic states might be important in determining product ion partitioning. Differences in initial excited electronic state population in electron capture versus electron transfer might therefore play a role in differences in product ion partitioning between ECD and ETD [55,56].

While differences in total internal energy between ECD and ETD, which may also be reflected in different partitioning in excited electronic states, might account for different extents of H[•]-loss, an alternative factor for the lesser extent of H[•]-loss from high charge per

residue peptides in ETD versus ECD is differences in the angular momenta, J, of the incipient charge reduced peptides (i.e., $[X(AG)_nX + 2H^+]^{+\bullet}$) produced via an ion/ion versus ion/electron process, as indicated in the following reaction:



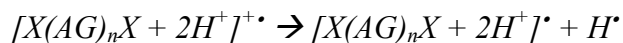
The angular momentum, L, associated with a two-body interaction is given by:

$$L = \mu v_{rel} b$$

where μ is the reduced mass of the collision partners, b is the impact parameter, and v_{rel} is the initial relative velocity of the collision partners. It is apparent from the equation above that angular momentum effects are most likely to be observed with large impact parameter processes. At comparable impact parameters, however, the low mass of the electron leads to values of L for the ion/electron case that are orders of magnitude lower than those for the ion/ion case. The importance of angular momentum in dissociation, particularly for H^\bullet -loss, has already been discussed for ion/molecule reactions, which have substantially smaller impact parameters than ion/ion reactions [57-60]. Long-lived intermediates generated by reactions of ions with molecules can be formed in which all of the initial angular momentum of the pair, L, is converted to angular momentum of the collision complex, J. The magnitude of J is significant for subsequent dissociation of the complex as the rotation of the collision complex creates a rotational barrier in the exit channel. The magnitude of the rotational barrier, $V_{eff,rot}$, to a first approximation is given by:

$$V_{eff,rot} = \frac{J^4}{8\alpha e^2 \mu^2}$$

where α is the polarizability of the neutral and μ , in this case, is the reduced mass of the dissociation products. Of all of the ECD/ETD dissociation channels, the rotational barrier for the reaction:



is highest due to the low polarizability of the hydrogen atom (0.667 Å³) and the fact that the products have the lowest reduced mass of all of the possible dissociation channels. For these reasons, a higher degree of angular momentum in the incipient charge-reduced product formed via ETD versus ECD is deemed to be the most likely interpretation for why ECD leads to a higher degree of H[•]-loss for the high charge per residue peptides. This conclusion is tempered by a lack of detailed understanding of the dynamics of the ion/ion reaction that may otherwise allow for an estimation of the L→J efficiency (i.e., the extent to which angular momentum of the orbiting pair is partitioned into rotation of the charge reduced peptide versus the other available channels, such as the relative translation of the charge-reduced peptide and the neutralized reagent, rotation of the reagent, etc.).

2.3.4 *c/z[•]-Ion Channels in ECD/ETD*

Tables 2.1, 2.2, and 2.3 provide partitioning into the main categories of ECD/ETD dissociation channels (viz., H[•]-loss, side-chain loss, and c/z[•]-formation). It is also of interest to compare ECD and ETD for partitioning among these categories, particularly for the sequence informative channels that lead to c/z[•]-ions. Figures 2.5 and 2.6 compare the relative contributions of the various dissociation channels that lead to c/z[•]-ions from ECD and ETD (azobenzene reagent) for the various lysine containing model peptides.

Contributions from side-chain loss are also included for both cases and H[•]-loss is included in the ECD data. Comparison of the ECD and ETD data for KAGK

(Figure 2.5a), for example, shows preference for the same backbone cleavages; the c₃-ion is dominant in both ECD and ETD. The extent of side-chain loss is also very similar in both cases. Doubly protonated KAGAGK behaves in a similar manner as demonstrated in Figure

2.5b where the c_3 channel is again the most dominant. The pattern of abundances of the c_n ion is also similar.

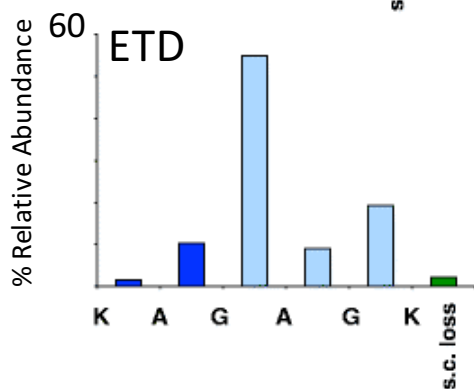
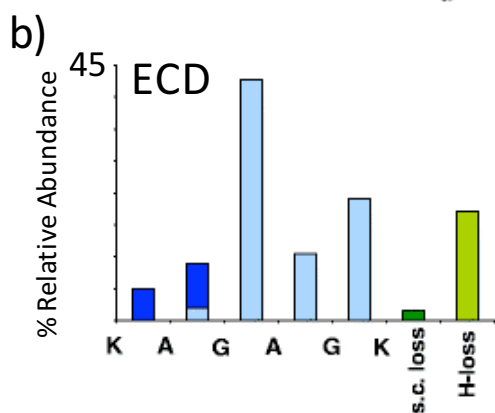
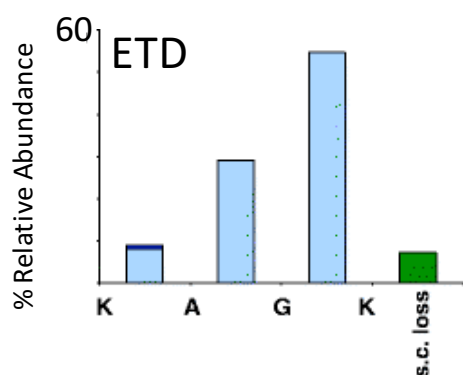
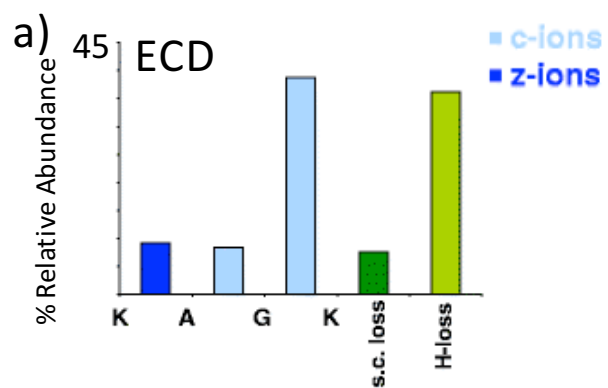


Figure 2.5. Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[KAGK + 2H^+]^{2+}$ (b) ECD (top) and ETD (bottom) of $[KAGAGK + 2H^+]^{2+}$

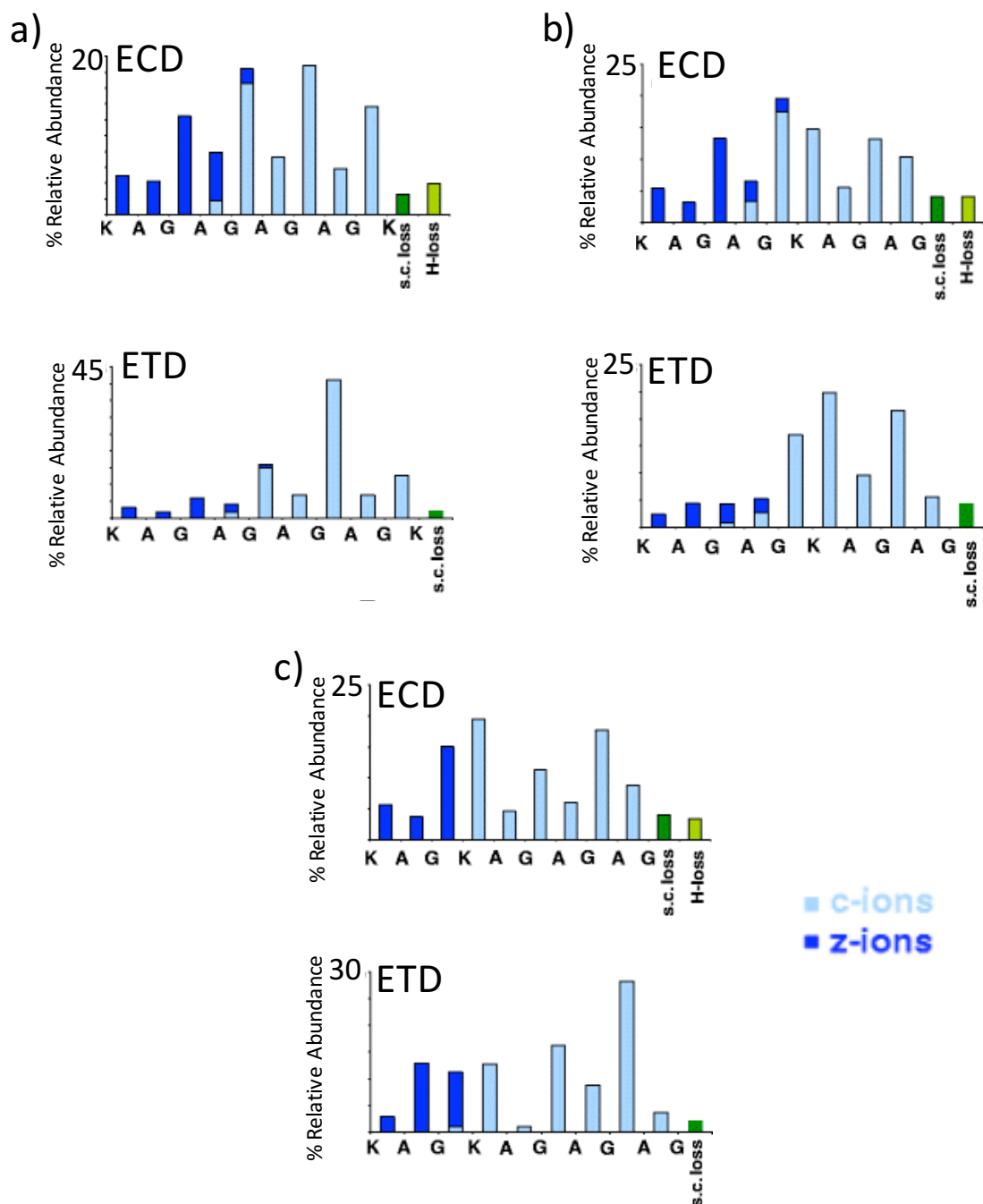


Figure 2.6. Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of [KAGAGAGAGK + 2H⁺]²⁺, (b) ECD (top) and ETD (bottom) of [KAGAGKAGAG + 2H⁺]²⁺, (c) ECD (top) and ETD (bottom) of [KAGKAGAGAG + 2H⁺]²⁺

In fact, the patterns of c/z⁺-ions in the ECD and ETD spectra were similar for all the lysine-containing peptides. Figure 2.6 compares the results for ECD and ETD (azobenzene as reagent) for the isomeric lysine-containing 10-mers. The similarities in the fragmentation

patterns are remarkable considering that the spectra were collected using different instrument platforms (viz., hybrid triple quadrupole/linear ion trap versus linear ion trap/time-of-flight). The ETD experiments generally yielded lower relative abundances for the products of low mass/charge than did the ECD experiments (compare, for example, the spectra of Figures 2.6a), which may reflect differences in collection/detection efficiencies in the two instruments. The pattern of c/z^{\bullet} -ions (i.e., the identities and abundances of the c/z^{\bullet} -ions relative to one another) was also remarkably similar in the ETD spectra generated with 1,3-dinitrobenzene anions (data not shown).

A summary of the dissociation product partitioning for the arginine and histidine containing model peptides is provided in Figures 2.7 and 2.8, respectively. In the case of the $R(AG)_nR$ peptides, the c/z^{\bullet} -ion patterns again appear to be similar for $n=1$ and $n=2$ peptides, recognizing the apparent lower relative collection/detection efficiencies for the ETD experiment. The ETD experiment yielded significantly fewer c/z^{\bullet} -ions, both in variety and relative abundance, than did the ECD experiment of the $R(AG)_4R$ dication, however. The $\%c/z^{\bullet}$ for this ion was the lowest of all the model peptides for both ETD reagents while the PT/H^{\bullet} -loss and side-chain loss channels were both relatively high. Furthermore, most of the c/z^{\bullet} -ion product signal was concentrated in the c_9 -ion.

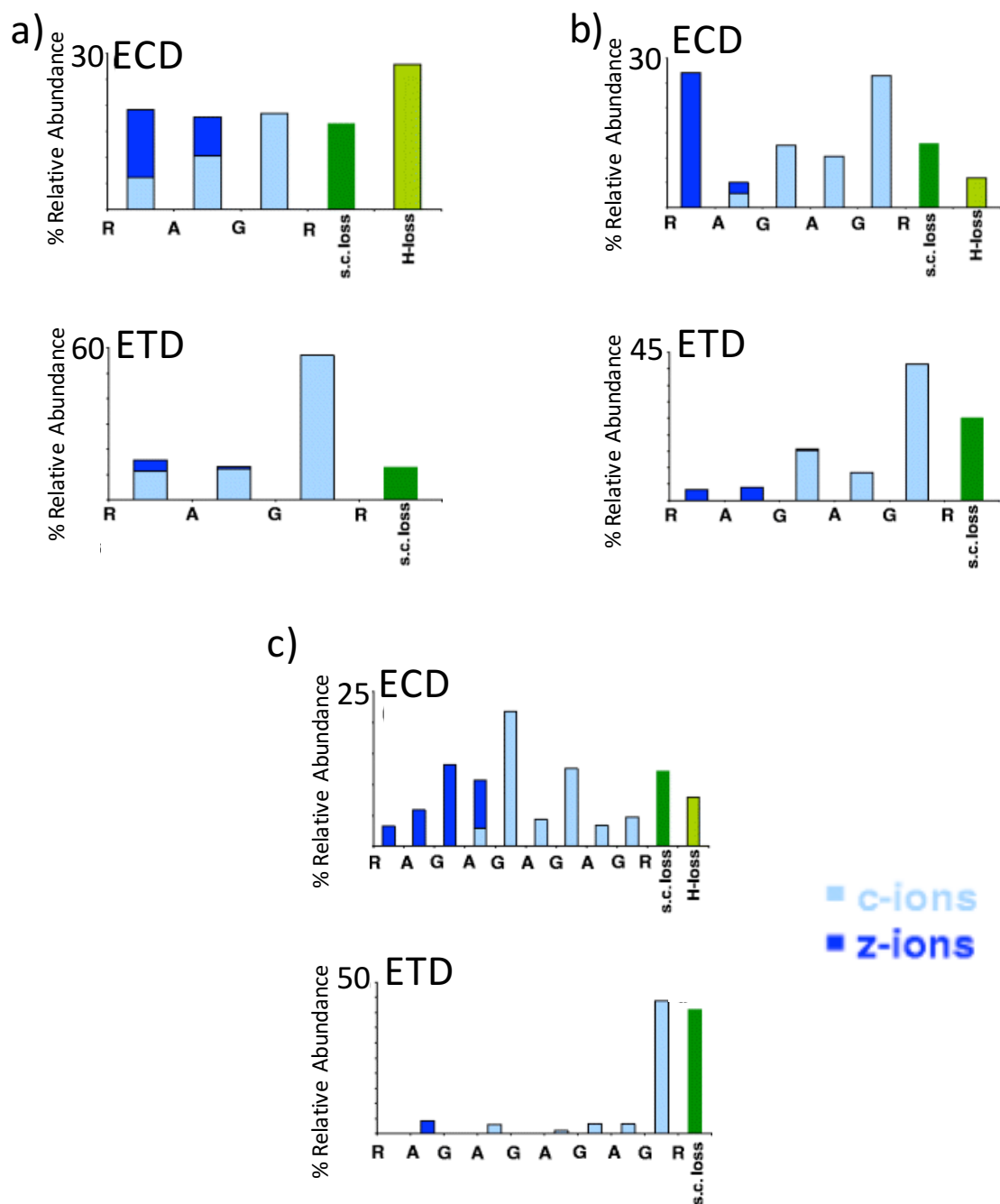


Figure 2.7. Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[RAGR + 2H^+]^{2+}$, (b) ECD (top) and ETD (bottom) of $[RAGAGR + 2H^+]^{2+}$, (c) ECD (top) and ETD (bottom) of $[RAGAGAGAGR + 2H^+]^{2+}$

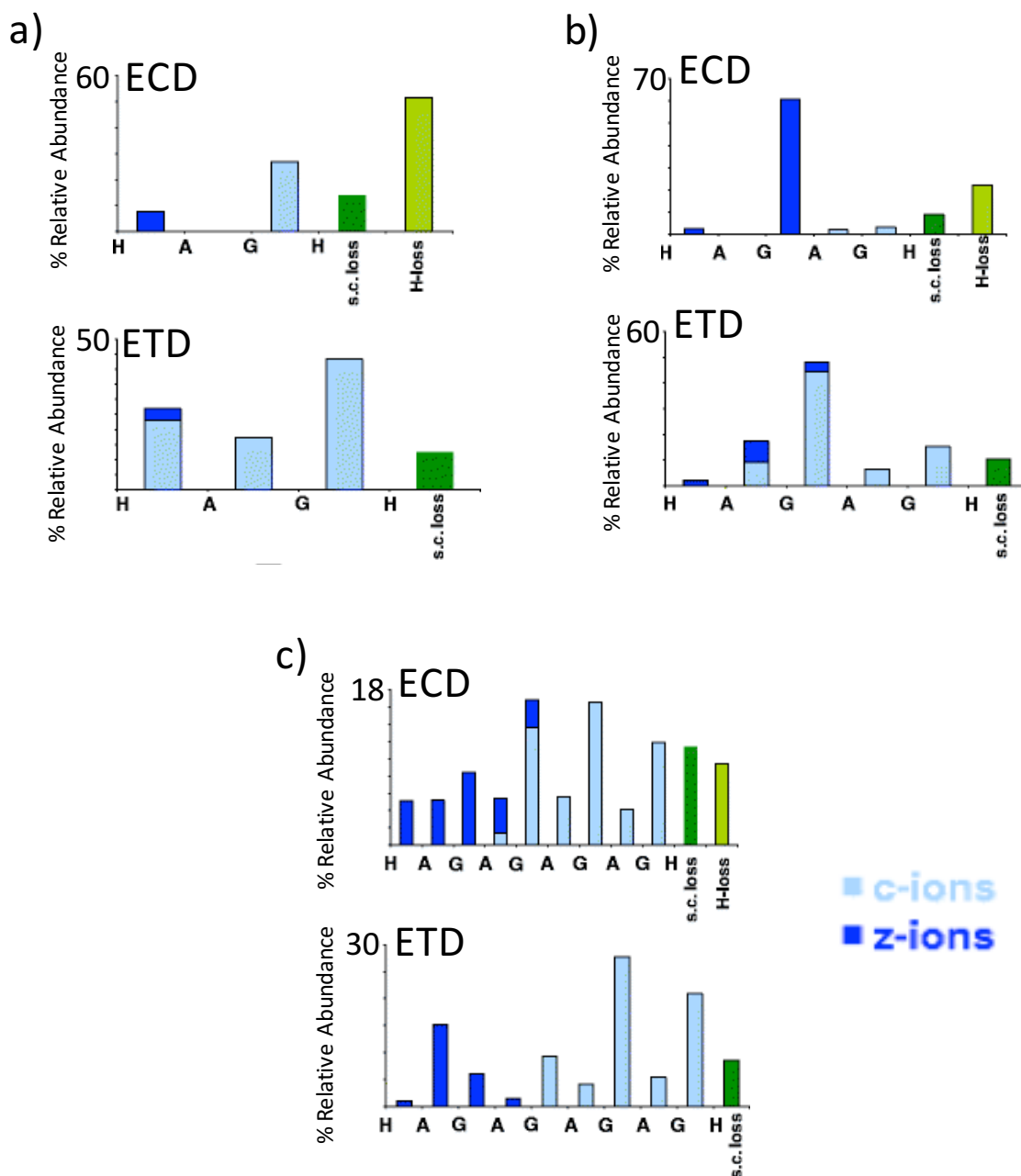


Figure 2.8. Relative abundances of product ions resulting from: (a) ECD (top) and ETD (bottom) of $[HAGH + 2H^+]^{2+}$, (b) ECD (top) and ETD (bottom) of $[HAGAGH + 2H^+]^{2+}$, (c) ECD (top) and ETD (bottom) of $[HAGAGAGAGH + 2H^+]^{2+}$

In the case of the $H(AG)_nH$ peptide dications (see Figure 2.8), the ECD and ETD c/z^{\bullet} -ion patterns were quite similar for the $H(AG)_4H$ ion, whereas ECD yielded a somewhat lower variety of c/z^{\bullet} -ions than did ETD for the smaller peptide ions. This observation may well be related to the fact that the small histidine containing ions yielded the highest contributions

from H[•]-loss under ECD conditions. If the angular momentum argument described above is a significant factor, an increased barrier for H[•]-loss in the charge-reduced peptide ions generated via ETD may allow more H[•]-transfer channels that lead to c/z[•]-ions competing with H[•]-loss.

2.4 Conclusions

A series of doubly protonated model peptides of the form X(AG)_nX have been subjected to ECD and ETD in electrodynamic ion traps operated in the presence of a bath gas in the mTorr pressure regime. A full accounting of the various competitive product ion channels was performed as a function of several experimental variables. These included the identity of the basic amino acids present in the peptide (X = K, R, or H), the length of the peptide (n=1, 2, or 4), and the reagent (near-thermal electrons, anions of azobenzene, or anions of 1,3-dinitrobenzene). By varying n, for a given charge state, it is possible to follow effects due to changes in recombination energy (or charge per residue). By varying X, it is possible to follow effects due to sites of protonation. By varying reagents, it is possible to follow effects due to changes in reagent electron affinity and reagent mass.

The results obtained in this work with both ECD and ETD are fully consistent with previous studies on the role of protonation site on product ion partitioning in ETD. The fact that proton transfer and electron transfer followed by H[•]-loss both lead to the same nominal products in ETD complicates interpretation of product channel partitioning in ETD studies. The ECD results were instructive in this regard in that they showed the tendency for H[•]-loss to correlate with charge per residue. Proton transfer, on the other hand, is expected to be inversely related to charge per residue (i.e., recombination energy). In general, the observed trends in %PT/H[•]-loss were consistent with expectation for n=1, 2, and 4 for the two anionic

reagents. The apparently anomalous ETD behavior of the $K(AG)_nK$ peptides, which showed relatively high %PT/ H^\bullet -loss values for the smallest peptides, can be rationalized by relatively high contributions from electron transfer followed by H^\bullet -loss, as suggested by the ECD data. Interestingly, ECD appears to lead to higher degrees of H^\bullet -loss from ions of high charge per residue than does ETD. This observation may arise from differences in internal energies, which would be reflected in different initial populations of different electronic excited states. An alternative possible factor introduced here is differences in angular momenta of the charge-reduced peptide ions generated by electron capture (low additional angular momentum) versus electron transfer (high additional angular momentum). The angular momentum effect may also enhance the likelihood for ET, noD relative to EC, noD, although the data set generated here is likely to be too small to draw a general conclusion. Finally, the partitioning among the various c/z $^\bullet$ -ion channels was remarkably similar in ECD and ETD for most of the model peptides studied despite the difference in the electron capture and electron transfer processes.

2.5 REFERENCES

1. Mitchell Wells, J.; McLuckey, S. A. Collision-Induced Dissociation (CID) of Peptides and Proteins. *Methods in Enzymology*. **2005**, *402*, 148-185.
2. Little, D. P.; Speir, J. P.; Senko, M. W.; O'Connor, P. B.; McLafferty, F. W. Infrared Multiphoton Dissociation of Large Multiply Charged Ions for Biomolecule Sequencing. *Analytical Chemistry*. **1994**, *66*, 2809-2815.
3. Zubarev, R. A.; Kelleher, N. L.; McLafferty, F. W. Electron Capture Dissociation of Multiply Charged Protein Cations. A Nonergodic Process. *Journal of the American Chemical Society*. **1998**, *120*, 3265-3266.
4. Coon, J. J. Collisions Or Electrons? Protein Sequence Analysis in the 21st Century. *Analytical Chemistry*. **2009**, *81*, 3208-3215.
5. Syka, J. E. P.; Coon, J. J.; Schroeder, M. J.; Shabanowitz, J.; Hunt, D. F. Peptide and Protein Sequence Analysis by Electron Transfer Dissociation Mass Spectrometry. *Proceedings of the National Academy of Science*. **2004**, *101*, 9528-9533.
6. Pitteri, S. J.; Chrisman, P. A.; McLuckey, S. A. Electron-Transfer Ion/Ion Reactions of Doubly Protonated Peptides: Effect of Elevated Bath Gas Temperature. *Analytical Chemistry*. **2005**, *77*, 5662-5669.
7. Coon, J. J.; Syka, J. E.; Schwartz, J. C.; Shabanowitz, J.; Hunt, D. F. Anion Dependence in the Partitioning between Proton and Electron Transfer in Ion/Ion Reactions. *International Journal of Mass Spectrometry*. **2004**, *236*, 33-42.
8. Breuker, K.; Oh, H.; Lin, C.; Carpenter, B. K.; McLafferty, F. W. Nonergodic and Conformational Control of the Electron Capture Dissociation of Protein Cations. *Proceedings of the National Academy of Science*. **2004**, *101*, 14011-14016.
9. Tureček, F. N C α Bond Dissociation Energies and Kinetics in Amide and Peptide Radicals is the Dissociation a Non-Ergodic Process? *Journal of the American Chemical Society*. **2003**, *125*, 5954-5963.
10. Zubarev, R. A.; Kruger, N. A.; Fridriksson, E. K.; Lewis, M. A.; Horn, D. M.; Carpenter, B. K.; McLafferty, F. W. Electron Capture Dissociation of Gaseous Multiply-Charged Proteins is Favored at Disulfide Bonds and Other Sites of High Hydrogen Atom Affinity. *Journal of the American Chemical Society*. **1999**, *121*, 2857-2862.
11. Zubarev, R. A. Reactions of Polypeptide Ions with Electrons in the Gas Phase. *Mass Spectrometry Reviews*. **2003**, *22*, 57-77.

12. Zubarev, R. A.; Haselmann, K. F.; Budnik, B.; Kjeldsen, F.; Jensen, F. Towards an Understanding of the Mechanism of Electron-Capture Dissociation: A Historical Perspective and Modern Ideas. *European Journal of Mass Spectrometry*. **2002**, *8*, 337-349.
13. Syrstad, E. A.; Turecek, F. Toward a General Mechanism of Electron Capture Dissociation. *Journal of the American Society for Mass Spectrometry*. **2005**, *16*, 208-224.
14. Sobczyk, M.; Anusiewicz, W.; Berdys-Kochanska, J.; Sawicka, A.; Skurski, P.; Simons, J. Coulomb-Assisted Dissociative Electron Attachment: Application to a Model Peptide. *Journal of Physical Chemistry A*. **2005**, *109*, 250-258.
15. Anusiewicz, I.; Berdys-Kochanska, J.; Simons, J. Electron Attachment Step in Electron Capture Dissociation (ECD) and Electron Transfer Dissociation (ETD). *The Journal of Physical Chemistry A*. **2005**, *109*, 5801-5813.
16. Scalf, M.; Westphall, M. S.; Krause, J.; Kaufman, S. L.; Smith, L. M. Controlling Charge States of Large Ions. *Science*. **1999**, *283*, 194-197.
17. Wells, J. M.; Chrisman, P. A.; McLuckey, S. A. "Dueling" ESI: Instrumentation to Study Ion/Ion Reactions of Electrospray-Generated Cations and Anions. *Journal of the American Society for Mass Spectrometry*. **2002**, *13*, 614-622.
18. Loo, R. R. O.; Udseth, H. R.; Smith, R. D. A New Approach for the Study of Gas-Phase Ion-Ion Reactions using Electrospray Ionization. *Journal of the American Society for Mass Spectrometry*. **1992**, *3*, 695-705.
19. Loo, R. R. O.; Udseth, H. R.; Smith, R. D. Evidence of Charge Inversion in the Reaction of Singly Charged Anions with Multiply Charged Macroions. *Journal of Physical Chemistry*. **1991**, *95*, 6412-6415.
20. Stephenson, J. L.; McLuckey, S. A. Ion/Ion Reactions in the Gas Phase: Proton Transfer Reactions Involving Multiply-Charged Proteins. *Journal of the American Chemical Society*. **1996**, *118*, 7390-7397.
21. Wells, J. M.; Chrisman, P. A.; McLuckey, S. A. Formation of Protein-Protein Complexes in Vacuo. *Journal of the American Chemical Society*. **2001**, *123*, 12428-12429.
22. He, M.; McLuckey, S. A. Two Ion/Ion Charge Inversion Steps to Form a Doubly Protonated Peptide from a Singly Protonated Peptide in the Gas Phase. *Journal of the American Chemical Society*. **2003**, *125*, 7756-7757.

23. He, M.; Emory, J. F.; McLuckey, S. A. Reagent Anions for Charge Inversion of Polypeptide/Protein Cations in the Gas Phase. *Analytical Chemistry*. **2005**, *77*, 3173-3182.
24. Zener, C. Non-Adiabatic Crossing of Energy Levels. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*. **1932**, 696-702.
25. Landau, L. D. Zue Theorie Der Energieubertragung. *II. Phys. Z.* **1932**, *2*, 46.
26. McLuckey, S. A. The Emerging Role of Ion/Ion Reactions in Biological Mass Spectrometry: Considerations for Reagent Ion Selection. *European Journal of Mass Spectrometry*. **2009**, *16*, 429-439.
27. Gunawardena, H. P.; He, M.; Chrisman, P. A.; Pitteri, S. J.; Hogan, J. M.; Hodges, B. D. M.; McLuckey, S. A. Electron Transfer Versus Proton Transfer in Gas-Phase Ion/Ion Reactions of Polyprotonated Peptides. *Journal of the American Chemical Society*. **2005**, *127*, 12627-12639.
28. Baba, T.; Hashimoto, Y.; Hasegawa, H.; Hirabayashi, A.; Waki, I. Electron Capture Dissociation in a Radio Frequency Ion Trap. *Analytical Chemistry*. **2004**, *76*, 4263-4266.
29. Silivra, O. A.; Kjeldsen, F.; Ivonin, I. A.; Zubarev, A. R. Electron Capture Dissociation of Polypeptides in a Three-Dimensional Quadrupole Ion Trap: Implementation and First Results. *Journal of the American Society for Mass Spectrometry*. **2005**, *16*, 22-27.
30. Satake, H.; Hasegawa, H.; Hirabayashi, A.; Hashimoto, Y.; Baba, T. Fast Multiple Electron Capture Dissociation in a Linear Radio Frequency Quadrupole Ion Trap. *Analytical Chemistry*. **2007**, *79*, 8755-8761.
31. Ding, L.; Brancia, F. L. Electron Capture Dissociation in a Digital Ion Trap Mass Spectrometer. *Analytical Chemistry*. **2006**, *78*, 1995-2000.
32. Reid, G.; Simpson, R.; O'Hair, R. J. A Mass Spectrometric and Ab Initio Study of the Pathways for Dehydration of Simple Glycine and Cysteine-Containing Peptide $[M+H]^+$ Ions. *Journal of the American Society for Mass Spectrometry*. **1998**, *9*, 945-956.
33. Hager, J. W. A New Linear Ion Trap Mass Spectrometer. *Rapid Communications in Mass Spectrometry*. **2002**, *16*, 512-526.

34. Liang, X.; Xia, Y.; McLuckey, S. A. Alternately Pulsed Nanoelectrospray Ionization/Atmospheric Pressure Chemical Ionization for Ion/Ion Reactions in an Electrodynamic Ion Trap. *Analytical Chemistry*. **2006**, *78*, 3208-3212.
35. Xia, Y.; Liang, X.; McLuckey, S. A. Pulsed Dual Electrospray Ionization for Ion/Ion Reactions. *Journal of the American Society for Mass Spectrometry*. **2005**, *16*, 1750-1756.
36. Xia, Y.; Wu, J.; McLuckey, S. A.; Londry, F. A.; Hager, J. W. Mutual Storage Mode Ion/Ion Reactions in a Hybrid Linear Ion Trap. *Journal of the American Society for Mass Spectrometry*. **2005**, *16*, 71-81.
37. Londry, F. A.; Hager, J. W. Mass Selective Axial Ion Ejection from a Linear Quadrupole Ion Trap. *Journal of the American Society for Mass Spectrometry*. **2003**, *14*, 1130-1147.
38. Mahan, B. H. Recombination of Gaseous Ions. *Advances in Chemical Physics*. **1973**, *23*.
39. Olson, R. E. Absorbing-Sphere Model for Calculating Ion—Ion Recombination Total Cross Sections. *Journal of Chemical Physics*. **1972**, *56*, 2979-2984.
40. Olson, R.; Smith, F.; Bauer, E. Estimation of the Coupling Matrix Elements for One-Electron Transfer Systems. *Applied Optics*. **1971**, *10*, 1848-1855.
41. Budnik, B. A.; Tsybin, Y. O.; Håkansson, P.; Zubarev, R. A. Ionization Energies of Multiply Protonated Polypeptides obtained by Tandem Ionization in Fourier Transform Mass Spectrometers. *Journal of Mass Spectrometry*. **2002**, *37*, 1141-1144.
42. Bates, D.; Morgan, W. L. New Recombination Mechanism: Tidal Termolecular Ionic Recombination. *Physical Review Letters*. **1990**, *64*, 2258.
43. Morgan, W.; Bates, D. R. Tidal Termolecular Ionic Recombination. *Journal of Physics B: Atomic, Molecular and Optical Physics*. **1992**, *25*, 5421.
44. Tureček, F.; Jones, J.; Towle, T.; Panja, S.; Nielsen, S.; Hvelplund, P.; Paizs, B. Hidden Histidine Radical Rearrangements upon Electron Transfer to Gas-Phase Peptide Ions. Experimental Evidence and Theoretical Analysis. *Journal of the American Chemical Society*. **2008**, *130*, 14584-14596.
45. Chen, X.; Hao, C. Where does the Electron Go? Electron Distribution and Reactivity of Peptide Cation Radicals Formed by Electron Transfer in the Gas Phase. *Journal of the American Chemical Society*. **2008**, *130*, 8818-8833.

46. Chung, T. W.; Tureček, F. Amplified Histidine Effect in Electron-Transfer Dissociation of Histidine-Rich Peptides from Histatin 5. *International Journal of Mass Spectrometry*. **2011**, *306*, 99-107.
47. Turecek, F.; Chung, T. W.; Moss, C. L.; Wyer, J. A.; Ehlerding, A.; Holm, A. I.; Zettergren, H.; Nielsen, S. B.; Hvelplund, P.; Chamot-Rooke, J. The Histidine Effect. Electron Transfer and Capture Cause Different Dissociations and Rearrangements of Histidine Peptide Cation-Radicals. *Journal of the American Chemical Society*. **2010**, *132*, 10728-10740.
48. Chen, X.; Turecek, F. The Arginine Anomaly: Arginine Radicals are Poor Hydrogen Atom Donors in Electron Transfer Induced Dissociations. *Journal of the American Chemical Society*. **2006**, *128*, 12520-12530.
49. Tsybin, Y. O.; Haselmann, K. F.; Emmett, M. R.; Hendrickson, C. L.; Marshall, A. G. Charge Location Directs Electron Capture Dissociation of Peptide Dications. *Journal of the American Society for Mass Spectrometry*. **2006**, *17*, 1704-1711.
50. Simons, J.; Ledvina, A. R. Spatial Extent of Fragment-Ion Abundances in Electron Transfer Dissociation and Electron Capture Dissociation Mass Spectrometry of Peptides. *International Journal of Mass Spectrometry*. **2012**, *330–332*, 85-94.
51. Tureček, F.; Moss, C. L.; Chung, T. W. Correlating ETD Fragment Ion Intensities with Peptide Ion Conformational and Electronic Structure. *International Journal of Mass Spectrometry*. **2012**, *330–332*, 207-219.
52. Vorobyev, A.; Hamidane, H. B.; Tsybin, Y. O. Electron Capture Dissociation Product Ion Abundances at the X Amino Acid in RAAAA-X-AAAK Peptides Correlate with Amino Acid Polarity and Radical Stability. *Journal of the American Society for Mass Spectrometry*. **2009**, *20*, 2273-2283.
53. Tureček, F.; Syrstad, E. A. Mechanism and Energetics of Intramolecular Hydrogen Transfer in Amide and Peptide Radicals and Cation-Radicals. *Journal of the American Chemical Society*. **2003**, *125*, 3353-3369.
54. Syrstad, E. A.; Stephens, D. D.; Tureček, F. Hydrogen Atom Adducts to the Amide Bond. Generation and Energetics of Amide Radicals in the Gas Phase. *The Journal of Physical Chemistry A*. **2003**, *107*, 115-126.
55. Moss, C. L.; Liang, W.; Li, X.; Tureček, F. The Early Life of a Peptide Cation-Radical. Ground and Excited-State Trajectories of Electron-Based Peptide Dissociations during the First 330 Femtoseconds. *Journal of the American Society for Mass Spectrometry*. **2012**, *23*, 446-459.

56. Simons, J. Analytical Model for Rates of Electron Attachment and Intramolecular Electron Transfer in Electron Transfer Dissociation Mass Spectrometry. *Journal of the American Chemical Society*. **2010**, *132*, 7074-7085.
57. Verboom, G.; Meisels, G. Angular Momentum Distributions in Collision Processes and the Location of the Transition State in Ion-molecule Reactions. *Journal of Chemical Physics*. **1978**, *68*, 2714-2717.
58. Chesnavich, W.; Bowers, M. Statistical Phase Space Theory of Polyatomic Systems. Application to the Cross Section and Product Kinetic Energy Distribution of the Reaction $\text{C}_2\text{H}_4^+ + \text{C}_2\text{H}_4 \rightarrow \text{C}_3\text{H}_5^+ + \text{CH}_3$. *Journal of the American Chemical Society*. **1976**, *98*, 8301-8309.
59. Chesnavich, W. J.; Bowers, M. T. Statistical Phase Space Theory of Polyatomic Systems: Rigorous Energy and Angular Momentum Conservation in Reactions Involving Symmetric Polyatomic Species. *Journal of Chemical Physics*. **2008**, *66*, 2306-2315.
60. Meisels, G.; Verboom, G.; Weiss, M.; Hsieh, T. Angular Momentum in Ion-Molecule Reactions. *Journal of the American Chemical Society*. **1979**, *101*, 7189-7195.

Chapter 3

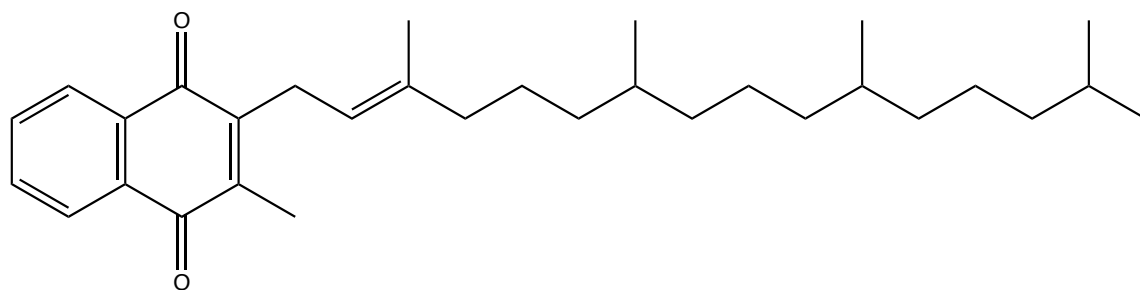
Separating Vitamin K₂ Protomers using Differential Ion Mobility Spectrometry

3.1 Introduction

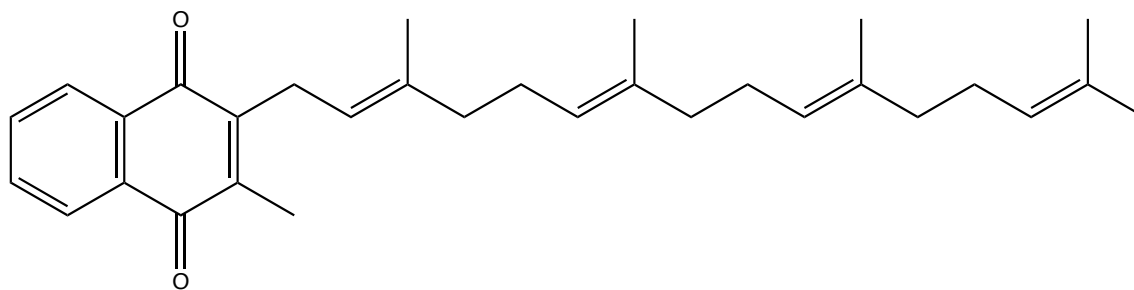
Vitamin K is a group of fat-soluble vitamins with similar structures that are essential to the formation of certain proteins necessary for blood coagulation and for proteins that the human body uses to bind calcium in bones and other tissues. There are two natural occurring vitamin K vitamers, vitamin K₁ and vitamin K₂. Vitamin K₁, typically found in plants in which it is synthesized, is most prevalent in green leafy vegetables and plays an integral role in the photosynthetic process. While vitamin K₁ can be found in animal species it is usually converted to vitamin K₂, which is the main storage form. Vitamin K₂ has several subtypes that differ in the number of isoprenoid groups in the side chain. These subtypes are referred to as menaquinones and are characterized by the nomenclature MK-n, where “M” stands for menaquinone, “K” for vitamin K, and “n” for the number of isoprenoid groups in the side chain. This study focuses on the MK-4 subtype of vitamin K₂, which is the main form found in humans. The work in this chapter is focused on separation of vitamin K₂ protomers (isomers that differ in protonation site location).

Gas phase separations have become more popular in recent times as a way to separate ions prior to mass analysis. These separations have traditionally been done using drift-tube ion mobility spectrometry (DTIMS) which separates ions in time.

DTIMS has been used in a variety of applications including separation of different isomers and conformations [1-5]. A drawback to using DTIMS is that it can only be coupled to time-of-flight mass spectrometers (TOFMS). Using an applied electric field DTIMS can separate ions moving through a buffer gas. Size, shape, charge state, and ion-molecule interactions govern the mobility of an ion traveling through a collision gas in a given electric field. The velocity of an ion is directly proportional to the reduced electric field, E/N , where E is the electric field and N is the gas number density. At low E/N , ion mobility is independent of E/N , but as E/N is increased, ion mobility becomes dependent on the reduced electric field [6-9]. Differential ion mobility spectrometry (DIMS) uses this mobility dependence on electric field to separate ions.



Vitamin K₁



Vitamin K₂

Figure 3.1. Structures of vitamin K₁ and vitamin K₂

DIMS has become popular due to its orthogonality with mass spectrometry (MS) and the fact that it can also be coupled with any mass analyzer type. DIMS separates ions based on their difference in mobility as electric field strength is alternated between high and low electric field strengths. This separation can be thought of as a separation of ions based on a "shape-to-charge" ratio. DIMS has been shown useful at separating a variety of molecules, including isobars, isomers, and conformers [10-16]. In this work, DIMS data and density functional theory (DFT) calculations are utilized to propose structures for protonated and sodiated vitamin K₂.

3.2 Experimental

3.2.1 Samples

Vitamin K₂ was purchased from Sigma Aldrich (St. Louis, MO, USA). Methanol, acetonitrile, water, ammonium formate and glacial acetic acid were obtained from Fisher Scientific (Waltham, MA, USA). Vitamin K₂ solutions were prepared to a concentration of 5 μ M in various solvent and salt conditions with 80/20 acetonitrile/water with 2.5 mM ammonium formate being the favored solution for most experiments.

3.2.2 Differential Ion Mobility Spectrometry

A planar DIMS assembly composed of two parallel stainless steel electrodes was used for these experiments (Figure 3.2). Experiments were completed on both the Generation 3 (G3) and Generation 4 (G4) DIMS devices. A Bruker Esquire 3000 ion trap MS was used for experiments using the G3 device while a Bruker HCT ion trap MS was used for experiments using the G4 device. The G3 device used has a 0.5 mm gap with electrodes that are 6 mm in width and 25 mm in length. The G4 device has a 0.3 mm gap with electrodes that are 4 mm in width and 10 mm in length. The instruments desolvation gas (Nitrogen) is

rerouted through the DIMS assembly and is used as both the carrier gas to pull ions through the DIMS device and the desolvation gas for ESI.

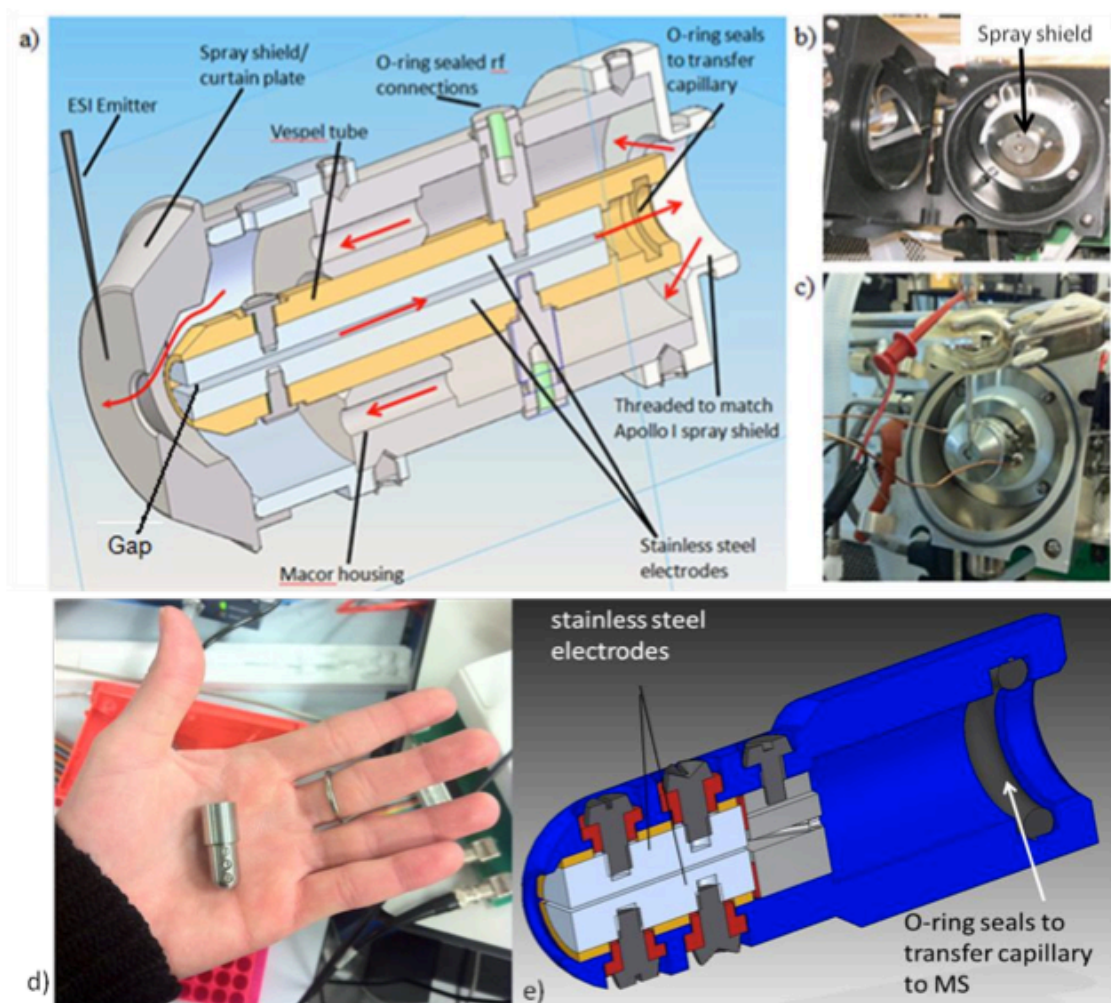


Figure 3.2. DIMS assemblies a) AutoCAD drawing of G3 b) Esquire 3000 source with spray shield c) picture of source with removed spray shield and DIMS assembly in its place d) G4 DIMS device without housing and e) AutoCAD drawing of G4 design.

A custom-built power supply was used for the G3 device. While a rectangular waveform would ideally be used to power a DIMS device the high power requirements make this unfeasible. In lieu of this, a bisinusoidal waveform that approximates a rectangular waveform is used [17]. To obtain this bisinusoidal waveform a sinusoidal wave is applied to one electrode with a given frequency and amplitude and a phase-shifted waveform at twice

the voltage and approximately half the amplitude is applied to the other electrode. This produces an electric field equivalent to the sum of the two individual waveforms (Figure 3.3) [18]. The dispersion voltage is defined as the maximum voltage (V_{0-p}) of the bisinusoidal waveform. The dispersion field (E_D) is defined as the dispersion voltage divided by the gap size of the DIMS device. The bisinusoidal waveform was tuned to a frequency of 1.7 MHz for the G3 device and 2 MHz for the G4 device.

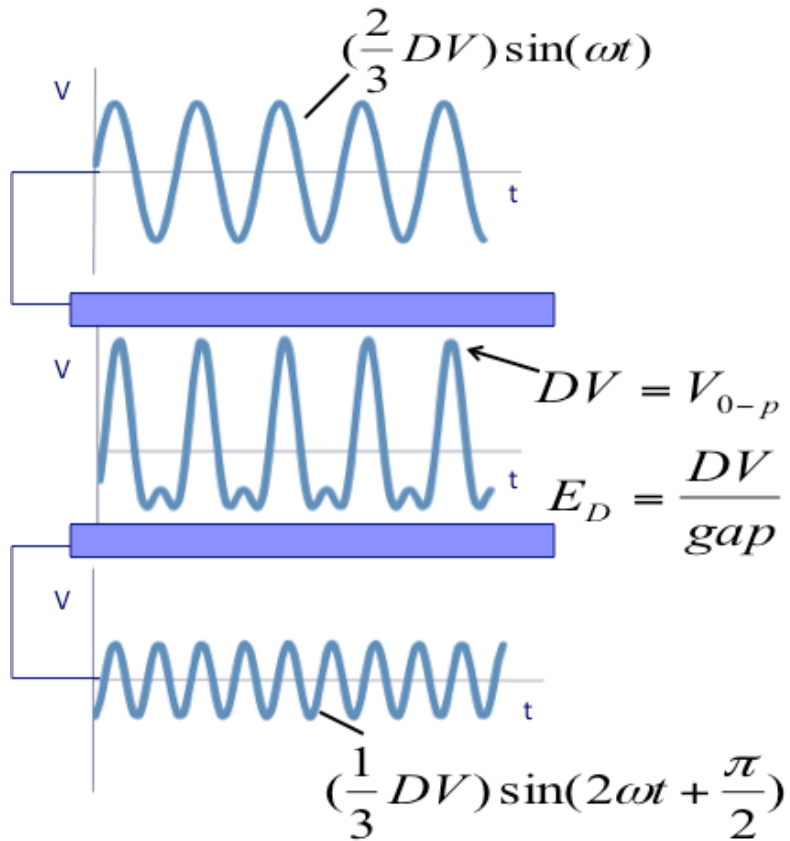


Figure 3.3. Summation of two sinusoidal waveforms to form a bisinusoidal waveform to approximate a rectangular wave.

A LabVIEW program linked to the instrument control software is used to set a range of compensation voltages or to define a static value. The DIMS device can be operated in

either filter or scanning mode. In scanning mode the compensation field (E_C) is stepped over a specific range as input into the LabVIEW program. By selecting a static voltage in the LabVIEW program the device can be operated in “filter mode”.

A difference of 4.25 kV is needed between the emitter and the spray shield for ESI. With the G3 device the voltage must be applied to the emitter as the G3 power supply is not capable of superimposing the bisinusoidal waveform on a high voltage. A second custom power supply was designed for the G4 device so that the waveform could be centered around the ESI voltage and operated with the ESI emitter at ground potential. This second supply is based off of the same bisinusoidal waveform design as the G3 power supply with a few minor improvements.

3.2.3 Computational Methods

Initial structures were drawn using GaussView 5. Structures were protonated or a sodium adduct was attached at either of the oxygens on the naphthoquinone unit. Structures were fully optimized at the HF/3-21G, B3LYP/3-21G, B3LYP/6-31G levels, before being finalized at the B3LYP/6-31G(d,p) level. After optimization at the B3LYP/6-31G(d,p) level the resulting structure was submitted to frequency analysis at the same level of theory. The Gaussian 03 suite of programs was used for all ab initio and DFT calculations [19]. All calculations were done on the KillDevil supercomputer cluster at UNC-Chapel Hill.

3.3 Results and Discussion

3.3.1 Solution Conditions

Initial studies with vitamin K₂ evolved around determining which solution conditions were best to use for studying the molecule of interest via ESI. Solution conditions were investigated with different amounts of water versus organic (methanol or acetonitrile) and

using various amounts of glacial acetic acid or ammonium formate as the charge carrier (Table 3.1). Ammonium formate was determined to be the best charge carrier as its use resulted in a higher intensity of the protonated structures and limited the amount of vitamin K₂ that was sodiated.

Table 3.1. Ratio of protonated species to sodiated species using various spray solution conditions

Solvent System	Ratio H⁺/Na⁺	Standard Deviation
80/20/1 ACN/H ₂ O/AcOH	1.79	0.24
60/40/1 ACN/H ₂ O/AcOH	1.70	0.53
50/50/1 ACN/H ₂ O/AcOH	2.79	0.74
40/60/1 ACN/H ₂ O/AcOH	1.48	0.03
80/20/1 MeOH/H ₂ O/AcOH	1.25	0.11
60/40/1 ACN/H ₂ O/AcOH	0.81	0.03
80/20 ACN/H ₂ O w/ 2.5 mM NH ₄ OOCH	4.97	1.36
60/40 ACN/H ₂ O w/ 2.5 mM NH ₄ OOCH	1.56	0.59
80/20 MeOH/H ₂ O w/ 2.5 mM NH ₄ OOCH	3.98	0.89
60/40 MeOH/H ₂ O w/ 2.5 mM NH ₄ OOCH	0.82	0.68
80/20 ACN/H ₂ O w/ 5.0 mM NH ₄ OOCH	3.53	0.65
80/20 ACN/H ₂ O w/ 7.5 mM NH ₄ OOCH	3.55	0.62
80/20 ACN/H ₂ O w/ 10 mM NH ₄ OOCH	1.30	0.60
80/20 MeOH/H ₂ O w/ 5.0 mM NH ₄ OOCH	2.63	0.56
80/20 MeOH/H ₂ O w/ 7.5 mM NH ₄ OOCH	1.63	0.40
80/20 MeOH/H ₂ O w/ 10 mM NH ₄ OOCH	1.24	0.21

3.3.2 Separation of Protonated and Sodiated Structures of Vitamin K₂

A DIMS-MS scan with the G3 DIMS device, where the CV is stepped after every 10 spectra, of vitamin K₂ (Figure 3.4) reveals two peaks for the protonated molecule (m/z 445) and one peak for the sodiated molecule (m/z 467). It is particularly interesting that the peak for the sodiated species appears at nearly the same compensation field as the peak for one of the protonated species. This would suggest that the structure of this protonated species and the sodiated species are similar. The DIMS-MS scan with the G4 device is similar, having two peaks for the protonated molecule and one peak for the sodiated molecule (Figure 3.5). The only difference being that the sodiated molecule appears at a compensation field between the two protonated peaks and thus no longer passes through the DIMS device at the same compensation field as one of the protonated species. DIMS-MS scans of vitamin K₂ with other metal adducts (Li⁺, K⁺, and Ag⁺) showed almost identical results as when sodium was used as the adduct.

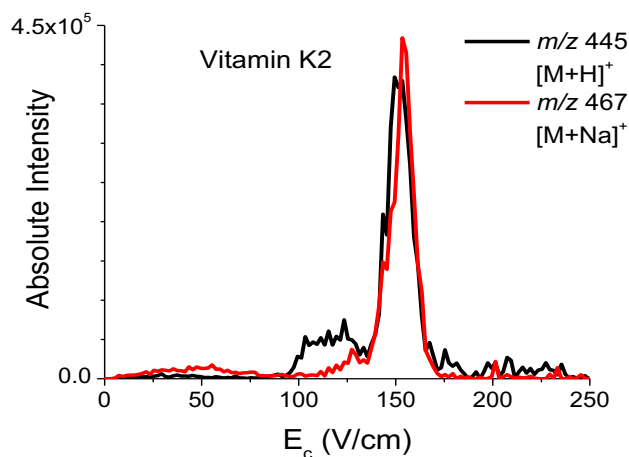


Figure 3.4. DIMS-MS scan with G3 device of 5 μ M vitamin K₂ at dispersion field strength of 30 kV/cm. Two peaks are observed for protonated vitamin K₂, while only one peak is observed for sodiated vitamin K₂

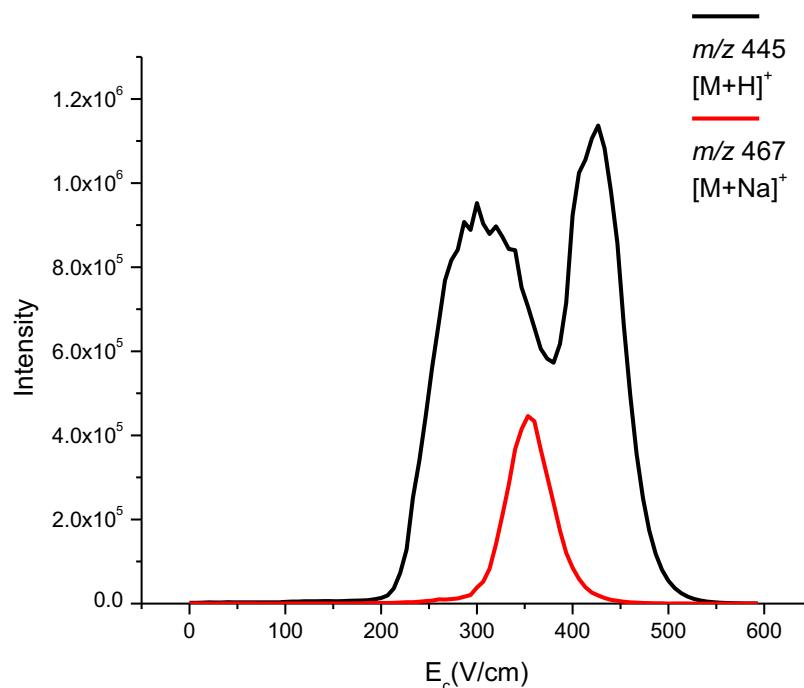


Figure 3.5. DIMS-MS scan with G4 device of 5 μ M vitamin K₂ at dispersion field strength of 36.7 kV/cm. Two peaks were observed for protonated vitamin K₂, while only one peak was observed for sodiated vitamin K₂.

With DIMS active and in filter mode (G3 device), the protonated molecule is activated by CID, resulting in one of the peaks having more extensive dissociation along the side chain than the other (Figure 3.6). Both protonated ions give rise to fragment ions that can be attributed to the naphthoquinone ring and the entire isoprenoid side chain. Additionally, both ions show loss of the very last isoprenoid group of the side chain, while the protonated ion that passes through DIMS at a higher compensation field also shows fragments that arise from cleavages that occur at the two other isoprenoid groups along the vitamin K₂ side chain. These data suggest that there are at least two distinct structures of protonated vitamin K₂ that are separable with DIMS. This phenomenon was confirmed using the G4 DIMS device, which showed similar fragmentation patterns for the protonated

vitamin K₂ peaks that pass through the DIMS device at different compensation fields.

Activation of sodiated vitamin K₂ was attempted using both the G3 and G4 devices however the only response observed was a loss in signal intensity of [vitamin K₂ + Na]⁺. This is possibly due to the sodium ion being bound very loosely to vitamin K₂ and the sodium ion being lost before any fragmentation occurs on the vitamin K₂ side chain. High amplitude short time excitation (HASTE) CID was also attempted on sodiated vitamin K₂, but the loss of the sodium ion was still the only fragmentation product that was observed [20].

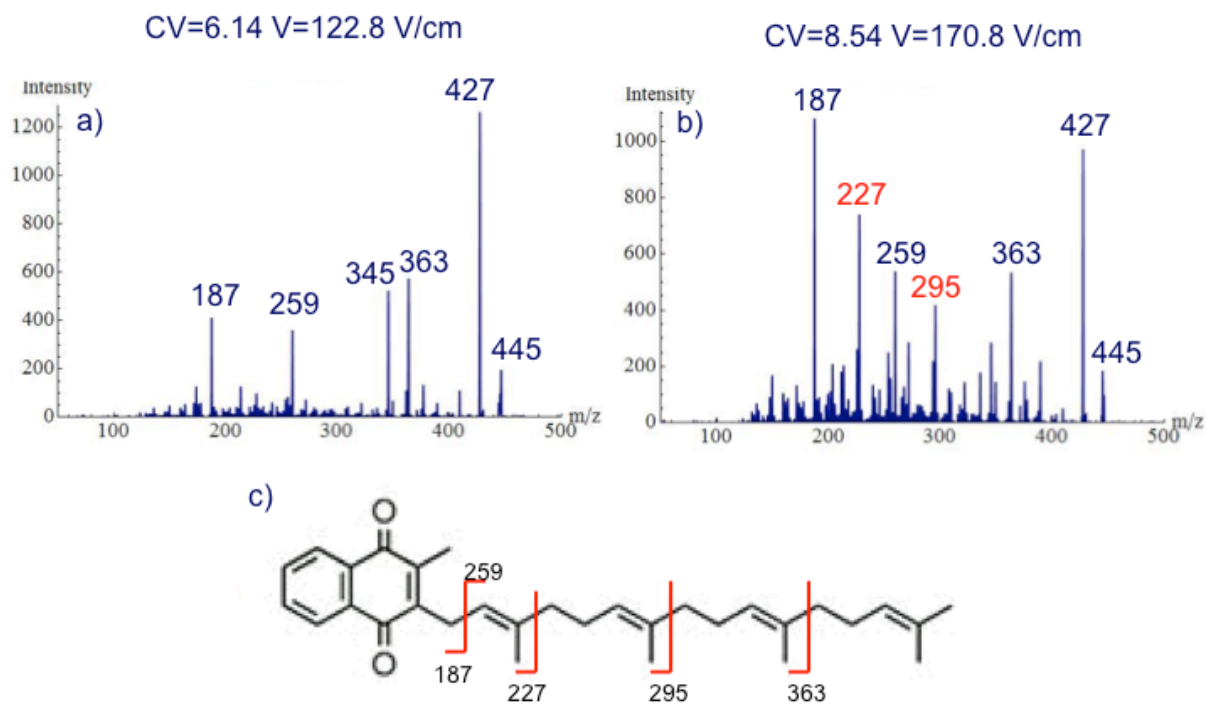


Figure 3.6. MS/MS of two vitamin K₂ protonated peaks. (a) MS/MS of protonated vitamin K₂ that passes at a lower CV shows less fragmentation along the side-chain (b) MS/MS of protonated vitamin K₂ that passes at a higher CV shows more fragmentation along the side-chain which corresponds to the lowest energy structure that has the extended side chain. (c) Fragmentation scheme of vitamin K₂

Theoretical calculations at the B3LYP/6-31G(d,p) level of theory resulted in two different structures of vitamin K₂ where the structure is protonated at either oxygen (Figure 3.7). The lowest energy structure is protonated on the oxygen closest to the methyl group.

The second structure, protonated on the oxygen closest to the isoprenoid side chain, is 8 kJ/mol higher in energy. The similarity in energy between the two protonated structures allows for the fact that two peaks were observed in the DIMS-MS scan. The lower energy structure has a more extended side chain; this suggests more extensive dissociation along the side chain, which was observed for one of the structures.

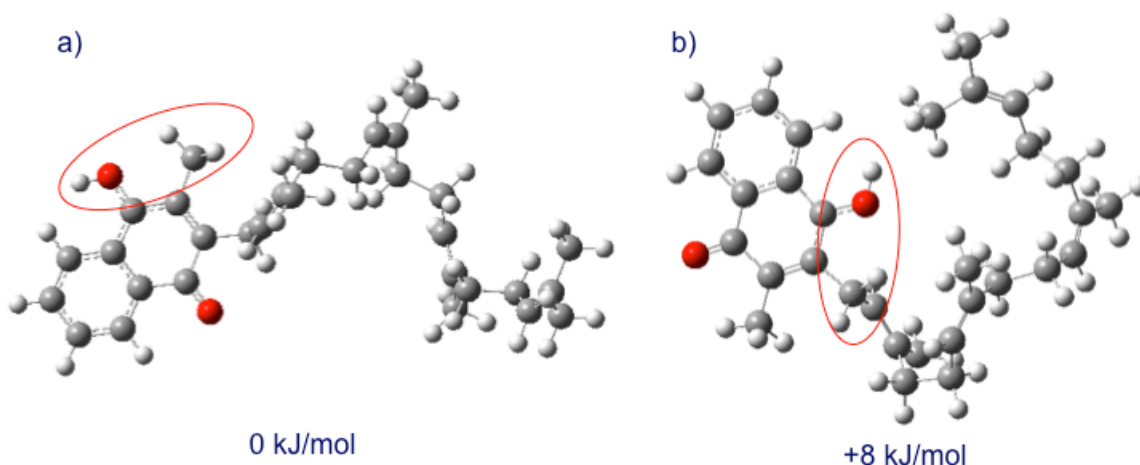


Figure 3.7. Proposed vitamin K₂ structures from DFT calculations fully optimized at B3LYP/6-31G(d,p) level of theory. (a) vitamin K₂ protonated near the methyl group (b) vitamin K₂ protonated near side-chain

DFT calculations for sodiated vitamin K₂ also resulted in two different structures depending upon which oxygen the sodium was initially bound (Figure 3.8). The lowest energy sodiated structure was determined to have the sodium attached to the oxygen closest to the methyl group and the second structure with a sodium initially attached to the oxygen closest to the side chain is 124 kJ/mol higher in energy. The large difference in energy between these two sodiated structures lends support to the experimental results because only one peak is observed for the sodiated molecule (m/z 467) in the DIMS-MS scan. Of note is that one of the oxygens in the naphthoquinone unit is now shown to have adapted a semiquinone structure formed via abstraction of a hydrogen from the isoprenoid side chain.

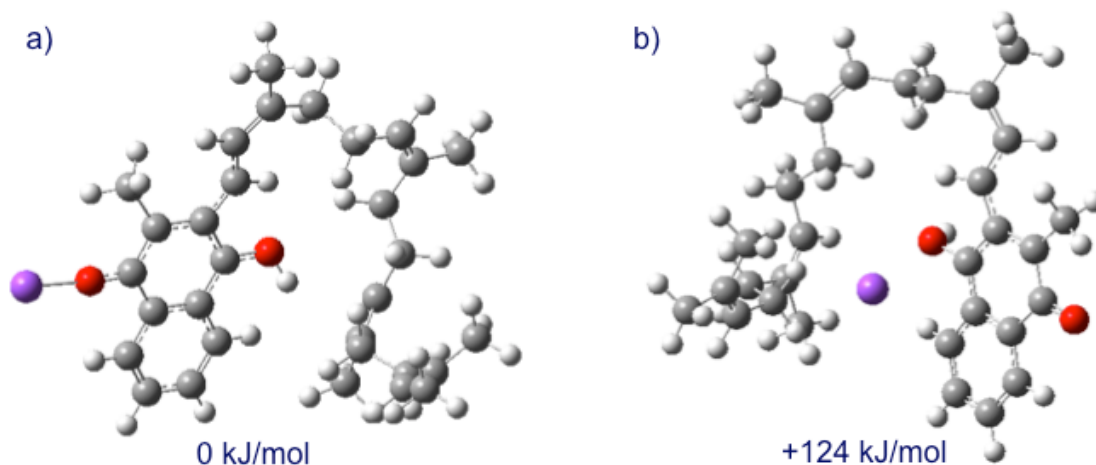


Figure 3.8. Proposed vitamin K₂ structures from DFT calculations fully optimized at B3LYP/6-31G(d,p) level of theory. (a) vitamin K₂ sodiated near the methyl group (b) vitamin K₂ sodiated near side-chain

3.3.3 Decomposition of Vitamin K₂

During this investigation of vitamin K₂ an unexpected sample decomposition occurred during storage at -18 °C, both as a pure solid and dissolved in ethanol. It was determined that solid vitamin K₂ was best kept in a freezer held at -80 °C where the sample remained in its original state for at least six months. These decompositions were so rapid that spray solutions of 5 µM vitamin K₂ in 80/20 acetonitrile/water with 2.5 mM ammonium formate showed evidence of decomposition in a matter of hours when placed in a glass syringe which is used to infuse the vitamin K₂ solution. This was due to photoactivation, which has been observed for vitamin K₁ previously and was circumvented by wrapping the glass syringe in aluminum foil, ultimately reducing the rate of decomposition [21-24].

A study was carried out to investigate the decomposition of vitamin K₂ in spray solutions. A set of spray solutions were made and wrapped in foil and placed in a freezer. Every day for one week a new sample was removed from the freezer and a DIMS-MS scan was collected on the G4 device at dispersion fields of 26.7, 30, 33.3, and 36.7 V/cm. Figure

3.9 shows two DIMS-MS scans, the first collected on the day the samples were initially made and the second collected five days later. The scan collected at the later date showed a decrease in intensity for the protonated species (m/z 445) while peaks at mass-to-charge ratios of 444 and 443 increased in intensity. It is of note that the decomposed species only appear at the higher compensation field while both protonated peaks in the DIMS-MS scan decrease in intensity.

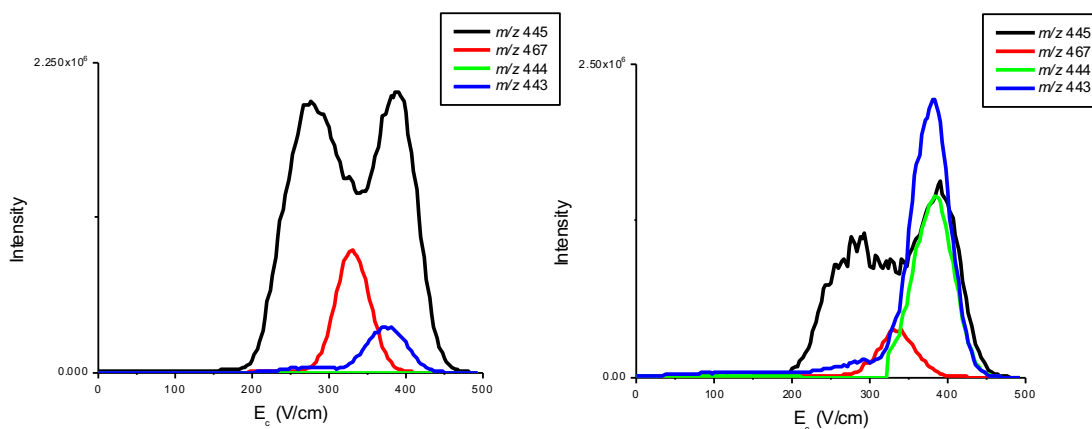


Figure 3.9. DIMS-MS scan with G4 device of 5 μ M Vitamin K₂ taken 5 days apart to show the decomposition of vitamin K₂ at dispersion field strength of 36.7 kV/cm. Fresh spray sample (left) and same sample used 5 days later (right).

To identify these decomposition products a high mass accuracy MS scan was obtained on a Thermo Scientific LTQ-FT Ultra (Figure 3.10). Analysis showed that mass-to-charge ratio 445 corresponds to a molecular formula of C₃₁H₄₁O₂ the formula for protonated vitamin K₂. The molecular formulas for mass-to-charge ratios 444 and 443 correspond to loss of either one or two hydrogen atoms respectively. The ion at mass-to-charge ratio 444 results from the loss of a hydrogen atom from vitamin K₂ and therefore is a radical species. The presence of a radical species in a mass spectrum using electrospray ionization is rare. In this case the radical species is observed due to the loss of a hydrogen atom in solution followed by protonation in the electrospray ionization process.

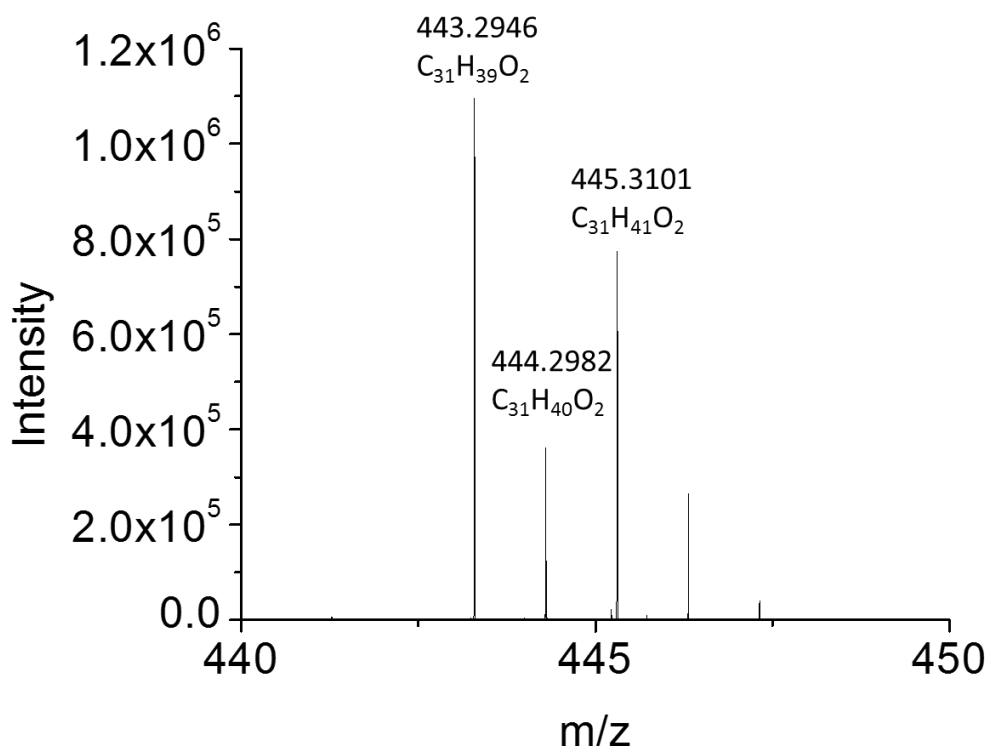


Figure 3.10. High mass accuracy MS scan of decomposed vitamin K₂ sample on LTQ-FT Ultra

To further validate this the nitrogen supply for the dry gas and nebulizing gas on the Bruker HCT was replaced with air to determine if the mass-to-charge ratio 444 species is indeed a radical ion as the high mass accuracy data suggest. The MS scan of this experiment is shown in Figure 3.11. The MS scan when nitrogen was used as the supply gas showed no mass-to-charge ratio 476 which corresponds to $[\text{vitamin K}_2 - \text{H}^\bullet + \text{H}^+ + \text{O}_2]^+$, while this species is observed in the MS scan when air was used as the supply gas. CID of the $[\text{vitamin K}_2 - \text{H}^\bullet + \text{H}^+ + \text{O}_2]^+$ peak gives rise to water loss and water loss plus hydroxyl radical peaks. This proved the addition of oxygen to vitamin K₂, which validates that the ion with a mass-to-charge ratio of 444 is a radical species.

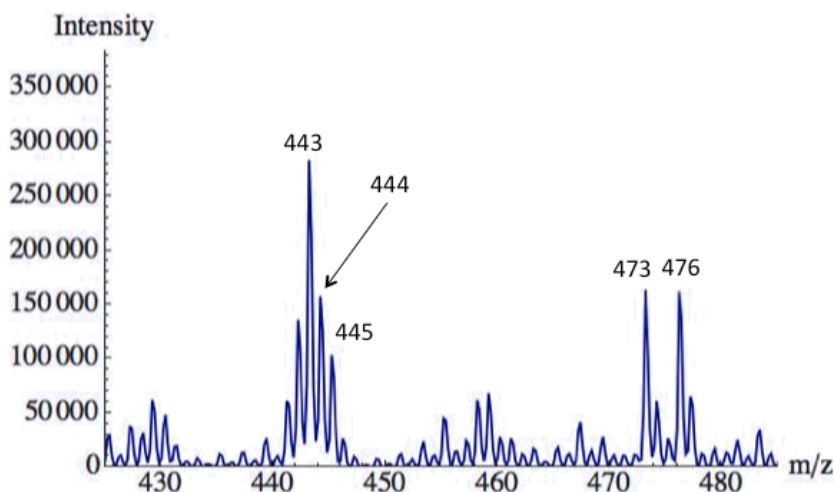


Figure 3.11. MS scan of decomposed vitamin K₂ sample with Bruker HCT's supply gas changed from nitrogen to air.

Both of the ions arising from decomposition of vitamin K₂ were activated using CID (Figure 3.12). Figure 3.12a shows the CID spectrum of the [vitamin K₂ - H[•] + H⁺]⁺ peak with only fragmentation around the double bond in the first isoprenoid group of the side chain as well as a peak that corresponded to water loss (m/z 426) observed. The decomposition product of vitamin K₂ that has lost two hydrogen atoms behaves almost identical under CID conditions to the non-decomposed protonated vitamin K₂ (Figure 3.12b). The only exception being that the water loss peak appears at mass-to-charge ratio 425 (as opposed to 427). Additionally, the peak that corresponds to loss of the entire side chain with the charge remaining on the side chain (mass-to-charge 259) for the non-decomposed protonated vitamin K₂ now appears at mass-to-charge ratio 257.

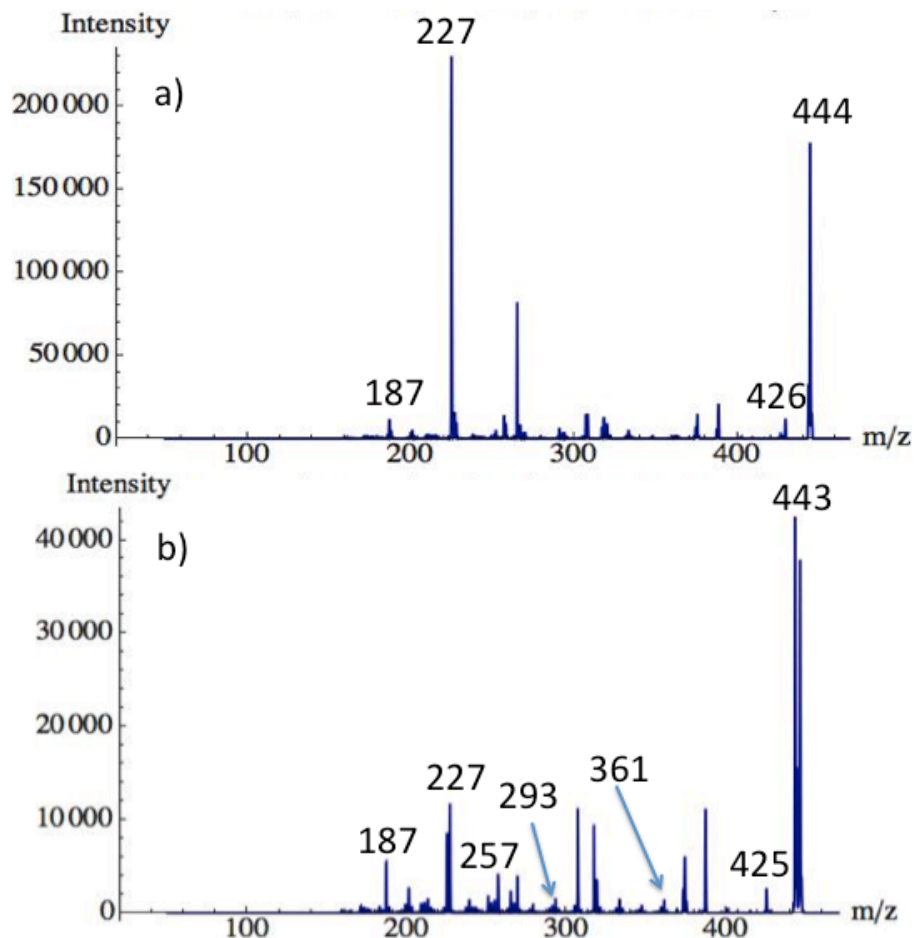


Figure 3.12. CID of products from the decomposition of vitamin K₂ taken at same activation voltage. (a) [vitamin K₂ - H• + H⁺]⁺ and (b) [vitamin K₂ - 2H• + H⁺]⁺

3.4 Conclusions

Spray solution composition was investigated to determine which solution condition was best for maximizing the amount of protonated vitamin K₂ observed in a DIMS-MS scan. It was determined that a solution composition of 80/20 acetonitrile/water with 2.5 mM ammonium formate best achieved this goal. This work shows that two structures of vitamin K₂ that differ by protonation site location are separable using DIMS. Proposed protonated structures, protonated at either oxygen are close enough in energy to both be accessible. The

large difference in energy between the two proposed sodiated structures agrees with the experimental data in that only one sodiated peak was observed in the DIMS-MS scan.

The decomposition of vitamin K₂ was found to be a point of interest as over a period of five days the appearance of the DIMS-MS changed drastically. It has been shown that the protonated ion of vitamin K₂ is no longer the most abundant species after five days with the loss of both a single and two hydrogen atoms from protonated vitamin K₂ becoming the more favored species in the DIMS-MS scan. The species resulting from loss of one hydrogen atom was of particular interest. This is due to the fact that when this species is protonated during the ESI process it shows up as a radical ion in the mass spectrum, which is a very rare occurrence in ESI mass spectrometry experiments.

3.5 REFERENCES

1. Clemmer, D. E.; Jarrold, M. F. Ion Mobility Measurements and their Applications to Clusters and Biomolecules. *Journal of Mass Spectrometry*. **1997**, *32*, 577-592.
2. Shelimov, K. B.; Clemmer, D. E.; Hudgins, R. R.; Jarrold, M. F. Protein Structure *in Vacuo*: Gas-Phase Conformations of BPTI and Cytochrome C. *Journal of the American Chemical Society*. **1997**, *119*, 2240-2248.
3. Valentine, S. J.; Anderson, J. G.; Ellington, A. D.; Clemmer, D. E. Disulfide-Intact and Reduced-Lysozyme in the Gas Phase: Conformations and Pathways of Folding and Unfolding. *Journal of Physical Chemistry B*. **1997**, *101*, 3891-3900.
4. Wu, C.; Siems, W. F.; Klasmeier, J.; Hill, H. H. Separation of Isomeric Peptides using Electrospray Ionization/High-Resolution Ion Mobility Spectrometry. *Analytical Chemistry*. **2000**, *72*, 391-395.
5. Ahonen, L.; Fasciotti, M.; Gennäs, G. B. a.; Kotiaho, T.; Daroda, R. J.; Eberlin, M.; Kostianen, R. Separation of Steroid Isomers by Ion Mobility Mass Spectrometry. *Journal of Chromatography A*. **2013**, *1310*, 133-137.
6. Eiceman, G. A.; Karpas, Z. *Ion Mobility Spectrometry*, Second ed.; CRC Press, **2005**; p. 337.
7. Mason, E. A.; McDaniel, E. W. *Transport Properties of Ions in Gases*; Wiley, New York, **1988**.
8. Purves, R. W.; Guevremont, R.; Day, S.; Pipich, C. W.; Matyjaszczyk, M. S. Mass Spectrometric Characterization of a High-Field Asymmetric Waveform Ion Mobility Spectrometer. *Review of Scientific Instruments*. **1998**, *69*, 4094-4105.
9. Shvartsburg, A. A. *Differential Ion Mobility Spectrometry : Nonlinear Ion Transport and Fundamentals of FAIMS*; CRC Press, Boca Raton, **2009**.
10. Purves, R. W.; Barnett, D. A.; Ells, B.; Guevremont, R. Elongated Conformers of Charge States +11 to +15 of Bovine Ubiquitin Studied using ESI-FAIMS-MS. *Journal of the American Society for Mass Spectrometry*. **2001**, *12*, 894-901.
11. Purves, R.; Guevremont, R. Electrospray Ionization High-Field Asymmetric Waveform Ion Mobility Spectrometry-Mass Spectrometry. *Analytical Chemistry*. **1999**, *71*, 2346-2357.
12. Barnett, D. A.; Ells, B.; Guevremont, R.; Purves, R. W. Separation of Leucine and Isoleucine by Electrospray Ionization-High Field Asymmetric Waveform Ion Mobility Spectrometry-Mass Spectrometry. *Journal of the American Society for Mass Spectrometry*. **1999**, *10*, 1279-1284.

13. Barnett, D. A.; Purves, R. W.; Ells, B.; Guevremont, R. Separation of O-, M- and P-Phthalic Acids by High-Field Asymmetric Waveform Ion Mobility Spectrometry (FAIMS) using Mixed Carrier Gases. *Journal of Mass Spectrometry*. **2000**, *35*, 976-980.
14. McCooeye, M.; Ding, L.; Gardner, G. J.; Fraser, C. A.; Lam, J.; Sturgeon, R. E.; Mester, Z. Separation and Quantitation of the Stereoisomers of Ephedra Alkaloids in Natural Health Products using Flow Injection-Electrospray Ionization-High Field Asymmetric Waveform Ion Mobility Spectrometry-Mass Spectrometry. *Analytical Chemistry*. **2003**, *75*, 2538-2542.
15. Guevremont, R.; Purves, R. W. High Field Asymmetric Waveform Ion Mobility Spectrometry-Mass Spectrometry: An Investigation of Leucine Enkephalin Ions Produced by Electrospray Ionization. *Journal of the American Society for Mass Spectrometry*. **1999**, *10*, 492-501.
16. Robinson, E. W.; Garcia, D. E.; Leib, R. D.; Williams, E. R. Enhanced Mixture Analysis of Poly(Ethylene Glycol) using High-Field Asymmetric Waveform Ion Mobility Spectrometry Combined with Fourier Transform Ion Cyclotron Resonance Mass Spectrometry. *Analytical Chemistry*. **2006**, *78*, 2190-2198.
17. Krylov, E. V.; Coy, S. L.; Vandermeij, J.; Schneider, B. B.; Covey, T. R.; Nazarov, E. G. Selection and Generation of Waveforms for Differential Mobility Spectrometry. *Review of Scientific Instruments*. **2010**, *81*.
18. Isenberg, S. L.; Armistead, P. M.; Glish, G. L. Optimization of Peptide Separations by Differential Ion Mobility Spectrometry. *Journal of the American Society for Mass Spectrometry*. **2014**, *25*, 1592-1599.
19. Frisch, M.; Trucks, G.; Schlegel, H.; Scuseria, G.; Robb, M.; Cheeseman, J.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. Gaussian 03, Revision D. 01; Gaussian: Wallingford, CT, USA, **2003**.
20. Connell, C., Jr; Glish, G. L.; Burinsky, D. J. High Amplitude Short Time Excitation: A Method to Form and Detect Low Mass Product Ions in a Quadrupole Ion Trap Mass Spectrometer. *Journal of the American Society for Mass Spectrometry*. **2006**, *17*, 81-84.
21. Balakrishnan, G.; Umapathy, S. Photogenerated Radical Intermediates of Vitamin K₁: A Time-Resolved Resonance Raman Study. *Journal of Molecular Structure*. **1999**, *475*, 5-11.
22. Hangarter, M.; Hörmann, A.; Kamdzhilov, Y.; Wirz, J. Primary Photoreactions of Phylloquinone (Vitamin K 1) and Plastoquinone-1 in Solution. *Photochemical & Photobiological Sciences*. **2003**, *2*, 524-535.

23. Snyder, C. D.; Rapoport, H. Photooxygenation of Phylloquinone and Menaquinones. *Journal of the American Chemical Society*. **1969**, *91*, 731-737.
24. Görner, H. Photoprocesses of P-Naphthoquinones and Vitamin K 1: Effects of Alcohols and Amines on the Reactivity in Solution. *Photochemical & Photobiological Sciences*. **2004**, *3*, 71-78.

Chapter 4

Analysis of Halobenzoquinones – Mass Spectrometry, Differential Ion Mobility Spectrometry, Electrochemistry, and Theoretical Calculations

4.1 Introduction

Halobenzoquinones (HBQs) are molecules that can be formed during the disinfection of drinking water [1-3]. Molecules formed during the water disinfection process are referred to as disinfection byproducts (DBPs). Some DBPs have raised health concerns as some epidemiological studies have reported an increase in the rate of bladder cancer possibly related to the consumption of chlorinated drinking water [4-6]. It is currently unknown which DBPs are possibly causing this increased risk of bladder cancer. This phenomenon is also associated with swimming in pools that are disinfected with chlorine-based products. Therefore, it is of extreme importance to be able to identify all possible DBPs. Some of the most commonly observed DBPs such as trihalomethanes and haloacetic acids are already regulated [7, 8]. However, a majority of possible organic halide DBPs have not been identified according to the Environmental Protection Agency.

HBQs have been classified as DBPs as they have been found in ng/L levels in drinking water. Additionally, HBQs have been predicted to be highly toxic using quantitative structure toxicity relationship analysis [2, 3]. The affinity of HBQs to DNA has been previously studied, and research has shown that HBQs can produce hydroxyl radicals which are known to cause DNA damage [9, 10]. One relation is the cytotoxicity and genotoxicity of benzoquinone which has been extensively studied.

Previously HBQs have been analyzed using ultra pressure liquid chromatography coupled with mass spectrometry with separations taking approximately 20 minutes [11]. To cut down on analysis time of HBQs differential ion mobility spectrometry (DIMS) was used as the separation method prior to mass analysis with this separation happening in milliseconds. The HBQs studied herein have been shown to produce two different ion species, $[M+H]^+$ and $M^{\bullet+}$. This study is focused on three different HBQs which are in Figure 4.1. The formation of the two ion types formed by these three HBQs will be analyzed in great detail using a variety of methods including mass spectrometry, differential ion mobility spectrometry, cyclic voltammetry, and density functional theory calculations.

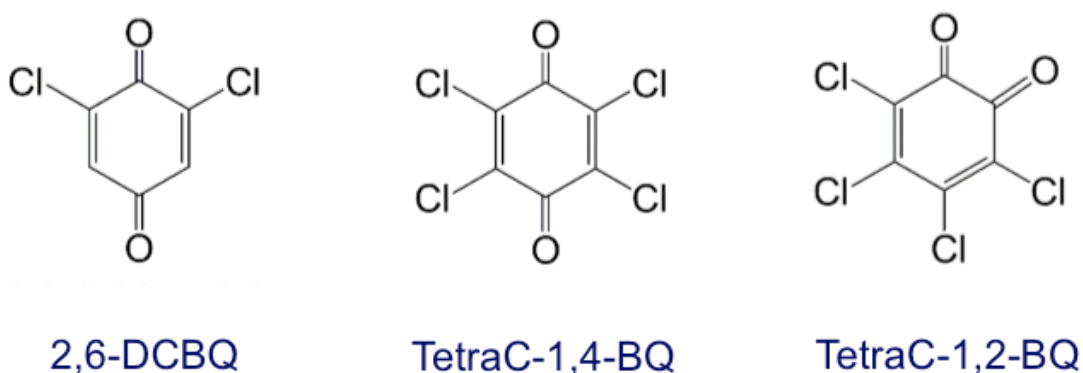


Figure 4.1. Halobenzoquinone molecules investigated in this study

4.2 Experimental

4.2.1 Samples

Tetrachloro-1,2-benzoquinone, tetrachloro-1,4-benzoquinone, and 2,6-dichlorobenzoquinone were all purchased from Sigma Aldrich (St. Louis, MO). Acetonitrile, methanol, ethanol, water, and formic acid were purchased from Fisher Scientific (Hampton, NH). Stock solutions for each halobenzoquinone were made by dissolving 3 mg of the halobenzoquinone in 10 mL of ethanol to create a 0.3 mg/mL solution. This solution was directly used for low temperature plasma ionization (LTPI) (or a similar solution with

acetonitrile as the solvent). The ethanol stock solutions were diluted to 10 μ M in 50/50 methanol/water with 0.1% formic acid added for ESI experiments.

4.2.2 Low Temperature Plasma Ionization

For some experiments, HBQ samples were ionized using a custom built low temperature plasma source like that depicted in Figure 4.2 [12]. LTPI utilizes a dielectric barrier discharge to generate a reaction cascade that results in the formation of ions [13, 14]. The LTPI source consists of a glass tube (i.d. 3.9 mm, o.d. 5.9 mm, length 73 mm) with a piece of copper tape wrapped around the end of the tube which is held at ground potential. A high ac voltage of approximately 4 kV_{0-p} is applied to the central tungsten electrode (diameter = 1.0 mm). The plasma gas used for these experiments was helium at a flow rate of approximately 3 L/min.

HBQs were analyzed for most experiments using a 0.3 mg/mL solution of the desired HBQ in ethanol or water. Additionally, to study the effect of solvent on the formation of different ion types experiments were also completed by ionizing solid samples of halobenzoquinones via LTPI. All samples were placed on copper tape on a glass microscope slide for analysis. Voltage and plasma gas flow settings were not fully optimized as settings were just varied until a significant amount of signal intensity was obtained for the HBQs studied.

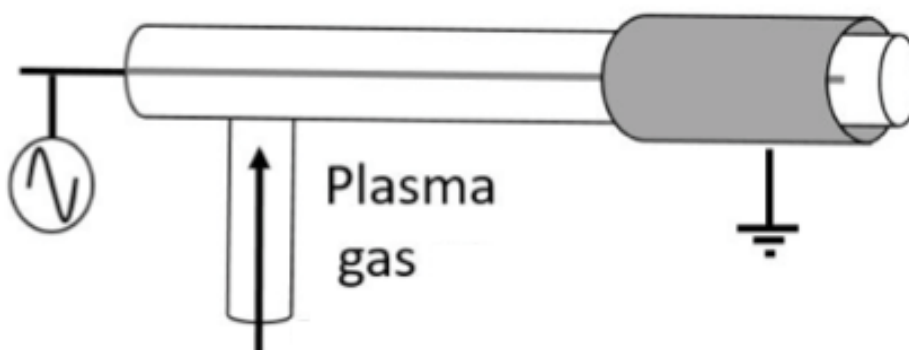


Figure 4.2. Low temperature plasma ionization (LTPI) probe schematic

4.2.3 Capillary-Skimmer Voltage Difference Studies

The capillary-skimmer voltage difference was varied to investigate the effect on the ions observed in the mass spectrometer. The voltage difference was determined by the difference in voltages applied to the exit of the glass transfer capillary and the skimmer as seen in the basic instrument schematic in Figure 4.3. For these experiments the voltage on the exit of the glass transfer capillary was kept at a constant value of -100 V while the voltage on the skimmer was varied from 0 to -100 V.

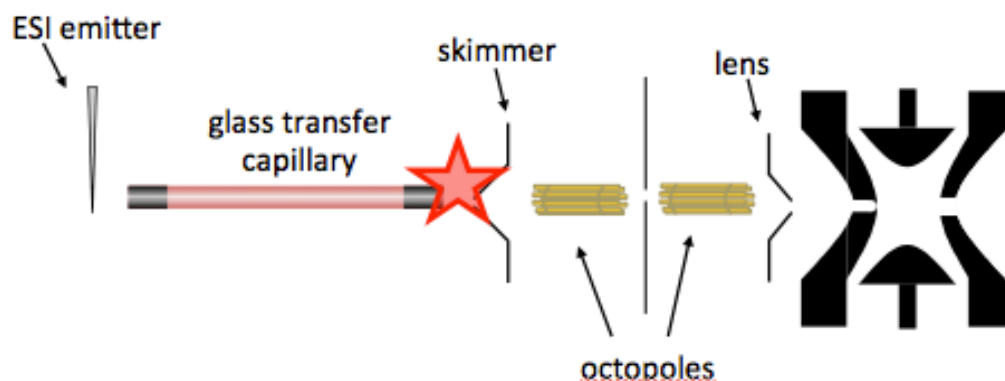


Figure 4.3. Bruker HCT ion trap schematic showing Capillary-Skimmer Voltage Difference

4.2.4 Differential Ion Mobility Spectrometry

A planar DIMS assembly composed of two parallel stainless steel electrodes was used for these experiments. Experiments were completed using the Generation 4 (G4) DIMS device coupled to a Bruker HCT ion trap mass spectrometer. The G4 device has a 0.3 mm gap with electrodes that are 4 mm in width and 10 mm in length. The mass spectrometer's desolvation gas (nitrogen) is rerouted through the DIMS assembly and is used as both the carrier gas to pull ions through the DIMS device and the desolvation gas for ESI.

A custom-built power supply was used for the G4 device. While a rectangular waveform would ideally be used to power a DIMS device, the high power requirements make this unfeasible. In lieu of this, a bisinusoidal waveform that approximates a rectangular waveform is used [15]. To generate this bisinusoidal waveform a sinusoidal wave is applied to one electrode with a given frequency and amplitude, and a phase-shifted waveform at twice the voltage and approximately half the amplitude is applied to the other electrode. This produces an electric field equivalent to the sum of the two individual waveforms [16]. The dispersion voltage is defined as the maximum voltage (V_{0-p}) of the bisinusoidal waveform. The dispersion field (E_D) is defined as the dispersion voltage divided by the gap size of the DIMS device. The bisinusoidal waveform was tuned to a frequency of 2 MHz for the G4 device. A custom power supply was designed for the G4 device so that the waveform could be centered around the ESI voltage and operated with the ESI emitter at ground potential.

A LabVIEW program linked to the instrument control software is used to set a range of compensation voltages or to define a static value. The DIMS device can be operated in either filter or scanning mode. In scanning mode the compensation field (E_C) is stepped over

a specific range as input into the LabVIEW program. By selecting a static voltage in the LabVIEW program the device can be operated in “filter mode”.

4.2.5 Cyclic Voltammetry

Cyclic voltammetry experiments were done using a CH Instrument Inc. model CH1030A multi-potentiostat. For all experiments a glassy carbon electrode was used as the working electrode. Scans for aqueous experiments were started at 0.6 V scanned to -0.4 V and then scanned back to 0.6 V at a scan rate of 25 mV/s. Non-aqueous experiments were started at 0.5 V scanned to -2.0 V and then scanned back to 0.5 V at a scan rate of 1 V/s. All experiments were done at room temperature. Experiments done in acetonitrile had 1 mM of halobenzoquinone and 0.100 M tetra-n-butyl-ammonium hexafluoro phosphate for a counter ion. Aqueous experiments were done using 0.5 mM halobenzoquinone in a 0.1 M phosphate buffer.

4.2.6 Computational Methods

Initial structures were drawn using GaussView 5. Structures were fully optimized at the HF/3-21G, B3LYP/3-21G, B3LYP/6-31G levels, before being finalized at the B3LYP/6-31G(d,p) level. After optimization at the B3LYP/6-31G(d,p) level the resulting structure was submitted to frequency analysis at the same level of theory. Calculations for adiabatic and vertical electron affinities were done at the B3LYP/6-311+G(d,p) level of theory to obtain more accurate energies for the electron affinity values. Transition state structures were determined using a synchronous transit-guided quasi-Newton method. These structures were confirmed to be transition states via frequency calculations to confirm that the structures had one imaginary frequency as well as via intrinsic reaction coordinate calculations. The Gaussian 03 suite of programs was used for all ab initio and DFT

calculations [17]. All calculations were carried out on the KillDevil supercomputer cluster at UNC-Chapel Hill.

4.3 Results and Discussion

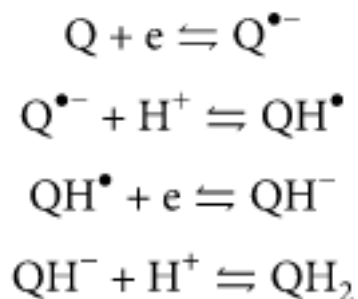
The interest in the analysis of halobenzoquinones is due largely in part to the interesting species that are observed in the mass spectrometer. The halobenzoquinones used in this study produce two different ion types when ionized via ESI or LTPI. Previous research on the analysis of halobenzoquinones using mass spectrometry has shown that the amount of the two different ion species could be controlled to some degree by experimental settings [11]. This research showed the formation of two species, $[M+H]^+$ and M^+ , when halobenzoquinones were ionized via ESI. They found that for most halobenzoquinones when they went from room temperature to high temperature source conditions they went from observing $[M+H]^+$ ions to M^+ ions. This was the case for 2,6-dichlorobenzoquinone and tetrachloro-1,4-benzoquinone. However, ionization of tetrachloro-1,2-benzoquinone produced abundant $[M+H]^+$ ions at both temperature settings. Further examination of this phenomenon is of interest. The goal is to determine why these different ion types are formed and why the two isomers, tetraC-1,4-BQ and tetraC-1,2-BQ, behave so differently. A variety of techniques are used including mass spectrometry, differential ion mobility spectrometry, cyclic voltammetry, and theoretical calculations to further understand the formation of the different ion types observed. Finally, the capability of DIMS at separating the two isomers, tetraC-1,4-BQ and tetraC-1,2-BQ is discussed.

4.3.1 Unique Ion Types in Mass Spectrometry Formed via Electrochemical Reduction

Previous research by Xing-Fang Li's group showed that as the source temperature of their instrument was changed they went from seeing the $[M+H]^+$ species at low temperatures

to predominantly having the $M^{\bullet-}$ species at higher temperatures [11]. Neither of these species are typical ion types observed when ionizing via ESI. For negative mode mass spectrometry, ESI would typically generate deprotonated ions of the form $[M-nH]^{n-}$ where n is the number of protons lost. It has been hypothesized that the ion types observed when ionizing halobenzoquinones are formed via electrochemical reduction [11].

Halobenzoquinones can be better understood by looking at the well-known and well-studied electrochemistry of benzoquinone and other quinone based systems [18-20]. The $M^{\bullet-}$ species is formed via a one-electron reduction. A proton can then be added to the $M^{\bullet-}$ species to form the $[M+H]^{\bullet}$ species also known as a semiquinone. Another one-electron reduction process results in the formation of the $[M+H]^-$ species. Addition of another proton creates the $[M+2H]$ species, called the hydroquinone. Also a second electron transferred to the $M^{\bullet-}$ species can result in the formation of the M^{2-} species.



The Li group reported data collected at source temperatures of 23 °C (room temperature) and 700 °C (this is the temperature set in the instrument software and therefore the actual temperature is unknown). The ion trap used in the present study is only capable of using a source temperature setting maximum of 300 °C (measured as 133.4 ± 1.0 °C using Klein Tool MM400 with a bead wire type K temperature probe). The halobenzoquinones were ionized via ESI at a source temperature setting of 300 °C (Figure 4.4). ESI mass

spectra for tetraC-1,4-BQ (a) and tetraC-1,2-BQ (b) exhibit drastically different behavior (the behavior of 2,6-DCBQ is identical to that of tetraC-1,4-BQ and therefore is not shown). The mass spectrum for tetraC-1,4-BQ (Figure 4.4a) has both the $[M+H]^-$ and M^{\bullet} species present. However, the mass spectrum for tetraC-1,2-BQ (Figure 4.4b) has very little of the M^{\bullet} species and is dominated by the $[M+H]^-$ species.

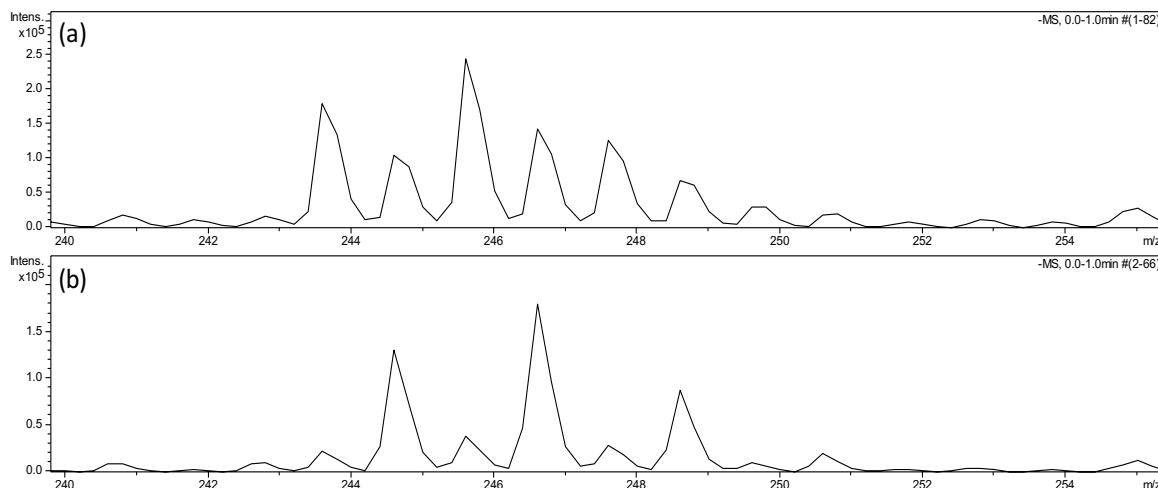


Figure 4.4. ESI of HBQs using settings of capillary exit at -150 V and skimmer at -60 V (a) tetraC-1,4-BQ and (b) tetraC-1,2-BQ (the behavior of 2,6-DCBQ is identical to that of tetraC-1,4-BQ and therefore is not shown)

This data is different from that of the Li group due to the amount of the M^{\bullet} species observed in the mass spectrum of tetraC-1,4-BQ. Since their data was collected at a much higher temperature setting the initial working hypothesis to describe the differences between experiments is that when the higher source temperatures are used more energy is imparted to the ions, thus forming the M^{\bullet} ion. Previous electrochemical studies on halobenzoquinones has shown that in protic solvents either the two-electron, one-proton reduction to form the $[M+H]^-$ species or the two-electron, two-proton reduction to form the $[M+2H]$ species can occur dependent on pH. The pH of solutions used for ionization is likely to lead to the

electrochemical formation of the $[M+2H]^+$ species which can lose a proton during the ESI process to form the $[M+H]^+$ species.

The instrument used in this study is not capable of source temperatures as high as those used in the previously published work. Therefore, an alternative way in which to impart more energy into the ions to manipulate the ion type observed was desired. This was obtained via changing the voltage difference between the exit of the capillary and the skimmer lens. For these experiments the capillary exit was set to a voltage of -100 V and the skimmer voltage was changed from 0 to -100 V creating a range of capillary/skimmer voltage differences of 0 to 100 V. To compare the intensity of the two species ($[M+H]^+$ and $M^{•+}$) at the various voltage settings, a ratio of the intensities was calculated (intensity of $M^{•+}$ /intensity of $[M+H]^+$). A summary of these findings collected at a source temperature setting of 300 °C can be found in Table 4.1. The first information to take from this data is that for tetraC-1,4-BQ all ratios are greater than one meaning that the $M^{•+}$ species is more abundant, while for tetraC-1,2-BQ all ratios are less than one meaning that the $[M+H]^+$ species is more abundant. For tetraC-1,4-BQ, an increase in the ratio is observed as the capillary/skimmer voltage difference is increased with a maximum ratio observed at a voltage difference of 70 V. Above a voltage difference of 70 V the $M^{•+}$ species begins to dissociate and therefore the ratios begin to decrease above this voltage setting. As the capillary/skimmer voltage difference is increased the ions are subjected to higher energies. The increase in the intensity of the $M^{•+}$ species relative to the $[M+H]^+$ species as the amount of energy increases supports the hypothesis that the $[M+H]^+$ ions can lose a hydrogen atom to form the $M^{•+}$ species. The intensity ratio changes very little for tetraC-1,2-BQ over the range of voltages studied and no increase in the ratio of intensities is observed as the amount of energy is increased.

Table 4.1. Ratio of intensities $M^{\bullet+}/[M+H]^+$ as the voltage difference between the capillary exit and skimmer is increased

Capillary/Skimmer Voltage Difference (V)	Ratio of Intensities $M^{\bullet+}/[M+H]^+$	
	TetraC-1,4-BQ	TetraC-1,2-BQ
0	1.4 ± 0.2	0.22 ± 0.01
10	1.5 ± 0.2	0.21 ± 0.01
20	1.4 ± 0.2	0.20 ± 0.01
30	1.4 ± 0.1	0.19 ± 0.03
40	1.4 ± 0.1	0.20 ± 0.02
50	1.3 ± 0.1	0.22 ± 0.01
60	2.1 ± 0.2	0.14 ± 0.01
70	2.9 ± 0.3	0.12 ± 0.02
80	2.7 ± 0.3	0.14 ± 0.01
90	2.1 ± 0.2	0.16 ± 0.01
100	1.3 ± 0.1	0.16 ± 0.01

4.3.2 Separation of Halobenzoquinone Isomers using Differential Ion Mobility Spectrometry

DIMS has previously been shown to impart energy to ions as they pass through the device into the mass spectrometer. HBQs were subjected to analysis via DIMS to see what affect the extra energy introduced to the system would have on the ion species observed as ions travel through the DIMS device. However, because DIMS is most commonly used as a method to separate isomers the ability of DIMS to distinguish between the two halobenzoquinone isomers, tetraC-1,4-BQ and tetraC-1,2-BQ was of interest. The two isomers were ionized via ESI and DIMS-MS scans were collected at five dispersion fields. The results of these studies are summarized in Table 4.2 for the $[M+H]^+$ species and in Table 4.3 for the $M^{\bullet+}$ species. For the $[M+H]^+$ species there is essentially no separation at the four lowest dispersion fields used, while at the highest dispersion field (40.0 kV/cm) slight separation is observed. However, for the $M^{\bullet+}$ species a significant increase in the ability to separate between the two isomers was observed. Especially at the highest dispersion field

where a difference in almost 80 V/cm was observed between the M^{\bullet} species for the two isomers.

Table 4.2. Compensation field at peak intensity for various dispersion fields for TetraC-1,2-BQ and TetraC-1,4-BQ for the $[M+H]^+$ species

Dispersion Field (kV/cm)	Compensation Field at Peak Intensity (V/cm)	
	TetraC-1,2-BQ	TetraC-1,4-BQ
26.7	102 ± 5	96 ± 3
30.0	146 ± 2	140 ± 2
33.3	220 ± 5	214 ± 8
36.7	297 ± 3	281 ± 7
40.0	385 ± 2	434 ± 14

Table 4.3. Compensation field at peak intensity for various dispersion fields for TetraC-1,2-BQ and TetraC-1,4-BQ for the M^{\bullet} species

Dispersion Field (kV/cm)	Compensation Field at Peak Intensity (V/cm)	
	TetraC-1,2-BQ	TetraC-1,4-BQ
26.7	85 ± 6	103 ± 3
30.0	127 ± 2	154 ± 2
33.3	200 ± 4	233 ± 7
36.7	277 ± 4	292 ± 3
40.0	362 ± 3	441 ± 13

Based on the better separation observed for the M^{\bullet} species there was interest in seeing the behavior of the halobenzoquinones when ionized via LTPI. When ionized via LTPI from a solution of ethanol an unexpected amount of the $[M+H]^+$ species was observed. DIMS-MS scans for LTPI ionized 2,6-DCBQ in various solvent conditions are presented in Figure 4.5. In Figure 4.5a for 2,6-DCBQ dissolved in ethanol, a significant amount of $[M+H]^+$ was observed. This was attributed to the protic nature of ethanol. Figure 4.5b is the DIMS-MS scan for 2,6-DCBQ dissolved in acetonitrile, an aprotic solvent, and no $[M+H]^+$ species was observed. Additionally, the DIMS-MS scan of solid 2,6-DCBQ ionized via LTPI had no $[M+H]^+$ present. These results support the fact that the $[M+H]^+$ species is formed via electrochemical reduction. The $[M+H]^+$ species is formed via loss of a proton

from the two-electron, two-proton reduction species ($[M+2H]^+$). When no protons are available (solid 2,6-DCBQ and 2,6-DCBQ dissolved in acetonitrile) none of the $[M+H]^+$ species was observed. This is due to the fact that there were no protons available to participate in the electrochemical reduction process to form the $[M+2H]^+$ species.

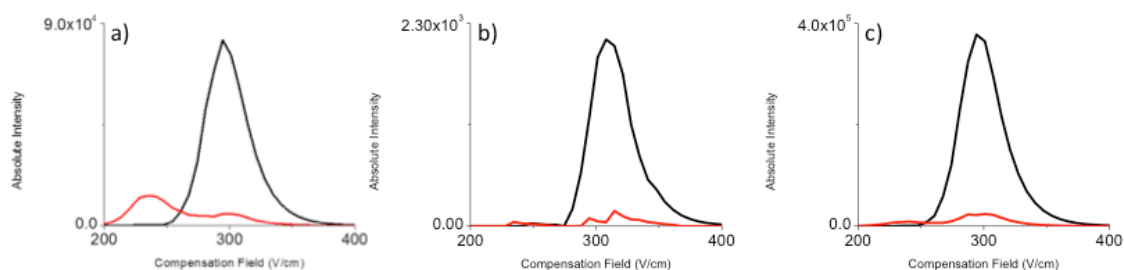


Figure 4.5. DIMS-MS scans for LTPI ionized 2,6-DCBQ at a dispersion field of 36.7 kV/cm in various solvents (a) ethanol (b) acetonitrile and (c) solid, no solvent

4.3.3 Effect of Increased Energy on Ion Types Observed using Differential Ion Mobility Spectrometry

To further study the effect of energy on the ion type observed, ESI ionized halobenzoquinones were subjected to DIMS-MS scans at various dry gas flow rates. The increase in dry gas flow rate has no effect on the ion residence time in the DIMS device. The DIMS device is connected to the glass transfer capillary of a mass spectrometer that has a conductance limit of approximately 1.5 L/min. Due to this conductance limit the gas flow rate through the DIMS device, which is what controls the amount of time ions spend in the DIMS devices, does not change as the dry gas flow rate is changed. The change in dry gas flow rate does however have an effect on the temperature of the DIMS electrodes. At a dry gas temperature setting of 300 °C, the temperature of the DIMS electrodes were measured to be 53 °C at a dry gas flow rate of 2.5 L/min, 84 °C at 5 L/min, and 111 °C at 7.5 L/min. As the temperature of the electrodes is increased there is a decrease in the gas number density (N) between the electrodes because the pressure is constant. Therefore, there is an increase

in the value of E/N (electric field strength/gas number density) which ion mobility is dependent on [21, 22]. As the dry gas flow rate is increased the effective temperature of the dry gas increases. Since pressure is constant in the DIMS device this increase in temperature corresponds to a decrease in gas number density. This decrease in gas number density causes the value of E/N to increase thus increasing the amount of energy that the ions are subjected to. Thus the effect of increased energy on the observance of the different ion types can be studied by collecting DIMS-MS scans at various dry gas flow rates.

DIMS-MS scans at various dry gas flow rates of ESI ionized 2,6-DCBQ are plotted in Figure 4.6. The black trace corresponds to m/z 175.8 (mass-to-charge ratio of the M^+ species) and the red trace corresponds to m/z 176.8 (mass-to-charge ratio of the $[M+H]^+$ species, as well as the ^{13}C peak of the M^+ species). At the two lowest dry gas flow rate settings of 2.5 L/min (Figure 4.6a) and 5.0 L/min (Figure 4.6b) both the M^+ species and $[M+H]^+$ species are observed. However, at the highest dry gas flow rate setting of 7.5 L/min only the M^+ species is observed (Figure 4.6c). The intensity of the red trace (m/z 176.8) relates to the intensity of the ^{13}C peak of the M^+ species expected. As was the case for the capillary/skimmer voltage difference studies discussed previously, these results again suggest that as energy is increased the $[M+H]^+$ species is converted to the M^+ species via loss of a hydrogen atom.

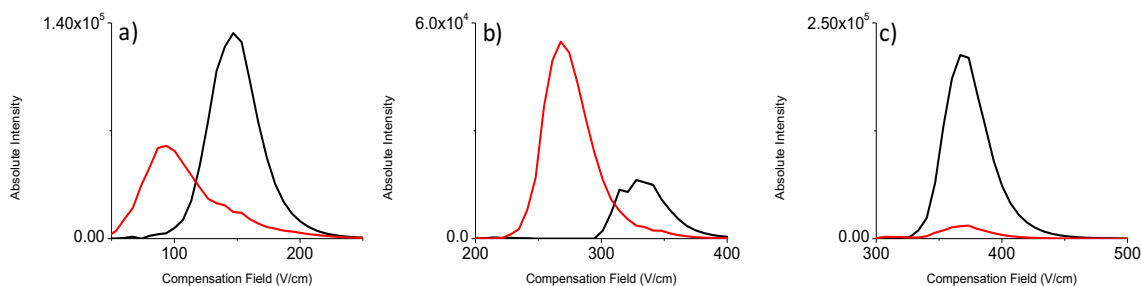


Figure 4.6. DIMS-MS scans for ESI ionized 2,6-DCBQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min

The DIMS-MS scans of ESI ionized tetraC-1,4-BQ at the various dry gas flow rates showed very similar results (Figure 4.7). The black trace corresponds to m/z 243.6 (mass-to-charge ratio of the M^{\bullet} species) and the red trace corresponds to m/z 244.6 (mass-to-charge ratio of the $[M+H]^+$ species, as well as the ^{13}C peak of the M^{\bullet} species). There is a significant amount of $[M+H]^+$ species observed at a flow rate of 2.5 L/min (Figure 4.7a). The amount of $[M+H]^+$ species present in the DIMS-MS scan at a flow rate of 5.0 L/min (Figure 4.7b) is significantly less than that observed at 2.5 L/min. Similar to 2,6-DCBQ, at the highest dry gas flow rate the DIMS-MS scan has no observable $[M+H]^+$ species. Again the increase in energy due to the increase in dry gas flow rate causes the $[M+H]^+$ species to lose a hydrogen atom to form the M^{\bullet} species.

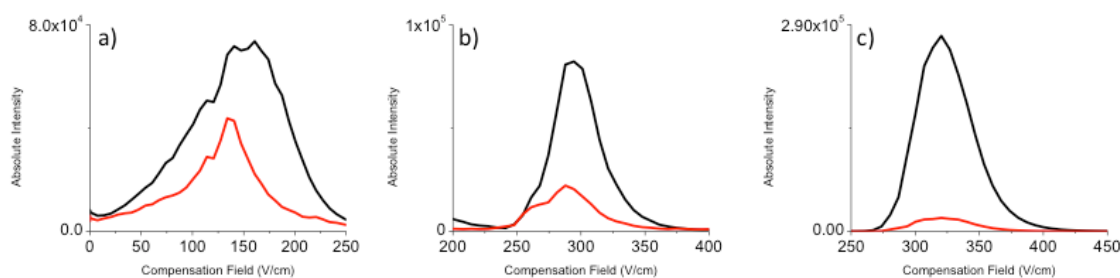


Figure 4.7. DIMS-MS scans for ESI ionized tetraC-1,4-BQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min

The DIMS-MS scans for ESI ionized tetraC-1,2-BQ differ drastically from the other two halobenzoquinones (Figure 4.8). The black trace corresponds to m/z 243.6 (mass-to-charge ratio of the M^{\bullet} species) and the red trace corresponds to m/z 244.6 (mass-to-charge ratio of the $[M+H]^+$ species, as well as the ^{13}C peak of the M^{\bullet} species). At all three dry gas flow rates used the $[M+H]^+$ species is most abundant. However, there is an observable increase in the amount of M^{\bullet} species as the dry gas flow rate is increased. Again suggesting

that inputting more energy corresponds to formation of the M^{\bullet} species from the $[M+H]^+$ species.

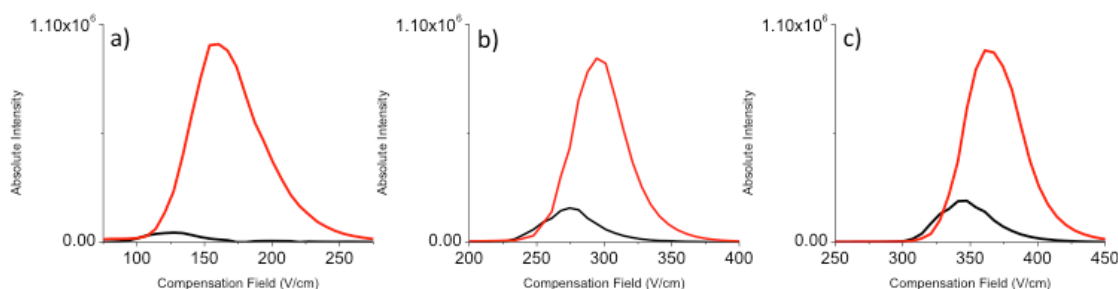


Figure 4.8. DIMS-MS scans for ESI ionized tetraC-1,2-BQ at a dispersion field of 36.7 kV/cm at varying dry gas flow rates of (a) 2.5 L/min (b) 5.0 L/min and (c) 7.5 L/min

An interesting aside is the difference in behavior between 2,6-DCBQ and tetraC-1,4-BQ relating to the difference in compensation field at which the two ion types are observed. The $[M+H]^+$ and M^{\bullet} ions are observed at significantly different compensation fields for 2,6-DCBQ while they are observed at almost the same compensation field for tetraC-1,4-BQ. The only difference between the two compounds is the addition of two chlorines in tetraC-1,4-BQ. If the proton is attached to the oxygen atom closer to the chlorine atoms in 2,6-DCBQ then the behavior of the two halobenzoquinones should be the same. However, theoretical calculations of the proton affinities for the two different possible protonated sites of 2,6-DCBQ are different by approximately 1 kcal/mol. Thus suggesting that the most likely protonation site for 2,6-DCBQ would be at the oxygen further from the chlorine atoms making it behave differently from tetraC-1,4-BQ.

4.3.4 Electrochemistry and Cyclic Voltammetry

Electrochemical studies using cyclic voltammetry were conducted to further investigate the processes taking place to form the $[M+H]^+$ and M^{\bullet} species. There are some electrochemical studies that have been done on two of the halobenzoquinones that are studied in this work. Both tetraC-1,2-BQ and tetraC-1,4-BQ have been studied via cyclic

voltammetry previously. TetraC-1,2-BQ has been studied in both acetonitrile and aqueous (phosphate buffered saline) environments [23, 24]. While tetraC-1,4-BQ has only been studied in acetonitrile [25]. Using these previous studies as a guide, cyclic voltammetry experiments on the three halobenzoquinones used in this study were completed. The cyclic voltammetry experiments were done in both aqueous and non-aqueous (acetonitrile) environments. This is necessary because different electrochemical reduction products are observed in the different solvent systems. This work made use of a silver/silver chloride reference electrode for the aqueous experiments and a silver/silver nitrate reference electrode for the non-aqueous experiments. A cyclic voltammetry scan of the ferrocene system was conducted in order to be able to convert all the experimental reduction potentials to a common reference electrode system.

Cyclic voltammograms for the three halobenzoquinones in an aqueous environment (pH 7 phosphate buffer) are presented in Figure 4.9. The voltammograms were collected at a scan rate of 25 mV/sec using a silver/silver chloride reference electrode and are shown background corrected. For 2,6-DCBQ (Figure 4.9a) there was a single peak corresponding to the electrochemical reduction to form the $[M+H]^-$ species and the reduction potential was determined to be 0.106 V. Like 2,6-DCBQ, tetraC-1,4-BQ (Figure 4.9b) also has a single peak corresponding to the reduction to form the $[M+H]^-$ species. The reduction potential for the formation of the $[M+H]^-$ species of tetraC-1,4-BQ was determined to be 0.116 V. The cyclic voltammogram of tetraC-1,2-BQ (Figure 4.9c) differs from those of the other two halobenzoquinones because it has two peaks that correspond to the reduction to form the $[M+H]^-$ species and the hydroquinone species. The reduction potentials for these two reductions were determined to be 0.118 V and 0.246 V, respectively.

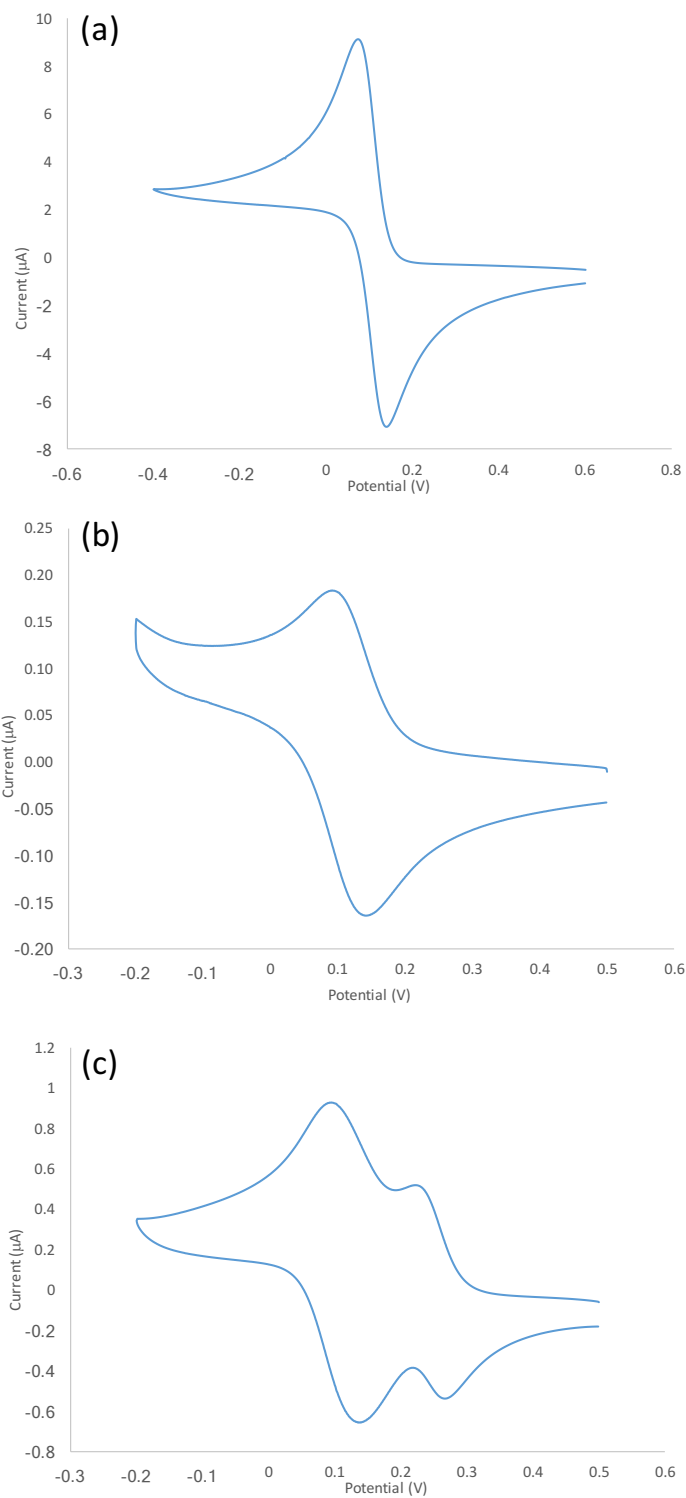


Figure 4.9. Cyclic voltammetry of (a) 2,6-DCBQ, (b) TetraC-1,4-BQ, and (C) TetraC-1,2-BQ in phosphate buffer at pH 7 with a scan rate of 25 mV/sec using a Ag/AgCl reference electrode. Background corrected.

The difference between the anodic and cathodic peak potentials were measured for all systems. It was determined that this difference was above 30 mV and below 60 mV for all of the redox couples. The number of electrons transferred is determined by this separation between the peak potentials. For a completely reversible electrochemical reaction the separation between the two peak potentials should be approximately 60 mV for a one-electron process and approximately 30 mV for a two-electron process. The values from the above experiments fall between this range, and this is due to some amount of electrochemical irreversibility. Slow electron transfer can cause the separation between peak potentials to be elevated. Peak cathodic and anodic currents should be equal for fast electrochemical couples. For these halobenzoquinones the ratio of the peak anodic and cathodic currents are not equal to one. Thus the reduction of halobenzoquinones is not a fast electron exchange process so higher than expected separation between peak potentials is often observed. Thus for these systems the studies above having a separation of peak potentials greater than 30 mV but less than 60 mV means that two-electron processes are occurring. Which supports the fact that the $[M+H]^-$ species is formed for all three halobenzoquinones and the second peak observed for tetraC-1,2-BQ corresponds to the electrochemical reduction to form the hydroquinone species ($[M+2H]^-$).

Cyclic voltammograms for the three halobenzoquinones in a non-aqueous environment (acetonitrile) are in Figure 4.10. Voltammograms were collected at a scan rate of 1 V/sec using a silver/silver nitrate reference electrode. The cyclic voltammograms are not shown background corrected in this case as the background has a large peak corresponding to the reduction of oxygen that greatly interferes at more negative potentials. For reference, Figure 4.11 is the cyclic voltammogram of acetonitrile with Bu_4NPF_6 . The

peak corresponds to a reduction potential that matches nicely with previously reported results for the reduction of oxygen [26]. This does not really matter as the species of interest is reduced at significantly less negative potentials and therefore is not affected by the reduction of oxygen. For 2,6-DCBQ (Figure 4.10a) the peak corresponding to the electrochemical reduction to form the M^{\bullet} species is at a reduction potential of -0.449 V. TetraC-1,4-BQ (Figure 4.10b) cyclic voltammogram has a peak corresponding to the reduction to form the M^{\bullet} species occurs at a reduction potential of -0.245 V. The cyclic voltammogram of tetraC-1,2-BQ (Figure 4.10c) has a peak corresponding to the reduction to the M^{\bullet} species, and the reduction potential was determined to be -0.149 V.

The separation of the peak anodic and cathodic peaks for the acetonitrile experiments were determined to be approximately 60 mV. Based upon the discussion above this means that the product formed during the electrochemical reduction of the halobenzoquinones in acetonitrile is a one-electron process. This would correspond to the formation of the M^{\bullet} species.

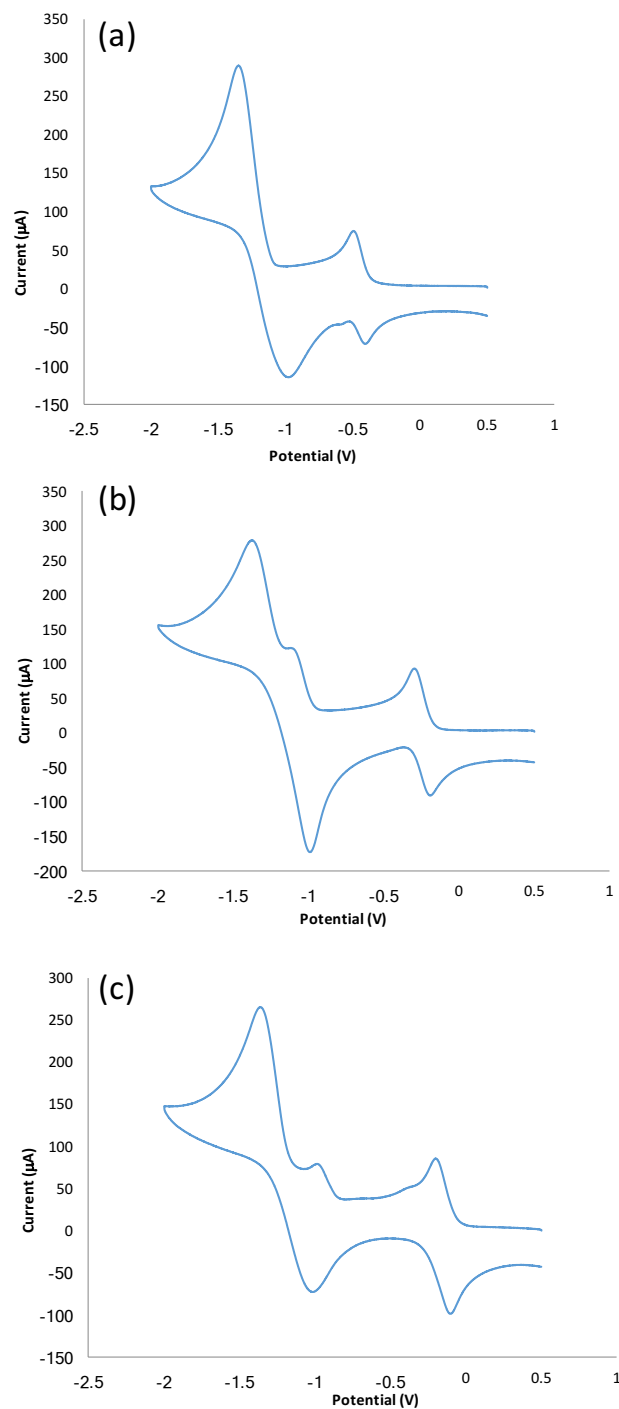


Figure 4.10. Cyclic voltammetry of (a) 2,6-DCBQ, (b) TetraC-1,4-BQ, and (c) TetraC-1,2-BQ in 0.1 M Bu_4NPF_6 in acetonitrile with a scan rate of 1 V/sec using a silver/silver nitrate reference electrode. Background uncorrected.

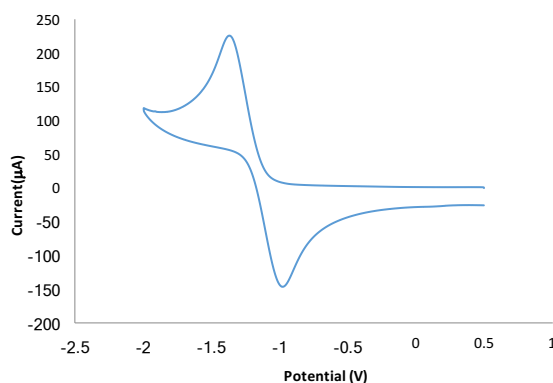


Figure 4.11. Cyclic voltammetry of 0.1 M Bu₄NPF₆ in acetonitrile with a scan rate of 1 V/sec using a silver/silver nitrate reference electrode.

All observed reduction potentials for the three halobenzoquinones were all converted to be referenced to the ferrocene couple reference electrode to allow for a direct comparison across all cyclic voltammetry experiments (Table 4.4). To understand this data, it is important to recall that a more negative reduction potential means that the electrochemical reduction is easier to occur. In the aqueous experiments for the formation of the [M+H]⁻ species, 2,6-DCBQ was found to have the most negative reduction potential (just slightly) while the reduction potentials for tetraC-1,4-BQ and tetraC-1,2-BQ were almost identical. The non-aqueous experiments showed that the reduction to form the M^{•-} species should be easiest for 2,6-DCBQ (-0.590 V) followed by tetraC-1,4-BQ (-0.386 V) and that leaves tetraC-1,2-BQ at -0.290 V to be the hardest to reduce.

The cyclic voltammetry studies in this work were done at a pH of 7. Other studies have found that in protic solvents at lower pH the [M+2H] (hydroquinone) species is readily formed via a two-electron, two-proton reduction. The solutions used for ESI are at a pH lower than 7, thus it is highly likely that the hydroquinone species is being formed. Then the [M+H]⁻ species is formed via loss of a proton during the ESI ionization process.

The other halobenzoquinone, tetraC-1,2-BQ, is drastically different as the reduction potential for the formation of the $[M+H]^-$ species is actually more negative than the reduction potential for the formation of the M^{\bullet} species. Additionally, the fact that the cyclic voltammetry experiments in an aqueous environment also showed formation of $[M+2H]$ (hydroquinone) is of importance. This species when ionized via ESI would lose a proton to form the $[M+H]^-$ species. This fully supports the mass spectrometry based experiments that were previously discussed in this work it was observed that TetraC-1,2-BQ showed a preference to form the $[M+H]^-$ species.

Table 4.4. Cyclic voltammetry reduction potentials (V) summary for all three halobenzoquinones studied where all values are referenced versus the ferrocene couple

	[M+H] ⁻	[M+2H]	M [•]
2,6-DCBQ	-0.324 ± 0.0005	Not observed	-0.590 ± 0.004
TetraC-1,4-BQ	-0.314 ± 0.0006	Not observed	-0.386 ± 0.002
TetraC-1,2-BQ	-0.312 ± 0.003	-0.184 ± 0.001	-0.290 ± 0.003

4.3.5 Calculations

Density functional theory calculations can aid in being better able to understand this data.

This can be particularly helpful in the interpretation of the behavior of the two halobenzoquinone isomers, tetraC-1,2-BQ and tetraC-1,4-BQ. The scheme below (Figure 4.12) for tetraC-1,4-BQ (tetraC-1,2-BQ is the same except for placement of oxygens) gives an overview of the important species that are subjected to DFT calculations.

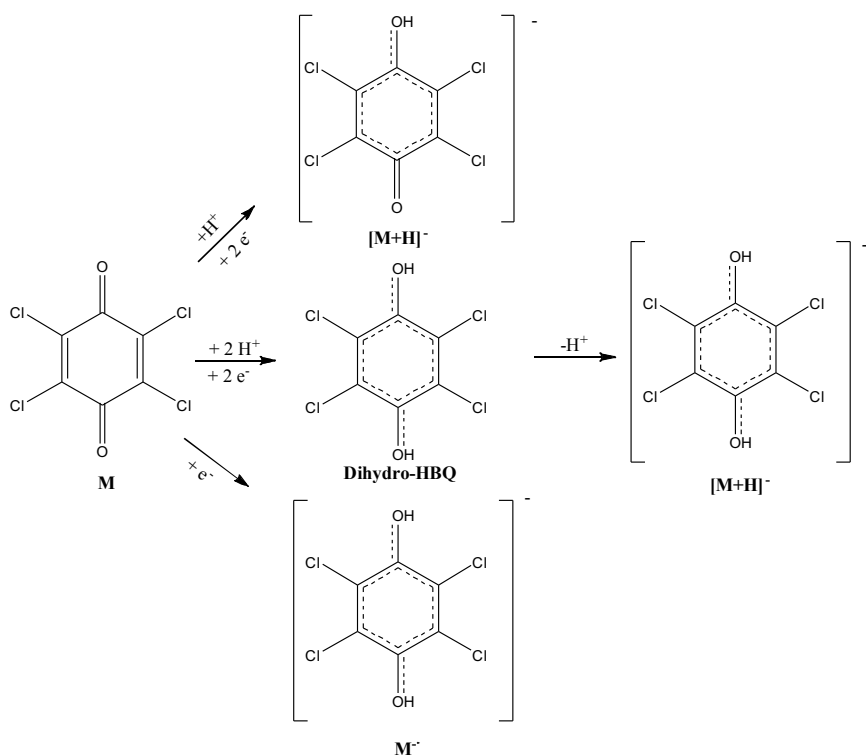


Figure 4.12. Schematic showing the important species that are formed via the electrochemical reduction of halobenzoquinones. These structures served as a starting points for theoretical calculations.

Calculation procedures are discussed in the experimental section of this chapter.

Important to note is the final optimized structures and energy values are from calculations at B3LYP/6-311G(d,p) level of theory. Summaries of the energy values for the two halobenzoquinone isomers can be seen in Table 4.5.

Table 4.5. Energetics summary from the theoretical calculations on tetraC-1,2-BQ and tetraC-1,4-BQ. All values are reported in eV. Energy values are reported as positive values for the $M^{\bullet-}$ species as that is the convention commonly used for reporting electron affinity values.

HBQ	Neutral	$M^{\bullet-}$	$[M+2H+2e]$	$[M+H]^{\bullet-}$
TetraC-1,4-BQ	0	2.84	-33.7	-19.4
TetraC-1,2-BQ	0	2.91	-33.2	-19.0

There are two important pieces of information gleaned from these calculations. One is related to the optimized structures in Figure 4.13. The $[M+H]^+$ ion of tetraC-1,2-BQ has a slightly elongated OH bond, which suggests that hydrogen bonding is occurring with the other oxygen. This explains why more abundant $[M+H]^+$ species versus the M^+ species for tetraC-1,2-BQ compared to tetraC-1,4-BQ was observed. Transition state structures were found using theoretical calculations for the formation of the M^+ species from the $[M+H]^+$ species. For tetraC-1,4-BQ the transition state was determined to have an activation energy barrier of 89.9 kcal/mol while for tetraC-1,2-BQ the transition state was found to have an activation energy barrier of 105.6 kcal/mol. This means that it takes approximately 15 kcal/mol less energy to form the M^+ species for tetraC-1,4-BQ than it does to form the same species for tetraC-1,2-BQ. Thus further supporting why the M^+ species is observed in abundance for tetraC-1,4-BQ, but is observed in very small amounts for tetraC-1,2-BQ.

Also the formation of the hydroquinone species, $[M+2H]^+$, is energetically favorable over the formation of the $[M+H]^+$ species. Therefore, it is most likely that the two-electron, two-proton reduction is taking place in solution to form the $[M+2H]^+$ species and then a proton is lost during the ESI ionization process to form the $[M+H]^+$ species. The $[M+H]^+$ species can then be observed in the mass spectrometry experiments and also if enough energy is present the $[M+H]^+$ species can lose a hydrogen atom to form the M^+ species.

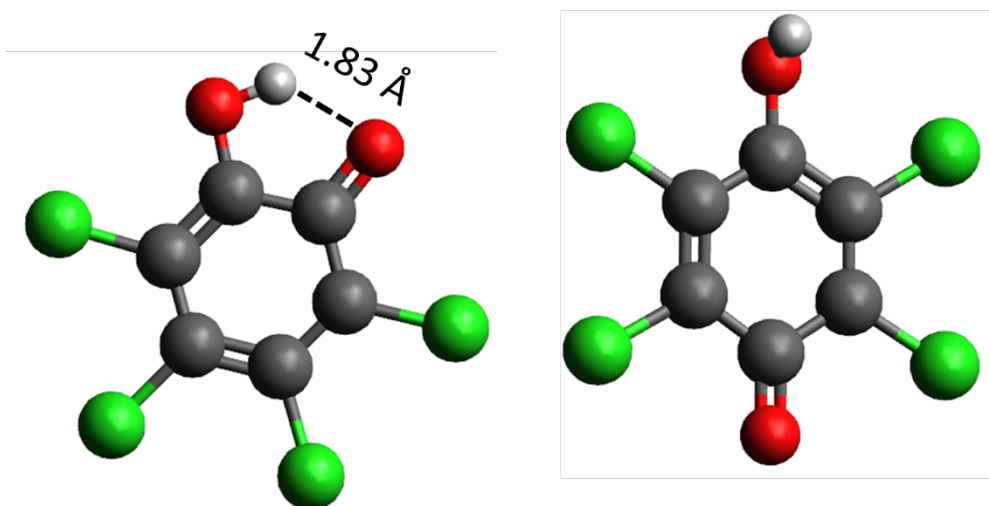


Figure 4.13. Optimized structures for the $[M+H]^+$ ions of tetraC-1,2-BQ (left) and tetraC-1,4-BQ (right).

4.4 Conclusions

The two ion types observed in the mass spectra of halobenzoquinone compounds are atypical for ESI generated ions. This is due to the fact that the species that are ionized first undergo electrochemical reduction in solution. In the protic solvents used to dissolve the halobenzoquinones studies in this work, electrochemical cyclic voltammetry experiments and theoretical calculations suggest that a two-electron, two-proton reduction takes place to the form the $M+2H$ (hydroquinone) species in solution. This species when ionized via ESI loses a proton to form the $[M+H]^+$ species. The $[M+H]^+$ species was the most abundant species observed for tetraC-1,2-BQ. However, for tetraC-1,4-BQ and 2,6-DCBQ in addition to the $[M+H]^+$ species the more abundant M^+ species was also observed. Increasing the voltage difference between the capillary exit and skimmer of the mass spectrometer resulted in more abundant M^+ species. Additionally, DIMS experiments showed that when using parameters that result in higher energies being imparted to the ions eventually only the M^+ species was observed for tetraC-1,4-BQ and 2,6-DCBQ. For tetraC-1,2-BQ the $[M+H]^+$ species remains

prevalent throughout all experiments. Theoretical calculations showed that this is likely due to the hydrogen bonding that takes place in the tetraC-1,2-BQ molecule that is not available to the other halobenzoquinone compounds. Furthermore, the process of removing a hydrogen atom to form the M^{\bullet} species from the $[M+H]^+$ species has a higher activation energy barrier (~ 15 kcal/mol) for tetraC-1,2-BQ when compared to tetraC-1,4-BQ based upon theoretical transition state calculations.

4.5 REFERENCES

1. Zhao, Y., Qin, F., Boyd, J.M., Anichina, J., Li, X.-F.; Characterization and Determination of Chloro- and Bromo-Benzoquinones as New Chlorination Disinfection Byproducts in Drinking Water. *Analytical Chemistry*. **2010**, *82*, 4599-4605.
2. Qin, F., Zhao, Y.-Y., Zhao, Y., Boyd, J.M., Zhou, W., Li, X.-F.; A Toxic Disinfection By-product, 2,6-Dichloro-1,4-benzoquinone, Identified in Drinking Water. *Angewandte Chemie International Edition*. **2010**, *49*, 790-792.
3. Zhao, Y., Anichina, J., Lu, X., Bull, R.J., Krasner, S.W., Hrudey, S.E., et al.; Occurrence and Formation of Chloro- and Bromo-benzoquinones During Drinking Water Disinfection. *Water Research*. **2012**, *46*, 4351-4360.
4. Krasner, S.W., Weinberg, H.S., Richardson, S.D., Pastor, S.J., Chinn, R., Scilimenti, M.J., et al.; Occurrence of a New Generation of Disinfection Byproducts. *Environmental Science & Technology*. **2006**, *40*, 7175-7185.
5. Richardson, S.D., Plewa, M.J., Wagner, E.D., Schoeny, R., DeMarini, D.M. Occurrence, Genotoxicity, and Carcinogenicity of Regulated and Emerging Disinfection By-Products in Drinking Water: A Review and Roadmap for Research. *Mutation Research*. **2007**, *636*, 178.
6. Mills, C.J., Bull, R.J., Cantor, K.P., Reif, J., Hrudey, S.E., Huston, P. Workshop Report. Health Risks of Drinking Water Chlorination By-Products: Report of an Expert Working Group. *Chronic Diseases in Canada*. **1998**, *19*, 91.
7. United States Environmental Protection Agency Fed. Regist. **2006**, *71*, 388.
8. European Union (Council Directive 98/83/EC) Off. J. Eur. Commun. **1998**, *41*, 32.
9. Anichina, J., Zhao, Y., Hrudey, S.E., Schreiber, A., Li, X.F. Electrospray Ionization Tandem Mass Spectrometry Analysis of the Reactivity of Structurally Related Bromo-methyl-benzoquinones toward oligonucleotides. *Analytical Chemistry*. **2011**, *83*, 8145.
10. Anichina, J., Zhao, Y., Hrudey, S.E., Le, X.C., Li, X.F. Electrospray Ionization Mass Spectrometry Characterization of Interactions of Newly Identified Water Disinfection Byproducts Halobenzoquinones with Oligodeoxynucleotides. *Environmental Science and Technology*. **2010**, *44*, 9557.
11. Huang, R., Wang, W., Qian, Y., Boyd, J.M., Zhao, Y., Li, X.-F.; Ultra Pressure Liquid Chromatography–Negative Electrospray Ionization Mass Spectrometry Determination of Twelve Halobenzoquinones at ng/L Levels in Drinking Water. *Analytical Chemistry*. **2013**, *85*, 4520-4529.

12. Spencer, S.E., Santiago, B.G., Glish, G.L.; Miniature Flow-Through Low-Temperature Plasma Ionization Source for Ambient Ionization of Gases and Aerosols. *Analytical Chemistry*. **2015**, *87*, 11887-11892.
13. Harper, J.D., Charipar, N.A., Mulligan, C.C., Zhang, X., Cooks, R.G., Ouyang, Z.; Low-Temperature Plasma Probe for Ambient Desorption Ionization. *Analytical Chemistry*. **2008**, *80*, 9097-9104.
14. Garcia-Reyes, J.F., Harper, J.D., Salazar, G.A., Charipar, N.A., Ouyang, Z., Cooks, R.G.; Detection of Explosives and Related Compounds by Low-Temperature Plasma Ambient Ionization Mass Spectrometry. *Analytical Chemistry*. **2011**, *83*, 1084-1092.
15. Krylov, E.V., Coy, S.L., Vandermeij, J., Schneider, B.B., Covey, T.R., Nazarov, E.G.; Selection and generation of waveforms for differential mobility spectrometry. *The Review of Scientific Instruments*. **2010**, *81*, 024101.
16. Isenberg, S.L., Armistead, P.M., Glish, G.L.; Optimization of Peptide Separations by Differential Ion Mobility Spectrometry. *Journal of the American Society for Mass Spectrometry*. **2014**, *25*, 1592-1599.
17. Frisch, M.J., Trucks, G.W., Schlegel, H.B., Scuseria, G.E., Robb, M.A., Cheeseman, J.R., et al.; Gaussian 09. **2009**.
18. Guin, P.S., Das, S., Mandal, P.C.; Electrochemical Reduction of Quinones in Different Media: A Review. *International Journal of Electrochemistry*. **2011**, *2011*, 22.
19. Wightman, R. M. Kinetic Investigation of the Protonation of Quinone Radical Anions Doctoral Dissertation, University of North Carolina at Chapel Hill. **1974**.
20. Wightman, R.M., Cockrell, J.R., Murray, R.W., Burnett, J.N., Jones, S.B.; Protonation Kinetics and Mechanism for 1,8-Dihydroxyanthraquinone and Anthraquinone Anion Radicals in Dimethylformamide Solvent. *Journal of the American Chemical Society*. **1976**, *98*, 2562-2570.
21. Krylov, E.V., Coy, S.L., Nazarov, E.G.; Temperature Effects in Differential Mobility Spectrometry. *International Journal of Mass Spectrometry*. **2009**, *279*, 119-125.
22. Barnett, D.A., Belford, M., Dunyach, J.-J., Purves, R.W.; Characterization of a Temperature-Controlled FAIMS System. *Journal of the American Society for Mass Spectrometry*. **2007**, *18*, 1653-1663.
23. Lehmann, M.W., Evans, D.H.; Anomalous Behavior in the Two-Step Reduction of Quinones in Acetonitrile. *Journal of Electroanalytical Chemistry*. **2001**, *500*, 12-20.

24. Zare, H.R., Eslami, M., Namazian, M., Coote, M.L.; Experimental and Theoretical Studies of Redox Reactions of o-Chloranil in Aqueous Solution. *Journal of Physical Chemistry B*. **2009**, *113*, 8080-8085.
25. Frontana, C., Vázquez-Mayagoitia, Á., Garza, J., Vargas, R., González, I.; Substituent Effect on a Family of Quinones in Aprotic Solvents: An Experimental and Theoretical Approach. *The Journal of Physical Chemistry A*. **2006**, *110*, 9411-9419.
26. Pegis, M.L., Roberts, J.A.S., Wasylenko, D.J., Mader, E.A., Appel, A.M., Mayer, J.M.; Standard Reduction Potentials for Oxygen and Carbon Dioxide Couples in Acetonitrile and N,N-Dimethylformamide. *Inorganic Chemistry*. **2015**, *54*, 11883-11888.

Chapter 5

A Program to Calculate Theoretical Properties of Molecules and Ions

5.1 Introduction

Theoretical calculations have been used for years to aid in the understanding of mass spectrometry data. However recent years have seen an increase in the utility of these calculations to better understand mass spectrometry experiments. This increase in use may be attributed to the use of new experimental methods related to identifying different ion structures in mass spectrometric experiments. One type of experiment that uses theoretical calculations are ion mobility experiments [1]. In an ion mobility experiment, ions are separated based on their shape to charge ratio and the drift times of the different ions can be converted to collisional cross sections which is a numerical value that relates to the shapes of the ions. Theoretical calculations for these experiments involve the optimization of various ion structures that can then be used to calculate theoretical cross sections [2-6]. Thus experimental collisional cross sections can be compared to theoretical cross sections to give an idea of what the structure looks like in the gas phase. Another example experiment that makes use of theoretical calculations are ion spectroscopy experiments. Theoretical calculations can also be used to create theoretical vibrational spectra that can be compared to the experimental spectra, thus allowing one to determine the structure of an ion [7-9].

Before going into detail about the problem discussed in this work, a basic overview of electronic structure theory is presented [10-12]. Electronic structure theory applies

fundamental laws of quantum mechanics to predict energies and properties of chemical structures. This involves the use of approximations to make it computationally possible to solve quantum mechanical equations. The goal is to solve the electronic Schrödinger equation. Choosing a computational model to solve the Schrödinger equation is dependent on the size of the molecule being studied as well as the computational power available to the user. The system chosen consists of a theoretical model to determine how to solve the Schrödinger equation and a basis set which is a set of wave functions used to describe the shape of atomic orbitals. For this work either a Hartree-Fock (HF) [13, 14] or density functional theory (DFT) [15] model was used as the theoretical model and split valence basis sets were used to describe the atomic orbitals [16]. A more thorough description of electronic structure theory can be found in Appendix 1.

This work focuses on fundamental properties of molecules and ions that can aid in determining what ions are observed in mass spectrometry experiments. For this work, the interest is in determining basic energetics and thermochemical values for gas-phase ions. The program is designed to make use of the Gaussian suite of programs [17]. A chemical formula is entered into the program where the program will access the NIST webbook database to find all chemicals of that specific formula. The names, 2D, and 3D structures are extracted from the NIST database. The 3D structure is then converted to a Gaussian input file and a series of calculations are run on each of the different molecules. The neutral molecules are then protonated and the protonated ions are run through the same series of calculations. Results from the calculations of molecules and ions are used to calculate a set of theoretical values of importance: ionization energy, proton affinity, and heats of formation. These values can then be used to aid in determining what species are observed in

mass spectrometry experiments. The main objective of using this program is to dramatically decrease the time that a user must spend on making structures, setting up calculations, handling inputs/outputs of calculations, and calculating the theoretical values of interest. Doing all of this manually would require a substantial amount of time.

For the past few years our group has been interested in the analysis of cigarette tobacco. This project started at the beginning by first understanding a basic major component of cigarette tobacco, cellulose. Cellulose is pyrolyzed and then the pyrolysate is sampled using a variety of ion sources. A mass spectrum of pyrolyzed cellulose is rather complex and hard to interpret with a peak present at essentially all odd mass-to-charge ratios (Figure 5.1). Analyzing the aerosolized cellulose with a high mass accuracy mass spectrometer (Fourier Transform Ion Cyclotron Resonance – FTICR) allows the chemical formula for all of the mass-to-charge ratio peaks to be determined. A table of these chemical formulas obtained from a high mass accuracy mass spectrum of cellulose can be found in Appendix 2. Considering mass accuracy error limits this analysis showed a total of 512 molecular species. However, each of these peaks could correspond to multiple compounds. Theoretical calculations could greatly help in this analysis; however, this would require a large number of calculations and a lot of theoretical data to process. Considering that for each chemical formula there are least 20 different compounds (some chemical formulas can have up to 50 different compounds) that could have that formula now there essentially over 10,000 molecular species to do calculations on. Each species requires four calculations to determine optimized structures and energies for the neutral molecules, now there are a total of approximately 40,000 calculations needed. Then each of these neutral molecules can have multiple possible sites of protonation thus leading to a major increase in

the number of ions for which calculations are run. The increase in the number of calculations to be run is dependent on the number of protonation sites (number of oxygens), so if compound has four oxygens then there are 4 protonated structures for each neutral molecule to be run at each level of theory. So just for cellulose a few hundred thousand calculations would have to be run to have a full set of theoretical data.

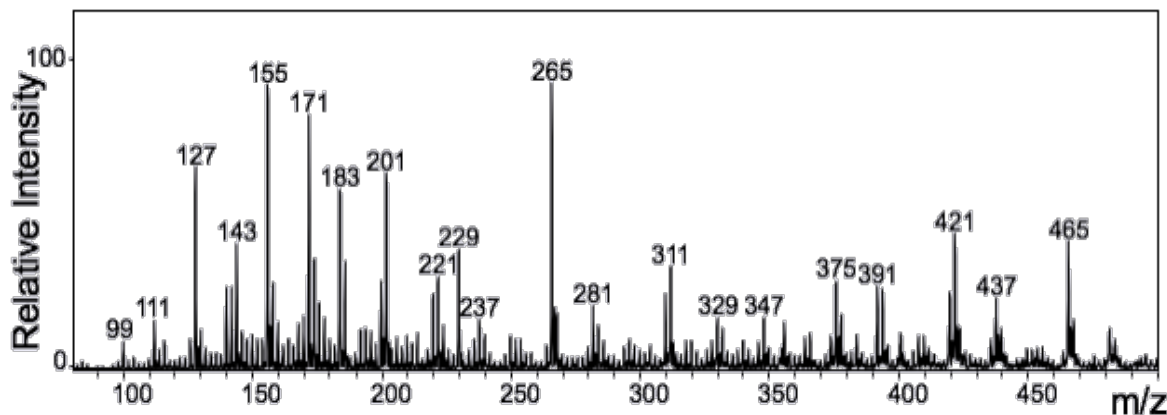


Figure 5.1. Example mass spectrum for pyrolyzed cellulose

The goal of this work is to use a program written mostly using Perl that takes care of all of the small tasks such as input file creation and file manipulations in an automated fashion. Thus a user only has to enter a chemical formula of interest and hit go and the program will run calculations for all molecules and ions and then output the desired theoretical data such as ionization energy, proton affinity, and heats of formation.

5.2 Experimental

The majority of the program is written using the Perl programming language which is a high-level, general purpose, interpreted, dynamic programming language. Perl was originally developed in 1987 to be used as a Unix scripting language. The language borrows some features from other popular programming languages such as C, shell script, AWK, and sed. Perl's power lies in its text processing abilities without the data-length limits of many Unix command line tools. Furthermore, the string parsing and regular expressions abilities

of Perl has greatly increased its use. Perl is heavily used in modern society including major websites that use Perl exclusively such as Priceline.com and Ticketmaster. The Perl program used in this work was composed using the Text Wrangler document editing software. This program makes extensive use of Perl modules. A Perl module is a discreet component that has program language built into the module that does a certain function. Modules serve as a way to do commonly used functions in Perl without having to write an entire new script to do these functions.

In addition to using Perl, the program makes use of some shell scripting language as well as a few Unix commands. Currently the program is designed to be run in a Unix environment. One benefit to this design is that it can be used in both Linux and Macintosh operating systems. Additionally, small modifications can be made to make the program capable of running on Windows operating systems. Windows is not capable of executing Unix commands so those commands would have to be converted into Perl code. Alternatively, the program could also be used in Windows using a program called Cygwin which creates a Linux-like environment in Windows.

The Gaussian 03 suite of programs was used for all ab initio and DFT calculations [17]. However the program is completely usable with the Gaussian 09 suite of programs with a few very simple modifications. All calculations were carried out on the KillDevil supercomputer cluster at UNC-Chapel Hill. Structures were fully optimized at the HF/3-21G, B3LYP/3-21G, B3LYP/6-31G levels, and B3LYP/6-31G(d,p) level in sequential order. After optimization at the B3LYP/6-31G(d,p) level the resulting structure was submitted to frequency analysis at the same level of theory. Energy values from the

frequency calculations can then be used to calculate the desired theoretical values of ionization energy, proton affinity, and heats of formation.

5.3 Results and Discussion

This program consists of seven main components:

1. Graphical user interface
2. Extracting important information from the NIST Webbook
3. Conversion of structure files to Gaussian input files using Babel and adding in Gaussian header information
4. Communicating with the KillDevil supercomputer to start calculation and how to know when calculations are finished running
5. Extracting structure geometries from Gaussian output files
6. Changing molecules into protonated ions
7. Extraction of important data from Gaussian output files and calculation of important energies

Each of these seven main components will be discussed in detail in the following sections.

The full program code can be found in Appendix 3. Besides the credentials needed to access the KillDevil supercomputer the only starting piece of information that is needed to run the program is the chemical formula that the user is interested in running calculations on. This chemical formula is saved as an input variable and is used throughout the entire program.

5.3.1 Graphical User Interface

To make this program as user friendly as possible a Perl script was written using Tk widgets to create a graphical user interface. The user must enter the command “./GUI.pl” at the command line of a terminal shell (in the directory where the code is located) and the

graphical user interface will appear (Figure 5.2). The graphical user interface requires the user to enter four pieces of information: the chemical formula to run calculations on, the user name for access to the KillDevil supercomputer, the password associated with that username, and an email address where the user can be informed that the program has finished running. The full Perl script used to create the graphical user interface can be found in Appendix 4.

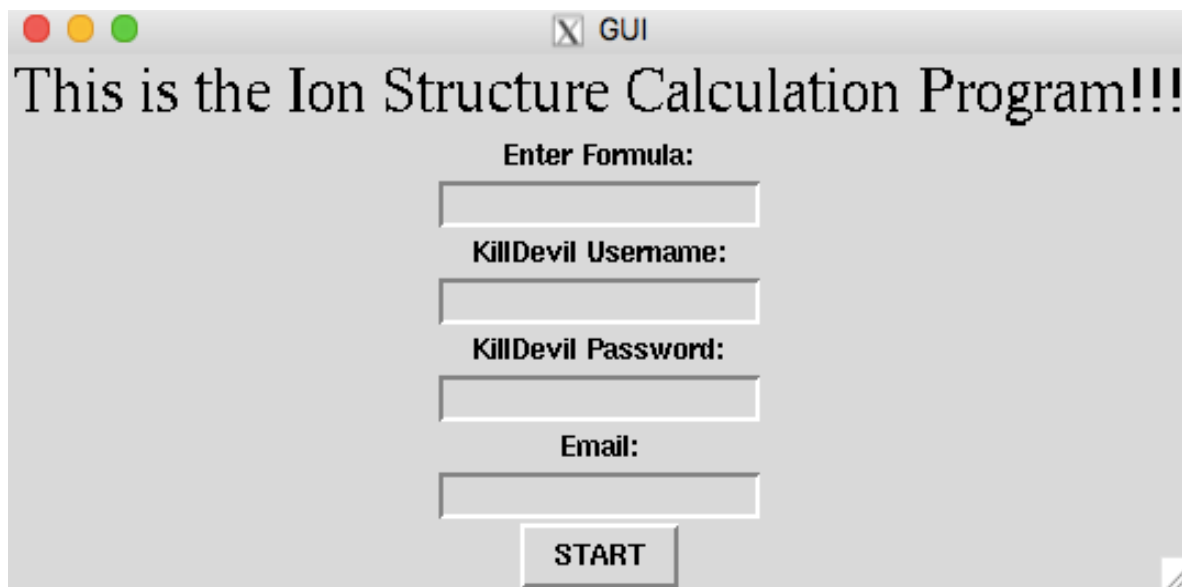


Figure 5.2. Graphical User Interface (GUI) for user input

5.3.2 *Extraction of Important Information from NIST Webbook*

The first component of this program concerns the use of the NIST Webbook as data source for the starting point of our calculations. The NIST webbook is a database of chemical properties. The database is used to determine the possible compounds for a particular chemical formula. The database is then accessed for each of those possible compounds to extract out the files for the 2D and 3D chemical structures.

This data is extracted from the NIST Webbook using a technique called web scraping. Web scraping is computer software technique that extracts information from websites. The program makes use of the Perl module WWW::Mechanize to do this web scraping. The first step is understanding the setup of the NIST Webbook website to understand how the pages of

interest can be accessed. For example, consider that the user is interested in looking at the different possible compounds that have the chemical formula C_6H_{12} . Entering in the formula in the NIST Webbooks search query pulls up a website with a url of, <http://webbook.nist.gov/cgi/cbook.cgi?Formula=C6H12&NoIon=on&Units=SI>. Of importance to note here is the format of the url. The url for every chemical formula is almost the same, the only difference is the chemical formula, in this example that is C_6H_{12} . So to access the webpage for a different chemical formula all that has to be done is to place a different chemical formula in the url.

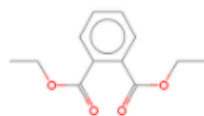
The first step in the program was to initialize the WWW::Mechanize module, line 25 in Figure 5.3. Next the website to be accessed is stored as a variable (line 26 Figure 5.3). Finally the WWW::Mechanize module was used to access the url of interest and the information from that url is stored as a new variable (line 27 Figure 5.3).

```
25 $m = WWW::Mechanize->new();  
26 $url = 'http://webbook.nist.gov/cgi/cbook.cgi?Formula='.$formula.'&NoIon=on&Units=SI';  
27 $m->get($url);
```

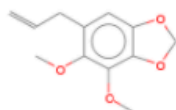
Figure 5.3. Using Perl module WWW::Mechanize to access the NIST Webbook

Next step is to access the information for all the compounds that have the specific chemical formula of interest. Figure 5.4 shows what the NIST Webbook website looks like for the chemical formula of $C_{12}H_{14}O_4$, one of the molecular formulas observed in the mass spectrum obtained from the pyrolysis of cellulose. Figure 5.4 shows just 6 of the 38 compounds that are listed in the NIST database.

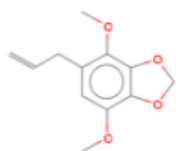
1. Diethyl Phthalate ($C_{12}H_{14}O_4$)



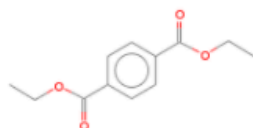
2. 1,3-Benzodioxole, 4,5-dimethoxy-6-(2-propenyl)- ($C_{12}H_{14}O_4$)



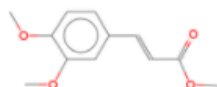
3. Apiol ($C_{12}H_{14}O_4$)



4. 1,4-Benzenedicarboxylic acid, diethyl ester ($C_{12}H_{14}O_4$)



5. 2-Propenoic acid, 3-(3,4-dimethoxyphenyl)-, methyl ester ($C_{12}H_{14}O_4$)



6. 1,3-Benzenedicarboxylic acid, diethyl ester ($C_{12}H_{14}O_4$)

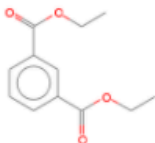


Figure 5.4. Example of NIST Webbook for compounds with formula $C_{12}H_{14}O_4$

The important thing to note from this webpage is that all of the different compounds have hyperlinks that take you to the webpage that contains information for that specific compound. To get that information, the name of each compound and the webpage url for each of those compounds is stored (Figure 5.5). The first line of this section of code (line 34) initializes the “find all links” function in the WWW::Mechanize module. The rest of this section of code then runs an if loop that accesses each link on the webpage and stores that url as a variable in an array. It then extracts the name of the compound (which is the text that is hyperlinked)

and the webpage url for each compound and then prints the name and url to output files called name.txt and output.txt.

```
34 my @links = $m->find_all_links();
35 for my $link ( @links ) {
36   if (index($link->url, 'cgi') != -1){
37     chdir "/Volumes/MAIN/Dropbox/DMC/Research//RJR/$formula" or die "$1";
38     open my $output, '>>', 'output.txt' or die "$1";
39     open my $output_name, '>>', 'name.txt' or die "$1";
40     printf "%s, webbook.nist.gov%s\n" , $link->text, $link->url;
41     printf $output "%s, webbook.nist.gov%s\n" , $link->text, $link->url;
42     printf $output_name "%s\n" , $link->text;
43   }
44 }
45 }
```

Figure 5.5. Web scraping url and compound name from NIST Webbook

The next step is to extract the identifier that the NIST Webbook uses to identify all of these different compounds. This can be figured out by understanding the format of the url for a specific compound. The webpage for the compound cyclohexane, has a url of

<http://webbook.nist.gov/cgi/cbook.cgi?ID=C110827&Units=SI>. The piece of information that is of interest is the ID number for the compound which appears in the url after “ID=”.

The next section of code (Figure 5.6) first goes through the array of compound url’s and separates each variable in that array (lines 50-54). For each variable (url) the program runs an if loop that searches for the ID number of the compound, which is easy to find since the ID number is found in a string between the string “ID=” and “&Units” (line 55). This ID number is then stored as a variable that can be used to access the desired information (line 57). The loop continues to run, accesses and saves the url of the compound with current ID number saved in the variable (lines 62-63). Next the loop continues to run and saves the url where the files for the 2D and 3D chemical structures are stored (lines 68 and 72). Finally, the loop runs and accesses the url of the 2D and 3D chemical structures and saves those files

to be used later (lines 79-80). The loop then goes back to the beginning where it will do the exact same set of steps for the next compound url stored in the initial array.

```

50 open my $fh, '<', 'output.txt';
51
52 my $ID;
53 chomp(my @lines = <$fh>);
54 for my $lines ( @lines ) {
55     if ( $lines =~ /ID=(.*)&Units/ ) {
56
57         $ID = $1;
58         open my $output_ID, '>>', 'ID.txt' or die "$1";
59         open my $output_site, '>>', 'site.txt' or die "$1";
60
61         print "$ID\n";
62         $m2 = WWW::Mechanize->new();
63         $url2 = 'http://webbook.nist.gov/cgi/cbook.cgi?ID=.'.$ID.'&Units=SI';
64         $m2->get($url2);
65         print "$url2\n";
66         printf $output_ID "$ID\n" ;
67         printf $output_site "$url2\n";
68         my $m3 = WWW::Mechanize->new();
69         $url3 = 'http://webbook.nist.gov/cgi/cbook.cgi?Str2File=.'.$ID.'';
70         $m3->get($url3);
71         print "$url3\n";
72         $m4 = WWW::Mechanize->new();
73         $url4 = 'http://webbook.nist.gov/cgi/cbook.cgi?Str3File=.'.$ID.'';
74         $m4->get($url4);
75         print "$url4\n";
76
77         my $file3 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $ID . '.mol';
78         my $file4 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $ID . '.sdf';
79
80         getstore($url3,$file3);
81         getstore($url4,$file4);

```

Figure 5.6. Accessing url for each compound and saving 2D and 3D chemical structures

To summarize, at this point the program has accessed the NIST webbook webpage for certain chemical formula and extracted all the names and url's to access the webpages for all the compounds with that specific chemical formula. The ID number (specific to the NIST database) for each of those compounds is then extracted. Finally, the webpages are accessed for the different compounds and the files of the 2D and 3D chemical structures for each compound are saved. The next step in the program is to prepare the different compounds for Gaussian calculations.

5.3.3 Conversion of NIST 3D structure files to Gaussian input files

The 3D chemical structures from the NIST Webbook are in a format called a structure data file (.sdf) which can not be used as an input to Gaussian. Therefore, all of the 3D chemical structures from the NIST Webbook must be converted into a format that Gaussian recognizes. This conversion can be accomplished using a program called Open Babel [18]. Open Babel is an open source chemical toolbox that can be used to search, convert, analyze, and store data from theoretical modeling programs. The conversion of chemical structures to a format usable with Gaussian is very straightforward making use of a Unix command line input (Figure 5.7). The first line of code in Figure 5.7 defines a variable which is essentially the command to execute on the Unix command line. The command essentially calls the Open Babel program tells it to convert all files of the structure data file (.sdf) format to the Gaussian input file (.com) format. The second line of code calls the system function to execute the command saved in the variable.

```
84 my $cmd = 'babel '.$ID.'.sdf '.$ID.'.com ';  
85 system($cmd);
```

Figure 5.7. Using Open babel to convert 3D chemical structures from NIST Webbook into Gaussian input file format

Now the header information required to tell the Gaussian program what type of calculation to run and how to run that calculation on the KillDevil supercomputer is added. The code to create this header is found in Figure 5.8. Lines 100-101 assigns the number of processors to use on the supercomputer to run the calculation to a variable (\$processors). Lines 102 and 103 give the amount of memory that the supercomputer can use to another variable (\$mem). Line 104 saves to a variable (\$key) the information that tells what type of calculation Gaussian should run and the level of theory and basis set to use. In line 105 the title of the calculation (the NIST Webbook ID number) is saved as another variable (\$title). The final

seven lines (106-112) opens a file called header.txt and then prints all of the variables for the header information into that file.

```
100     $proc = 4;
101     $processors = '%nprocshared='.$proc.'';
102     $mem = 40;
103     $memory = '%mem='.$mem.'MW';
104     $key = '# opt hf/3-21g';
105     $title = ''.$ID.'';
106     open my $input, '>>', 'header_'.$ID.'.txt' or die "$1";
107     print $input "$processors\n";
108     print $input "$memory\n";
109     print $input "$key\n";
110     print $input "\n";
111     print $input "$title\n";
112     print $input "\n";
113
```

Figure 5.8. Creating header information for Gaussian input file

The final step in preparing the Gaussian input file is to combine the header information (header.txt) with the Gaussian input structure file (.com). This combination of files is achieved using a separate Perl program called *merge_files.pl*. Basically the *merge_files.pl* program opens a new file and then copies the information from the header information file (header.txt) and the Gaussian input structure file (.com) to the new file and saves it. Again this makes use of a Unix command line input (Figure 5.9). This section of code functions identically to the code used to convert files using Open Babel that was previously described.

```
119 my $cmd2 = 'perl merge_files.pl header_'.$ID.'.txt '.$ID.'_1.com '.$ID.'_2.com';
120 system($cmd2);
```

Figure 5.9. Using Perl program *merge_files.pl* to combine header and geometry information to create a complete Gaussian input file

An example of a complete file (for a compound with chemical formula C₁₄H₁₄O₄) ready be used for an input for a Gaussian calculation is shown in Figure 5.10. The next step is to

transfer the files that are ready for input to the Gaussian program onto the KillDevil server, starting the calculations, and then monitoring the calculations until they are finished running.

```
%nprocshared=4
%mem=40MW
# opt hf/3-21g

C484311

0 1
C      -3.70540      -3.75480      -1.23480
C      -3.23380      -2.43830      -1.33200
C      -5.07760      -4.04570      -0.99260
O      -2.83780      -4.80270      -1.45820
C      -4.12840      -1.35640      -1.18710
C      -1.76930      -2.16190      -1.63920
C      -5.45090      -1.65130      -0.94610
C      -5.92400      -2.95680      -0.83900
O      -6.49110      -0.77200      -0.71790
O      -7.27380      -2.93570      -0.53110
O      -5.43760      -5.35830      -0.89580
C      -2.50270      -5.57850      -0.30450
C      -1.13260      -1.19280      -0.67580
C      -0.56060      -0.03730      -1.01750
C      -7.66830      -1.57560      -0.74640
C      -6.73640      -5.72880      -1.36220
H      -3.77720      -0.33240      -1.24970
H      -1.23060      -3.11520      -1.62540
H      -1.67530      -1.76710      -2.66040
H      -1.79020      -6.33400      -0.64470
H      -2.02920      -4.95330      0.46480
H      -3.38560      -6.06990      0.11540
H      -1.16060      -1.48400      0.37520
H      -0.10840      0.61880      -0.27850
H      -0.51570      0.29330      -2.05360
H      -8.15380      -1.48680      -1.73090
H      -8.34280      -1.26620      0.05510
H      -6.75130      -6.82050      -1.35150
H      -7.52690      -5.33980      -0.71420
H      -6.89920      -5.37130      -2.38650
```

Figure 5.10. Example of complete Gaussian input file. Blank lines around the title of the calculation and the blank line at the end of the file are necessary for correct submission to Gaussian

5.3.4 Communicating with KillDevil: Transferring Files, Starting Calculations, and

Monitoring Progress of Calculations

The first step in getting the Gaussian calculations ready to run is to transfer the input files to the KillDevil supercomputer. To accomplish this, the first task is to place all the

Gaussian input files in an archive file (tarball) (Figure 5.11). This section of code is again a Unix command line input and functions just like the other command line inputs previously discussed.

```
131 my $cmd3 = 'tar -cvvf '.$formula.'.tar /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/';  
132 system($cmd3);
```

Figure 5.11. Creating a tarball (archive of files, similarly to a zipped folder of files)

Next another Perl module, Net::OpenSSH, is used to transfer files securely from our local computer to the KillDevil supercomputer using the section of code in Figure 5.12. The first line of this code (line 136) assigns a variable to the supercomputer host (\$host). The username of the user (\$user), and the password for the user's account (\$password) are taken from user input via the graphical user interface. The next two lines (137-138) assigns variables for the location on our computer where the archived file is currently located (\$local_path) and the location of where to place the archived file on the supercomputer (\$host_path). Line 139 of the code initializes the Net::OpenSSH module and gives the modules information about the host, the username, and the password for that username. The next to last line of code (line 140) tells the Net::OpenSSH module to securely copy the archived file from the local path (user's computer) to the remote path (KillDevil supercomputer). The final line of code (line 141) in this section just tells the program to print an error message to the screen if the transfer of files does not happen correctly.

```
136 my $host = "killdevil.unc.edu";  
137 my $local_path = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'.tar/';  
138 my $remote_path = "/netscr/dcrizer";  
139 my $ssh = Net::OpenSSH->new(host => $host, user => $user, password => $password);  
140 $ssh->scp_put($local_path, $remote_path)  
141 or die "scp failed: " . $ssh->error;
```

Figure 5.12. Secure copy of files from user's computer to the KillDevil supercomputer

Now that the archived file of Gaussian input files has been moved to the KillDevil supercomputer the files then have to be unarchived to be able to use them. To communicate

with the KillDevil supercomputer another Perl module, Net::SSH::Perl is used. Figure 5.13 contains the section of code needed to unarchive the Gaussian input files and to move them to a new location to start Gaussian calculations. All lines of code in this section submits a command to run on the KillDevil supercomputer command line. Line 148 makes a new directory to place the Gaussian input files in. The second line (149) moves into the new directory and then unarchives the Gaussian input files. With the third line of code (line 150) these unarchived input files are moved into the directory where the Gaussian calculations will be run. Finally, the last line of code (line 151) moves the specific Gaussian input files to use to the appropriate directory. This last line of code is necessary due to the way in which the Gaussian input files were created. This was done by creating multiple text files that end in “.com”; however, calculations only need to be run on a subset of files that end in “.com”.

```

146 my $ssh = Net::SSH::Perl->new($host);
147 $ssh->login($user, $password);
148 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/.$user.'/'.$formula.'');
149 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/ && tar -xvf
... /netscr/.$user.'/'.$formula.'.tar');
150 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'
... /netscr/.$user.'/'.$formula.'');
151 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.$formula.'/'.$formula.'/*_2.com
... /netscr/.$user.'/'.$formula.'');

```

Figure 5.13. Commands to move Gaussian input files to appropriate directory on KillDevil

The next step is to submit all of the Gaussian input files to begin calculations. To do this a shell script, courtesy of Dr. Shubin Liu, is used that finds all the possible Gaussian input files and then executes a command on the KillDevil supercomputer command line to start each of these calculations (Figure 5.14). The first line of this code (line 3) lists all the files in the current directory and then copies all of the file names that end in “.com” to a file list. The rest of the code (lines 6-11) is a loop that goes through this list and then submits a command to the KillDevil supercomputer command line that starts a calculation for each input file.

```

1 #!/bin/sh -x
2
3 `ls -l *.com| awk '{print $9}' |sed 's/.com//g' > file_list`
4
5 cat file_list|while read each_file
6 do
7
8 bsub -q week -n 4 -R "span[ptile=4]" -M 40 -W 120 g03 $each_file".com"
9
10 done
11 /bin/rm file_list

```

Figure 5.14. Shell script to start Gaussian jobs on KillDevil

The final step in communicating with the KillDevil supercomputer is to monitor the progress of the calculation submitted to run and keep the program from moving forward until all calculations are finished (Figure 5.15). This is done via communicating with the KillDevil supercomputer using the Net::SSH::Perl module. The first line of code (line 165) submits a command to the KillDevil supercomputer command line that lists all the jobs currently running and prints it to file (output2.txt). The next section of the code is a loop that continuously runs. Line 167 makes the loop hold for five minutes which is necessary so that the program isn't constantly accessing the KillDevil server. The loop first deletes the file output2.txt (line 169) and then submits the command to list all the jobs currently running while printing to that output2.txt file (line 170). This file is then securely copied from the KillDevil supercomputer to the local computer where it is read (lines 171-175). This loop of code keeps running until that file of list of jobs currently running (output2.txt) is read to be empty which is what the "-Z" command in the last line of code does (line 176). At that point all calculations have finished running and the program can move on to the next step.

```

165 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output2.txt');
166 do{
167 sleep(300);
168 print "Running neutral compounds at HF/3-21G\n";
169 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user./ && rm output2.txt');
170 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user./ && bjobs -l > output2.txt');
171 my $local_path3 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
172 my $remote_path3 = '/netscr/.$user./output2.txt';
173 my $ssh3 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
174 $ssh3->scp_get($remote_path3, $local_path3)
175 or die "scp failed: " . $ssh3->error;
176 } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output2.txt");

```

Figure 5.15. Communicating with KillDevil to determine when the calculations have finished running

Now the first series of Gaussian calculations has finished running. The next step is to extract the optimized geometries from the output files (.log) to set up next series of calculations to run at the next level of theory.

5.3.5 Extracting Structure Geometries from Gaussian Output Files

Before extracting structure geometries from Gaussian output files the files must first get downloaded from the KillDevil supercomputer. The specifics of this process has been previously discussed. In short, the Gaussian output files are archived into a single file and then securely transferred from the KillDevil supercomputer to the local computer. These files are then unarchived on the local computer. A modified Perl script called *geometry_extract.plx* is used to extract the optimized geometries from the Gaussian output files. The full code from this script can be found in Appendix 5. This script is executed using the code found in Figure 5.16. This again is the same format of code previously discussed which submits a command to the Unix command line. The first line (line 218) tells Perl to run the script *geometry_extract.plx* on the Gaussian output files (.log) and then to copy and save those coordinates to a new file name “ID number”_3.com.

```

218 my $cmd5 = 'perl geometry_extract.plx < '.$ID.'_2.log > '.$ID.'_3.com';
219 system($cmd5);

```

Figure 5.16 Extracting optimized geometries from Gaussian output file into new file to be used for Gaussian calculation at next level of theory

The next step in the program is to go through processes that have been previously discussed in detail. The Gaussian header information is added to each of the new Gaussian input files and the only change is to change to the next highest level of theory in the \$key variable (opt b3lyp/3-21g). This set of input files is then moved to the KillDevil supercomputer where they are submitted to run Gaussian calculations. This process, geometry extraction, adding header information, moving files to KillDevil, and submit calculations to run, is repeated two more times with the variable \$key being changed to “opt b3lyp/6-31g” and then “opt freq b3lyp/6-31g(d,p)” for the last two calculations. The process of running calculations at increasing levels of theory is to decrease the number of failed calculations which is important due to the fact that the entire calculation process is automated.

This completes the Gaussian calculations for the neutral molecules. The next step in the program is to turn the molecules into protonated ions which will be run through the same Gaussian calculation flow.

5.3.6 Changing Neutral Molecules into Protonated Ions

Now that a neutral structure for each compound has been optimized at the highest level of theory in the calculation scheme the structures now need to be protonated. Since all compounds of interest only contain carbon, hydrogen, and oxygens atoms, the sites of protonation will be the oxygen atoms. The first step in creating protonated molecules is to find all of the oxygens in the structure. The code for this task is in Figure 5.17. The first part of this code is the if loop that results in all of the compound ID numbers being read from the output.txt file (lines 559-566). The important part of this code is the last two lines (567-568) where the Unix grep command is used to search through the compound’s geometry file to

find all lines that begin with “O” which signifies an oxygen atom. The grep command copies each line that corresponds to an oxygen atom which includes the Cartesian coordinates of the oxygen atom to a file that is saved as “ID Number_convert.txt”.

```

559 open my $fh, '<', 'output.txt';
560
561 my $ID;
562 chomp(my @lines = <$fh>);
563 for my $lines ( @lines ) {
564     if ( $lines =~ /ID=(.*)&Units/ ) {
565
566         $ID = $1;
567 my $cmd18 = 'grep -o "O.*" '.$ID.'_12.com > '.$ID.'_convert.txt';
568     system($cmd18);

```

Figure 5.17. Finding all available oxygens in the neutral structure

The next step in the conversion from neutral to protonated structure is to convert the information extracted about the oxygen atoms to the appropriate information to describe a hydrogen atom and its location in the structure (Figure 5.18). The first two lines of code open the “ID Number_convert.txt” file that contains a list of oxygen atoms in the structure and then separate each of these lines into separate variables in the array “@convertline” (lines 583-584). Next for each variable in this array (each oxygen atom) the x, y, and z coordinates for the oxygen atom are saved as the variables \$x, \$y, and \$z using the split function (line 586). The split function uses the spaces between each coordinate to separate the values from a string into the specific variable (\$x, \$y, and \$z). To convert to the appropriate coordinates for the added hydrogen atom the three dimensional distance formula is used.

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

The accepted value for the bond length of an OH bond is approximately 1 Å. So the sum of the squares of the differentials must be equal to 1 and solving this equation gives that each coordinate needs to be shifted by a value 0.58 to give a distance of 1 Å between the oxygen

and hydrogen atom. So a value of 0.58 is added to each of the variables \$x, \$y, and \$z to give the new variables \$d, \$e, and \$f (lines 588-593). Then a string is printed into a new file “ID Number_convertout.txt” that has the atom label from hydrogen followed by the coordinates for the hydrogen atom (\$d, \$e, and \$f) (lines 595-602).

```

583 open my $convertin, '<', '$ID.'_convert.txt';
584 chomp(my @convertline = <$convertin>);
585 for my $convertline(@convertline){
586     my ($w, $x, $y, $z) = split(/ /, $convertline);
587     print "H\t$x\t$y\t$z\n";
588     my $a = 0.58;
589     my $b = 0.58;
590     my $c = 0.58;
591     my $d = $x+$a;
592     my $e = $y+$b;
593     my $f = $z+$c;
594     print "H\t\t$d\t\t$e\t\t$f\n";
595     open my $convertout, '>>', '$ID.'_convertout.txt';
596     print $convertout "H\t\t";
597     my $junk1 = sprintf("%f", $d);
598     print $convertout "$junk1\t\t";
599     my $junk2 = sprintf("%f", $e);
600     print $convertout "$junk2\t\t";
601     my $junk3 = sprintf("%f", $f);
602     print $convertout "$junk3\n";

```

Figure 5.18. Converting coordinates for oxygens atoms into coordinates for hydrogen atom

Now the number of new protonated structures that will be created is dependent on the number of oxygen atoms in the molecule so this number has to be determined to run the loop to make the correct number of protonated structures (Figure 5.19). First, the number of oxygen atoms variable (\$lc) is set to a value of zero (line 619). Then the variable \$lc is increased by one for each line in the file “ID Number_convert.txt” by reading each line in a while loop (lines 621-622).

```

619 my $lc = 0;
620 open my $convertout2, '<', '$ID._convert.txt' or die($!);
621 $lc++ while <$convertout2>;
622 close $convertout2;

```

Figure 5.19. Determining the number of oxygens in the structure

Next a do loop is executed until all of the appropriate hydrogen atom coordinates are added to the input geometry file to create input geometry files for all of the possible protonated structures. As previously discussed, a header that contains the information to run the Gaussian calculation is added to the input geometry files. The only change from the calculations on the neutral species is the charge and multiplicity values must be changed from zero and one to one and one for the protonated species. The input files are then submitted to KillDevil supercomputer to run. As was the case for the neutral molecules each of the protonated species was optimized at the four levels of theory used in the calculation scheme.

5.3.7 *Changing Neutral Molecules into Molecular Ions*

The final set of calculations to be run are structures that have an electron removed to form the molecular ion ($M^{+\bullet}$). This requires a very simple change of the neutral molecules. The geometry of the molecule is extracted from the highest level of theory calculation for the neutral molecule. The spin and multiplicity values are then changed from zero and one to one and two. The molecular ion structures are then submitted to Gaussian calculations at each of the four levels of theory used previously for both the neutral and protonated structures. At this point all Gaussian calculations needed to calculate the properties of interest have been completed, and all that is left to do is to extract important energy values and calculate the properties of interest.

5.3.8 Extraction of Energies from Gaussian Output Files and Calculation of Important Properties

The first step in the data workup process is to move all needed Gaussian output files to a single folder. At this point, the program executes a command than runs another Perl program called *workup.pl* (Figure 5.20). The full code for this program can be found in Appendix 6.

```
1640 my $cmd50 = 'perl workup.pl '.$formula.'';  
1641 system($cmd50);
```

Figure 5.20. Command to start program to workup data from calculations

This program makes use of two Perl modules Excel::Writer::XLSX and Excel::Writer::XLSX::Utility for moving and processing data in Excel. The first part of the program sets up a worksheet to calculate ionization energies and proton affinities (Figure 5.21). The first two lines of code create a new Excel workbook with the name “Chemical Formula.xlsx” and a worksheet in that worksheet that is labeled as “Energies” (lines 13-14). Next the format command is used so that certain columns with the most important information can be highlighted by coloring the text red (lines 15-16). The rest of the code in this section is to create labels for the columns of data to be pulled into the Excel workbook (lines 17-23).

```
13 my $workbook = Excel::Writer::XLSX->new( '.$formula.'.xlsx );  
14 $worksheet = $workbook->add_worksheet('Energies');  
15 $format = $workbook->add_format();  
16 $format->set_color( 'red' );  
17 $worksheet->write( 'A1', 'ID Number' );  
18 $worksheet->write( 'B1', 'Website' );  
19 $worksheet->write( 'C1', 'Compound Name' );  
20 $worksheet->write( 'D1', 'Energy (Hartrees)' );  
21 $worksheet->write( 'E1', 'M+ Energy (Hartrees)' );  
22 $worksheet->write( 'F1', 'IE (Hartrees)' );  
23 $worksheet->write( 'G1', 'IE (kcal/mol)', $format );
```

Figure 5.21. Creating and formatting Excel workbook for data entry and processing

Next the ID number, the link to the NIST Webbook for the compound, and the name of the compound are placed into the Excel workbook for each compound. The code for this is the same for all of the information with just modifications to the file the information is being pulled from and where the information is placed in the Excel file. The code to do this for the ID Number is in Figure 5.22. The file “ID.txt” which contains a lists of all of the ID numbers is opened and each line is read into the array @lines (lines 34-40). The variable \$lines_ref is then created and each element in the array @lines is places into this variable so that the ID numbers can be copied into Excel (line 42). The last line of code (line 43) writes the variable \$lines_ref for each element in the array @lines into the first column of the Excel worksheet starting at the second row.

```

34 my $file = "ID.txt";
35 open (FH, "< $file") or die "Can't open $file for read: $!";
36 my @lines;
37 while (<FH>) {
38     push (@lines, $_);
39 }
40 close FH or die "Cannot close $file: $!";
41 print @lines;
42 $lines_ref = \@lines;
43 $worksheet->write_col( 1, 0, $lines_ref );

```

Figure 5.22. Extracting compound ID numbers and writing them into the Excel workbook

Now the energies from the Gaussian output files need to be stripped out and copied into Excel (Figure 5.23). The first two lines run another Perl program called “find_energy.plx” (lines 81-82), the complete code of this program can be found in Appendix 7. This program extracts the energies from the output files and prints each energy value to the file “energy.txt”. The next section of code reads each line of the “energy.txt” file and places each value into an array, @lines3 (lines 85-91). Finally, this array is copied into Excel into the third column beginning in the second row (line 94).

```

81 my $cmd1 = 'perl find_energy.plx < '.$ID.'_11.log';
82 system($cmd1);
85 my $file3 = "energy.txt";
86 open (FH, "< $file3") or die "Can't open $file3 for read: $!";
87 my @lines3;
88 while (<FH>) {
89     push (@lines3, $_);
90 }
91 close FH or die "Cannot close $file3: $!";
92 print @lines3;
93 $lines3_ref = \@lines3;
94 $worksheet->write_col( 1, 3, $lines3_ref );

```

Figure 5.23. Extracting total energy values from Gaussian output files and writing the values into Excel workbook

This section of code is done first for all of the neutral compounds and then for all of the molecular ions. These energy values can then be used to calculate the ionization energy of each compound (Figure 5.24). First the number of compounds that were run through the program needs to be determined so that the ionization energy calculation can be completed the correct number of times (lines 132-137). The file containing a list of the ID numbers is opened and each line is read. As each line is read the variable for the number of rows is increased by one each time. The ionization energy of a compound is defined as the energy required to remove the most loosely bound electron of the molecule to create a cation.

$$IE = E(cation) - E(neutral)$$

The calculation for ionization energy is done in the last section of code in Figure 5.24. First a formula is written into the Excel workbook in column F that subtracts the energy of the neutral molecule from the energy of the molecular cation (line 143). Then another formula is written into column G that converts the energy values from Hartrees (energy units used in Gaussian) to a more conventional energy unit in kcal/mol (line 144). The column that contains the ionization energy values in kcal/mol is highlighted by changing the font color to red.

```

132 open my $ID_out, '<', 'ID.txt' or die($!);
133 my $end = 1;
134 $end++ while <$ID_out>;
135 close $ID_out;
136 print "Number of rounds:\n";
137 print $end, "\n";
140 $row=2;
141 $realend=($end)+1;
142 do{
143 $worksheet->write( 'F'.$row.', '=E'.$row.'-D'.$row.' ');
144 $worksheet->write( 'G'.$row.', '=F'.$row.'*627.503', $format );
145 $row++;
146 }while($row < $realend);

```

Figure 5.24. Inserting formulas into Excel to calculate ionization energy

The next step is to calculate the proton affinities for the various compounds. The general process is the same as that for the calculation of ionization energies with a few small differences. The first important difference is that the number of oxygens has to be determined to make sure that the proton affinity calculation is done for each of the protonated structures that are generated from the neutral molecules. An important thing to be careful of in this case is to be sure that the file that is being read to determine the number of oxygens correctly actually has oxygens in it. The reason for this need is that if one of the compounds doesn't actually have a 3D structure in the NIST database the file that is checked to count oxygens would actually be empty since there isn't a structure. This problem is avoided using the first two lines of code in Figure 5.25. A file containing a list of the different oxygens in the compound is opened if the file opened is associated with a compound that doesn't have a structure then the file will be empty. So the command "if(-s \$fh1)" is used to make sure that the file isn't empty (line 161). A file that isn't empty can then be opened and the number of lines in the file corresponds to the number of oxygens in the molecule and as each line is read a variable is increased in count by one thus determining the number of protonated structures that exist for each compound (lines 164-167).

```

160 open my $fh11, '<', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/' . $ID . '_convert.txt';
161 if(!s $fh11)
162 {
163 open my $convertout2, '<', '' . $ID . '_convert.txt' or die($!);
164 my $lc = 0;
165 $lc++ while <$convertout2>;
166 close $convertout2;
167 print "Total number of lines:";

```

Figure 5.25. Determining number of protonated structures for each compound

Now the energies can be stripped out of the Gaussian output files and written into the Excel file essentially the same in which this was done for the calculation of ionization energies. The code to calculate the proton affinities is a little different from the ionization energies in that in the case of proton affinities each formula ran through the program has a different number of protonation sites therefore that must be accounted for making the code a little different (Figure 5.26). The first section of this code determines which columns the energies values for the neutral and various protonated species are in (lines 248-257). The “xl_rowcol_to_cell” function converts the row and column numbers into a single string which can be saved as a variable. For example, the cell in row zero and column zero is also known as the A1 cell. The proton affinity is defined as the energy released when a proton is added to a species and is calculated using the following equation.

$$PA = E(neutral) - E(protonated)$$

The end of the code in Figure 5.26 places a formula in the correct Excel column to subtract the energy of the protonated species from the energy of the neutral species (line 258). Again these energy values, as was the case for ionization energy, are converted from Hartrees to kcal/mol and the columns containing the proton affinities in kcal/mol are highlighted in red font (line 259).

```

248 $row=1;
249 do{
250 $row2=($row);
251 $str=xl_rowcol_to_cell( $row2, 3 );
252 print "$str\n";
253 $str2=xl_rowcol_to_cell( $row2, $column );
254 print "$str2\n";
255 $newcol2=($newcol1)-1;
256 $str3=xl_rowcol_to_cell( $row2, $newcol2);
257 print "$str2\n";
258 $worksheet->write( $row, $newcol, '='.$str.' - '.$str2.' ');
259 $worksheet->write( $row, $newcol1, '='.$str3.'* 627.503',$format );
260 $row++;
261 }while($row < $end);

```

Figure 5.26. Writing formulas into Excel to calculate proton affinity

An example of the Excel workbook that contains the summary of the compounds ID numbers, link to NIST webbook site for that compound, name of compound, all energies stripped out of Gaussian output files, ionization energies, and proton affinities can be found in Figure 5.27. As can be seen from these results a number of these compounds don't result in energy values. This is due to these compounds not having 3D structures in the NIST database.

	A	B	C	D	E	F
1	ID Number	Website	Compound Name	Energy (Hartrees)	M+ Energy (Hartrees)	IE (Hartrees)
2	C84662	http://webbook.nist.gov/cgi/cbook.cgi?ID=C84662&Units=SI	Diethyl Phthalate			#VALUE!
3	C484311	http://webbook.nist.gov/cgi/cbook.cgi?ID=C484311&Units=SI	1,3-Benzodioxole, 4,5-dimethoxy-6-(2-propenyl)-	-766.5347837	-766.2903966	0.244387143
4	C523808	http://webbook.nist.gov/cgi/cbook.cgi?ID=C523808&Units=SI	ApioI	-766.5351602	-766.2888823	0.246277884
5	C636099	http://webbook.nist.gov/cgi/cbook.cgi?ID=C636099&Units=SI	1,4-Benzenedicarboxylic acid, diethyl ester	-766.6651757	-766.344757	0.320418706
6	C5396645	http://webbook.nist.gov/cgi/cbook.cgi?ID=C5396645&Units=SI	2-Propenoic acid, 3-(3,4-dimethoxyphenyl)-, methyl ester	-766.5881969	-766.3272232	0.260973757
7	C636533	http://webbook.nist.gov/cgi/cbook.cgi?ID=C636533&Units=SI	1,3-Benzenedicarboxylic acid, diethyl ester			#VALUE!
8	C20803887	http://webbook.nist.gov/cgi/cbook.cgi?ID=C20803887&Units=SI	1,1-Ethanediol, 2-phenyl-, diacetate	-766.6363093	-766.3455351	0.290774241
9	R290625	http://webbook.nist.gov/cgi/cbook.cgi?ID=R290625&Units=SI	R,5,4'-Methoxy- α -pyrrolidinopropiophenone-M (desmethyl-3-methoxy-desamino-oxo-), ethylated	-766.593783	-766.3251587	0.268624255
10	U373892	http://webbook.nist.gov/cgi/cbook.cgi?ID=U373892&Units=SI	Methyl propyl phthalate			#VALUE!
11	C4046020	http://webbook.nist.gov/cgi/cbook.cgi?ID=C4046020&Units=SI	ethyl ferulate	-766.6162913	-766.3541189	0.26217236
12	U352415	http://webbook.nist.gov/cgi/cbook.cgi?ID=U352415&Units=SI	trans-2,4-Dimethoxycinnamic acid, methyl ester			#VALUE!
13	U307541	http://webbook.nist.gov/cgi/cbook.cgi?ID=U307541&Units=SI	Acetoxyacetic acid, 3,5-dimethylphenyl ester	-766.6535892	-766.3658291	0.287760133
14	C119450112	http://webbook.nist.gov/cgi/cbook.cgi?ID=C119450112&Units=SI	Acetoxycetic acid, methyl phenylmethyl ester	-766.6367351	-766.3461434	0.290591612
15	C131704	http://webbook.nist.gov/cgi/cbook.cgi?ID=C131704&Units=SI	1,2-Benzenedicarboxylic acid, monobutyl ester	-766.6481303	-766.3393452	0.308785083
16	R275106	http://webbook.nist.gov/cgi/cbook.cgi?ID=R275106&Units=SI	Fenproporex-M (desamino-oxo-HO-methoxy), AC	-766.6254632	-766.3574722	0.279909983
17	U355700	http://webbook.nist.gov/cgi/cbook.cgi?ID=U355700&Units=SI	Acetoxyacetic acid, 2,3-dimethylphenyl ester	-766.6525879	-766.3580271	0.294560761
18	R298436	http://webbook.nist.gov/cgi/cbook.cgi?ID=R298436&Units=SI	2-Propenoic acid, 3-(3,4,5-trimethoxyphenyl)-, methyl ester, cis	-766.5809152	-766.3282037	0.252711514
19	U355699	http://webbook.nist.gov/cgi/cbook.cgi?ID=U355699&Units=SI	Acetoxyacetic acid, 3-ethylphenyl ester	-766.6491769	-766.3568718	0.292305161
20	R298421	http://webbook.nist.gov/cgi/cbook.cgi?ID=R298421&Units=SI	2-Propenoic acid, 3-(3,4-dimethoxyphenyl)-, methyl ester, trans	-766.5858547	-766.3298997	0.255865034
21	R84007	http://webbook.nist.gov/cgi/cbook.cgi?ID=R84007&Units=SI	Methyl 4-(3,4-methylenedioxyphenyl)butanoate	-766.6139317	-766.3491832	0.264748518
22	R399534	http://webbook.nist.gov/cgi/cbook.cgi?ID=R399534&Units=SI	4-(1-Carboxyethyl)benzoic acid, dimethyl	-766.6499474	-766.3442454	0.305701971
23	R34384	http://webbook.nist.gov/cgi/cbook.cgi?ID=R34384&Units=SI	Trimethoxycinnamaldehyde	-766.5554467	-766.3028761	0.252570642
24	C3083355	http://webbook.nist.gov/cgi/cbook.cgi?ID=C3083355&Units=SI	isobutyl hydrogen phthalate			#VALUE!
25	C71369192	http://webbook.nist.gov/cgi/cbook.cgi?ID=C71369192&Units=SI	isopropyl methyl phthalate	-766.6420691	-766.3383364	0.303732663
26	U372945	http://webbook.nist.gov/cgi/cbook.cgi?ID=U372945&Units=SI	isopropyl 4-acetoxybenzoate			#VALUE!
27	R490850	http://webbook.nist.gov/cgi/cbook.cgi?ID=R490850&Units=SI	tyrosol acetate	-766.6545316	-766.3669755	0.287556155
28	C28153244	http://webbook.nist.gov/cgi/cbook.cgi?ID=C28153244&Units=SI	1,1-Diacetoxy-1-phenylethane	-766.651311	-766.3602209	0.291090132
29	C5600560	http://webbook.nist.gov/cgi/cbook.cgi?ID=C5600560&Units=SI	2,2'-(1,4-phenylene)bis-1,3-dioxolane			#VALUE!
30	C500025901	http://webbook.nist.gov/cgi/cbook.cgi?ID=C500025901&Units=SI	1,4-benzenediol dipropionate			#VALUE!
31	T999913563	http://webbook.nist.gov/cgi/cbook.cgi?ID=T999913563&Units=SI	3,5-dimethylbenzylmalonic acid	-766.6713204	-766.3857285	0.285591961
32	T999917872	http://webbook.nist.gov/cgi/cbook.cgi?ID=T999917872&Units=SI	1,4-benzenediol dipropionate	-766.6369544	-766.3708461	0.26610825
33	C101906	http://webbook.nist.gov/cgi/cbook.cgi?ID=C101906&Units=SI	Oxirane, 2,2'-(1,3-phenylenebis(oxyethylene))bis-	-766.5017426	-766.2214321	0.280310544
34	T999909374	http://webbook.nist.gov/cgi/cbook.cgi?ID=T999909374&Units=SI	DL-mono-sec-butyl phthalate			#VALUE!
35	T999909396	http://webbook.nist.gov/cgi/cbook.cgi?ID=T999909396&Units=SI	D-mono-sec-butyl phthalate			#VALUE!
36	B6006780	http://webbook.nist.gov/cgi/cbook.cgi?ID=B6006780&Units=SI	3-Methyl-1,4-benzodioxan-2-carboxylic acid, ethyl ester			#VALUE!
37	B6000663	http://webbook.nist.gov/cgi/cbook.cgi?ID=B6000663&Units=SI	Octatetraene dicarboxylic acid, dimethyl ester			#VALUE!
38	C36076252	http://webbook.nist.gov/cgi/cbook.cgi?ID=C36076252&Units=SI	P-benzenediacetic acid, dimethyl ester	-766.6457723	-766.3506489	0.295123398
39	U308750	http://webbook.nist.gov/cgi/cbook.cgi?ID=U308750&Units=SI	1-(4-Acetoxy-3-methoxyphenyl)-2-propanone	-766.6175315	-766.3466792	0.270852305

G	H	I	J	K	L	M	N	O
IE (kcal/mol)	H+ ₀ Energies (Hartrees)	H+ ₁ Energies (Hartrees)	H+ ₂ Energies (Hartrees)	H+ ₃ Energies (Hartrees)	PA_0 Energies (Hartrees)	PA_0 Energies (kcal/mol)	PA_1 Energies (Hartrees)	PA_1 Energies (kcal/mol)
#VALUE!	-766.9617102	-766.922274	-766.9788654	-766.9660875	#VALUE!	#VALUE!	#VALUE!	#VALUE!
153.3536654	-766.8759154	-766.8478619	-766.8826774	-766.8715647	0.341131697	214.0611633	0.313078211	196.4575166
154.540111	-766.8644684	-766.879438	-766.8855241	-766.8998822	0.329308137	206.6418439	0.344277789	216.0353454
201.0636993	-766.9502871	-767.0059657	-766.9624812	-767.0097179	0.285111414	178.9082676	0.340790036	213.84677
163.7618154	-766.9251422	-766.9251471	-766.9191725	-766.919207	0.33694527	211.4341678	0.336950138	211.4372224
#VALUE!	-766.9251422	-766.9251471	-766.9191725	-766.919207	#VALUE!	#VALUE!	#VALUE!	#VALUE!
182.4617086	-766.9835529	-766.9762543	-766.9987053	-767.0045556	0.347243649	217.8964315	0.339944995	213.3165042
168.5625259	-766.9547748	-766.9094538	-766.9283893	-766.867213	0.360991865	226.5234783	0.315670882	198.0844255
#VALUE!	-766.9547748	-766.9094538	-766.9283893	-766.867213	#VALUE!	#VALUE!	#VALUE!	#VALUE!
164.5139424	-766.9349565	-766.9349497	-766.9529174	-766.9820403	0.318665266	199.9634104	0.318658401	199.9591026
#VALUE!	-766.9349565	-766.9349497	-766.9529174	-766.9820403	#VALUE!	#VALUE!	#VALUE!	#VALUE!
180.5703467	-767.0057529	-766.9911283	-766.9993005	-766.9992697	0.352163684	220.9837682	0.337539087	211.8067897
182.3471083	-767.009846	-766.9917787	-766.9295883	-766.9853208	0.373110993	234.1282674	0.355043679	222.7909737
193.7635659	-767.0164817	-767.0133885	-766.9997444	-767.0026896	0.368351382	231.1415973	0.365258199	229.2006156
168.1651458	-766.9784325	-766.9754262	-766.9340686	-766.9645112	0.352969294	221.4892909	0.349962963	219.6028092
184.8377612	-766.9975711	-766.9842843	-766.9889357	-767.0035718	0.344983243	216.4780199	0.331696482	208.1405375
158.5722332	-766.9486091	-766.9522954	-766.9194281	-766.919437	0.367693835	230.7289845	0.371380117	233.0421376
183.4223654	-766.9858763	-766.9858889	-766.985787	-766.9997143	0.336699374	211.2798673	0.336711926	211.2877437
160.5560764	-766.9191675	-766.9191689	-766.9268582	-766.9268642	0.333312774	209.1547656	0.333314163	209.1556372
166.1304893	-766.9417652	-766.9224268	-766.9205345	-766.9268642	0.3278335	205.7165048	0.308495135	193.5816227
191.8289039	-766.993902	-766.9939048	-766.9619988	-766.9619839	0.343954645	215.8325716	0.343957392	215.8342954
158.4888356	-766.9404473	-766.8827163	-766.8829324	-766.8456647	0.385000578	241.5890177	0.327269648	205.3626859
#VALUE!	-766.9404473	-766.8827163	-766.8829324	-766.8456647	#VALUE!	#VALUE!	#VALUE!	#VALUE!
190.5931572	-767.007709	-767.0359505	-767.0359779	-767.0179101	0.365639899	229.4401335	0.393881423	247.1617746
#VALUE!	-767.007709	-767.0359505	-767.0359779	-767.0179101	#VALUE!	#VALUE!	#VALUE!	#VALUE!
180.4423499	-766.961172	-766.9615544	-766.9905063	-766.9883151	0.306640427	192.4177879	0.307022826	192.6577444
182.6599311	-767.0131296	-767.0206137	-767.0131227	-766.9883151	0.361818648	227.0422871	0.369302731	231.7385716
#VALUE!	-767.0131296	-767.0206137	-767.0131227	-766.9883151	#VALUE!	#VALUE!	#VALUE!	#VALUE!
#VALUE!	-767.0131296	-767.0206137	-767.0131227	-766.9883151	#VALUE!	#VALUE!	#VALUE!	#VALUE!
179.2098123	-767.0131296	-767.0081214	-767.0119374	-767.0081139	0.341809197	214.4862965	0.336800934	211.3435965
166.9837252	-766.9735016	-766.9792935	-766.9922005	-766.9632835	0.336547211	211.1843845	0.342339179	214.8188618
175.8957073	-766.819546	-766.8240477	-766.7903266	-766.8511177	0.317803359	199.4225612	0.322305089	202.2474103
#VALUE!	-766.819546	-766.8240477	-766.7903266	-766.8511177	#VALUE!	#VALUE!	#VALUE!	#VALUE!
#VALUE!	-766.819546	-766.8240477	-766.7903266	-766.8511177	#VALUE!	#VALUE!	#VALUE!	#VALUE!
#VALUE!	-766.819546	-766.8240477	-766.7903266	-766.8511177	#VALUE!	#VALUE!	#VALUE!	#VALUE!
#VALUE!	-766.819546	-766.8240477	-766.7903266	-766.8511177	#VALUE!	#VALUE!	#VALUE!	#VALUE!
185.1908176	-766.9638202	-766.9822707	-766.9966209	-766.9822952	0.318047859	199.5759857	0.336498342	211.1537191
169.9606339	-766.922274	-766.9788654	-766.9660875	-766.9617102	0.30520848	191.5192368	0.361333888	226.7380987

	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE
1	PA_2 Energies (Hartrees)	PA_2 Energies (kcal/mol)	PA_3 Energies (Hartrees)	PA_3 Energies (kcal/mol)												
2	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
3	0.347893637	218.3043009	0.336780985	211.3310784												
4	0.350963851	219.8543676	0.364822028	228.926917												
5	0.297305554	186.5601271	0.344542236	216.2012867												
6	0.330975544	207.6881468	0.331010061	207.7098063												
7	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
8	0.362395955	227.404549	0.368246324	231.075673												
9	0.334606335	209.966479	0.273430061	171.5781836												
10	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
11	0.336626097	211.2338857	0.36574902	229.5086073												
12	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
13	0.345711328	216.9348955	0.345680484	216.9155408												
14	0.292853294	183.7663205	0.34858579	218.738629												
15	0.351614086	220.6388938	0.35455934	222.4870495												
16	0.308605421	193.6508275	0.339048012	212.7536447												
17	0.336347866	211.059295	0.35098395	220.2434816												
18	0.338512859	212.4178346	0.338521717	212.423393												
19	0.336610018	211.2237961	0.350537346	219.9632362												
20	0.341003502	213.9807205	0.341009453	213.9844548												
21	0.306602806	192.3941806	0.312932518	196.3660938												
22	0.312051479	195.8132392	0.312036567	195.8038819												
23	0.327485694	205.4982554	0.290217984	182.1126556												
24	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
25	0.393908811	247.1789606	0.375841023	235.8413695												
26	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
27	0.335974674	210.8251159	0.333783451	209.4501169												
28	0.361811712	227.0379347	0.337004077	211.4710693												
29	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
30	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
31	0.340616986	213.7381806	0.336793499	211.338931												
32	0.355246169	222.9180368	0.326329126	204.7725056												
33	0.288584042	181.0873521	0.349375147	219.2339529												
34	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
35	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
36	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
37	#VALUE!	#VALUE!	#VALUE!	#VALUE!												
38	0.350848604	220.1585516	0.336522847	211.1690961												
39	0.34855596	218.7199106	0.344178702	215.973168												

Figure 5.27. Example output of calculation data for the compound C₁₂H₁₄O₄

5.3.9 Calculation of Heats of Formation

The final thing that is determined from these calculations is the heat of formation (or enthalpy of formation) for each of the compounds [19]. Heat of formation is defined as the

change in enthalpy as one mole of a molecule is formed from elements. This calculation uses a combination of the theoretical data that is generated from the Gaussian calculations in this program as well as some experimental data. The first value that needs to be calculated is the heat of formation for a molecule at 0 K.

$$\Delta_f H^\circ(M, 0K) = \sum_{atoms} x \Delta_f H^\circ(X, 0K) - \sum D_o(M)$$

$$\Delta_f H^\circ(M, 0K) = \sum_{atoms} x \Delta_f H^\circ(X, 0K) - (\sum_{atoms} x \varepsilon_o(X) - \varepsilon_o(M) - \varepsilon_{ZPE}(M))$$

where

$$x = \text{number of each atom}$$

$$\Delta_f H^\circ(X, 0K) = \text{heat of formation of the atoms at 0K}$$

$$\varepsilon_o(X) = \text{total energy of atoms}$$

$$\varepsilon_o(M) = \text{total energy of molecule}$$

$$\varepsilon_{ZPE}(M) = \text{zero - point energy of molecule}$$

The values for the heat of formation of the atoms and the total energy of the atoms are taken from experimental results from literature. The total energy of molecule and zero-point energy of molecule are taking from Gaussian calculations done in this program. The next calculation is the heat of formation at 298 K which is the value of interest.

$$\Delta_f H^\circ = \Delta_f H^\circ(M, 0K) + \left(H_M^\circ(298K) - H_M^\circ(0K) \right) - \sum_{atoms} x \left(H_X^\circ(298K) - H_X^\circ(0K) \right)$$

where

$$\Delta_f H^\circ(M, 0K) = \text{heat of formation of molecule at 0K}$$

$$H_M^\circ(298K) - H_M^\circ(0K) = \text{Enthalpy correction for molecule}$$

$$H_X^\circ(298K) - H_X^\circ(0K) = \text{Enthalpy correction for atomic elements}$$

The calculation for the heat of formation of the molecule at 0K was previously discussed.

The enthalpy correction for atomic elements is taken from experimental studies in the literature and the enthalpy correction for the molecule is determined using the results from Gaussian calculations.

The code to setup and calculate heats of formations begins by setting up the Excel worksheet (Figure 5.28). The first line of code creates a new worksheet in the workbook “Formula.xlsx” and labels it as “Enthalpies of Formation” (line 285). The rest of this code makes labels for each of the values that will be imported into the Excel worksheet (lines 286-309).

```
285 $worksheet2 = $workbook->add_worksheet('Ethalpies of Formation');
286 $worksheet2->write( 'A1', '# Carbon Atoms' );
287 $worksheet2->write( 'B1', '# Hydrogen Atoms' );
288 $worksheet2->write( 'C1', '# Oxygen Atoms' );
289 $worksheet2->write( 'D1', 'DH for Carbon' );
290 $worksheet2->write( 'E1', 'DH for Hydrogen' );
291 $worksheet2->write( 'F1', 'DH for Oxygen' );
292 $worksheet2->write( 'H1', 'DH(X,0K)', $format );
293 $worksheet2->write( 'I1', 'E for Carbon' );
294 $worksheet2->write( 'J1', 'E for Hydrogen' );
295 $worksheet2->write( 'K1', 'E for Oxygen' );
296 $worksheet2->write( 'M1', 'Sum of E for all atoms' );
297 $worksheet2->write( 'N1', 'Enthapy correction for Carbon' );
298 $worksheet2->write( 'O1', 'Enthapy correction for Hydrogen' );
299 $worksheet2->write( 'P1', 'Enthapy correction for Oxygen' );
300 $worksheet2->write( 'R1', 'Sum of Enthalpy corrections for atoms' );
301 $worksheet2->write( 'S1', 'Energy Converstion Factor (Hartrees to kcal/mol)' );
302 $worksheet2->write( 'A4', 'ID Number' );
303 $worksheet2->write( 'B4', 'Sum of Electronic and ZPE (Hartrees)' );
304 $worksheet2->write( 'C4', 'ZPE (Hartrees)' );
305 $worksheet2->write( 'D4', 'Thermal correction to Enthalpy (Hartrees)' );
306 $worksheet2->write( 'E4', 'Electronic Energy (Hartrees)' );
307 $worksheet2->write( 'F4', '298K - 0K Enthalpy correction (Hartrees)' );
308 $worksheet2->write( 'G4', 'Enthalpy of formation at 0K (kcal/mol)' );
309 $worksheet2->write( 'H4', 'Enthalpy of formation at 298K (kcal/mol)' );
```

Figure 5.28. Creating and setting up new worksheet in Excel workbook to calculate heats of formation

The next section of the program sets up the Excel worksheet with values from literature and constants that will be used for any formula run through this program (Figure 5.29). The first part of this code (lines 312-318) takes the formula input by the user at the

very beginning and strips out the number of each atom present and saves those values as variables (\$C, \$H, and \$O). The values for these variables are then written into the Excel worksheet (lines 320-322). Next the values for the heat of formation and the total energy of the different atoms are copied into Excel (lines 324-326 and 329-331). Then sums of these values for all atoms are calculated and placed into the Excel worksheet (line 327 and 332). The values for the enthalpy corrections for the atomic elements and the sum of these values for all of the atoms are then written into Excel (lines 334-337). Finally, the conversion factor to convert from Hartrees to kcal/mol is placed into a cell in the Excel worksheet (line 339). Now all values taken from the literature of experimentally determined values is in Excel and ready to be used for the heat of formation calculations.

```

312 $formula = ' '.$formula.'';
313 ($C)= $formula =~ /C(.*)H/;
314 print "$C\n";
315 ($H)= $formula =~ /H(.*)O/;
316 print "$H\n";
317 ($O)= $formula =~ /O(.*)/;
318 print "$O\n";
319
320 $worksheet2->write( 'A2', ' '.$C.' ' );
321 $worksheet2->write( 'B2', ' '.$H.' ' );
322 $worksheet2->write( 'C2', ' '.$O.' ' );
323
324 $worksheet2->write( 'D2', '169.98' );
325 $worksheet2->write( 'E2', '51.63' );
326 $worksheet2->write( 'F2', '58.99' );
327 $worksheet2->write( 'H2', '=( (A2*D2)+(B2*E2)+(C2*F2))',$format );
328
329 $worksheet2->write( 'I2', '-37.776009' );
330 $worksheet2->write( 'J2', '-0.500273' );
331 $worksheet2->write( 'K2', '-74.957398' );
332 $worksheet2->write( 'M2', '=( (A2*I2)+(B2*J2)+(C2*K2))',$format );
333
334 $worksheet2->write( 'N2', '0.25' );
335 $worksheet2->write( 'O2', '1.01' );
336 $worksheet2->write( 'P2', '1.04' );
337 $worksheet2->write( 'R2', '=( (A2*N2)+(B2*O2)+(C2*P2))',$format );
338
339 $worksheet2->write( 'S2', '627.503' );

```

Figure 5.29. Importing literature values needed to calculate heats of formation into the Excel worksheet

Next the ID numbers for all of the compounds processed by the program are placed into the Excel worksheet just like it was discussed for the “Energies” worksheet. Then the total energies, zero-point energies, and enthalpy correction for each compound is extracted from the Gaussian output files and written into the Excel worksheet. The extraction of total energies is done using the same code (find_energy.plx) discussed in the previous section. Zero-point energies are extracted using the program called find_zpe2.plx (Appendix 8) and the enthalpy corrections are extracted using the program called find_thermal.plx (Appendix 9).

The only thing left to do is to calculate the heats of formation at 0K and 298K (the value of interest) for each compound using the data that has been extracted into Excel (Figure 5.30). The first line of code writes a formula into Excel that calculates the total energy of the molecule by subtracting the zero-point energy of the molecule from the sum of the total energy and zero-point energy (line 458). Next the enthalpy correction for the molecule is determined by subtracting the zero-point energy from the thermal correction value (line 459). Now the heat of formation of the molecule at 0 K is calculated. First, the value of the total energy of the atomic elements minus the total energy of the molecule minus the zero-point energy of the molecule is determined and converted from Hartrees to kcal/mol. This value is then subtracted from the heat of formation of the atomic elements at 0 K to give the heat of formation of the molecule at 0 K (line 460). The final formula written into the Excel worksheet is the final calculation for the heat of formation at 298 K (line 461). Here both the sum of the enthalpy correction for the atoms and the enthalpy correction for the molecule is subtracted from the value of heat of formation of the molecule at 0 K.

```

454 $row=5;
455 $realend=($end)+1;
456
457 do{
458 $worksheet2->write( 'E'.$row.',', '=B'.$row.'-C'.$row.' ');
459 $worksheet2->write( 'F'.$row.',', '=D'.$row.'-C'.$row.' ');
460 $worksheet2->write( 'G'.$row.',', '=H2-(S2*(M2-E'.$row.'-C'.$row.''))');
461 $worksheet2->write( 'H'.$row.',', '=G'.$row.'+(S2*(F'.$row.'))-R2',$format);
462 $row++;
463 }while($row < $realend);

```

Figure 5.30. Inserting formulas into excel to calculated heats of formation at 0 K and 298 K

An example of this output in the Excel worksheet can be found in Figure 5.31, where the important information (the heat of formation of the molecule at 298 K) is highlighted in red font for each compound.

A	B	C	D	E	F	G	H	I	J	K
# Carbon Atoms	# Hydrogen Atoms	# Oxygen Atoms	DH for Carbon	DH for Hydrogen	DH for Oxygen		DH(X,OK)	E for Carbon	E for Hydrogen	E for Oxygen
12	14	4		169.98	51.63	58.99		2998.54	-37.776009	-0.500273
ID Number	Sum of Electronic and ZPE (Hartrees)	ZPE (Hartrees)	Thermal correction to Enthalpy (Hartrees)	Electronic Energy (Hartrees)	298K - OK Enthalpy correction (Hartrees)	Enthalpy of formation at OK (kcal/mol)	Enthalpy of formation at 298K (kcal/mol)			
C8462				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
C484311	-766.5347837	0.24191	0.258899	-766.7766937	0.016989	-1010.740906	-1021.380257			
C523808	-766.5351602	0.241885	0.258941	-766.7770452	0.017056	-1010.777151	-1021.57446			
C636099	-766.6651757	0.243829	0.260338	-766.9090047	0.016709	-1092.562249	-1103.377301			
C339645	-766.5881969	0.241785	0.259346	-766.8299819	0.017561	-1044.257848	-1054.538268			
C63653				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
C20803887	-766.6363093	0.241932	0.258655	-766.8782413	0.016723	-1074.448502	-1085.25477			
R290625	-766.593783	0.240713	0.258325	-766.834496	0.017612	-1047.763091	-1058.01508			
U373892				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
C4046020	-766.6162913	0.242809	0.259787	-766.8591003	0.016978	-1061.887133	-1072.533387			
U552415				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
U307541	-766.6335892	0.240167	0.258049	-766.8937562	0.017882	-1085.291686	-1095.370677			
C119450112	-766.6367351	0.243372	0.260381	-766.8804551	0.016661	-1074.715663	-1085.560836			
C131704	-766.6481303	0.243539	0.260359	-766.8916693	0.01682	-1081.866206	-1092.611606			
R275106	-766.6254632	0.240478	0.258033	-766.8659412	0.017555	-1067.642543	-1077.926728			
U355700	-766.625879	0.241006	0.258306	-766.8935939	0.0173	-1084.663349	-1095.107547			
R298436	-766.5809152	0.242106	0.259482	-766.8230212	0.017376	-1039.688567	-1050.080575			
U555699	-766.6491769	0.241741	0.258878	-766.8909179	0.017137	-1082.522986	-1093.09467			
R298421	-766.5858547	0.241495	0.259156	-766.8273497	0.017661	-1042.788112	-1053.005782			
R84007	-766.6139317	0.244142	0.260455	-766.8580737	0.016313	-1060.406473	-1071.470017			
R399534	-766.6499474	0.243185	0.260403	-766.8931324	0.017218	-1083.006429	-1093.502083			
R343484	-766.554467	0.241064	0.258546	-766.7965107	0.017482	-1023.706978	-1034.03697			
C3083535				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
C71369192	-766.6420691	0.242222	0.259331	-766.8842911	0.017109	-1078.062798	-1088.626849			
U372945				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
R490850	-766.6545316	0.241738	0.259099	-766.8962696	0.017361	-1085.883059	-1096.28898			
C28153244	-766.651311	0.24071	0.257788	-766.892021	0.017078	-1083.862107	-1094.44461			
C5660560				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
C500025901				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
T999913563	-766.6713204	0.242948	0.260055	-766.9142684	0.017107	-1096.418097	-1106.983403			
T999917872	-766.6369544	0.24063	0.257843	-766.8775844	0.017213	-1074.853285	-1085.352076			
C101906	-766.5017426	0.24311	0.259047	-766.7448526	0.015937	-990.0074964	-1001.306981			
T999909374				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
T999909396				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
B6006780				#VALUE!	#VALUE!	#VALUE!	#VALUE!			
B6000663				#VALUE!	#VALUE!	#VALUE!	#VALUE!			

Figure 5.31. Example Excel worksheet output for the heats of formation calculations for formula $C_{12}H_{14}O_4$

5.4 Conclusions

This chapter discusses a Perl program that was written to obtain theoretical data for all compounds in the NIST database that have a specific molecular formula. The program asks the user for the chemical formula to run calculations for and the information required to access the KillDevil supercomputer to run the calculations. The program accesses the NIST Webbook database to search for all compounds with that particular molecular formula. The 3D chemical structures are then extracted from the NIST webbook website and prepares Gaussian input files to run calculations. Neutral structures are then optimized at four levels of theory. The neutral species are then converted to protonated species, and those protonated species are optimized at the same four levels of theory. Then the program runs the same optimization calculations for the molecular cations of each neutral compound.

All needed energy values from the Gaussian calculations are then extracted from the Gaussian output files and written into an Excel workbook. In Excel these energy values are

used to calculate ionization energies, proton affinities, and heats of formation for all the possible compounds.

The significance of this program is the reduced amount of time that it requires the user to run all of these calculations and workup the data from the calculations into useful theoretical values. To do all the needed steps in the process by hand typically would require many hours of work. Additionally, the time it takes to run all of these calculations by hand for any formula can take up to a couple of days. This program allows the user to start a calculation and walk away. Within approximately 12-16 hours, depending on the size of the compounds, a user will have all the desired theoretical results for a formula. Furthermore, the fact that this whole process is completely automated requires no knowledge about electronic structure theory calculations for the user to be able to obtain quality theoretical data. Results for chemical formulas that have been run through this program can be found in Appendix 10.

5.5 REFERENCES

1. von Helden, G., Hsu, M.T., Kemper, P.R., Bowers, M.T.; Structures of Carbon Cluster Ions from 3 to 60 Atoms: Linears to Rings to Fullerenes. *The Journal of Chemical Physics*. **1991**, *95*, 3835-3837.
2. Wyttenbach, T., Helden, G.v., Batka, J.J., Carlat, D., Bowers, M.T.; Effect of the Long-Range Potential on Ion Mobility Measurements. *Journal of the American Society for Mass Spectrometry*. **1997**, *8*, 275-282.
3. Wyttenbach, T., Witt, M., Bowers, M.T.; On the Stability of Amino Acid Zwitterions in the Gas Phase: The Influence of Derivatization, Proton Affinity, and Alkali Ion Addition. *Journal of the American Chemical Society*. **2000**, *122*, 3458-3464.
4. von Helden, G., Hsu, M.T., Gotts, N., Bowers, M.T.; Carbon Cluster Cations with Up to 84 Atoms: Structures, Formation Mechanism, and Reactivity. *The Journal of Physical Chemistry*. **1993**, *97*, 8182-8192.
5. Mesleh, M.F., Hunter, J.M., Shvartsburg, A.A., Schatz, G.C., Jarrold, M.F.; Structural Information from Ion Mobility Measurements: Effects of the Long-Range Potential. *The Journal of Physical Chemistry*. **1996**, *100*, 16082-16086.
6. Shvartsburg, A.A., Jarrold, M.F.; An Exact Hard-Spheres Scattering Model for the Mobilities of Polyatomic Ions. *Chemical Physics Letters*. **1996**, *261*, 86-91.
7. Bakker, J.M., Aleese, L.M., Meijer, G., von Helden, G.; Fingerprint IR Spectroscopy to Probe Amino Acid Conformations in the Gas Phase. *Physical Review Letters*. **2003**, *91*, 203003.
8. Kapota, C., Lemaire, J., Maître, P., Ohanessian, G.; Vibrational Signature of Charge Solvation vs Salt Bridge Isomers of Sodiated Amino Acids in the Gas Phase. *Journal of the American Chemical Society*. **2004**, *126*, 1836-1842.
9. Simons, J.; Bio-Active Molecules in the Gas Phase. *Physical Chemistry Chemical Physics-Cambridge-Royal Society of Chemistry*. **2004**, *7*, E7-E7.
10. Jensen, F. *Introduction to Computational Chemistry*. John Wiley & Sons, **1999**.
11. Helgaker, T., Jorgensen, P., Olsen. *Molecular Electronic-Structure Theory*. J. John Wiley & Sons, **2000**.
12. Cramer, C.J., Bickelhaupt, F.; *Essentials of Computational Chemistry*. John Wiley & Sons. **2002**.
13. Hartree, D. R. The Wave Mechanics of an Atom with a Non-Coulomb Central Field. *Mathematical Proceedings of the Cambridge Philosophical Society*. **1928**, *24*, 111.

14. Slater, J.C.; Note on Hartree's method. *Physical Review*. **1930**, 35, 210.
15. Parr, R.G.; *Density Functional Theory of Atoms and Molecules*. Springer. **1980**.
16. Ditchfield, R., Hehre, W.J., Pople, J.A.; Self-Consistent Molecular-Orbital Methods. IX. An Extended Gaussian-Type Basis for Molecular-Orbital Studies of Organic Molecules. *The Journal of Chemical Physics*. **1971**, 54, 724-728.
17. Frisch, M.J., Trucks, G.W., Schlegel, H.B., Scuseria, G.E., Robb, M.A., Cheeseman, J.R., et al.; *Gaussian 09*. **2009**.
18. O'Boyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T., Hutchison, G.R.; Open Babel: An Open Chemical Toolbox. *Journal of Cheminformatics*. **2011**, 3, 33.
19. Ochterski, J.W.; *Thermochemistry in Gaussian*. Gaussian Inc. **2000**, 1-19.

Chapter 6

Summary and Future Directions

6.1 General Summary

The work presented in this dissertation is focused on the study of novel chemistries using a variety of techniques. First an introduction to these various techniques is given in Chapter 1. Fundamentals of the electron capture dissociation and electron transfer dissociation processes are studied in Chapter 2 with a focus on the effect that cation recombination energy has in the processes. The ability of differential ion mobility spectrometry to separate protomers (isomers differing in protonation site) is discussed in Chapter 3. The work in Chapter 4 is focused on the unique ion types observed in mass spectra of halobenzoquinones ionized via electrospray ionization and the role that electrochemical reduction plays in the formation of these ions. A program designed to automate the calculation of useful theoretical energy values is discussed in Chapter 5. This final chapter provides a summary of the results in each chapter as well as potential future directions that can follow from this work.

6.2 Cation Recombination Energy/Coulomb Repulsion Effects in Electron Capture Dissociation and Electron Transfer Dissociation

6.2.1 Summary

Chapter 2 discusses the effects of cation recombination energy and Coulomb repulsion in the electron capture dissociation (ECD) and electron transfer dissociation (ETD) processes. A series of doubly protonated model peptides of the form $X(AG)_nX$ were

subjected to ECD and ETD. Experimental variables changed in these experiments include the identity of the basic amino acids in the peptides ($X = K, R, \text{ or } H$), the length of the peptide ($n = 1, 2, \text{ or } 4$), and the reagent (near-thermal electrons, anions of azobenzene, or anions of 1,3-dinitrobenzene). By varying n for a given charge state, the effects due to changes in recombination energy were studied. By varying X , it was possible to follow effects due to sites of protonation. By varying reagents, it was possible to study effects due to changes in reagent electron affinity and reagent mass.

Results obtained in this work with both ECD and ETD are consistent with previous studies on the role of protonation site on product ion partitioning in ETD. The fact that proton transfer and electron transfer followed by hydrogen atom loss both lead to the same nominal products complicated interpretation of product channel partitioning in ETD studies. The ECD results were helpful in this regard because they showed that there was a tendency for hydrogen atom loss to correlate with charge per residue. The observed trends in the percentage of proton transfer or hydrogen atom loss were consistent with expectations. The relatively high percentage of proton transfer or hydrogen atom loss for the $K(AG)_nK$ peptides can be rationalized by the relatively high contribution of electron transfer followed by hydrogen atom loss observed in the ECD data. ECD appears to lead to higher degrees of hydrogen atom loss from ions of high charge per residue than does ETD. It is hypothesized that this could be due to the differences in angular momenta of the charge-reduced peptide ions generated by electron capture (low additional angular momentum) versus electron transfer (high additional angular momentum). The angular momentum effect may also enhance the likelihood of electron transfer with no dissociation relative to electron capture with no dissociation, although the data set generated here is likely too small to draw general

conclusions. Finally, the partitioning among the various c-type and z-types ion channels was very similar for ECD and ETD for most of the model peptides studied despite the differences in the electron capture and electron transfer processes.

6.2.2 Future Directions

A further investigation into the effect of angular momentum on the likelihood of electron transfer with no dissociation relative to electron capture with no dissociation is of interest. This would require the collection of ETD and ECD spectra for a large set of peptides. To draw general conclusions, it may be of use to generate a large set of peptides from a protein digest.

6.3 Separating Vitamin K₂ Protomers using Differential Ion Mobility Spectrometry

6.3.1 Summary

The work in Chapter 3 discusses the use of differential ion mobility spectrometry (DIMS) to separate different isomers of vitamin K₂ that differ by location of protonation site. Structures that are protonated at either oxygen are proposed. Theoretical calculations on these two structures show that they are close enough in energy to both be observed in mass spectrometry experiments. A DIMS-MS scan of sodiated vitamin K₂ had only a single peak. Theoretical calculations on the sodiated structures gave a large energy difference between the two structures thus agreeing with only one sodiated peak being observed experimentally.

Studies in the chapter showed that vitamin K₂ decomposes over time. Over a period of five days the DIMS-MS scan of vitamin K₂ changed drastically. The DIMS-MS scan after five days showed that the protonated ion of vitamin K₂ was no longer the most abundant species with loss of both a single and two hydrogen atoms from protonated vitamin K₂ being the more favored species. These species were confirmed using high mass accuracy MS.

6.3.2 Future Directions

The dissociation of the two different protonated structures into different product ions could be furthered studied using theoretical calculations. The energy surface between parent protonated ions and products ions could be studied to determine transition state structures. Determination of transition state structures may give some information that aids in determining why the different protonated structures of vitamin K₂ give rise to different product ions when subjected to collision induced dissociation. Additionally, the further investigation into the ability of DIMS to separate protomers (isomers that differ in protonation site) of different molecules is of interest.

6.4 Analysis of Halobenzoquinones – Mass Spectrometry, Differential Ion Mobility Spectrometry, Electrochemistry, and Theoretical Calculations

6.4.1 Summary

Unique ion types observed from halobenzoquinones were discussed in Chapter 4. The species observed are atypical for ESI generated ions. These species are observed due to the fact that they first undergo electrochemical reduction in solution prior to ionization. In the protic solvents used to dissolve the halobenzoquinones studied in Chapter 4, cyclic voltammetry experiments and theoretical calculations suggest that a two-electron, two-proton reductions takes place to form the [M+2H] species in solution. When ionized via ESI, this species loses a proton to form the [M+H]⁺ species. The [M+H]⁺ species was the most abundant species observed for tetraC-1,2-BQ. However, for the other two halobenzoquinones used in this study the M^{•+} species was observed in addition to the [M+H]⁺ species, with the M^{•+} species being most abundant. When higher energy was imparted to the system, via difference in voltage between the capillary exit and skimmer of the mass spectrometer or via DIMS experiments, eventually only the M^{•+} species was observed for

tetraC-1,4-BQ and 2,6-DCBQ. However, for tetraC-1,2-BQ the $[M+H]^+$ species remained prevalent throughout all experiments. Theoretical calculations showed that this was likely due to the hydrogen bonding that takes place in the tetraC-1,2-BQ molecule that is not available to the other halobenzoquinone compounds. Additionally, calculations also showed the energy required to remove a hydrogen atom to form the M^{\bullet} species from the $[M+H]^+$ species was approximately 15 kcal/mol higher for tetraC-1,2-BQ compared to tetraC-1,4-BQ.

6.4.2 Future Directions

An extension of the study contained in Chapter 4 would be to study more halobenzoquinone compounds. Looking at a more diverse set of halobenzoquinone compounds may give information that would allow for a more thorough understanding of the processes that are taking place to form these unique ion types observed. It is also of interest to see how well DIMS can separate these other halobenzoquinone compounds.

6.5 A Program to Calculate Theoretical Properties of Molecules and Ions

6.5.1 Summary

Chapter 5 involved the writing of a Perl program that prepare input files and controls Gaussian calculations to obtain theoretical data for all compounds in the NIST database that have a specific chemical formula. The program requires very little user input to run all calculations on the KillDevil supercomputer. Required information from the user is the chemical formula and the information needed to access the KillDevil supercomputer (username and password). The program accesses the NIST database and finds all compounds that have a particular molecular formula. The 3D chemical structures are extracted from the NIST webbook website, and then the program converts these structures into Gaussian input files. Neutral structures are optimized at four levels of theory. The neutral species are then

converted to protonated species which are optimized at the same four levels of theory. Next the program runs optimization calculations again at the same four level of theories for the molecular cations of each neutral compound.

All needed energy values are then extracted from the Gaussian output files and written into an Excel workbook. These energy values are then used to calculate ionization energies, proton affinities, and heats of formations for all of the possible compounds. This program is significant because it reduces the amount of time that it would require the user to run all of these calculations and then workup the data from these calculations into useful theoretical information. To do all of these steps by hand can take up to a couple of days for each chemical formula, but with this program the user can start a set of calculations, walk away, and within approximately 12 hours a user will have all the desired theoretical results for a specific chemical formula. Additionally, because this whole process is fully automated, no knowledge about electronic structure theory is needed by the user to obtain useful theoretical results.

6.5.2 Future Directions

The NIST database contains a relatively small number of compounds for each possible chemical formula so using this program with a bigger database is of interest. The database ChemSpider has a significantly higher number of compounds. However, ChemSpider does seem to have a limit to how much information you can pull from their website so it would be necessary to find a way to work around that issue. It may be possible to contact the administrators of the ChemSpider database and ask for permission to allow to access all of the information that the program would need to run. Switching to using the ChemSpider database would require some modifications to the program. The program

currently uses the ID numbers from the NIST database to name the various input files so all instances of using NIST ID numbers in the program would have to be changed to something that corresponds to the ChemSpider database.

Currently the program only works for compounds that contain carbon, hydrogen, and oxygen. A small modification will need to be made to make this program useable for compounds that also have nitrogen. Comments in the program file will aid future users of the program in how to go about doing this step.

Finally, the current supercomputer system at UNC, KillDevil, is going to be phased out in the coming year. The big issue with this is that the new supercomputer system, LongLeaf, will make use of a different cluster management and scheduling software. Thus some small modifications will need to be made to get this program to work on the new supercomputer.

APPENDIX 1

AN INTRODUCTION TO ELECTRONIC STRUCTURE THEORY

A1.1 Introduction

Electronic structure theory is concerned with the motion of electrons in atoms and molecules. It applies fundamental laws of quantum mechanics to predict energies and properties. Approximations are used to make it possible to solve difficult mathematical equations. This introduction will begin with a discussion on fundamental quantum mechanics. Following that will be a discussion on Hartree/Fock (HF) theory and density functional theory (DFT) which are the two theoretical models that are of most use to future Glish lab members. This discussion concludes with an introduction to basis sets which are used in conjunction with different theoretical models to form the complete method for theoretical calculations.

A1.2 Basic Quantum Mechanics

The Schrödinger equation governs systems of electrons and nuclei

$$\begin{aligned}\hat{H}\Psi &= E\Psi \\ \hat{H} &= \text{Hamiltonian of the system} \\ E &= \text{Energy} \\ \Psi &= \text{many particle wave function}\end{aligned}$$

The system obeys the time-dependent Schrödinger equation

$$\hat{H}\Psi = i\frac{\partial}{\partial t}\Psi(t)$$

The general Hamiltonian for this system is shown below.

$$\begin{aligned}\hat{H} &= \hat{T}_n + \hat{T}_e + \hat{V}_{ne} + \hat{V}_{ee} + \hat{V}_{nn} \\ \hat{T}_n &= \text{kinetic energy of nuclei} \\ \hat{T}_e &= \text{kinetic energy of electrons} \\ \hat{V}_{ne} &= \text{potential energy of electron – nuclei interactions} \\ \hat{V}_{ee} &= \text{potential energy of electron – electron interactions} \\ \hat{V}_{nn} &= \text{potential energy of nuclei – nuclei interaction}\end{aligned}$$

The Hamiltonian can be simplified by using the Born-Oppenheimer approximation. This approximation is based on the fact that nuclei are much heavier than electrons; meaning that nuclei react much slower to perturbations than do the electrons. Thus the electrons act instantaneously to nuclear motions and the nuclear and electronic motions can be decoupled. Thus the energies associated with the nuclei can be considered fixed and then only the electronic components of the Hamiltonian have to be considered. This simplifies the Hamiltonian as shown below.

$$\hat{H} = \hat{T}_e + \hat{V}_{ne} + \hat{V}_{ee}$$

This Hamiltonian is then used to solve the electronic Schrödinger equation. Exact solutions to the electronic Schrödinger equation can only be determined for one electron systems. Therefore, approximations must be made to solve the equation. The following sections discuss two different approximation methods that can be used to solve the electronic Schrödinger equation.

A1.3 Hartree/Fock Theory

Hartree/Fock theory (also known as self-consistent field theory) is a method used to make approximations to solve the electronic Schrödinger equation. HF Theory uses a wave function based approach to solve this equation. The HF method is referred to as an *ab initio* method meaning that it is based on first principles and does not include any experimental information other than fundamental constants. For this method, the wave function, Ψ_0 , is written as the Slater determinant of one electron wave functions.

$$\Psi_0 = \|\psi_1\psi_2 \dots \psi_n\|$$

The one electron wave functions, ψ_i , are expanded as the linear combinations of basis functions, ϕ_μ .

$$\psi_i = \sum_{\mu} c_{i\mu} \phi_{\mu}$$

The values for the above equation are determined by using the variational method. The variational method is a way of determining approximations of the ground state. It is based upon the variational principle and allows for the calculation of approximated wave functions. This essentially determines an upper limit of the true ground state method. The process is iterative and the steps taken in this process are summarized in Figure A1.1.

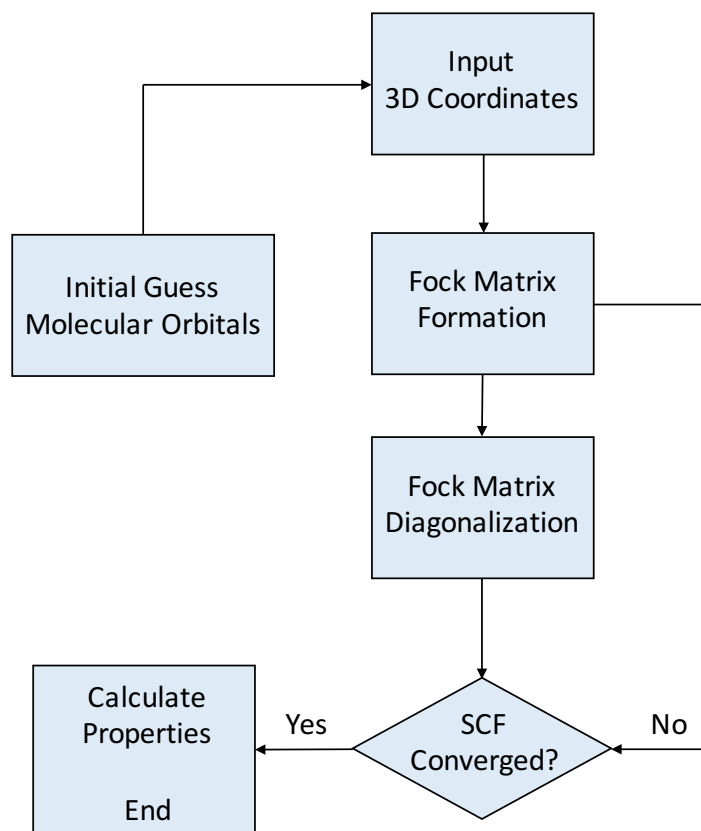


Figure A1.1. Schematic of the Hartree-Fock iterative process

HF theory takes into consideration the correlation between electrons that have the same spin.

However, it does not consider the correlation between electron motion of different spins.

This is the main drawback of HF theory. This issue has been solved with the development of density functional theory, which is discussed in the next section.

A1.4 Density Functional Theory

Density functional theory (DFT) is based on the Hohenberg-Kohn theorem. This theorem shows that the ground state properties of a many electron system can be determined by electron density, ρ . Using HF theory, an *ab initio* method, requires $3N$ coordinates, where N is the number of electrons, to determine the ground state energy. For DFT only three terms are needed to determine the ground state energy. This energy is a combination of kinetic energy, attraction between electrons and nuclei, and electron-electron repulsion.

$$E(\rho) = T(\rho) + V_{ne}(\rho) + V_{ee}(\rho)$$

The use of DFT for chemical systems was furthered by the ability to describe orbitals using the Kohn-Sham equations. These equations allow for the determination of electron density using a one-electron orbital.

$$\rho(\vec{r}) = \sum_1^N |\phi_i(\vec{r})|^2$$

This allows for the calculation of ground state energy using the equation below.

$$E(\rho) = \min_{\Psi} \langle \Psi | T + V_{ee} | \Psi \rangle + \int \rho(\vec{r}) V_{ne} d^3r$$

Ψ = wave function of the non – interacting reference system of Kohn – Sham theory

As was the case for HF theory, a set of orthogonal orbitals needs to be determined to minimize the energy. This again is an iterative process, much the same as that used in HF theory.

The advantage of DFT is that it requires a similar amount of computational cost when compared to HF theory, but it takes into account most of the electron correlation energy that

HF theory neglects. There are a variety of DFT methods available to use. It has been determined that for gas-phase ion chemistry the best performance has been determined to come from using the B3LYP hybrid method. This method includes Becke's three parameter exchange potential and the correlation functional of Lee, Yang, and Parr.

A1.5 Basis Sets

Basis sets are a group of wave functions that describes the shape of atomic orbitals. Molecular orbitals are computed using the selected theoretical model by these atomic orbitals. The level of approximation in a calculation is directly related to the basis set that is chosen. It must be noted that there is a trade-off between accuracy of calculations and the required computational cost. This discussion will focus on three different categories of basis sets: minimal, split valence or Pople, and correlation-consistent.

A1.5.1 Minimal Basis Sets

Minimal basis sets use one function for each atomic orbital. They are named according to the following nomenclature, STO-nG ($n=2,\dots,6$). This means that n gaussian type orbitals are used to describe one Slater type orbital (STO). Only one STO is used to describe each atomic orbital. Minimal basis sets can be used for qualitative results for very large molecules or quantitative results for very small systems like atoms.

A1.5.2 Pople Basis Sets

Pople basis sets are the choice that will work best for most applications in the Glish group. These basis sets allow for the specification of the number of Gaussian type orbitals (GTOs) to use for core and valence electrons separately. The notation of Pople basis is shown below:

K-LMG

Where K is the number of sp-type inner shell GTOs

L is the number of inner valence s-type and p-type GTOs
M is the number of outer valence s-type and p-type GTOs
And G indicates that GTOs are used

There are certain modifications that can be added to Pople basis sets to make them more accurate. Polarized functions can be added to allow atomic orbitals to distort from their original shape. When the user wants to add d-type functions to atoms other than hydrogen and f-type functions to transition metals they would use the additional code of (d) or * to the Pople basis set nomenclature discussed above. If the user wants to add p-type functions to atoms other than hydrogen, d-type functions to other atoms, and f-type functions to transition metals they would use the additional code of (d,p) or ** to the Pople basis set nomenclature. Diffuse functions can also be added to allow electrons to move far away from the nucleus. Adding + to the basis set nomenclature will add diffuse functions to atoms other than hydrogen and adding ++ to the basis set will add diffuse functions to all atoms.

A1.5.3 Correlation-consistent Basis Sets

Correlation-consistent basis sets are optimized using correlated wave functions.

These basis sets are denoted as follows:

cc-pVXZ
where cc indicates that is a correlation-consistent basis
pV indicates that it is a polarized valence basis
and XZ indicates the zeta number (X= D for double, T for triple, and Q for quadrupole)

Using these basis sets can become very computationally expensive. However, they can be used to give very accurate results for some systems.

APPENDIX 2

MOLECULAR FORMULAS DETERMINED BY HIGH MASS ACCURACY MASS SPECTROMETRY OF PYROLIZED CELLULOSE

<i>m/z</i>	Intensity	Formula	PPM Error
97.06	136.52	C ₆ H ₉ O	0.99
111.04	912.28	C ₆ H ₇ O ₂	1.26
111.08	247.52	C ₇ H ₁₁ O	1.03
113.06	1444.05	C ₆ H ₉ O ₂	1.41
125.06	1306.34	C ₇ H ₉ O ₂	0.62
125.10	131.76	C ₈ H ₁₃ O	1.15
127.04	694.02	C ₆ H ₇ O ₃	1.33
127.08	351.29	C ₇ H ₁₁ O ₂	1.20
137.06	1067.09	C ₈ H ₉ O ₂	0.63
137.10	376.11	C ₉ H ₁₃ O	1.01
139.08	2215.65	C ₈ H ₁₁ O ₂	0.91
139.11	160.21	C ₉ H ₁₅ O	0.30
141.05	584.88	C ₇ H ₉ O ₃	1.14
141.09	811.62	C ₈ H ₁₃ O ₂	0.81
143.03	219.87	C ₆ H ₇ O ₄	0.79
143.07	152.39	C ₇ H ₁₁ O ₃	0.22
147.04	601.15	C ₉ H ₇ O ₂	0.62
149.02	337.73	C ₈ H ₅ O ₃	0.96
149.06	1312.52	C ₉ H ₉ O ₂	1.17
149.10	557.38	C ₁₀ H ₁₃ O	0.36
151.04	525.79	C ₈ H ₇ O ₃	0.91
151.08	7690.09	C ₉ H ₁₁ O ₂	0.97
151.11	739.82	C ₁₀ H ₁₅ O	0.91
153.05	1231.12	C ₈ H ₉ O ₃	0.72
153.09	4592.20	C ₉ H ₁₃ O ₂	0.89
155.07	1854.34	C ₈ H ₁₁ O ₃	0.58
155.11	683.26	C ₉ H ₁₅ O ₂	1.11
157.05	602.49	C ₇ H ₉ O ₄	0.55
157.09	264.14	C ₈ H ₁₃ O ₃	0.10
159.07	184.90	C ₇ H ₁₁ O ₄	1.04
161.06	1215.40	C ₁₀ H ₉ O ₂	1.06
163.04	4587.49	C ₉ H ₇ O ₃	1.18
163.08	2542.41	C ₁₀ H ₁₁ O ₂	0.96
163.11	281.91	C ₁₁ H ₁₅ O	0.64
165.05	2398.73	C ₉ H ₉ O ₃	0.39
165.09	3474.44	C ₁₀ H ₁₃ O ₂	0.68
165.13	190.51	C ₁₁ H ₁₇ O	0.88

167.03	104.45	C ₈ H ₇ O ₄	0.32
167.07	2636.56	C ₉ H ₁₁ O ₃	0.42
167.11	2291.63	C ₁₀ H ₁₅ O ₂	0.43
169.05	1085.94	C ₈ H ₉ O ₄	0.95
169.09	2038.48	C ₉ H ₁₃ O ₃	0.96
169.12	273.34	C ₁₀ H ₁₇ O ₂	0.15
171.06	420.59	C ₈ H ₁₁ O ₄	2.09
171.07	607.98	C ₈ H ₁₁ O ₄	0.02
171.10	296.08	C ₉ H ₁₅ O ₃	0.11
173.08	122.51	C ₈ H ₁₃ O ₄	0.29
175.04	168.96	C ₁₀ H ₇ O ₃	0.39
175.08	1642.22	C ₁₁ H ₁₁ O ₂	0.26
175.11	100.46	C ₁₂ H ₁₅ O	0.05
177.05	8237.67	C ₁₀ H ₉ O ₃	0.33
177.09	1811.69	C ₁₁ H ₁₃ O ₂	0.80
179.07	4313.65	C ₁₀ H ₁₁ O ₃	0.74
179.11	1267.47	C ₁₁ H ₁₅ O ₂	0.91
181.05	1141.05	C ₉ H ₉ O ₄	0.52
181.09	4500.73	C ₁₀ H ₁₃ O ₃	0.62
181.12	702.70	C ₁₁ H ₁₇ O ₂	0.65
183.07	3959.79	C ₉ H ₁₁ O ₄	0.66
183.10	1491.79	C ₁₀ H ₁₅ O ₃	0.75
185.04	108.50	C ₈ H ₉ O ₅	1.01
185.08	1068.39	C ₉ H ₁₃ O ₄	0.20
185.12	242.75	C ₁₀ H ₁₇ O ₃	0.43
187.08	312.43	C ₁₂ H ₁₁ O ₂	0.79
187.10	128.70	C ₉ H ₁₅ O ₄	0.33
189.05	1031.58	C ₁₁ H ₉ O ₃	0.34
189.09	1629.37	C ₁₂ H ₁₃ O ₂	0.24
191.07	5720.77	C ₁₁ H ₁₁ O ₃	1.20
191.11	1156.11	C ₁₂ H ₁₅ O ₂	0.50
193.05	1203.65	C ₁₀ H ₉ O ₄	0.63
193.09	5572.59	C ₁₁ H ₁₃ O ₃	0.68
193.12	1799.57	C ₁₂ H ₁₇ O ₂	0.65
195.07	6764.56	C ₁₀ H ₁₁ O ₄	0.17
195.10	7918.21	C ₁₁ H ₁₅ O ₃	1.03
195.14	124.76	C ₁₂ H ₁₉ O ₂	0.64
197.04	316.67	C ₉ H ₉ O ₅	0.65
197.08	10850.94	C ₁₀ H ₁₃ O ₄	0.14
197.12	1022.63	C ₁₁ H ₁₇ O ₃	0.77
199.06	531.86	C ₉ H ₁₁ O ₅	0.59
199.10	1530.74	C ₁₀ H ₁₅ O ₄	1.15
201.08	184.93	C ₉ H ₁₃ O ₅	0.57
201.09	299.51	C ₁₃ H ₁₃ O ₂	0.87
203.07	904.77	C ₁₂ H ₁₁ O ₃	0.01

203.11	752.08	C ₁₃ H ₁₅ O ₂	0.67
205.05	430.91	C ₁₁ H ₉ O ₄	0.08
205.09	3615.72	C ₁₂ H ₁₃ O ₃	0.84
205.12	646.97	C ₁₃ H ₁₇ O ₂	0.46
207.10	8493.91	C ₁₂ H ₁₅ O ₃	1.16
207.14	2285.56	C ₁₃ H ₁₉ O ₂	1.07
207.16	391.48	C ₁₀ H ₂₃ O ₄	0.11
209.04	354.04	C ₁₀ H ₉ O ₅	0.76
209.08	15518.97	C ₁₁ H ₁₃ O ₄	1.19
209.12	11460.12	C ₁₂ H ₁₇ O ₃	0.30
211.06	3734.38	C ₁₀ H ₁₁ O ₅	0.68
211.10	13622.09	C ₁₁ H ₁₅ O ₄	0.33
211.13	602.11	C ₁₂ H ₁₉ O ₃	0.07
213.08	1883.29	C ₁₀ H ₁₃ O ₅	1.11
213.11	1111.85	C ₁₁ H ₁₇ O ₄	0.08
215.05	125.91	C ₉ H ₁₁ O ₆	1.09
215.07	314.47	C ₁₃ H ₁₁ O ₃	0.33
215.09	393.62	C ₁₀ H ₁₅ O ₅	0.79
215.11	173.41	C ₁₄ H ₁₅ O ₂	1.31
217.09	828.07	C ₁₃ H ₁₃ O ₃	0.87
217.12	301.76	C ₁₄ H ₁₇ O ₂	0.02
219.06	775.58	C ₁₂ H ₁₁ O ₄	1.28
219.10	1949.31	C ₁₃ H ₁₅ O ₃	1.37
219.14	582.35	C ₁₄ H ₁₉ O ₂	0.07
221.04	119.41	C ₁₁ H ₉ O ₅	0.40
221.08	8171.38	C ₁₂ H ₁₃ O ₄	0.18
221.12	8561.14	C ₁₃ H ₁₇ O ₃	1.25
221.15	1874.50	C ₁₄ H ₂₁ O ₂	0.92
223.06	1739.99	C ₁₁ H ₁₁ O ₅	0.52
223.10	22557.76	C ₁₂ H ₁₅ O ₄	0.32
223.13	8826.11	C ₁₃ H ₁₉ O ₃	1.11
225.04	150.22	C ₁₀ H ₉ O ₆	0.83
225.08	7762.30	C ₁₁ H ₁₃ O ₅	0.17
225.11	12125.35	C ₁₂ H ₁₇ O ₄	0.82
225.15	365.30	C ₁₃ H ₂₁ O ₃	0.05
227.05	780.60	C ₁₀ H ₁₁ O ₆	0.89
227.09	2984.85	C ₁₁ H ₁₅ O ₅	0.08
227.13	903.49	C ₁₂ H ₁₉ O ₄	0.61
227.16	183.59	C ₁₃ H ₂₃ O ₃	1.08
229.07	441.63	C ₁₀ H ₁₃ O ₆	0.20
229.09	322.26	C ₁₄ H ₁₃ O ₃	0.14
229.11	495.10	C ₁₁ H ₁₇ O ₅	0.72
229.12	172.92	C ₁₅ H ₁₇ O ₂	0.65
231.06	211.09	C ₁₃ H ₁₁ O ₄	1.15
231.09	158.29	C ₁₀ H ₁₅ O ₆	0.15

231.10	947.43	C ₁₄ H ₁₅ O ₃	0.33
231.14	256.82	C ₁₅ H ₁₉ O ₂	0.86
233.08	1308.07	C ₁₃ H ₁₃ O ₄	0.38
233.12	2715.03	C ₁₄ H ₁₇ O ₃	1.05
233.15	666.45	C ₁₅ H ₂₁ O ₂	0.27
235.06	416.31	C ₁₂ H ₁₁ O ₅	0.31
235.10	7303.13	C ₁₃ H ₁₅ O ₄	1.19
235.13	6393.82	C ₁₄ H ₁₉ O ₃	0.61
235.17	912.96	C ₁₅ H ₂₃ O ₂	0.02
237.08	6552.71	C ₁₂ H ₁₃ O ₅	0.64
237.11	14175.27	C ₁₃ H ₁₇ O ₄	0.34
237.15	4177.66	C ₁₄ H ₂₁ O ₃	0.02
239.05	1017.34	C ₁₁ H ₁₁ O ₆	0.07
239.09	10645.52	C ₁₂ H ₁₅ O ₅	0.10
239.13	5995.46	C ₁₃ H ₁₉ O ₄	0.09
239.16	150.44	C ₁₄ H ₂₃ O ₃	1.48
240.09	1161.20	C ₁₉ H ₁₂	4.92
241.07	2390.90	C ₁₁ H ₁₃ O ₆	0.48
241.11	2691.07	C ₁₂ H ₁₇ O ₅	0.87
241.14	366.12	C ₁₃ H ₂₁ O ₄	1.23
242.07	274.77	C ₁₈ H ₁₀ O	4.58
243.05	140.39	C ₁₀ H ₁₁ O ₇	1.02
243.05	201.32	C ₁₀ H ₁₁ O ₇	1.93
243.09	952.83	C ₁₁ H ₁₅ O ₆	0.13
243.10	549.56	C ₁₅ H ₁₅ O ₃	0.74
243.12	221.47	C ₁₂ H ₁₉ O ₅	0.71
243.14	149.25	C ₁₆ H ₁₉ O ₂	1.56
245.07	176.80	C ₁₀ H ₁₃ O ₇	0.34
245.08	793.99	C ₁₄ H ₁₃ O ₄	0.20
245.08	191.94	C ₁₄ H ₁₃ O ₄	4.63
245.10	264.46	C ₁₁ H ₁₇ O ₆	0.24
245.12	1279.58	C ₁₅ H ₁₇ O ₃	0.09
245.15	201.58	C ₁₆ H ₂₁ O ₂	0.07
246.06	338.69	C ₈ H ₆ O ₂ N ₈	0.16
247.06	389.10	C ₁₃ H ₁₁ O ₅	0.07
247.08	148.10	C ₁₀ H ₁₅ O ₇	0.08
247.10	2546.56	C ₁₄ H ₁₅ O ₄	0.42
247.13	2571.33	C ₁₅ H ₁₉ O ₃	0.87
247.17	508.80	C ₁₆ H ₂₃ O ₂	1.28
249.08	1021.82	C ₁₃ H ₁₃ O ₅	0.26
249.11	6371.46	C ₁₄ H ₁₇ O ₄	0.86
249.15	3787.24	C ₁₅ H ₂₁ O ₃	0.43
249.18	407.68	C ₁₆ H ₂₅ O ₂	0.04
251.05	323.04	C ₁₂ H ₁₁ O ₆	1.09
251.09	4706.79	C ₁₃ H ₁₅ O ₅	0.10

251.13	7535.73	C ₁₄ H ₁₉ O ₄	0.21
251.16	2014.66	C ₁₅ H ₂₃ O ₃	0.47
253.07	2839.33	C ₁₂ H ₁₃ O ₆	0.99
253.11	4957.25	C ₁₃ H ₁₇ O ₅	0.56
253.11	917.76	C ₁₃ H ₁₇ O ₅	4.06
253.14	2269.57	C ₁₄ H ₂₁ O ₄	0.08
255.05	276.85	C ₁₁ H ₁₁ O ₇	0.93
255.09	2524.59	C ₁₂ H ₁₅ O ₆	1.34
255.10	143.16	C ₁₆ H ₁₅ O ₃	2.39
255.12	1221.68	C ₁₃ H ₁₉ O ₅	0.17
256.09	335.54	C ₁₉ H ₁₂ O	4.21
257.07	601.57	C ₁₁ H ₁₃ O ₇	0.77
257.08	347.19	C ₁₅ H ₁₃ O ₄	0.90
257.10	800.92	C ₁₂ H ₁₇ O ₆	0.50
257.12	770.51	C ₁₆ H ₁₇ O ₃	0.61
257.14	128.14	C ₁₃ H ₂₁ O ₅	0.18
257.15	162.80	C ₁₇ H ₂₁ O ₂	0.28
259.06	158.55	C ₁₄ H ₁₁ O ₅	0.83
259.08	390.96	C ₁₁ H ₁₅ O ₇	0.73
259.10	1469.26	C ₁₅ H ₁₅ O ₄	0.06
259.13	1286.82	C ₁₆ H ₁₉ O ₃	0.58
259.17	134.48	C ₁₇ H ₂₃ O ₂	1.18
261.08	1004.91	C ₁₄ H ₁₃ O ₅	0.34
261.11	2969.74	C ₁₅ H ₁₇ O ₄	0.43
261.15	1897.69	C ₁₆ H ₂₁ O ₃	0.47
261.18	179.33	C ₁₇ H ₂₅ O ₂	0.49
263.06	178.02	C ₁₃ H ₁₁ O ₆	0.29
263.09	2279.92	C ₁₄ H ₁₅ O ₅	0.89
263.13	4425.83	C ₁₅ H ₁₉ O ₄	0.43
263.16	1670.46	C ₁₆ H ₂₃ O ₃	1.53
263.20	129.96	C ₁₇ H ₂₇ O ₂	0.99
264.09	299.90	C ₂₁ H ₁₂	3.82
265.07	651.30	C ₁₃ H ₁₃ O ₆	0.22
265.11	3728.88	C ₁₄ H ₁₇ O ₅	0.45
265.14	3362.55	C ₁₅ H ₂₁ O ₄	1.09
265.18	708.39	C ₁₆ H ₂₅ O ₃	0.09
265.18	470.48	C ₁₆ H ₂₅ O ₃	3.29
265.18	211.64	C ₁₆ H ₂₅ O ₃	4.89
266.11	459.85	C ₂₁ H ₁₄	4.25
267.05	181.67	C ₁₂ H ₁₁ O ₇	1.22
267.09	1927.02	C ₁₃ H ₁₅ O ₆	0.13
267.12	2756.49	C ₁₄ H ₁₉ O ₅	0.09
267.16	1039.00	C ₁₅ H ₂₃ O ₄	0.27
268.09	260.08	C ₂₀ H ₁₂ O	4.11
268.13	274.50	C ₂₁ H ₁₆	4.92

269.07	969.27	C ₁₂ H ₁₃ O ₇	1.15
269.10	1630.72	C ₁₃ H ₁₇ O ₆	0.99
269.12	106.55	C ₁₇ H ₁₇ O ₃	0.59
269.14	621.23	C ₁₄ H ₂₁ O ₅	0.80
269.25	106.56	C ₁₇ H ₃₃ O ₂	0.00
270.10	142.54	C ₂₀ H ₁₄ O	3.65
271.08	1003.21	C ₁₂ H ₁₅ O ₇	0.28
271.10	526.42	C ₁₆ H ₁₅ O ₄	0.13
271.12	470.78	C ₁₃ H ₁₉ O ₆	0.83
271.13	580.12	C ₁₇ H ₁₉ O ₃	0.42
273.06	176.62	C ₁₁ H ₁₃ O ₈	0.26
273.08	591.68	C ₁₅ H ₁₃ O ₅	0.14
273.10	324.83	C ₁₂ H ₁₇ O ₇	0.57
273.11	1540.04	C ₁₆ H ₁₇ O ₄	0.66
273.15	938.23	C ₁₇ H ₂₁ O ₃	1.43
275.05	151.39	C ₁₄ H ₁₁ O ₆	0.45
275.09	1559.17	C ₁₅ H ₁₅ O ₅	1.00
275.13	2117.80	C ₁₆ H ₁₉ O ₄	1.52
275.16	878.47	C ₁₇ H ₂₃ O ₃	0.34
277.07	480.87	C ₁₄ H ₁₃ O ₆	1.19
277.11	2022.78	C ₁₅ H ₁₇ O ₅	0.18
277.14	2404.89	C ₁₆ H ₂₁ O ₄	0.09
277.18	640.05	C ₁₇ H ₂₅ O ₃	0.33
279.09	1152.44	C ₁₄ H ₁₅ O ₆	0.20
279.12	2366.53	C ₁₅ H ₁₉ O ₅	0.30
279.16	31852.19	C ₁₆ H ₂₃ O ₄	0.37
279.20	294.01	C ₁₇ H ₂₇ O ₃	0.39
280.13	261.97	C ₂₂ H ₁₆	4.66
281.07	351.46	C ₁₃ H ₁₃ O ₇	1.50
281.10	1619.24	C ₁₄ H ₁₇ O ₆	1.46
281.14	1305.49	C ₁₅ H ₂₁ O ₅	1.39
281.17	314.89	C ₁₆ H ₂₅ O ₄	1.28
282.11	180.65	C ₂₁ H ₁₄ O	4.57
282.14	108.83	C ₂₂ H ₁₈	3.85
283.08	950.90	C ₁₃ H ₁₅ O ₇	0.59
283.12	996.98	C ₁₄ H ₁₉ O ₆	0.42
283.13	252.83	C ₁₈ H ₁₉ O ₃	0.61
283.15	190.18	C ₁₅ H ₂₃ O ₅	0.21
284.12	134.93	C ₂₁ H ₁₆ O	4.85
285.06	344.17	C ₁₂ H ₁₃ O ₈	0.27
285.10	636.53	C ₁₃ H ₁₇ O ₇	1.22
285.11	611.87	C ₁₇ H ₁₇ O ₄	1.16
285.13	225.75	C ₁₄ H ₂₁ O ₆	0.95
285.15	575.39	C ₁₈ H ₂₁ O ₃	0.88
287.08	303.36	C ₁₂ H ₁₅ O ₈	0.68

287.09	801.16	C ₁₆ H ₁₅ O ₅	1.61
287.11	250.10	C ₁₃ H ₁₉ O ₇	0.38
287.13	1299.89	C ₁₇ H ₁₉ O ₄	1.30
287.16	425.80	C ₁₈ H ₂₃ O ₃	0.96
288.09	176.80	C ₂₃ H ₁₂	4.40
289.07	284.63	C ₁₅ H ₁₃ O ₆	0.97
289.09	135.03	C ₁₂ H ₁₇ O ₈	0.47
289.11	1567.03	C ₁₆ H ₁₇ O ₅	0.67
289.14	1281.90	C ₁₇ H ₂₁ O ₄	0.34
289.18	356.80	C ₁₈ H ₂₅ O ₃	0.03
290.11	214.13	C ₂₃ H ₁₄	4.91
291.09	868.90	C ₁₅ H ₁₅ O ₆	0.92
291.12	1571.35	C ₁₆ H ₁₉ O ₅	0.63
291.16	916.67	C ₁₇ H ₂₃ O ₄	0.31
291.20	176.68	C ₁₈ H ₂₇ O ₃	0.04
292.13	281.87	C ₂₃ H ₁₆	3.47
293.07	330.03	C ₁₄ H ₁₃ O ₇	0.45
293.10	1286.75	C ₁₅ H ₁₇ O ₆	0.24
293.14	1193.75	C ₁₆ H ₂₁ O ₅	0.00
293.17	402.81	C ₁₇ H ₂₅ O ₄	1.50
294.11	124.47	C ₂₂ H ₁₄ O	4.11
295.08	891.61	C ₁₄ H ₁₅ O ₇	1.02
295.12	1127.51	C ₁₅ H ₁₉ O ₆	0.90
295.15	627.17	C ₁₆ H ₂₃ O ₅	0.75
295.19	123.61	C ₁₇ H ₂₇ O ₄	0.57
296.12	130.31	C ₂₂ H ₁₆ O	3.58
297.06	335.88	C ₁₃ H ₁₃ O ₈	1.26
297.10	850.89	C ₁₄ H ₁₇ O ₇	1.29
297.13	663.91	C ₁₅ H ₂₁ O ₆	1.29
297.15	152.72	C ₁₉ H ₂₁ O ₃	0.40
299.08	432.71	C ₁₃ H ₁₅ O ₈	1.16
299.09	250.90	C ₁₇ H ₁₅ O ₅	0.41
299.11	608.31	C ₁₄ H ₁₉ O ₇	1.34
299.13	606.56	C ₁₈ H ₁₉ O ₄	0.23
299.15	104.42	C ₁₅ H ₂₃ O ₆	1.50
299.16	183.52	C ₁₉ H ₂₃ O ₃	1.72
301.09	249.54	C ₁₃ H ₁₇ O ₈	0.84
301.11	787.02	C ₁₇ H ₁₇ O ₅	0.41
301.14	837.83	C ₁₈ H ₂₁ O ₄	0.77
303.07	123.12	C ₁₂ H ₁₅ O ₉	1.51
303.09	438.45	C ₁₆ H ₁₅ O ₆	0.41
303.12	1247.81	C ₁₇ H ₁₉ O ₅	1.03
303.16	647.33	C ₁₈ H ₂₃ O ₄	0.21
304.13	165.41	C ₂₄ H ₁₆	4.36
305.07	151.81	C ₁₅ H ₁₃ O ₇	1.47

305.10	966.66	C ₁₆ H ₁₇ O ₆	0.54
305.14	947.55	C ₁₇ H ₂₁ O ₅	0.43
305.17	408.16	C ₁₈ H ₂₅ O ₄	0.43
307.08	778.33	C ₁₅ H ₁₅ O ₇	0.94
307.12	1023.21	C ₁₆ H ₁₉ O ₆	0.29
307.15	450.21	C ₁₇ H ₂₃ O ₅	0.39
308.12	161.85	C ₂₃ H ₁₆ O	3.81
309.06	352.70	C ₁₄ H ₁₃ O ₈	0.15
309.10	1122.97	C ₁₅ H ₁₇ O ₇	0.16
309.13	649.83	C ₁₆ H ₂₁ O ₆	1.39
309.17	188.45	C ₁₇ H ₂₅ O ₅	1.02
310.08	254.58	C ₂₅ H ₁₀	4.15
310.10	157.80	C ₂₂ H ₁₄ O ₂	2.88
311.08	1047.30	C ₁₄ H ₁₅ O ₈	0.17
311.11	644.51	C ₁₅ H ₁₉ O ₇	0.13
311.13	126.18	C ₁₉ H ₁₉ O ₄	0.12
311.15	208.07	C ₁₆ H ₂₃ O ₆	0.10
312.08	142.64	C ₂₁ H ₁₂ O ₃	4.52
312.09	137.33	C ₂₅ H ₁₂	2.76
313.09	572.57	C ₁₄ H ₁₇ O ₈	0.38
313.11	250.41	C ₁₈ H ₁₇ O ₅	1.66
313.13	245.58	C ₁₅ H ₂₁ O ₇	0.81
313.14	414.57	C ₁₉ H ₂₁ O ₄	0.19
315.07	105.76	C ₁₃ H ₁₅ O ₉	0.67
315.11	242.17	C ₁₄ H ₁₉ O ₈	0.36
315.12	568.63	C ₁₈ H ₁₉ O ₅	0.30
315.16	389.16	C ₁₉ H ₂₃ O ₄	1.15
317.10	442.56	C ₁₇ H ₁₇ O ₆	0.17
317.14	628.88	C ₁₈ H ₂₁ O ₅	1.16
317.17	180.00	C ₁₉ H ₂₅ O ₄	0.54
319.08	284.88	C ₁₆ H ₁₅ O ₇	0.24
319.12	808.72	C ₁₇ H ₁₉ O ₆	0.15
319.15	384.39	C ₁₈ H ₂₃ O ₅	0.03
321.06	104.05	C ₁₅ H ₁₃ O ₈	0.88
321.10	919.04	C ₁₆ H ₁₇ O ₇	1.51
321.13	665.82	C ₁₇ H ₂₁ O ₆	0.02
321.17	200.47	C ₁₈ H ₂₅ O ₅	0.37
322.08	403.30	C ₂₆ H ₁₀	2.78
322.10	134.45	C ₂₃ H ₁₄ O ₂	3.86
323.08	1141.45	C ₁₅ H ₁₅ O ₈	0.84
323.11	1085.46	C ₁₆ H ₁₉ O ₇	0.13
323.15	161.16	C ₁₇ H ₂₃ O ₆	0.83
324.08	136.80	C ₂₂ H ₁₂ O ₃	3.57
324.09	294.65	C ₂₆ H ₁₂	2.26
324.12	147.61	C ₂₃ H ₁₆ O ₂	3.27

325.05	264.32	C ₁₄ H ₁₃ O ₉	1.47
325.09	1780.32	C ₁₅ H ₁₇ O ₈	1.09
325.11	205.71	C ₁₉ H ₁₇ O ₅	0.73
325.13	369.10	C ₁₆ H ₂₁ O ₇	0.69
326.10	273.98	C ₂₂ H ₁₄ O ₃	4.31
327.07	327.53	C ₁₄ H ₁₅ O ₉	1.10
327.11	585.35	C ₁₅ H ₁₉ O ₈	1.31
327.12	348.56	C ₁₉ H ₁₉ O ₅	0.36
327.14	113.48	C ₁₆ H ₂₃ O ₇	1.50
329.09	496.69	C ₁₄ H ₁₇ O ₉	0.09
329.12	180.71	C ₁₅ H ₂₁ O ₈	0.93
329.14	288.72	C ₁₉ H ₂₁ O ₅	1.40
329.17	141.32	C ₂₀ H ₂₅ O ₄	0.21
331.10	132.16	C ₁₄ H ₁₉ O ₉	1.84
331.12	293.93	C ₁₈ H ₁₉ O ₆	0.25
331.12	195.30	C ₁₈ H ₁₉ O ₆	3.75
331.15	290.99	C ₁₉ H ₂₃ O ₅	1.24
333.10	443.66	C ₁₇ H ₁₇ O ₇	0.89
333.13	457.15	C ₁₈ H ₂₁ O ₆	1.09
333.17	141.17	C ₁₉ H ₂₅ O ₅	1.25
334.08	290.33	C ₂₇ H ₁₀	3.13
335.08	435.29	C ₁₆ H ₁₅ O ₈	0.07
335.11	791.74	C ₁₇ H ₁₉ O ₇	1.00
335.15	205.57	C ₁₈ H ₂₃ O ₆	1.91
335.21	378.65	C ₁₆ H ₃₁ O ₇	1.22
336.09	320.26	C ₂₇ H ₁₂	2.41
336.12	158.21	C ₂₄ H ₁₆ O ₂	4.01
337.09	1666.82	C ₁₆ H ₁₇ O ₈	0.20
337.10	156.18	C ₂₇ H ₁₃	4.68
337.13	822.16	C ₁₇ H ₂₁ O ₇	0.16
337.16	190.37	C ₁₈ H ₂₅ O ₆	1.49
338.09	186.66	C ₂₃ H ₁₄ O ₃	3.10
338.11	223.16	C ₂₇ H ₁₄	3.08
338.13	105.86	C ₂₄ H ₁₈ O ₂	4.12
339.07	816.54	C ₁₅ H ₁₅ O ₉	0.09
339.11	1726.30	C ₁₆ H ₁₉ O ₈	0.34
339.12	225.08	C ₂₇ H ₁₅	4.90
339.14	179.03	C ₁₇ H ₂₃ O ₇	0.74
339.14	111.25	C ₁₇ H ₂₃ O ₇	4.83
340.11	318.08	C ₂₃ H ₁₆ O ₃	3.32
341.09	748.15	C ₁₅ H ₁₇ O ₉	0.54
341.12	426.26	C ₁₆ H ₂₁ O ₈	1.76
343.10	478.85	C ₁₅ H ₁₉ O ₉	1.62
343.12	243.13	C ₁₉ H ₁₉ O ₆	0.34
343.14	138.16	C ₁₆ H ₂₃ O ₈	0.48

343.15	212.06	C ₂₀ H ₂₃ O ₅	0.30
345.08	126.37	C ₁₄ H ₁₇ O ₁₀	0.62
345.10	186.32	C ₁₈ H ₁₇ O ₇	0.34
345.12	150.50	C ₁₅ H ₂₁ O ₉	0.23
345.13	300.76	C ₁₉ H ₂₁ O ₆	0.50
346.08	166.42	C ₂₈ H ₁₀	1.50
347.11	373.93	C ₁₈ H ₁₉ O ₇	0.22
347.15	213.70	C ₁₉ H ₂₃ O ₆	1.49
348.09	427.51	C ₂₈ H ₁₂	2.32
349.09	548.65	C ₁₇ H ₁₇ O ₈	1.35
349.13	541.87	C ₁₈ H ₂₁ O ₇	1.88
350.10	124.63	C ₂₄ H ₁₄ O ₃	3.90
350.11	260.76	C ₂₈ H ₁₄	3.60
351.07	308.32	C ₁₆ H ₁₅ O ₉	0.00
351.11	1331.55	C ₁₇ H ₁₉ O ₈	1.49
351.12	226.11	C ₂₈ H ₁₅	3.72
351.14	452.36	C ₁₈ H ₂₃ O ₇	0.83
352.09	155.31	C ₂₇ H ₁₂ O	3.51
352.11	176.30	C ₂₄ H ₁₆ O ₃	4.79
353.09	1247.76	C ₁₆ H ₁₇ O ₉	1.37
353.10	118.15	C ₂₇ H ₁₃ O	4.17
353.12	1116.19	C ₁₇ H ₂₁ O ₈	1.68
353.16	111.19	C ₁₈ H ₂₅ O ₇	1.97
354.09	218.52	C ₂₃ H ₁₄ O ₄	2.62
354.13	169.56	C ₂₄ H ₁₈ O ₃	2.89
355.07	199.15	C ₁₅ H ₁₅ O ₁₀	0.48
355.10	1087.87	C ₁₆ H ₁₉ O ₉	0.37
355.14	320.17	C ₁₇ H ₂₃ O ₈	0.92
356.11	187.43	C ₂₃ H ₁₆ O ₄	4.28
357.08	345.88	C ₁₅ H ₁₇ O ₁₀	0.01
357.12	578.78	C ₁₆ H ₂₁ O ₉	0.17
358.08	105.76	C ₂₉ H ₁₀	2.62
359.10	222.50	C ₁₅ H ₁₉ O ₁₀	0.75
359.11	114.68	C ₁₉ H ₁₉ O ₇	0.30
359.15	170.46	C ₂₀ H ₂₃ O ₆	0.88
360.09	508.46	C ₂₉ H ₁₂	2.58
361.13	273.13	C ₁₉ H ₂₁ O ₇	0.07
362.11	453.42	C ₂₉ H ₁₄	3.16
363.11	492.42	C ₁₈ H ₁₉ O ₈	0.48
363.14	201.04	C ₁₉ H ₂₃ O ₇	0.57
364.09	157.95	C ₂₈ H ₁₂ O	2.63
364.12	183.24	C ₂₉ H ₁₆	3.24
365.09	474.28	C ₁₇ H ₁₇ O ₉	0.89
365.12	912.08	C ₁₈ H ₂₁ O ₈	0.95
365.16	185.58	C ₁₉ H ₂₅ O ₇	1.01

366.10	144.56	C ₂₈ H ₁₄ O	2.10
366.13	161.11	C ₂₅ H ₁₈ O ₃	2.32
367.07	223.08	C ₁₆ H ₁₅ O ₁₀	2.26
367.10	1238.43	C ₁₇ H ₁₉ O ₉	1.26
367.14	540.45	C ₁₈ H ₂₃ O ₈	0.23
369.08	604.96	C ₁₆ H ₁₇ O ₁₀	1.28
369.12	787.75	C ₁₇ H ₂₁ O ₉	1.42
371.10	536.19	C ₁₆ H ₁₉ O ₁₀	1.04
371.12	127.97	C ₁₃ H ₂₃ O ₁₂	3.97
371.13	316.84	C ₁₇ H ₂₃ O ₉	0.11
372.09	485.38	C ₃₀ H ₁₂	3.48
373.11	310.58	C ₁₆ H ₂₁ O ₁₀	0.27
374.11	605.00	C ₃₀ H ₁₄	3.16
375.11	200.72	C ₁₉ H ₁₉ O ₈	0.45
376.09	219.79	C ₂₉ H ₁₂ O	2.75
376.12	415.33	C ₃₀ H ₁₆	3.74
377.12	368.90	C ₁₉ H ₂₁ O ₈	0.20
378.10	300.13	C ₂₉ H ₁₄ O	1.64
379.10	532.91	C ₁₈ H ₁₉ O ₉	0.30
379.11	150.78	C ₂₉ H ₁₅ O	3.13
379.14	449.80	C ₁₉ H ₂₃ O ₈	0.84
380.12	125.56	C ₂₉ H ₁₆ O	3.26
381.08	220.19	C ₁₇ H ₁₇ O ₁₀	0.24
381.12	828.90	C ₁₈ H ₂₁ O ₉	1.01
381.13	146.21	C ₂₉ H ₁₇ O	4.42
381.15	180.97	C ₁₉ H ₂₅ O ₈	0.56
382.12	151.68	C ₂₅ H ₁₈ O ₄	3.54
383.10	712.12	C ₁₇ H ₁₉ O ₁₀	0.26
383.13	512.66	C ₁₈ H ₂₃ O ₉	0.50
385.08	154.54	C ₁₆ H ₁₇ O ₁₁	0.81
385.11	557.77	C ₁₇ H ₂₁ O ₁₀	0.41
387.09	253.28	C ₁₆ H ₁₉ O ₁₁	0.55
387.11	101.64	C ₂₀ H ₁₉ O ₈	0.07
387.13	174.48	C ₁₇ H ₂₃ O ₁₀	1.67
388.09	254.79	C ₃₀ H ₁₂ O	2.63
389.12	192.58	C ₂₀ H ₂₁ O ₈	1.49
390.10	244.21	C ₃₀ H ₁₄ O	1.43
391.10	194.30	C ₁₉ H ₁₉ O ₉	0.57
391.14	202.51	C ₂₀ H ₂₃ O ₈	1.13
391.28	482.04	C ₂₄ H ₃₉ O ₄	0.87
392.12	172.51	C ₃₀ H ₁₆ O	3.58
393.12	416.20	C ₁₉ H ₂₁ O ₉	0.21
393.15	233.14	C ₂₀ H ₂₅ O ₈	0.18
394.11	170.79	C ₁₅ H ₂₂ O ₁₂	4.78
395.10	397.45	C ₁₈ H ₁₉ O ₁₀	0.99

395.13	558.70	C ₁₉ H ₂₃ O ₉	0.32
396.13	259.41	C ₁₅ H ₂₄ O ₁₂	4.71
397.08	183.63	C ₁₇ H ₁₇ O ₁₁	0.51
397.11	442.14	C ₁₈ H ₂₁ O ₁₀	0.35
397.15	177.08	C ₁₉ H ₂₅ O ₉	0.20
399.09	199.83	C ₁₇ H ₁₉ O ₁₁	0.49
399.13	221.08	C ₁₈ H ₂₃ O ₁₀	0.27
401.11	124.51	C ₁₇ H ₂₁ O ₁₁	1.29
403.10	140.78	C ₂₀ H ₁₉ O ₉	1.54
405.12	213.16	C ₂₀ H ₂₁ O ₉	0.16
406.10	137.96	C ₃₀ H ₁₄ O ₂	3.66
407.13	228.17	C ₂₀ H ₂₃ O ₉	0.18
409.11	245.44	C ₁₉ H ₂₁ O ₁₀	1.20
409.15	186.24	C ₂₀ H ₂₅ O ₉	0.86
411.09	188.68	C ₁₈ H ₁₉ O ₁₁	0.05
411.13	238.45	C ₁₉ H ₂₃ O ₁₀	1.34
411.13	146.83	C ₁₉ H ₂₃ O ₁₀	3.66
413.11	299.27	C ₁₈ H ₂₁ O ₁₁	0.32
415.12	105.21	C ₁₈ H ₂₃ O ₁₁	0.55
417.12	220.37	C ₂₁ H ₂₁ O ₉	1.17
419.13	130.91	C ₂₁ H ₂₃ O ₉	1.56
421.11	120.35	C ₂₀ H ₂₁ O ₁₀	0.94
423.13	123.36	C ₂₀ H ₂₃ O ₁₀	0.32
427.12	148.60	C ₁₉ H ₂₃ O ₁₁	1.61

APPENDIX 3

FULL CODE OF EXAMPLE.PL

```

1  #!/usr/bin/perl
2  use warnings;
3  use WWW::Mechanize;
4  use LWP::Simple;
5  use File::Copy;
6  use Net::OpenSSH;
7  use Net::SSH::Perl;
8  use Tie::File;
9
10
11 my $formula = <$ARGV[0]>;
12 my $user = <$ARGV[1]>;
13 my $password = <$ARGV[2]>;
14 my $email = <$ARGV[3]>;
15 #this takes values that were input from the GUI and assigns them to variables
16
17
18
19 $dirname = "/Volumes/MAIN/Dropbox/DMC/Research/RJR/$formula";
20 mkdir $dirname;
21 chdir "/Volumes/MAIN/Dropbox/DMC/Research/RJR/$formula" or die "$1";
22 #gives name to a directory that is made for storing everything, changes to this directory
23
24
25 $m = WWW::Mechanize->new();
26 $url = 'http://webbook.nist.gov/cgi/cbook.cgi?Formula='.$formula.'&NoIon=on&Units=SI';
27 $m->get($url);
28 #url with formula(the entered variable) is stored as a variable $m
29
30 print "$url\n";
31 #prints the url link
32
33
34 my @links = $m->find_all_links();
35 for my $link ( @links ) {
36   if (index($link->url, 'cgi') != -1){
37     chdir "/Volumes/MAIN/Dropbox/DMC/Research/RJR/$formula" or die "$1";
38     open my $output, '>>', 'output.txt' or die "$1";
39     open my $output_name, '>>', 'name.txt' or die "$1";
40     printf "%s, webbook.nist.gov%s\n" , $link->text, $link->url;
41     printf $output "%s, webbook.nist.gov%s\n" , $link->text, $link->url;
42     printf $output_name "%s\n" , $link->text;
43   }
44 }
45
46 #okay so the above section finds all the links for compounds of a specific chemical formula it then
47 # looks for the links that have a certain keyword and then prints the name of compound and the website
48 # to acces that compound to the screen and to an output text file
49
50 open my $fh, '<', 'output.txt';
51
52 my $ID;
53 chomp(my @lines = <$fh>);
54 for my $lines ( @lines ) {
55   if ( $lines =~ /ID=(.+)&Units/ ) {
56
57     $ID = $1;
58     open my $output_ID, '>>', 'ID.txt' or die "$1";
59     open my $output_site, '>>', 'site.txt' or die "$1";

```

```

60     print "$ID\n";
61     $m2 = WWW::Mechanize->new();
62     $url2 = 'http://webbook.nist.gov/cgi/cbook.cgi?ID=',$ID.'&Units=SI';
63     $m2->get($url2);
64     print "$url2\n";
65     printf $output_ID "$ID\n" ;
66     printf $output_site "$url2\n";
67     my $m3 = WWW::Mechanize->new();
68     $url3 = 'http://webbook.nist.gov/cgi/cbook.cgi?Str2File=',$ID.'';
69     $m3->get($url3);
70     print "$url3\n";
71     $m4 = WWW::Mechanize->new();
72     $url4 = 'http://webbook.nist.gov/cgi/cbook.cgi?Str3File=',$ID.'';
73     $m4->get($url4);
74     print "$url4\n";
75
76     my $file3 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $ID . '.mol';
77     my $file4 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $ID . '.sdf';
78
79     getstore($url3,$file3);
80     getstore($url4,$file4);
81
82     #the above searches file and prints what's between two strings, prints the ID number and then
83     ... accesses the website for that particular compound the ID numbers are printed to a file named ID.txt
84     ... and the websites for the compounds are printed to site.txtIt then saves the 2D chemical structure
85     ... file and the 3D chemical structure to the folder
86
87     my $cmd = 'babel ' . $ID . '.sdf ' . $ID . '.com ' ;
88     system($cmd);
89
90     #uses babel to convert from the format from NIST to a gaussian input (.com) file
91
92     open (DATAIN, '<' . $ID . '.com');
93     open (DATAOUT, '>' . $ID . '_1.com');
94     while (<DATAIN>){
95         print DATAOUT unless ($. == 1..4);
96     }
97     close DATAOUT;
98     close DATAIN;
99
100    #Removes the unwanted information from the .com file
101
102    $proc = 4;
103    $processors = '%nprocshared=',$proc.'';
104    $mem = 40;
105    $memory = '%mem=',$mem.'MW';
106    $key = '# opt hf/3-21g';
107    $title = '',$ID.'';
108    open my $input, '>>', 'header_' . $ID . '.txt' or die "$1";
109    print $input "$processors\n";
110    print $input "$memory\n";
111    print $input "$key\n";
112    print $input "\n";
113    print $input "$title\n";
114    print $input "\n";
115
116    #This creates a header file for every ID number and saves that so we can use it
117
118    copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
119    ... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/merge_files.pl') or die "Copy failed: $!";
120    #This copies a script called merge_files.pl to the specific formula folder we are working in
121
122    my $cmd2 = 'perl merge_files.pl header_' . $ID . '.txt ' . $ID . '_1.com ' . $ID . '_2.com';
123    system($cmd2);
124    print "$cmd2\n";
125    #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
126    ... input file

```

```

123
124
125     }
126
127 }
128 close $fh;
129
130
131 my $cmd3 = 'tar -cvvf '.$formula.'.tar /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/';
132 system($cmd3);
133 #creates a tarbell of all the files
134
135
136 my $host = "killdevil.unc.edu";
137 my $local_path = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'.tar/';
138 my $remote_path = "/netscr/dcrizer";
139 my $ssh = Net::OpenSSH->new(host => $host, user => $user, password => $password);
140 $ssh->scp_put($local_path, $remote_path)
141     or die "scp failed: " . $ssh->error;
142
143 #Just does a straight secure copy from local path to the remote path (netscr/$user)
144
145
146 my $ssh = Net::SSH::Perl->new($host);
147 $ssh->login($user, $password);
148 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'');
149 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
150     /netscr/'.$user.'/'.$formula.'.tar');
151 my($stdout, $stderr, $exit) = $ssh->cmd('mv
152     /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'
153     /netscr/'.$user.'/'.$formula.'');
154 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/'.$user.'/'.$formula.'/'.$formula.'/*_2.com
155     /netscr/'.$user.'/'.$formula.'');
156 #accesses the killdevil server via ssh and then moves important data to the server, the untars the
157 #files and copies the input files that we want to submit to a specific folder
158
159
160 my $local_path2 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
161 my $remote_path2 = '/netscr/'.$user.'/'.$formula.'/';
162 my $ssh2 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
163 $ssh2->scp_put($local_path2, $remote_path2)
164     or die "scp failed: " . $ssh2->error;
165 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/ && ./run.sh');
166 #okay so this section accesses KillDevil and then executes commands. It makes a directory
167 #for the formula of interest, it then changes to the netscr directory and then untar's the
168 #file containing input files and then it moves that directory to where we want it.
169
170
171 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output2.txt');
172 do{
173     sleep(300);
174     print "Running neutral compounds at HF/3-21G\n";
175     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && rm output2.txt');
176     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && bjobs -l > output2.txt');
177     my $local_path3 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
178     my $remote_path3 = '/netscr/'.$user.'/output2.txt';
179     my $ssh3 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
180     $ssh3->scp_get($remote_path3, $local_path3)
181         or die "scp failed: " . $ssh3->error;
182     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output2.txt");
183
184
185 #This above section is really the heart and soul that makes this program work. The idea is
186 #that once you have a set of calculations submitted you need to wait until they are done
187 #running until you can move on to the next level of theory. This sections basically monitors the
188 #output for bjobs on killdevil and writes that into and output file. Then the output file is
189 #monitored on the computer where you submit the program and looks for the output file to be empty
190 #which means that all of the calculations are done!!

```

```

181
182
183
184 print "DONE\n";
185
186 #####
187
188 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G/');
189 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'$formula.'/*_2.log
... /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G/');
190 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'$formula.'/ && tar -cvvf
... '$formula.'_HF-3-21G.tar /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G/');
191
192 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
193
194 my $local_path4 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'';
195 my $remote_path4 = '/netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G.tar';
196 my $ssh4 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
197 $ssh4->scp_get($remote_path4, $local_path4)
198     or die "scp failed: " . $ssh4->error;
199 my $cmd4 = 'tar -xvf /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G.tar';
200 system($cmd4);
201
202 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/netscr/dcrizer/.$formula.'/'.'$formula.'
... _HF-3-21G/', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G/') or die
... "Copy failed: $!";
203 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
... then moves them to an appropriate directory
204
205 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G/geometry_extract.plx') or
... die "Copy failed: $!";
206 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G/output.txt') or die "Copy
... failed: $!";
207 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G/' or die '$!';
208 #moves script and needed list of ID numbers to the folder where output files are and then we move to
... that directory
209
210 open my $fh, '<', 'output.txt';
211
212 my $ID;
213 chomp(my @lines = <$fh>);
214 for my $lines ( @lines ) {
215     if ( $lines =~ /ID=(.*)&Units/ ) {
216
217         $ID = $1;
218 my $cmd5 = 'perl geometry_extract.plx < '.$ID.'_2.log > '.$ID.'_3.com';
219 system($cmd5);
220 #extracts geometries
221 open (DATAIN, '<'.$ID.'_3.com');
222 open (DATAOUT, '>'.$ID.'_4.com');
223 while (<DATAIN>){
224     print DATAOUT unless ($. == 1);
225 }
226 close DATAOUT;
227 close DATAIN;
228
229     $proc2 = 4;
230     $processors2 = '%nprocshared='.$proc.'';
231
232     $mem2 = 40;
233     $memory2 = '%mem='.$mem.'MW';

```



```

234
235     $key2 = '# opt b3lyp/3-21g';
236     $title2 = ''.$ID.'';
237     open my $input2, '>>', 'header2_'. $ID.'.txt' or die "$1";
238     print $input2 "$processors2\n";
239     print $input2 "$memory2\n";
240
241     print $input2 "$key2\n";
242     print $input2 "\n";
243     print $input2 "$title2\n";
244     print $input2 "\n";
245
246 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
...   '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_HF-3-21G/merge_files.pl') or die
...   "Copy failed: $!";
247 #This copies a script called merge_files.pl to the specific formula folder we are working in
248
249 my $cmd6 = 'perl merge_files.pl header2_'. $ID.'.txt ' . $ID . '_4.com ' . $ID . '_5.com';
250 system($cmd6);
251 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
... input file
252 }
253 }
254 }
255 close $fh;
256
257
258 my $cmd7 = 'tar -cvvf ' . $formula . '_2.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_HF-3-21G/';
259 system($cmd7);
260
261 my $local_path5 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_HF-3-21G/' . $formula . '_2.tar/';
262 my $remote_path5 = "/netscr/dcrizer";
263 my $ssh5 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
264 $ssh5->scp_put($local_path5, $remote_path5)
265     or die "scp failed: " . $ssh5->error;
266
267 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/' . $user . '/' . $formula . '/run2/');
268 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_HF-3-21G/');
269 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && tar -xvf
... /netscr/' . $user . '/' . $formula . '_2.tar');
270 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/' . $user . '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_HF-3-21G/
270... /netscr/' . $user . '/' . $formula . '/' . $formula . '_HF-3-21G/');
271 print "$stdout\n";
272 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_HF-3-21G/' . $formula . '_HF-3-21G/*_5.com
... /netscr/' . $user . '/' . $formula . '/run2/');
273
274 my $local_path8 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
275 my $remote_path8 = '/netscr/' . $user . '/' . $formula . '/run2/';
276 my $ssh8 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
277 $ssh8->scp_put($local_path8, $remote_path8)
278     or die "scp failed: " . $ssh8->error;
279 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' . $formula . '/run2/ && ./run.sh');
280
281 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output3.txt');
282 do{
283     sleep(300);
284     print "Running neutral compounds at B3LYP/3-21G\n";
285     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && rm output3.txt');
286     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && bjobs -l > output3.txt');
287     my $local_path7 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
288     my $remote_path7 = '/netscr/' . $user . '/output3.txt';
289     my $ssh7 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
290     $ssh7->scp_get($remote_path7, $local_path7)
291         or die "scp failed: " . $ssh7->error;
292     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output3.txt");

```

```

293
294
295 print "DONE*2\n";
296
297 #####
298
299 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-3-21G');
300 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/' . $user . '/' . $formula . '/run2/*_5.log
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-3-21G/');
301 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' . $formula . '/' && tar -cvvf
... '$formula._B3LYP-3-21G.tar /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-3-21G/');
302
303 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
304
305 my $local_path9 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/';
306 my $remote_path9 = '/netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-3-21G.tar';
307 my $ssh9 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
308 $ssh9->scp_get($remote_path9, $local_path9)
309     or die "scp failed: " . $ssh9->error;
310 my $cmd8 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G.tar';
311 system($cmd8);
312
313 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_HF-3-21G/netscr/dcrizer/' . $
... formula . '/' . $formula . '_B3LYP-3-21G/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G/') or die "Copy
... failed: $!";
314 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
... then moves them to an appropriate directory
315
316 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G/geometry_extract.plx')
... or die "Copy failed: $!";
317 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G/output.txt') or die
... "Copy failed: $!";
318 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G/' or die '$!';
319 #moves script and needed list of ID numbers to the folder where output files are and then we move to
... that directory
320
321 open my $fh, '<', 'output.txt';
322
323 my $ID;
324 chomp(my @lines = <$fh>);
325 for my $lines ( @lines ) {
326     if ( $lines =~ /ID=(.+)&Units/ ) {
327
328         $ID = $1;
329 my $cmd9 = 'perl geometry_extract.plx < '$ID.'_5.log > '$ID.'_6.com';
330 system($cmd9);
331 open (DATAIN, '<' . $ID . '_6.com');
332 open (DATAOUT, '>' . $ID . '_7.com');
333 while (<DATAIN>){
334     print DATAOUT unless ($. == 1);
335 }
336 close DATAOUT;
337 close DATAIN;
338
339     $proc3 = 4;
340     $processors3 = '%nprocshared=' . $proc . ' ';
341
342     $mem3 = 40;
343     $memory3 = '%mem=' . $mem . 'MW';
344

```

```

345     $key3 = '# opt b3lyp/6-31g';
346     $title3 = ".$ID.";
347     open my $input3, '>', 'header3_.$ID.'.txt' or die "$1";
348     print $input3 "$processors3\n";
349     print $input3 "$memory3\n";
350
351     print $input3 "$key3\n";
352     print $input3 "\n";
353     print $input3 "$title3\n";
354     print $input3 "\n";
355
356 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
357     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.$formula.'_B3LYP-3-21G/merge_files.pl') or
358     die "Copy failed: $!";
359 #This copies a script called merge_files.pl to the specific formula folder we are working in
360
361 my $cmd10 = 'perl merge_files.pl header3_.$ID.'.txt ".$ID.'_7.com '.$ID.'_8.com';
362 system($cmd10);
363 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
364 input file
365 }
366 }
367 close $fh;
368
369 #executes the script to strip geometries for all output files and places the results into new input
370 files.
371
372 my $cmd11 = 'tar -cvvf '.$formula.'_3.tar
373     /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.$formula.'_B3LYP-3-21G/';
374 system($cmd11);
375
376 my $local_path10 =
377     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.$formula.'_B3LYP-3-21G/.$formula.'_3.tar/';
378 my $remote_path10 = "/netscr/dcrizer";
379 my $ssh10 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
380 $ssh10->scp_put($local_path10, $remote_path10)
381     or die "scp failed: " . $ssh10->error;
382
383 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/.$user.'/'.$formula.'/run3/');
384 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
385     /netscr/.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G/');
386 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/ && tar -xvf
387     /netscr/.$user.'/'.$formula.'_3.tar');
388 my($stdout, $stderr, $exit) = $ssh->cmd('mv
389     /netscr/.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.$formula.'_B3LYP-3-21G/
390     /netscr/.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G/');
391 print "$stdout\n";
392 my($stdout, $stderr, $exit) = $ssh->cmd('mv
393     /netscr/.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G/.$formula.'_B3LYP-3-21G/*_8.com
394     /netscr/.$user.'/'.$formula.'/run3/');
395
396 my $local_path11 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
397 my $remote_path11 = '/netscr/.$user.'/'.$formula.'/run3/';
398 my $ssh11 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
399 $ssh11->scp_put($local_path11, $remote_path11)
400     or die "scp failed: " . $ssh11->error;
401 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.$formula.'/run3/ && ./run.sh');
402
403 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output4.txt');
404 do{
405     sleep(300);
406     print "Running neutral compounds at B3LYP/6-31G\n";
407     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/ && rm output4.txt');
408     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/ && bjobs -l > output4.txt');
409     my $local_path12 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
410     my $remote_path12 = '/netscr/.$user.'/output4.txt';

```



```

400 my $ssh12 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
401 $ssh12->scp_get($remote_path12, $local_path12)
402     or die "scp failed: " . $ssh12->error;
403     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output4.txt");
404
405 print "DONE*3\n";
406
407 #####
408 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
...
/netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31G');
409 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/' . $user . '/' . $formula . '/run3/*_8.log
...
/netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31G/');
410 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' . $formula . '/' && tar -cvvf
...
' . $formula . '_B3LYP-6-31G.tar /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31G/');
411
412 #makes a directory for the output files (.log) and then moves to another folder. We then change over
...
to that directory and tar that directory to be easier to transfer
413
414 my $local_path13 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/';
415 my $remote_path13 = '/netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31G.tar';
416 my $ssh13 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
417 $ssh13->scp_get($remote_path13, $local_path13)
418     or die "scp failed: " . $ssh13->error;
419 my $cmd12 = 'tar -xvf
...
/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31G.tar';
420 system($cmd12);
421
422 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G/netscr/dcrizer/' .
...
$formula . '/' . $formula . '_B3LYP-6-31G/',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31G/') or die "Copy
...
failed: $!";
423 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
...
then moves them to an appropriate directory
424
425 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31G/geometry_extract.plx')
...
or die "Copy failed: $!";
426 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/output.txt',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31G/output.txt') or die
...
"Copy failed: $!";
427 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31G/' or die '$!';
428 #moves script and needed list of ID numbers to the folder where output files are and then we move to
...
that directory
429
430 open my $fh, '<', 'output.txt';
431
432 my $ID;
433 chomp(my @lines = <$fh>);
434 for my $lines ( @lines ) {
435     if ( $lines =~ /ID=(.+)&Units/ ) {
436
437         $ID = $1;
438 my $cmd13 = 'perl geometry_extract.plx < ' . $ID . '_8.log > ' . $ID . '_9.com';
439 system($cmd13);
440 open (DATAIN, '<' . $ID . '_9.com');
441 open (DATAOUT, '>' . $ID . '_10.com');
442 while (<DATAIN){
443     print DATAOUT unless ($. == 1);
444 }
445 close DATAOUT;
446 close DATAIN;
447
448 $proc4 = 4;
449 $processors4 = '%nprocshared=' . $proc . ' ';
450
451 $mem4 = 40;
452 $memory4 = '%mem=' . $mem . 'MW';
453

```

```

454     $key4 = '# opt freq b3lyp/6-31g(d,p)';
455     $title4 = ' '.$ID.'';
456     open my $input4, '>', 'header4_'.$ID.'.txt' or die "$1";
457     print $input4 "$processors4\n";
458     print $input4 "$memory4\n";
459
460     print $input4 "$key4\n";
461     print $input4 "\n";
462     print $input4 "$title4\n";
463     print $input4 "\n";
464
465     copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
466         '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G/merge_files.pl') or
467         die "Copy failed: $!";
468     #This copies a script called merge_files.pl to the specific formula folder we are working in
469
470     my $cmd14 = 'perl merge_files.pl header4_'.$ID.'.txt '.$ID.'_10.com '.$ID.'_11.com';
471     system($cmd14);
472
473     #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
474     input file
475 }
476 }
477 close $fh;
478
479 #executes the script to strip geometries for all output files and places the results into new input
480 files.
481
482 my $cmd15 = 'tar -cvf '.$formula.'_4.tar
483 /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G/';
484 system($cmd15);
485
486 my $local_path14 =
487     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G/'.$formula.'_4.tar/';
488 my $remote_path14 = "/netscr/dcrizer";
489 my $ssh14 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
490 $ssh14->scp_put($local_path14, $remote_path14)
491     or die "scp failed: " . $ssh14->error;
492
493 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run4/');
494 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
495 /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G/');
496 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
497 /netscr/'.$user.'/'.$formula.'_4.tar');
498 my($stdout, $stderr, $exit) = $ssh->cmd('mv
499 /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G/
500 /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G/');
501 print "$stdout\n";
502 my($stdout, $stderr, $exit) = $ssh->cmd('mv
503 /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G/'.$formula.'_B3LYP-6-31G/*_11.com
504 /netscr/'.$user.'/'.$formula.'/run4/');
505
506 my $local_path15 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
507 my $remote_path15 = '/netscr/'.$user.'/'.$formula.'/run4/';
508 my $ssh15 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
509 $ssh15->scp_put($local_path15, $remote_path15)
510     or die "scp failed: " . $ssh15->error;
511 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/run4/ && ./run.sh');
512
513 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output5.txt');
514 do{
515     sleep(300);
516     print "Running neutral compounds at B3LYP/6-31G(d,p)\n";
517     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && rm output5.txt');
518     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && bjobs -l > output5.txt');
519     my $local_path16 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
520     my $remote_path16 = '/netscr/'.$user.'/output5.txt';

```

```

509 my $ssh16 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
510 $ssh16->scp_get($remote_path16, $local_path16)
511     or die "scp failed: " . $ssh16->error;
512 } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output5.txt");
513
514 print "DONE*4\n";
515
516 #so at this point we have completed the runs for the neutral molecules and next have to convert to
517 # protonated species
518 #####
519
520 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
521     /netscr/.$user.'/'.'.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp');
522 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'.$formula.'_B3LYP-6-31Gdp/run4/*_11.log
523     /netscr/.$user.'/'.'.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/');
524 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'.$formula.'/' && tar -cvvf
525     '$formula.'_B3LYP-6-31Gdp.tar /netscr/.$user.'/'.'.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/');
526
527 #makes a directory for the output files (.log) and then moves to another folder. We then change over
528 # to that directory and tar that directory to be easier to transfer
529
530 my $local_path17 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'';
531 my $remote_path17 = '/netscr/.$user.'/'.'.$formula.'_B3LYP-6-31Gdp.tar';
532 my $ssh17 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
533 $ssh17->scp_get($remote_path17, $local_path17)
534     or die "scp failed: " . $ssh17->error;
535 my $cmd16 = 'tar -xvf
536     /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp.tar';
537 system($cmd16);
538
539 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/netscr/dcrizer/.'
540     '$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/',
541     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/') or die "Copy
542     failed: $!";
543 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
544 # then moves them to an appropriate directory
545
546 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
547     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/geometry_extract.plx
548     ') or die "Copy failed: $!";
549 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/output.txt',
550     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/output.txt') or die
551     "Copy failed: $!";
552 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'.$formula.'_B3LYP-6-31Gdp/' or die
553     '$!';
554 #moves script and needed list of ID numbers to the folder where output files are and then we move to
555 # that directory
556
557 open my $fh, '<', 'output.txt';
558
559 my $ID;
560 chomp(my @lines = <$fh>);
561 for my $lines ( @lines ) {
562     if ( $lines =~ /ID=(.+)&Units/ ) {
563
564         $ID = $1;
565     }
566 }
567 my $cmd17 = 'perl geometry_extract.plx < '.$ID.'_11.log > '.$ID.'_12.com';
568 system($cmd17);
569
570 }
571 }
572 close $fh;
573 my $cmd17b = 'grep -l "^0$" *.com | xargs rm';
574 system($cmd17b);
575
576 open my $fh, '<', 'output.txt';

```

```

561 my $ID;
562 chomp(my @lines = <$fh>);
563     for my $lines ( @lines ) {
564         if ( $lines =~ /ID=(.+)&Units/ ) {
565
566             $ID = $1;
567 my $cmd18 = 'grep -o "O.*" '.$ID.'_12.com > '.$ID.'_convert.txt';
568     system($cmd18);
569
570 #this greps a file and print the whole line after "O" so in otherwords it will print out all lines
... that are oxygen with the coordinats that can be modified to add a hydrogen
571 }
572 }
573 close my $fh;
574
575 open my $fh, '<', 'output.txt';
576
577 my $ID;
578 chomp(my @lines = <$fh>);
579     for my $lines ( @lines ) {
580         if ( $lines =~ /ID=(.+)&Units/ ) {
581
582             $ID = $1;
583 open my $convertin, '<', ''.$ID.'_convert.txt';
584 chomp(my @convertline = <$convertin>);
585     for my $convertline(@convertline){
586         my ($w, $x , $y, $z) = split(/      /, $convertline);
587         print "H\t$x\t$y\t$z\n";
588         my $a = 0.58;
589         my $b = 0.58;
590         my $c = 0.58;
591         my $d = $x+$a;
592         my $e = $y+$b;
593         my $f = $z+$c;
594         print "H\t$d\t$e\t$f\n";
595         open my $convertout, '>>', ''.$ID.'_convertout.txt';
596         print $convertout "H\t\t";
597         my $junk1 = sprintf("%f", $d);
598         print $convertout "$junk1\t\t";
599         my $junk2 = sprintf("%f", $e);
600         print $convertout "$junk2\t\t";
601         my $junk3 = sprintf("%f", $f);
602         print $convertout "$junk3\n";
603
604     }
605 close $convertin;
606 close $convertout;
607 }
608 }
609 close my $fh;
610 #so this will read the oxygen containing lines one by one and then string out each line and then
... assign the numbers to variables and currently print a new line to the screen need to figure out what
... to put in // to separate out the different numbers. Now this looks like it is splitting and
... assigning variables properly
611 open my $fh, '<', 'output.txt';
612
613 my $ID;
614 chomp(my @lines = <$fh>);
615     for my $lines ( @lines ) {
616         if ( $lines =~ /ID=(.+)&Units/ ) {
617
618             $ID = $1;
619 my $lc = 0;
620 open my $convertout2, '<', ''.$ID.'_convert.txt' or die($!);
621 $lc++ while <$convertout2>;
622 close $convertout2;

```



```

623 print $lc, "\n";
624
625
626 my $line_number = 0;
627 do{
628 my $filetolocate =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/' . $ID . '_12.com';
629 if (-e $filetolocate) {
630 copy(''. $ID . '_12.com', ''. $ID . '_12_' . $line_number . '.com') or die "Copy failed: $!";
631 tie @dellines, Tie::File, ''. $ID . '_12_' . $line_number . '.com' or die "can't update $file: $!";
632 delete $dellines[-1];
633 }
634 if (-e $filetolocate) {
635 open my $convertout, '<', ''. $ID . '_convertout.txt' or die($!);
636 open my $in2, '>>', ''. $ID . '_transer_' . $line_number . '.com';
637 chomp(my @convertlines = <$convertout>);
638 print "@convertlines[$line_number]\n";
639 print $in2 "@convertlines[$line_number]\n";
640 print $in2 "\n";
641 }
642
643 open (DATAIN, '<' . $ID . '_12_' . $line_number . '.com');
644 open (DATAOUT, '>' . $ID . '_13_' . $line_number . '.com');
645 while (<DATAIN>){
646     print DATAOUT unless ($.==1..2);
647 }
648 close DATAOUT;
649 close DATAIN;
650 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/merge_files.pl') or
... die "Copy failed: $!";
651 my $cmd19 = 'perl merge_files.pl ' . $ID . '_13_' . $line_number . '.com
... ' . $ID . '_transer_' . $line_number . '.com ' . $ID . '_14_' . $line_number . '.com';
652 system($cmd19);
653
654
655     $proc5 = 4;
656     $processors5 = '%nprocshared=' . $proc . ' ';
657
658     $mem5 = 40;
659     $memory5 = '%mem=' . $mem . 'MW';
660
661     $key5 = '# opt hf/3-21g';
662     $title5 = ' ' . $ID . ' ';
663     open my $input5, '>>', 'header5_' . $ID . '_' . $line_number . '.txt' or die "$1";
664     print $input5 "$processors5\n";
665     print $input5 "$memory5\n";
666
667     print $input5 "$key5\n";
668     print $input5 "\n";
669     print $input5 "$title5\n";
670     print $input5 "\n";
671     print $input5 "1\t1\n";
672
673 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/merge_files.pl') or
... die "Copy failed: $!";
674 #This copies a script called merge_files.pl to the specific formula folder we are working in
675
676
677 my $cmd20 = 'perl merge_files.pl header5_' . $ID . '_' . $line_number . '.txt
677... ' . $ID . '_14_' . $line_number . '.com ' . $ID . '_15_' . $line_number . '.com';
678 system($cmd20);
679 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
... input file
680 print "stuff\n";
681 print "$line_number\n";

```

```

682
683 $line_number=$line_number+1;
684 print "stuff2\n";
685 print "$line_number\n";
686 print "$lc\n";
687 my $lc2 = $lc;
688 print "$lc2\n";
689 }while($line_number < $lc);
690
691
692 #so this separates the different lines of the hydrogens to be added prints the line and then loops
... back and prints the next one and so on until we reach the max number of lines need to add in from
... like line 484 right before we make header and merge files
693
694 }
695 }
696
697
698 close my $fh;
699
700 my $cmd15 = 'tar -cvvf '.$formula.'_5.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/';
701 system($cmd15);
702
703 my $local_path18 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/'.$formula.'_5.tar/'
... ;
704 my $remote_path18 = "/netscr/dcrizer";
705 my $ssh18 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
706 $ssh18->scp_put($local_path18, $remote_path18)
707     or die "scp failed: " . $ssh18->error;
708
709 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run5/');
710 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/');
711 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
... /netscr/'.$user.'/'.$formula.'_5.tar');
712 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/');
713 print "$stdout\n";
714 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31Gdp/'.$formula.'_B3LYP-6-31Gdp/*_15_*.com
... /netscr/'.$user.'/'.$formula.'/run5/');
715
716 my $local_path19 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
717 my $remote_path19 = '/netscr/'.$user.'/'.$formula.'/run5/';
718 my $ssh19 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
719 $ssh19->scp_put($local_path19, $remote_path19)
720     or die "scp failed: " . $ssh19->error;
721 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/run5/ && ./run.sh');
722
723 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output6.txt');
724 do{
725     sleep(300);
726     print "Running protonated compounds at HF/3-21G\n";
727     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && rm output6.txt');
728     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && bjobs -l > output6.txt');
729     my $local_path20 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
730     my $remote_path20 = '/netscr/'.$user.'/output6.txt';
731     my $ssh20 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
732     $ssh20->scp_get($remote_path20, $local_path20)
733         or die "scp failed: " . $ssh20->error;
734     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output6.txt");
735
736 print "DONE*5\n";
737 #####

```

```

738
739
740 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_H');
741 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'$formula.'/'run5/*_15*.log
... /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_H/');
742 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'$formula.'/' && tar -cvvf
... '$formula.'_HF-3-21G_H.tar /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_H/');
743
744 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
745
746 my $local_path21 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'';
747 my $remote_path21 = '/netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_H.tar';
748 my $ssh21 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
749 $ssh21->scp_get($remote_path21, $local_path21)
750     or die "scp failed: " . $ssh21->error;
751 my $cmd21 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H.tar';
752 system($cmd21);
753
754 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-6-31Gdp/netscr/dcrizer/
... '$formula.'/'.'$formula.'_HF-3-21G_H/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/') or die "Copy failed:
... $!";
755 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
... then moves them to an appropriate directory
756
757 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/geometry_extract.plx')
... or die "Copy failed: $!";
758 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/output.txt') or die
... "Copy failed: $!";
759 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/' or die '$!';
760 #moves script and needed list of ID numbers to the folder where output files are and then we move to
... that directory
761
762
763 print "$lc\n";
764 print "$lc2\n";
765 open my $fh, '<', 'output.txt';
766
767 my $ID;
768 chomp(my @lines = <$fh>);
769 for my $lines ( @lines ) {
770     if ( $lines =~ /ID=(.+)&Units/ ) {
771
772         $ID = $1;
773         my $lc = 0;
774 copy
... ('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-6-31Gdp/.$ID.'_convert.txt
... ',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/.$ID.'_convert.txt')
... or die "Copy failed $!";
775 open my $convertout2, '<', '$ID.'_convert.txt' or die($!);
776 $lc++ while <$convertout2>;
777 close $convertout2;
778 print $lc, "\n";
779
780 my $line_number2 = 0;
781 do{
782 my $cmd22 = 'perl geometry_extract.plx < '$ID.'_15_.$line_number2.'.log >
... '$ID.'_16_.$line_number2.'.com';
783 system($cmd22);
784 open (DATAIN, '<'.$ID.'_16_.$line_number2.'.com');
785 open (DATAOUT, '>'.$ID.'_17_.$line_number2.'.com');
786 while (<DATAIN>){
787     print DATAOUT unless ($. == 1);
788 }

```

```

789 close DATAOUT;
790 close DATAIN;
791
792 $proc6 = 4;
793 $processors6 = '%nprocshared='.$proc6.'';
794
795 $mem6 = 40;
796 $memory6 = '%mem='.$mem6.'MW';
797
798 $key6 = '# opt b3lyp/3-21g';
799 $title6 = '.$ID.'';
800 open my $input6, '>', 'header6_'.$ID.'_'.$line_number2.'.txt' or die "$1";
801 print $input6 "$processors6\n";
802 print $input6 "$memory6\n";
803
804 print $input6 "$key6\n";
805 print $input6 "\n";
806 print $input6 "$title6\n";
807 print $input6 "\n";
808 print $input6 "1\t1\n";
809
810 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_H/merge_files.pl') or die
... "Copy failed: $!";
811 #This copies a script called merge_files.pl to the specific formula folder we are working in
812
813 my $cmd23 = 'perl merge_files.pl header6_'.$ID.'_'.$line_number2.'.txt
... '.$ID.'_17_'.$line_number2.'.com '.$ID.'_18_'.$line_number2.'.com';
814 system($cmd23);
815 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
... input file
816
817 $line_number2=($line_number2)+1;
818 print "$lc\n";
819 print "$line_number2\n";
820 }while($line_number2 < $lc);
821
822 }
823 }
824
825 close $fh;
826
827
828
829 #executes the script to strip geometries for all output files and places the results into new input
... files.
830
831 my $cmd24 = 'tar -cvvf '.$formula.'_6.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_H/';
832 system($cmd24);
833
834 my $local_path22 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_H/'.$formula.'_6.tar/';
835 my $remote_path22 = "/netscr/dcrizer";
836 my $ssh22 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
837 $ssh22->scp_put($local_path22, $remote_path22)
838 or die "scp failed: " . $ssh22->error;
839
840 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run6/');
841 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_HF-3-21G_H/');
842 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
... /netscr/'.$user.'/'.$formula.'_6.tar');
843 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_H/
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_HF-3-21G_H/');

```



```

844 print "$stdout\n";
845 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/.$user.'/'.'$formula.'/'_HF-3-21G_H/'.'$formula.'_HF-3-21G_H/*_18_*.com
... /netscr/.$user.'/'.'$formula.'/'run6/');
846
847 my $local_path23 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
848 my $remote_path23 = '/netscr/.$user.'/'.'$formula.'/'run6/';
849 my $ssh23 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
850 $ssh23->scp_put($local_path23, $remote_path23)
851     or die "scp failed: " . $ssh23->error;
852 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'$formula.'/'run6/ && ./run.sh');
853
854 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output7.txt');
855 do{
856     sleep(300);
857     print "Running protonated compounds at B3LYP/3-21G\n";
858     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/' && rm output7.txt');
859     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/' && bjobs -l > output7.txt');
860     my $local_path24 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
861     my $remote_path24 = '/netscr/.$user.'/'output7.txt';
862     my $ssh24 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
863     $ssh24->scp_get($remote_path24, $local_path24)
864         or die "scp failed: " . $ssh24->error;
865     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output7.txt");
866
867
868 print "DONE*6\n";
869 #####
870
871 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/.$user.'/'.'$formula.'/'_B3LYP-3-21G_H');
872 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'$formula.'/'run6/*_18_*.log
... /netscr/.$user.'/'.'$formula.'/'_B3LYP-3-21G_H/');
873 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'$formula.'/' && tar -cvvf
... '$formula.'_B3LYP-3-21G_H.tar /netscr/.$user.'/'.'$formula.'/'_B3LYP-3-21G_H/');
874
875 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
876
877 my $local_path25 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'';
878 my $remote_path25 = '/netscr/.$user.'/'.'$formula.'/'_B3LYP-3-21G_H.tar';
879 my $ssh25 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
880 $ssh25->scp_get($remote_path25, $local_path25)
881     or die "scp failed: " . $ssh25->error;
882 my $cmd25 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-3-21G_H.tar';
883 system($cmd25);
884
885 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_H/netscr/dcrizer/.$
... formula.'/'.'$formula.'_B3LYP-3-21G_H/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-3-21G_H/') or die "Copy
... failed: $!";
886 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
... then moves them to an appropriate directory
887
888 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-3-21G_H/geometry_extract.plx
... ') or die "Copy failed: $!";
889 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-3-21G_H/output.txt') or die
... "Copy failed: $!";
890 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-3-21G_H/' or die
... '$!';
891 #moves script and needed list of ID numbers to the folder where output files are and then we move to
... that directory
892

```

```

893 open my $fh, '<', 'output.txt';
894
895 my $ID;
896 chomp(my @lines = <$fh>);
897 for my $lines ( @lines ) {
898     if ( $lines =~ /ID=(.+)&Units/ ) {
899
900         $ID = $1;
901         my $lc = 0;
902     copy
903     ( '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/' . $ID . '_convert.txt'
904     ,
905     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G_H/' . $ID . '_convert.txt'
906     ) or die "Copy failed $1";
907 open my $convertout2, '<', '$ID._convert.txt' or die($!);
908 $lc++ while <$convertout2>;
909 close $convertout2;
910 print $lc, "\n";
911
912 my $line_number3 = 0;
913 do{
914 my $cmd26 = 'perl geometry_extract.plx < '$ID.'_18_'$line_number3.'.log >
915 '$ID.'_19_'$line_number3.'.com';
916 system($cmd26);
917 open (DATAIN, '<' . $ID . '_19_' . $line_number3 . '.com');
918 open (DATAOUT, '>' . $ID . '_20_' . $line_number3 . '.com');
919 while (<DATAIN>){
920     print DATAOUT unless ($. == 1);
921 }
922 close DATAOUT;
923 close DATAIN;
924
925 $proc7 = 4;
926 $processors7 = '%nprocshared=' . $proc . ' ';
927
928 $mem7 = 40;
929 $memory7 = '%mem=' . $mem . 'MW';
930
931 $key7 = '# opt b3lyp/6-31g';
932 $title7 = '$ID.';
933 open my $input7, '>>', 'header7_' . $ID . '_' . $line_number3 . '.txt' or die "$1";
934 print $input7 "$processors7\n";
935 print $input7 "$memory7\n";
936
937 print $input7 "$key7\n";
938 print $input7 "\n";
939 print $input7 "$title7\n";
940 print $input7 "\n";
941 print $input7 "1\t1\n";
942
943 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
944     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-3-21G_H/merge_files.pl') or
945     die "Copy failed: $!";
946 #This copies a script called merge_files.pl to the specific formula folder we are working in
947
948 my $cmd27 = 'perl merge_files.pl header7_' . $ID . '_' . $line_number3 . '.txt
949 '$ID.'_20_'$line_number3.'.com '$ID.'_21_'$line_number3.'.com';
950 system($cmd27);
951 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
952 input file
953
954 $line_number3=($line_number3)+1;
955 print "$lc\n";
956 print "$line_number3\n";
957 }while($line_number3 < $lc);
958 }
959 }
960 close $fh;

```

```

952
953
954 my $cmd28 = 'tar -cvvf '.$formula.'_7.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/';
955 system($cmd28);
956
957 my $local_path26 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/'.$formula.'_7.tar/'
... ;
958 my $remote_path26 = "/netscr/dcrizer";
959 my $ssh26 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
960 $ssh26->scp_put($local_path26, $remote_path26)
961     or die "scp failed: " . $ssh26->error;
962
963 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run7/');
964 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/');
965 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
... /netscr/'.$user.'/'.$formula.'_7.tar');
966 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/');
967 print "$stdout\n";
968 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_H/'.$formula.'_B3LYP-3-21G_H/*_21_*.com
... /netscr/'.$user.'/'.$formula.'/run7/');
969
970 my $local_path27 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
971 my $remote_path27 = '/netscr/'.$user.'/'.$formula.'/run7/';
972 my $ssh27 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
973 $ssh27->scp_put($local_path27, $remote_path27)
974     or die "scp failed: " . $ssh27->error;
975 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/run7/ && ./run.sh');
976
977 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output8.txt');
978 do{
979     sleep(300);
980     print "Running protonated compounds at B3LYP/6-31G\n";
981     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && rm output8.txt');
982     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && bjobs -l > output8.txt');
983     my $local_path28 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
984     my $remote_path28 = '/netscr/'.$user.'/output8.txt';
985     my $ssh28 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
986     $ssh28->scp_get($remote_path28, $local_path28)
987         or die "scp failed: " . $ssh28->error;
988     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output8.txt");
989
990 print "DONE*7\n";
991 #####
992
993 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G_H/');
994 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/'.$user.'/'.$formula.'/run7/*_21_*.log
... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G_H/');
995 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/ && tar -cvvf
... '.$formula.'_B3LYP-6-31G_H.tar /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G_H/');
996
997 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
998
999 my $local_path29 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/';
1000 my $remote_path29 = '/netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-6-31G_H.tar';
1001 my $ssh29 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1002 $ssh29->scp_get($remote_path29, $local_path29)
1003     or die "scp failed: " . $ssh29->error;
1004 my $cmd29 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G_H.tar';
1005 system($cmd29);
1006

```

```

1007 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-3-21G_H/netscr/dcrizer/
... '$formula.'/',$formula.'_B3LYP-6-31G_H/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31G_H/') or die "Copy
failed: $!";
1008 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
... then moves them to an appropriate directory
1009
1010 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31G_H/geometry_extract.plx
... ') or die "Copy failed: $!";
1011 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31G_H/output.txt') or die
"COPY failed: $!";
1012 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31G_H/' or die
... '$!.';
1013 #moves script and needed list of ID numbers to the folder where output files are and then we move to
... that directory
1014
1015 open my $fh, '<', 'output.txt';
1016
1017 my $ID;
1018 chomp(my @lines = <$fh>);
1019 for my $lines ( @lines ) {
1020     if ( $lines =~ /ID=(.+)&Units/ ) {
1021
1022         $ID = $1;
1023         my $lc = 0;
1024
1025         copy
... ('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp/',$ID.'_convert.txt
... ',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31G_H/',$ID.'_convert.txt'
... ) or die "Copy failed $1";
1025 open my $convertout2, '<', '$ID.'_convert.txt' or die($!);
1026 $lc++ while <$convertout2>;
1027 close $convertout2;
1028 print $lc, "\n";
1029
1030 my $line_number4 = 0;
1031 do{
1032 my $cmd30 = 'perl geometry_extract.plx < '$ID.'_21_',$line_number4.'.log >
... '$ID.'_22_',$line_number4.'.com';
1033 system($cmd30);
1034 open (DATAIN,'<',$ID.'_22_',$line_number4.'.com');
1035 open (DATAOUT,'>',$ID.'_23_',$line_number4.'.com');
1036 while (<DATAIN>){
1037     print DATAOUT unless ($. == 1);
1038 }
1039 close DATAOUT;
1040 close DATAIN;
1041
1042     $proc8 = 4;
1043     $processors8 = '%nprocshared='.$proc.'';
1044
1045     $mem8 = 40;
1046     $memory8 = '%mem='.$mem.'MW';
1047
1048     $key8 = '# opt freq b3lyp/6-31g(d,p)';
1049     $title8 = '$ID.';
1050     open my $input8, '>',$ID.'_',$line_number4.'.txt' or die "$1";
1051     print $input8 "$processors8\n";
1052     print $input8 "$memory8\n";
1053
1054     print $input8 "$key8\n";
1055     print $input8 "\n";
1056     print $input8 "$title8\n";
1057     print $input8 "\n";
1058     print $input8 "1\t1\n";
1059

```



```

1060 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/merge_files.pl') or
... die "Copy failed: $!";
1061 #This copies a script called merge_files.pl to the specific formula folder we are working in
1062
1063 my $cmd31 = 'perl merge_files.pl header8'.'$ID.'_'.'$line_number4.'.txt
... '$ID.'_'_23_'.'$line_number4.'.com '$ID.'_'_24_'.'$line_number4.'.com';
1064 system($cmd31);
1065 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
... input file
1066
1067 $line_number4=($line_number4)+1;
1068 print "$lc\n";
1069 print "$line_number4\n";
1070 }while($line_number4 < $lc);
1071 }
1072 }
1073 close $fh;
1074
1075 #executes the script to strip geometries for all output files and places the results into new input
... files.
1076
1077 my $cmd32 = 'tar -cvvf '$formula.'_8.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/';
1078 system($cmd32);
1079
1080 my $local_path30 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/'.'$formula.'_8.tar/'
... ;
1081 my $remote_path30 = "/netscr/dcrizer";
1082 my $ssh30 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1083 $ssh30->scp_put($local_path30, $remote_path30)
1084 or die "scp failed: " . $ssh30->error;
1085
1086 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.'$user.'/'.'$formula.'/run8/');
1087 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/'.'$user.'/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/');
1088 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.'$user.'/' && tar -xvf
... /netscr/'.'$user.'/'.'$formula.'_8.tar');
1089 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.'$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/
... /netscr/'.'$user.'/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/');
1090 print "$stdout\n";
1091 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/'.'$user.'/'.'$formula.'/'.'$formula.'_B3LYP-6-31G_H/'.'$formula.'_B3LYP-6-31G_H/*_24*.com
... /netscr/'.'$user.'/'.'$formula.'/run8/');
1092
1093 my $local_path31 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
1094 my $remote_path31 = '/netscr/'.'$user.'/'.'$formula.'/run8/';
1095 my $ssh31 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1096 $ssh31->scp_put($local_path31, $remote_path31)
1097 or die "scp failed: " . $ssh31->error;
1098 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.'$user.'/'.'$formula.'/run8/ && ./run.sh');
1099
1100 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output9.txt');
1101 do{
1102 sleep(300);
1103 print "Running protonated compounds at B3LYP/6-31G(d,p)\n";
1104 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.'$user.'/' && rm output9.txt');
1105 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.'$user.'/' && bjobs -l > output9.txt');
1106 my $local_path32 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
1107 my $remote_path32 = '/netscr/'.'$user.'/'output9.txt';
1108 my $ssh32 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1109 $ssh32->scp_get($remote_path32, $local_path32)
1110 or die "scp failed: " . $ssh32->error;
1111 } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output9.txt");
1112
1113 print "DONE*8\n";
1114 #####

```

```

1115 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H');
1116 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/' . $user . '/' . $formula . '/run8/*_24*.log
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H/');
1117 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' . $formula . '/' && tar -cvvf
... '$formula.' . '_B3LYP-6-31Gdp_H.tar' /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H/');
1118
1119 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
1120
1121 my $local_path33 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/';
1122 my $remote_path33 = '/netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H.tar';
1123 my $ssh33 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1124 $ssh33->scp_get($remote_path33, $local_path33)
1125     or die "scp failed: " . $ssh33->error;
1126 my $cmd33 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H.tar';
1127 system($cmd33);
1128
1129 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H/netscr/dcrizer/
... '$formula . '/' . $formula . '_B3LYP-6-31Gdp_H/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp_H/') or die "Copy
... failed: $!";
1130 #secure copies output files to a directory on you computer when they are then untared and then moves
... them to an appropriate directory
1131
1132 # so that at this point all calculations are done for the neutral and protonated species and all
... files have been moved to the local computer
1133
1134 #####
1135 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/' or die
... '$!';
1136 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/geometry_extract.plx
... ') or die "Copy failed: $!";
1137
1138 open my $fh, '<', 'output.txt';
1139
1140 my $ID;
1141 chomp(my @lines = <$fh>);
1142 for my $lines ( @lines ) {
1143     if ( $lines =~ /ID=(.+)&Units/ ) {
1144
1145         $ID = $1;
1146 my $cmd34 = 'perl geometry_extract.plx < '$ID.'_11.log > '$ID.'_25.com';
1147 system($cmd34);
1148
1149
1150 open (DATAIN, '<' . $ID . '_25.com');
1151 open (DATAOUT, '>' . $ID . '_26.com');
1152 while (<DATAIN>){
1153     print DATAOUT unless ($. == 1..2);
1154 }
1155 close DATAOUT;
1156 close DATAIN;
1157
1158
1159 $proc9 = 4;
1160 $processors9 = '%nprocshared=' . $proc . '';
1161
1162 $mem9 = 40;
1163 $memory9 = '%mem=' . $mem . 'MW';
1164
1165 $key9 = '# opt hf/3-21g';
1166 $title9 = '$ID.';
1167 open my $input9, '>>', 'header9_' . $ID . '.txt' or die "$!";

```

```

1168     print $input9 "$processors9\n";
1169     print $input9 "$memory9\n";
1170
1171     print $input9 "$key9\n";
1172     print $input9 "\n";
1173     print $input9 "$title9\n";
1174     print $input9 "\n";
1175     print $input9 "1\t2\n";
1176
1177 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/merge_files.pl') or
... die "Copy failed: $!";
1178 #This copies a script called merge_files.pl to the specific formula folder we are working in
1179
1180
1181 my $cmd35 = 'perl merge_files.pl header9_.$ID.'.txt '.$ID.'.26.com '.$ID.'.27.com';
1182 system($cmd35);
1183 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
... input file
1184
1185 }
1186 }
1187 }
1188
1189
1190 close my $fh;
1191
1192 my $cmd36 = 'tar -cvvf ' . $formula . '_9.tar
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/';
1193 system($cmd36);
1194
1195 my $local_path34 =
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/' . $formula . '_9.tar/'
... ;
1196 my $remote_path34 = "/netscr/dcrizer";
1197 my $ssh34 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1198 $ssh34->scp_put($local_path34, $remote_path34)
1199     or die "scp failed: " . $ssh34->error;
1200
1201 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/' . $user . '/' . $formula . '/run9/');
1202 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/' . $user . '/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/');
1203 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && tar -xvf
... /netscr/' . $user . '/' . $formula . '_9.tar');
1204 print "$stdout\n";
1205 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/' . $user . '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/' . $formula . '_B3LYP-6-31Gdp/*_27
... .com /netscr/' . $user . '/' . $formula . '/run9/');
1206
1207 my $local_path35 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
1208 my $remote_path35 = '/netscr/' . $user . '/' . $formula . '/run9/';
1209 my $ssh35 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1210 $ssh35->scp_put($local_path35, $remote_path35)
1211     or die "scp failed: " . $ssh35->error;
1212 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' . $formula . '/run9/ && ./run.sh');
1213
1214 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output10.txt');
1215 do{
1216     sleep(300);
1217     print "Running molecular cations at HF/3-21G\n";
1218     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && rm output10.txt');
1219     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/' . $user . '/' && bjobs -l > output10.txt');
1220     my $local_path36 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
1221     my $remote_path36 = '/netscr/' . $user . '/output10.txt';
1222 my $ssh36 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1223 $ssh36->scp_get($remote_path36, $local_path36)
1224     or die "scp failed: " . $ssh36->error;
1225 } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output10.txt");

```



```

1226 print "DONE*9\n";
1227 #####
1228 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
1229 /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_M-');
1230 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'$formula.'/'run9/*_27.log
1231 /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_M-');
1232 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'$formula.'/' && tar -cvvf
1233 '$formula.'_HF-3-21G_M-.tar /netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_M-');
1234 #makes a directory for the output files (.log) and then moves to another folder. We then change over
1235 to that directory and tar that directory to be easier to transfer
1236 my $local_path37 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-6-31Gdp/netscr/dcrizer/
1237 my $remote_path37 = '/netscr/.$user.'/'.'$formula.'/'.'$formula.'_HF-3-21G_M-.tar';
1238 my $ssh37 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1239 $ssh37->scp_get($remote_path37, $local_path37)
1240 or die "scp failed: " . $ssh37->error;
1241 my $cmd37 = 'tar -xvf
1242 /Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_M-.tar';
1243 system($cmd37);
1244 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_B3LYP-6-31Gdp/netscr/dcrizer/
1245 '$formula.'/'.'$formula.'_HF-3-21G_M-/',
1246 '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_M-') or die "Copy
1247 failed: $!";
1248 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
1249 then moves them to an appropriate directory
1250 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
1251 '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_M-/geometry_extract.plx')
1252 or die "Copy failed: $!";
1253 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'output.txt',
1254 '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_M-/output.txt') or die
1255 "Copy failed: $!";
1256 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/'.'$formula.'_HF-3-21G_M-' or die '$!';
1257 #moves script and needed list of ID numbers to the folder where output files are and then we move to
1258 that directory
1259 open my $fh, '<', 'output.txt';
1260 my $ID;
1261 chomp(my @lines = <$fh>);
1262 for my $lines ( @lines ) {
1263     if ( $lines =~ /ID=(.+)&Units/ ) {
1264         $ID = $1;
1265     }
1266 }
1267 my $cmd38 = 'perl geometry_extract.plx < '.$ID.'_27.log > '.$ID.'_28.com';
1268 system($cmd38);
1269 open (DATAIN, '<'.$ID.'_28.com');
1270 open (DATAOUT, '>'.$ID.'_29.com');
1271 while (<DATAIN>){
1272     print DATAOUT unless ($. == 1);
1273 }
1274 close DATAOUT;
1275 close DATAIN;
1276
1277 $proc10 = 4;
1278 $processors10 = '%nprocshared='.$proc.'';
1279
1280 $mem10 = 40;
1281 $memory10 = '%mem='.$mem.'MW';

```



```

1277     $key10 = '# opt b3lyp/3-21g';
1278     $title10 = ' '.$ID.'';
1279     open my $input10, '>>', 'header10_'.$ID.'.txt' or die "$1";
1280     print $input10 "$processors10\n";
1281     print $input10 "$memory10\n";
1282
1283     print $input10 "$key10\n";
1284     print $input10 "\n";
1285     print $input10 "$title10\n";
1286     print $input10 "\n";
1287     print $input10 "1\t2\n";
1288
1289     copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
1290         '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_M-/merge_files.pl') or
1291         die "Copy failed: $!";
1292     #This copies a script called merge_files.pl to the specific formula folder we are working in
1293
1294     my $cmd39 = 'perl merge_files.pl header10_'.$ID.'.txt '.$ID.'_29.com '.$ID.'_30.com';
1295     system($cmd39);
1296     #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
1297     input file
1298 }
1299 }
1300
1301 close $fh;
1302
1303
1304
1305 #executes the script to strip geometries for all output files and places the results into new input
1306 files.
1307
1308 my $cmd40 = 'tar -cvvf '.$formula.'_10.tar
1309 /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_M-/';
1310 system($cmd40);
1311
1312 my $local_path38 =
1313     '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_M-/'.$formula.'_10.tar/';
1314 my $remote_path38 = '/netscr/dcrizer';
1315 my $ssh38 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1316 $ssh38->scp_put($local_path38, $remote_path38)
1317     or die "scp failed: " . $ssh38->error;
1318
1319 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run10/');
1320 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
1321     /netscr/'.$user.'/'.$formula.'/'.$formula.'_HF-3-21G_M-/');
1322 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
1323     /netscr/'.$user.'/'.$formula.'_10.tar');
1324 my($stdout, $stderr, $exit) = $ssh->cmd('mv
1325     /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_HF-3-21G_M-/
1326     /netscr/'.$user.'/'.$formula.'/'.$formula.'_HF-3-21G_M-/');
1327 print "$stdout\n";
1328 my($stdout, $stderr, $exit) = $ssh->cmd('mv
1329     /netscr/'.$user.'/'.$formula.'/'.$formula.'_HF-3-21G_M-/'.$formula.'_HF-3-21G_M-/*_30.com
1330     /netscr/'.$user.'/'.$formula.'/run10/');
1331
1332 my $local_path39 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
1333 my $remote_path39 = '/netscr/'.$user.'/'.$formula.'/run10/';
1334 my $ssh39 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1335 $ssh39->scp_put($local_path39, $remote_path39)
1336     or die "scp failed: " . $ssh39->error;
1337 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/'.$formula.'/run10/ && ./run.sh');
1338
1339 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output11.txt');

```

```

1331 do{
1332 sleep(300);
1333 print "Running molecular cations at B3LYP/3-21G\n";
1334 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user./ && rm output11.txt');
1335 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user./ && bjobs -l > output11.txt');
1336 my $local_path40 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
1337 my $remote_path40 = '/netscr/.$user./output11.txt';
1338 my $ssh40 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1339 $ssh40->scp_get($remote_path40, $local_path40)
1340 or die "scp failed: " . $ssh40->error;
1341 } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output11.txt");
1342
1343
1344 print "DONE*10\n";
1345 #####
1346 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
...
/netscr/.$user./'.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-');
1347 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user./'.'.$formula.'/'_run10/*_30.log
...
/netscr/.$user./'.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-');
1348 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user./'.'.$formula.'/' && tar -cvvf
...
'.'.$formula.'_'_B3LYP-3-21G_M-.tar /netscr/.$user./'.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-');
1349
1350 #makes a directory for the output files (.log) and then moves to another folder. We then change over
...
to that directory and tar that directory to be easier to transfer
1351
1352 my $local_path41 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.';
1353 my $remote_path41 = '/netscr/.$user./'.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-.tar';
1354 my $ssh41 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1355 $ssh41->scp_get($remote_path41, $local_path41)
1356 or die "scp failed: " . $ssh41->error;
1357 my $cmd41 = 'tar -xvf
...
/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-.tar';
1358 system($cmd41);
1359
1360 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_HF-3-21G_M-/netscr/dcrizer/.'
...
.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-/',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-/') or die "Copy
...
failed: $!";
1361 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
...
then moves them to an appropriate directory
1362
1363 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-/geometry_extract.
...
plx') or die "Copy failed: $!";
1364 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'_output.txt',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-/output.txt') or die
...
"Copy failed: $!";
1365 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'.$formula.'/'.'.$formula.'_'_B3LYP-3-21G_M-/ or die
...
'.'.$1.';
1366 #moves script and needed list of ID numbers to the folder where output files are and then we move to
...
that directory
1367
1368
1369 open my $fh, '<', 'output.txt';
1370
1371 my $ID;
1372 chomp(my @lines = <$fh>);
1373 for my $lines ( @lines ) {
1374     if ( $lines =~ /ID=(.+)&Units/ ) {
1375
1376         $ID = $1;
1377
1378 my $cmd42 = 'perl geometry_extract.plx < ' . $ID . '_30.log > ' . $ID . '_31.com';
1379 system($cmd42);
1380 open (DATAIN, '<' . $ID . '_31.com');
1381 open (DATAOUT, '>' . $ID . '_32.com');
1382 while (<DATAIN>){

```

```

1383     print DATAOUT unless ($. == 1..2);
1384 }
1385 close DATAOUT;
1386 close DATAIN;
1387
1388     $proc11 = 4;
1389     $processors11 = '%nprocshared=%. $proc.';
1390
1391     $mem11 = 40;
1392     $memory11 = '%mem=%. $mem.'MW';
1393
1394     $key11 = '# opt b3lyp/6-31g';
1395     $title11 = ' '.$ID.'';
1396     open my $input11, '>>', 'header11_'.$ID.'.txt' or die "$1";
1397     print $input11 "$processors11\n";
1398     print $input11 "$memory11\n";
1399
1400     print $input11 "$key11\n";
1401     print $input11 "\n";
1402     print $input11 "$title11\n";
1403     print $input11 "\n";
1404     print $input11 "1\t2\n";
1405
1406     copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
1407 ... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/merge_files.pl') or
1408 ... die "Copy failed: $!";
1409 #This copies a script called merge_files.pl to the specific formula folder we are working in
1410
1411 my $cmd43 = 'perl merge_files.pl header11_'.$ID.'.txt '.$ID.'_32.com '.$ID.'_33.com';
1412 system($cmd43);
1413 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
1414 ... input file
1415
1416 }
1417
1418 close $fh;
1419
1420
1421
1422 #executes the script to strip geometries for all output files and places the results into new input
1423 ... files.
1424
1425 my $cmd44 = 'tar -cvvf '.$formula.'_11.tar
1426 ... /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/';
1427 system($cmd44);
1428
1429 my $local_path42 =
1430 ... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/'.$formula.'_11.tar
1431 ... /';
1432
1433 my $remote_path42 = "/netscr/dcrizer";
1434 my $ssh42 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1435 $ssh42->scp_put($local_path42, $remote_path42)
1436 ... or die "scp failed: " . $ssh42->error;
1437
1438 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/'.$user.'/'.$formula.'/run11/');
1439 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
1440 ... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/');
1441 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/'.$user.'/ && tar -xvf
1442 ... /netscr/'.$user.'/'.$formula.'_11.tar');
1443 my($stdout, $stderr, $exit) = $ssh->cmd('mv
1444 ... /netscr/'.$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/
1445 ... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/');
1446 print "$stdout\n";
1447 my($stdout, $stderr, $exit) = $ssh->cmd('mv
1448 ... /netscr/'.$user.'/'.$formula.'/'.$formula.'_B3LYP-3-21G_M-/'.$formula.'_B3LYP-3-21G_M-/*_33.com
1449 ... /netscr/'.$user.'/'.$formula.'/run11/');

```



```

1439
1440 my $local_path43 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
1441 my $remote_path43 = '/netscr/.$user.'/'.'.$formula.'/run11/';
1442 my $ssh43 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1443 $ssh43->scp_put($local_path43, $remote_path43)
1444     or die "scp failed: " . $ssh43->error;
1445 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'.$formula.'/run11/ && ./run.sh');
1446
1447 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output12.txt');
1448 do{
1449     sleep(300);
1450     print "Running molecular cations at B3LYP/6-31G\n";
1451     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/' && rm output12.txt');
1452     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/' && bjobs -l > output12.txt');
1453     my $local_path44 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
1454     my $remote_path44 = '/netscr/.$user.'/'output12.txt';
1455     my $ssh44 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1456     $ssh44->scp_get($remote_path44, $local_path44)
1457         or die "scp failed: " . $ssh44->error;
1458     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output12.txt");
1459
1460
1461 print "DONE*11\n";
1462 #####
1463 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
...
/netscr/.$user.'/'.'.$formula.'/.'.$formula.'_B3LYP-6-31G_M-');
1464 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.$user.'/'.'.$formula.'/run11/*_33.log
...
/netscr/.$user.'/'.'.$formula.'/.'.$formula.'_B3LYP-6-31G_M-');
1465 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.$user.'/'.'.$formula.'/ && tar -cvvf
...
'.'.$formula.'_B3LYP-6-31G_M-.tar /netscr/.$user.'/'.'.$formula.'/.'.$formula.'_B3LYP-6-31G_M-');
1466
1467 #makes a directory for the output files (.log) and then moves to another folder. We then change over
...
to that directory and tar that directory to be easier to transfer
1468
1469 my $local_path45 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/';
1470 my $remote_path45 = '/netscr/.$user.'/'.'.$formula.'/.'.$formula.'_B3LYP-6-31G_M-.tar';
1471 my $ssh45 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1472 $ssh45->scp_get($remote_path45, $local_path45)
1473     or die "scp failed: " . $ssh45->error;
1474 my $cmd45 = 'tar -xvf
...
/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-6-31G_M-.tar';
1475 system($cmd45);
1476
1477 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-3-21G_M-/netscr/dcrizer
...
/.'.$formula.'/.'.$formula.'_B3LYP-6-31G_M-/',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-6-31G_M-') or die "Copy
...
failed: $!";
1478 #secure copies HF/3-21G output files to a directory on you computer when they are then untared and
...
then moves them to an appropriate directory
1479
1480 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/geometry_extract.plx',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-6-31G_M-/geometry_extract.
...
plx') or die "Copy failed: $!";
1481 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/output.txt',
...
'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-6-31G_M-/output.txt') or die
...
"Copy failed: $!";
1482 chmod '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula.'/.'.$formula.'_B3LYP-6-31G_M-/ or die
...
'.'.$1.';
1483 #moves script and needed list of ID numbers to the folder where output files are and then we move to
...
that directory
1484
1485
1486 open my $fh, '<', 'output.txt';
1487
1488 my $ID;
1489 chomp(my @lines = <$fh>);
1490 for my $lines ( @lines ) {
1491     if ( $lines =~ /ID=(.+)&Units/ ) {

```

```

1492
1493     $ID = $1;
1494
1495 my $cmd46 = 'perl geometry_extract.plx < '.$ID.'_33.log > '.$ID.'_34.com';
1496 system($cmd46);
1497 open (DATAIN,'<'.$ID.'_34.com');
1498 open (DATAOUT,'>'.$ID.'_35.com');
1499 while (<DATAIN){
1500     print DATAOUT unless ($. == 1..2);
1501 }
1502 close DATAOUT;
1503 close DATAIN;
1504
1505 $proc12 = 4;
1506 $processors12 = '%nprocshared='.$proc.'';
1507
1508 $mem12 = 40;
1509 $memory12 = '%mem='.$mem.'MW';
1510
1511 $key12 = '# opt freq b3lyp/6-31g(d,p)';
1512 $title12 = ''.$ID.'';
1513 open my $input12, '>>', 'header12_'.$ID.'.txt' or die "$1";
1514 print $input12 "$processors12\n";
1515 print $input12 "$memory12\n";
1516
1517 print $input12 "$key12\n";
1518 print $input12 "\n";
1519 print $input12 "$title12\n";
1520 print $input12 "\n";
1521 print $input12 "1\t2\n";
1522
1523 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/merge_files.pl',
...
1524 #This copies a script called merge_files.pl to the specific formula folder we are working in
1525
1526 my $cmd47 = 'perl merge_files.pl header12_'.$ID.'.txt '.$ID.'_35.com '.$ID.'_36.com';
1527 system($cmd47);
1528 #these above 2 lines combine the header and the geometry info together to make the perfect gaussian
...
1529 input file
1530
1531 }
1532 }
1533 }
1534
1535 close $fh;
1536
1537
1538
1539 #executes the script to strip geometries for all output files and places the results into new input
...
1540 files.
1541
1542 my $cmd48 = 'tar -cvvf '.$formula.'_12.tar
...
1543 /Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G_M-/';
1544 system($cmd48);
1545
1546 my $local_path46 =
...
1547 '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/'.$formula.'_B3LYP-6-31G_M-/'.$formula.'_12.tar
...
1548 /';
1549 my $remote_path46 = "/netscr/dcrizer";
1550 my $ssh46 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1551 $ssh46->scp_put($local_path46, $remote_path46)
1552 or die "scp failed: " . $ssh46->error;

```

```

1549
1550 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p /netscr/.'$user.'/'$formula.'/run12/');
1551 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31G_M-/');
1552 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.'$user.'/ && tar -xvf
... /netscr/.'$user.'/'$formula.'_12.tar');
1553 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/.'$user.'/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/.'$formula.'_B3LYP-6-31G_M-/
... /netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31G_M-/');
1554 print "$stdout\n";
1555 my($stdout, $stderr, $exit) = $ssh->cmd('mv
... /netscr/.'$user.'/'$formula.'_B3LYP-6-31G_M-/.'$formula.'_B3LYP-6-31G_M-/*_36.com
... /netscr/.'$user.'/'$formula.'/run12/');
1556
1557 my $local_path47 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/run.sh';
1558 my $remote_path47 = '/netscr/.'$user.'/'$formula.'/run12/';
1559 my $ssh47 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1560 $ssh47->scp_put($local_path47, $remote_path47)
1561     or die "scp failed: " . $ssh47->error;
1562 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.'$user.'/'$formula.'/run12/ && ./run.sh');
1563
1564 my($stdout, $stderr, $exit) = $ssh->cmd('bjobs -l > output13.txt');
1565 do{
1566     sleep(300);
1567     print "Running molecular cations at B3LYP/6-31G(d,p)\n";
1568     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.'$user.'/ && rm output13.txt');
1569     my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.'$user.'/ && bjobs -l > output13.txt');
1570     my $local_path48 = '/Volumes/MAIN/Dropbox/DMC/Research/Perl/';
1571     my $remote_path48 = '/netscr/.'$user.'/output13.txt';
1572     my $ssh48 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1573     $ssh48->scp_get($remote_path48, $local_path48)
1574         or die "scp failed: " . $ssh48->error;
1575     } until (-z "/Volumes/MAIN/Dropbox/DMC/Research/Perl/output13.txt");
1576
1577
1578 print "DONE*12\n";
1579 #####
1580 my($stdout, $stderr, $exit) = $ssh->cmd('mkdir -p
... /netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-');
1581 my($stdout, $stderr, $exit) = $ssh->cmd('mv /netscr/.'$user.'/'$formula.'/run12/*_36.log
... /netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-/');
1582 my($stdout, $stderr, $exit) = $ssh->cmd('cd /netscr/.'$user.'/'$formula.'/ && tar -cvvf
... '$formula.'_B3LYP-6-31Gdp_M-.tar /netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-/');
1583
1584 #makes a directory for the output files (.log) and then moves to another folder. We then change over
... to that directory and tar that directory to be easier to transfer
1585
1586 my $local_path49 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/';
1587 my $remote_path49 = '/netscr/.'$user.'/'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-.tar';
1588 my $ssh49 = Net::OpenSSH->new(host => $host, user => $user, password => $password);
1589 $ssh49->scp_get($remote_path49, $local_path49)
1590     or die "scp failed: " . $ssh49->error;
1591 my $cmd49 = 'tar -xvf
... /Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-.tar';
1592 system($cmd49);
1593
1594 move('/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/.'$formula.'_B3LYP-6-31G_M-/netscr/dcrizer
... /.'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-/',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/.'$formula.'_B3LYP-6-31Gdp_M-/') or die "Copy
... failed: $!";
1595 #####
1596 $dirname2 = '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/Workup';
1597 mkdir $dirname2;
1598 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.'$formula.'/Workup/' or die '$!';
1599

```



```

1600 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/output.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/output.txt') or die "Copy failed: $!";
1601
1602 open my $fh, '<', 'output.txt';
1603
1604 my $ID;
1605 chomp(my @lines = <$fh>);
1606     for my $lines ( @lines ) {
1607         if ( $lines =~ /ID=(.+)&Units/ ) {
1608
1609             $ID = $1;
1610
1611 copy
... ('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp/',$ID.'_convert.txt
... ', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/',$ID.'_convert.txt') or die "Copy
... failed $1";
1612 open my $convertout2, '<', '$ID.'_convert.txt' or die($!);
1613 $lc++ while <$convertout2>;
1614 close $convertout2;
1615 print $lc, "\n";
1616
1617 my $line_number5 = 0;
1618 do{
1619 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp_M-/',$ID.'_36.
... log', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/',$ID.'_36.log') or die "Copy
... failed: $!";
1620
1621 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp/',$ID.'_11.log'
... ', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/',$ID.'_11.log') or die "Copy failed:
... $!";
1622
1623 open my $fh10, '<',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp_H/',$ID.'_24_'.
... $line_number5.'.log';
1624 if (-e $fh10){
1625
1626
1627
1628 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/',$formula.'_B3LYP-6-31Gdp_H/',$ID.'_24_'.
... $line_number5.'.log',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/',$ID.'_24_'. $line_number5.'.log') or
... die "Copy failed: $!";}
1629
1630 $line_number5=($line_number5)+1;
1631 print "$lc\n";
1632 print "$line_number5\n";
1633 }while($line_number5 < $lc);
1634 }
1635 }
1636 close $fh;
1637
1638 chdir '/Volumes/MAIN/Dropbox/DMC/Research/Perl' or die "$1";
1639
1640 my $cmd50 = 'perl workup.pl '$formula.'';
1641 system($cmd50);
1642 #submits command to run the program to workup data
1643
1644 my $cmd51 = 'perl email.pl '$formula.' '$email.'';
1645 system($cmd51);
1646 #runs program to tell user that program is finished running

```

APPENDIX 4

FULL CODE OF GUI.PL

```
1 #!/usr/bin/perl
2 use Tk;
3
4 # Main Window
5 my $mw = new MainWindow;
6
7 my $lab = $mw -> Label(-text=>"This is the Ion Structure Calculation Program!!!", -font =>"Times 20") ->
8
9 my $lab1 = $mw -> Label(-text=>"Enter Formula:") -> pack();
10 my $ent1 = $mw -> Entry() -> pack();
11 my $lab2 = $mw -> Label(-text=>"KillDevil Username:") -> pack();
12 my $ent2 = $mw -> Entry() -> pack();
13 my $lab3 = $mw -> Label(-text=>"KillDevil Password:") -> pack();
14 my $ent3 = $mw -> Entry(-show=>"*") -> pack();
15 my $lab4 = $mw -> Label(-text=>"Email:") -> pack();
16 my $ent4 = $mw -> Entry() -> pack();
17 my $but = $mw -> Button(-text => "START",
18                         -command =>\&push_button);
19 $but -> pack();
20
21 MainLoop;
22
23 #This is executed when the button is pressed
24 sub push_button {
25
26     $formula = $ent1 -> get();
27     print "$formula\n";
28     $user = $ent2 -> get();
29     print "$user\n";
30     $password = $ent3 -> get();
31     print "$password\n";
32     $email = $ent4 -> get();
33     print "$email\n";
34
35     my $cmd1 = 'perl example.pl '.$formula.' '.$user.' '.$password.' '.$email.'';
36     system($cmd1);
37 };
```


APPENDIX 5

FULL CODE OF GEOMETRY_EXTRACT.PLX

```
1  #!/usr/bin/perl -w
2
3  # This program is a modified version of a program from
4  #Maciej Haranczyk and Sanliang Ling, 2007
5
6
7  $n_atoms = 0;
8  while ($line = <STDIN>) {
9      if($line =~ /Input orientation/) {
10
11          $line = <STDIN>;
12          $line = <STDIN>;
13          $line = <STDIN>;
14          $line = <STDIN>;
15
16          $line = <STDIN>;
17
18          $n_atoms = 0;
19
20          do {
21
22              chop($line);
23
24              if ($line =~ /\S/) {
25                  $n_atoms++;
26                  @entries = split(/\s+/, $line);
27                  for($j = 1; $j <= $n_atoms; $j++) {
28                      if($entries[2] == 1) {
29                          $symb[$n_atoms] = "H";}
30                      elsif ($entries[2] == 2) {
31                          $symb[$n_atoms] = "He";}
32                      elsif ($entries[2] == 3) {
33                          $symb[$n_atoms] = "Li";}
34                      elsif ($entries[2] == 4) {
35                          $symb[$n_atoms] = "Be";}
36                      elsif ($entries[2] == 5) {
37                          $symb[$n_atoms] = "B";}
38                      elsif ($entries[2] == 6) {
39                          $symb[$n_atoms] = "C";}
40                      elsif ($entries[2] == 7) {
41                          $symb[$n_atoms] = "N";}
42                      elsif ($entries[2] == 8) {
43                          $symb[$n_atoms] = "O";}
44                      elsif ($entries[2] == 9) {
45                          $symb[$n_atoms] = "F";}
46                      elsif ($entries[2] == 10) {
47                          $symb[$n_atoms] = "Ne";}
48                      elsif ($entries[2] == 11) {
49                          $symb[$n_atoms] = "Na";}
50                      elsif ($entries[2] == 12) {
51                          $symb[$n_atoms] = "Mg";}
52                      elsif ($entries[2] == 13) {
53                          $symb[$n_atoms] = "Al";}
54                      elsif ($entries[2] == 14) {
```

```

55     $symb[$n_atoms] = "Si";}
56     elseif ($entries[2] == 15) {
57     $symb[$n_atoms] = "P";}
58     elseif ($entries[2] == 16) {
59     $symb[$n_atoms] = "S";}
60     elseif ($entries[2] == 17) {
61     $symb[$n_atoms] = "Cl";}
62
63     elseif ($entries[2] == 18) {
64     $symb[$n_atoms] = "Ar";}
65     elseif ($entries[2] == 19) {
66     $symb[$n_atoms] = "K";}
67     elseif ($entries[2] == 20) {
68     $symb[$n_atoms] = "Ca";}
69     elseif ($entries[2] == 21) {
70     $symb[$n_atoms] = "Sc";}
71     elseif ($entries[2] == 22) {
72     $symb[$n_atoms] = "Ti";}
73     elseif ($entries[2] == 23) {
74     $symb[$n_atoms] = "V";}
75     elseif ($entries[2] == 24) {
76     $symb[$n_atoms] = "Cr";}
77     elseif ($entries[2] == 25) {
78     $symb[$n_atoms] = "Mn";}
79     elseif ($entries[2] == 26) {
80     $symb[$n_atoms] = "Fe";}
81     elseif ($entries[2] == 27) {
82     $symb[$n_atoms] = "Co";}
83     elseif ($entries[2] == 28) {
84     $symb[$n_atoms] = "Ni";}
85     elseif ($entries[2] == 29) {
86     $symb[$n_atoms] = "Cu";}
87     elseif ($entries[2] == 30) {
88     $symb[$n_atoms] = "Zn";}
89     elseif ($entries[2] == 31) {
90     $symb[$n_atoms] = "Ga";}
91     elseif ($entries[2] == 32) {
92     $symb[$n_atoms] = "Ge";}
93     elseif ($entries[2] == 33) {
94     $symb[$n_atoms] = "As";}
95     elseif ($entries[2] == 34) {
96     $symb[$n_atoms] = "Se";}
97     elseif ($entries[2] == 35) {
98     $symb[$n_atoms] = "Br";}
99     elseif ($entries[2] == 36) {
100    $symb[$n_atoms] = "Kr";}
101    elseif ($entries[2] == 37) {
102    $symb[$n_atoms] = "Rb";}
103    elseif ($entries[2] == 38) {
104    $symb[$n_atoms] = "Sr";}
105    elseif ($entries[2] == 39) {
106    $symb[$n_atoms] = "Y";}
107    elseif ($entries[2] == 40) {
108    $symb[$n_atoms] = "Zr";}
109    elseif ($entries[2] == 41) {
110    $symb[$n_atoms] = "Nb";}
111    elseif ($entries[2] == 42) {
112    $symb[$n_atoms] = "Mo";}
113    elseif ($entries[2] == 43) {
114    $symb[$n_atoms] = "Tc";}
115    elseif ($entries[2] == 44) {
116    $symb[$n_atoms] = "Ru";}
117    elseif ($entries[2] == 45) {
118    $symb[$n_atoms] = "Rh";}
119    elseif ($entries[2] == 46) {
120    $symb[$n_atoms] = "Pd";}
121    elseif ($entries[2] == 47) {
122    $symb[$n_atoms] = "Ag";}
    elseif ($entries[2] == 48) {

```

```

123     $symb[$n_atoms] = "Cd";}
124     elsif ($entries[2] == 49) {
125     $symb[$n_atoms] = "In";}
126     elsif ($entries[2] == 50) {
127     $symb[$n_atoms] = "Sn";}
128     elsif ($entries[2] == 51) {
129     $symb[$n_atoms] = "Sb";}
130     elsif ($entries[2] == 52) {
131     $symb[$n_atoms] = "Te";}
132     elsif ($entries[2] == 53) {
133     $symb[$n_atoms] = "I";}
134     elsif ($entries[2] == 54) {
135     $symb[$n_atoms] = "Xe";}
136     elsif ($entries[2] == 55) {
137     $symb[$n_atoms] = "Cs";}
138     elsif ($entries[2] == 56) {
139     $symb[$n_atoms] = "Ba";}
140     elsif ($entries[2] == 57) {
141     $symb[$n_atoms] = "La";}
142     elsif ($entries[2] == 58) {
143     $symb[$n_atoms] = "Ce";}
144     elsif ($entries[2] == 59) {
145     $symb[$n_atoms] = "Pr";}
146     elsif ($entries[2] == 60) {
147     $symb[$n_atoms] = "Nd";}
148     elsif ($entries[2] == 61) {
149     $symb[$n_atoms] = "Pm";}
150     elsif ($entries[2] == 62) {
151     $symb[$n_atoms] = "Sm";}
152     elsif ($entries[2] == 63) {
153     $symb[$n_atoms] = "Eu";}
154     elsif ($entries[2] == 64) {
155     $symb[$n_atoms] = "Gd";}
156     elsif ($entries[2] == 65) {
157     $symb[$n_atoms] = "Tb";}
158     elsif ($entries[2] == 66) {
159     $symb[$n_atoms] = "Dy";}
160     elsif ($entries[2] == 67) {
161     $symb[$n_atoms] = "Ho";}
162     elsif ($entries[2] == 68) {
163     $symb[$n_atoms] = "Er";}
164     elsif ($entries[2] == 69) {
165     $symb[$n_atoms] = "Tm";}
166     elsif ($entries[2] == 70) {
167     $symb[$n_atoms] = "Yb";}
168     elsif ($entries[2] == 71) {
169     $symb[$n_atoms] = "Lu";}
170     elsif ($entries[2] == 72) {
171     $symb[$n_atoms] = "Hf";}
172     elsif ($entries[2] == 73) {
173     $symb[$n_atoms] = "Ta";}
174     elsif ($entries[2] == 74) {
175     $symb[$n_atoms] = "W";}
176     elsif ($entries[2] == 75) {
177     $symb[$n_atoms] = "Re";}
178     elsif ($entries[2] == 76) {
179     $symb[$n_atoms] = "Os";}
180     elsif ($entries[2] == 77) {
181     $symb[$n_atoms] = "Ir";}
182     elsif ($entries[2] == 78) {
183     $symb[$n_atoms] = "Pt";}
184
185     elsif ($entries[2] == 79) {
186     $symb[$n_atoms] = "Au";}
187     elsif ($entries[2] == 80) {
188     $symb[$n_atoms] = "Hg";}
189     elsif ($entries[2] == 81) {
190     $symb[$n_atoms] = "Tl";}
191     elsif ($entries[2] == 82) {

```

```

191     $symb[$n_atoms] = "Pb";}
192     elsif ($entries[2] == 83) {
193     $symb[$n_atoms] = "Bi";}
194     elsif ($entries[2] == 84) {
195     $symb[$n_atoms] = "Po";}
196     elsif ($entries[2] == 85) {
197     $symb[$n_atoms] = "At";}
198     elsif ($entries[2] == 86) {
199     $symb[$n_atoms] = "Rn";}
200     elsif ($entries[2] == 87) {
201     $symb[$n_atoms] = "Fr";}
202     elsif ($entries[2] == 88) {
203     $symb[$n_atoms] = "Ra";}
204     elsif ($entries[2] == 89) {
205     $symb[$n_atoms] = "Ac";}
206     elsif ($entries[2] == 90) {
207     $symb[$n_atoms] = "Th";}
208     elsif ($entries[2] == 91) {
209     $symb[$n_atoms] = "Pa";}
210     elsif ($entries[2] == 92) {
211     $symb[$n_atoms] = "U";}
212     elsif ($entries[2] == 93) {
213     $symb[$n_atoms] = "Np";}
214     elsif ($entries[2] == 94) {
215     $symb[$n_atoms] = "Pu";}
216     elsif ($entries[2] == 95) {
217     $symb[$n_atoms] = "Am";}
218     elsif ($entries[2] == 96) {
219     $symb[$n_atoms] = "Cm";}
220     elsif ($entries[2] == 97) {
221     $symb[$n_atoms] = "Bk";}
222     elsif ($entries[2] == 98) {
223     $symb[$n_atoms] = "Cf";}
224     elsif ($entries[2] == 99) {
225     $symb[$n_atoms] = "Es";}
226     elsif ($entries[2] == 100) {
227     $symb[$n_atoms] = "Fm";}
228     elsif ($entries[2] == 101) {
229     $symb[$n_atoms] = "Md";}
230     elsif ($entries[2] == 102) {
231     $symb[$n_atoms] = "No";}
232     elsif ($entries[2] == 103) {
233     $symb[$n_atoms] = "Lr";}
234     elsif ($entries[2] == 104) {
235     $symb[$n_atoms] = "Rf";}
236     elsif ($entries[2] == 105) {
237     $symb[$n_atoms] = "Db";}
238     elsif ($entries[2] == 106) {
239     $symb[$n_atoms] = "Sg";}
240     elsif ($entries[2] == 107) {
241     $symb[$n_atoms] = "Bh";}
242     elsif ($entries[2] == 108) {
243     $symb[$n_atoms] = "Hs";}
244     elsif ($entries[2] == 109) {
245     $symb[$n_atoms] = "Mt";}
246     else {
247         $symb[$n_atoms] = "Bq";
248     }
249     }
250     $x[$n_atoms] = $entries[4];
251     $y[$n_atoms] = $entries[5];
252     $z[$n_atoms] = $entries[6];
253     $line = <STDIN>;
254

```

```

255     } }
256     while(!($line =~ /--/));
257     }
258 }
259
260
261 printf STDOUT "%d\n", $n_atoms;
262 print ("0 1\n");
263 for($i = 1; $i <= $n_atoms; $i++) {
264     printf STDOUT "%s %16.7f %16.7f %16.7f\n",
265         $symb[$i], $x[$i], $y[$i], $z[$i];
266     }
267 print "\n";

```

APPENDIX 6

FULL CODE OF WORKUP.PL

```
1 #!/usr/bin/perl
2 use warnings;
3 use LWP::Simple;
4 use File::Copy;
5 use Excel::Writer::XLSX;
6 use Excel::Writer::XLSX::Utility;
7
8 my $formula = <$ARGV[0]>;
9 print "$formula\n";
10
11
12 chdir '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/' or die "$1";
13 my $workbook = Excel::Writer::XLSX->new( '$formula.xlsx' );
14 $worksheet = $workbook->add_worksheet('Energies');
15 $format = $workbook->add_format();
16 $format->set_color( 'red' );
17 $worksheet->write( 'A1', 'ID Number' );
18 $worksheet->write( 'B1', 'Website' );
19 $worksheet->write( 'C1', 'Compound Name' );
20 $worksheet->write( 'D1', 'Energy (Hartrees)' );
21 $worksheet->write( 'E1', 'M+ Energy (Hartrees)' );
22 $worksheet->write( 'F1', 'IE (Hartrees)' );
23 $worksheet->write( 'G1', 'IE (kcal/mol)', $format );
24
25 #opens a new Excel workbook that is named for the chemical formula, then opens a worksheet
26 #called Energies where we write into cell A1 the title "ID Number" the rest of the commands continue to
27 #label the columns
28 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/ID.txt',
29      '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/ID.txt') or die "Copy failed: $!";
30 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/site.txt',
31      '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/site.txt') or die "Copy failed: $!";
32 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/name.txt',
33      '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/name.txt') or die "Copy failed: $!";
34 copy('/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/output.txt',
35      '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/output.txt') or die "Copy failed: $!";
36 #copies files made previously in example.pl that can be used to extract important information into the
37 #excel file. These commands give us access to the ID number, the NIST webbook hyperlink for each
38 #compound, and the name of the compound
39
40 my $file = "ID.txt";
41 open (FH, "< $file") or die "Can't open $file for read: $!";
42 my @lines;
43 while (<FH>) {
44     push (@lines, $_);
45 }
46 close FH or die "Cannot close $file: $!";
47 print @lines;
48 $lines_ref = \@lines;
49 $worksheet->write_col( 1, 0, $lines_ref );
50 #A file with a list of compound ID numbers is opened, each line is read into an array (@lines). Then
51 #each element in the array is written into the excel worksheet starting in row 1 (actually the second row)
52 #and column 0 (the first column)
53
54 my $file1 = "site.txt";
55 open (FH, "< $file1") or die "Can't open $file1 for read: $!";
56 my @lines1;
57 while (<FH>) {
58     push (@lines1, $_);
59 }
60 close FH or die "Cannot close $file1: $!";
61 print @lines1;
62 $lines1_ref = \@lines1;
63 $worksheet->write_col( 1, 1, $lines1_ref );
64 #A file with a list of compound NIST webbook links is opened, each line is read into an array (@lines1).
```

```

56... Then each element in the array is written into the excel worksheet starting in row 1(actualy the second
... row) and column 1 (the second column)
57
58 my $file2 = "name.txt";
59 open (FH, "< $file2") or die "Can't open $file2 for read: $!";
60 my @lines2;
61 while (<FH>) {
62     push (@lines2, $_);
63 }
64 close FH or die "Cannot close $file2: $!";
65 print @lines2;
66 $lines2_ref = \@lines2;
67 $worksheet->write_col( 1, 2, $lines2_ref );
68 #A file with a list of compound names is opened, each line is read into an array (@lines2). Then each
... element in the array is written into the excel worksheet starting in row 1(actualy the second row) and
... column 2 (the third column)
69
70 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/find_energy.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/find_energy.plx') or die "Copy failed: $!";
71 #this copies the program that extracts energies from Gaussian output files into the "Workup" directory
72
73 open my $fh, '<', 'output.txt';
74
75 my $ID;
76 chomp(my @lines = <$fh>);
77 for my $lines ( @lines ) {
78     if ( $lines =~ /ID=(.+)&Units/ ) {
79
80         $ID = $1;
81 my $cmd1 = 'perl find_energy.plx < ' . $ID . '_11.log';
82 system($cmd1);
83 #this is the command that runs the program that extracts energies. In the program energies are extracted
... and written to the file energy.txt
84
85 my $file3 = "energy.txt";
86 open (FH, "< $file3") or die "Can't open $file3 for read: $!";
87 my @lines3;
88 while (<FH>) {
89     push (@lines3, $_);
90 }
91 close FH or die "Cannot close $file3: $!";
92 print @lines3;
93 $lines3_ref = \@lines3;
94 $worksheet->write_col( 1, 3, $lines3_ref );
95 }
96 }
97 close $fh;
98 #A file with a list neutral molecules energies is opened, each line is read into an array (@lines3).
... Then each element in the array is written into the excel worksheet starting in row 1(actualy the second
... row) and column 3 (the fourth column)
99
100 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/find_energy4.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/' . $formula . '/Workup/find_energy4.plx') or die "Copy failed: $!";
101 #this copies the program that extracts energies from Gaussian output files into the "Workup" directory
102
103 open my $fh, '<', 'output.txt';
104
105 my $ID;
106 chomp(my @lines = <$fh>);
107 for my $lines ( @lines ) {
108     if ( $lines =~ /ID=(.+)&Units/ ) {
109
110         $ID = $1;
111 my $cmd1 = 'perl find_energy4.plx < ' . $ID . '_36.log';
112 system($cmd1);
113 #this is the command that runs the program that extracts energies. In the program energies are extracted
... and written to the file energy.txt
114
115 }
116 }
117 close $fh;
118

```



```

119 my $file4 = "energy4.txt";
120 open (FH, "< $file4") or die "Can't open $file4 for read: $!";
121 my @lines4;
122 while (<FH>) {
123     push (@lines4, $_);
124 }
125 close FH or die "Cannot close $file4: $!";
126 print @lines4;
127 $lines4_ref = \@lines4;
128 $worksheet->write_col( 1, 4, $lines4_ref );
129 #A file with a list molecular ions (M+(dot))energies is opened, each line is read into an array
... (@lines4). Then each element in the array is written into the excel worksheet starting in row 1
... (actually the second row) and column 4 (the fifth column)
130
131
132 open my $ID_out, '<', 'ID.txt' or die($!);
133 my $end = 1;
134 $end++ while <$ID_out>;
135 close $ID_out;
136 print "Number of rounds:\n";
137 print $end, "\n";
138 #This is a loop that determines the number of compounds that we have run calculations on. This is needed
... so that we know into how many rows we need to place the calculation to calculate ionization energy
139
140 $row=2;
141 $realend=($end)+1;
142 do{
143     $worksheet->write( 'F'.$row.', 'E'.$row.'-D'.$row.' ' );
144     $worksheet->write( 'G'.$row.', 'F'.$row.*627.503', $format );
145     $row++;
146 }while($row < $realend);
147 #This is a loop that calculates the ionization energy for each compound. First a formula is written into
... a cell that calculates =E(cation)-E(neutral). This formula is placed into every row where there are
... compound energies. The second formula written into excel converts the ionization energy from Hartrees to
... kcal/mol
148
149 open my $fh, '<', 'output.txt';
150
151 my $ID;
152
153 chomp(my @lines = <$fh>);
154 for my $lines ( @lines ) {
155     if ( $lines =~ /ID=(.+)&Units/ ) {
156
157         $ID = $1;
158         copy
... ('/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'_'.$formula.'_B3LYP-6-31Gdp/'.$ID.'_convert.txt',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/Workup/'.$ID.'_convert.txt') or die "Copy failed
... $1";
159
160 open my $fh11, '<', '/Volumes/MAIN/Dropbox/DMC/Research/RJR/'.$formula.'/Workup/'.$ID.'_convert.txt';
161 if(-s $fh11)
162 {
163     open my $convertout2, '<', '._ID._convert.txt' or die($!);
164     my $lc = 0;
165     $lc++ while <$convertout2>;
166     close $convertout2;
167     print "Total number of lines:";
168     print $lc, "\n";
169 #For the proton affinities we can have a number of protonated species per neutral compound. Essentially
... every oxygen can be protonated thus creating a protonated structure for each oxygen in the compound.
... This piece of code reads a file that was generated in example.pl and counts the number of lines. Thus
... telling us the number of protonated species we have of each compound. A pretty important piece of code
... here is "if(-s $fh11)" this makes sure that the file has something in it. If we were to look at a file
... that corresponded to a compound that didn't have a 3D structure available on NIST then the program would
... think they that were no protonated species. So by making sure that the file has information in it then
... we know we are correctly counting the number of protonated species.
170
171
172 my $line_number2 = 0;
173 $increase = 0;
174 #$row = 1;
175 $add=1;

```



```

176 do{
177 open my $fh, '<', 'output.txt';
178
179 my $ID;
180
181 chomp(my @lines = <$fh>);
182 for my $lines ( @lines ) {
183     if ( $lines =~ /ID=(.+)&Units/ ) {
184
185         $ID = $1;
186
187         open my $FH, '<', '$ID.'_24_'$line_number2.'.log';
188
189
190
191         while($line = <$FH>) {
192             if($line =~ /SCF Done/) {
193                 @entries = split(/\s+/, $line);
194                 $energy3 = $entries[5];
195             }
196             else {
197                 next;
198             }
199         }
200         #printf STDOUT "%5.10f\n ", $energy3;
201         open my $energies3, '>>', 'energy3.txt';
202         print $energies3 "$energy3\n";
203
204     close $FH;
205     #The above section strips out the energies from the Gaussian output files for the protonated species.
206     ... This is the code that is also used in the program "find_energy.plx". It is written out here in order to
207     ... be able to appropriately loop through all of the protonated structures for each compounds. Again these
208     ... energies are written into a text file named "energy3".
209 }
210 }
211 close $fh;
212
213 my $file5 = "energy3.txt";
214 open (FH, "< $file5") or die "Can't open $file5 for read: $!";
215 my @lines5;
216 while (<FH>) {
217     push (@lines5, $_);
218 }
219 close FH or die "Cannot close $file5: $!";
220 print "These are the lines:\n";
221 print @lines5;
222 $lines5_ref = \@lines5;
223 $column=($line_number2)+7;
224 print "column#\n";
225 print "$column#\n";
226 $worksheet->write_col( 1, $column, $lines5_ref );
227 $worksheet->write( 0, $column, 'H+'.$line_number2.' Energies (Hartrees)' );
228 $newcol=($column)+($lc)+($increase));
229 $worksheet->write( 0, $newcol, 'PA+'.$line_number2.' Energies (Hartrees)' );
230 #this reads the energies of the protonated species into an array. The columns are labeled with
231 ... appropriate headers. The column numbers where things are placed are dependent on how many protonated
232 ... structures there are for each compound.
233
234 if ($line_number2 <= 1){
235     $newcol1=($newcol)+($increase)+($add);
236 } else{
237     $newcol1=($newcol)+1+($add);
238 }
239 $worksheet->write( 0, $newcol1, 'PA+'.$line_number2.' Energies (kcal/mol)', $format );
240 #again this is just a loop that controls where to write things into the excel file. This helps to make
241 ... this work for any number of protonated structures.
242
243 #put the array into the excel workbook
244

```

```

241 open my $ID_out, '<', 'ID.txt' or die($!);
242 my $end = 1;
243 $end++ while <$ID_out>;
244 close $ID_out;
245 print "Number of rounds:\n";
246 print $end, "\n";
247
248 $row=1;
249 do{
250 $row2=($row);
251 $str=xl_rowcol_to_cell( $row2, 3 );
252 print "$str\n";
253 $str2=xl_rowcol_to_cell( $row2, $column );
254 print "$str2\n";
255 $newcol2=($newcol1)-1;
256 $str3=xl_rowcol_to_cell( $row2, $newcol2);
257 print "$str2\n";
258 $worksheet->write( $row, $newcol, '='.$str.' - '.$str2.' ' );
259 $worksheet->write( $row, $newcol1, '='.$str3.* 627.503',$format );
260 $row++;
261 }while($row < $end);
262 #this section of code does the calculations of proton affinities. The "xl_rowcol_to_cell" function is
... important in that it allows us convert from row,column number format to the name of cell format. For
... example in row column format of (0,0) would be cell A1. This function allows us to save the cell value
... as a string to identify where to get values from for the calculations. The first formula written into
... excel is PA=E(neutral)-E(protonated). The other formula is again the conversion from Hartrees to
... kcal/mol.
263
264 print "Is this working??\n";
265 print "$line_number2\n";
266
267 unlink $file5;
268 #this is very important. It essentially deletes the files containing the Gaussian energy for the
... protonated structures. So that each time we go through the loop to extract energies from the various
... protonated species we get the correct energies written into excel.
269
270
271 $line_number2=($line_number2)+1;
272 print "$lc\n";
273 print "$line_number2\n";
274 $increase++;
275 print "The increase is:\n";
276 print "$increase\n";
277 $add = 0;
278 }while($line_number2 < $lc); }
279 }
280 }
281 close $fh;
282
283
284 #####
285 $worksheet2 = $workbook->add_worksheet('Ethalpies of Formation');
286 $worksheet2->write( 'A1', '# Carbon Atoms' );
287 $worksheet2->write( 'B1', '# Hydrogen Atoms' );
288 $worksheet2->write( 'C1', '# Oxygen Atoms' );
289 $worksheet2->write( 'D1', 'DH for Carbon' );
290 $worksheet2->write( 'E1', 'DH for Hydrogen' );
291 $worksheet2->write( 'F1', 'DH for Oxygen' );
292 $worksheet2->write( 'H1', 'DH(X,0K)',$format );
293 $worksheet2->write( 'I1', 'E for Carbon' );
294 $worksheet2->write( 'J1', 'E for Hydrogen' );
295 $worksheet2->write( 'K1', 'E for Oxygen' );
296 $worksheet2->write( 'M1', 'Sum of E for all atoms' );
297 $worksheet2->write( 'N1', 'Enthapy correction for Carbon' );
298 $worksheet2->write( 'O1', 'Enthapy correction for Hydrogen' );
299 $worksheet2->write( 'P1', 'Enthapy correction for Oxygen' );
300 $worksheet2->write( 'R1', 'Sum of Enthalpy corrections for atoms' );
301 $worksheet2->write( 'S1', 'Energy Converstion Factor (Hartrees to kcal/mol)' );
302 $worksheet2->write( 'A4', 'ID Number' );
303 $worksheet2->write( 'B4', 'Sum of Electronic and ZPE (Hartrees)' );
304 $worksheet2->write( 'C4', 'ZPE (Hartrees)' );
305 $worksheet2->write( 'D4', 'Thermal correction to Enthalpy (Hartrees)' );

```

```

306 $worksheet2->write( 'E4', 'Electronic Energy (Hartrees)' );
307 $worksheet2->write( 'F4', '298K - 0K Enthalpy correction (Hartrees)' );
308 $worksheet2->write( 'G4', 'Enthalpy of formation at 0K (kcal/mol)' );
309 $worksheet2->write( 'H4', 'Enthalpy of formation at 298K (kcal/mol)' );
310 #the above section of code makes a new worksheet in the excel file that will be used to calculated heats
    of formation. The rest of the code is writing in labels for the data that will be placed into excel.
311
312 $formula = '$formula.';
313 ($C)= $formula =~ /C(.*)H/;
314 print "$C\n";
315 ($H)= $formula =~ /H(.*)O/;
316 print "$H\n";
317 ($O)= $formula =~ /O(.*)/;
318 print "$O\n";
319
320 $worksheet2->write( 'A2', '$C.' );
321 $worksheet2->write( 'B2', '$H.' );
322 $worksheet2->write( 'C2', '$O.' );
323
324 $worksheet2->write( 'D2', '169.98' );
325 $worksheet2->write( 'E2', '51.63' );
326 $worksheet2->write( 'F2', '58.99' );
327 $worksheet2->write( 'H2', '=((A2*D2)+(B2*E2)+(C2*F2))',$format );
328
329 $worksheet2->write( 'I2', '-37.776009' );
330 $worksheet2->write( 'J2', '-0.500273' );
331 $worksheet2->write( 'K2', '-74.957398' );
332 $worksheet2->write( 'M2', '=((A2*I2)+(B2*J2)+(C2*K2))',$format );
333
334 $worksheet2->write( 'N2', '0.25' );
335 $worksheet2->write( 'O2', '1.01' );
336 $worksheet2->write( 'P2', '1.04' );
337 $worksheet2->write( 'R2', '=((A2*N2)+(B2*O2)+(C2*P2))',$format );
338
339 $worksheet2->write( 'S2', '627.503' );
340 #the first chunk of code(320-326) takes the formula that the user input at the beginning and determines
    the number of each atom in the formula. These values are then set in variables ($C,$H, and $O). The
341
342 next chunk(328-330) writes the values of those variables into excel. Lines 332-335 writes in the heat of
    formation of the atoms and calculates the sum over all atoms. Lines 337-340 writes in the total energy
    of the atoms and again calculates the sum over all atoms. Lines 342-345 writes in the enthalpy
    corrections for the atoms and the sum of these values over all atoms. Finally line 347 is the the
    conversion factor to go from Hartrees to kcal/mol.
341
342 my $file = "ID.txt";
343 open (FH, "< $file") or die "Can't open $file for read: $!";
344 my @lines;
345 while (<FH>) {
346     push (@lines, $_);
347 }
348 close FH or die "Cannot close $file: $!";
349 print @lines;
350 $lines_ref = \@lines;
351 $worksheet2->write_col( 4, 0, $lines_ref );
352 #A file with a list of compound ID numbers is opened, each line is read into an array (@lines). Then
    each element in the array is written into the excel worksheet starting in row 4(actually the firth row)
    and column 0 (the first column)
353
354 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/find_energy5.plx',
    '/Volumes/MAIN/Dropbox/DMC/Research/RJR/.$formula./Workup/find_energy5.plx') or die "Copy failed: $!";
355 #copies the program to extract energies into the Workup foldeer
356
357 open my $fh, '<', 'output.txt';
358
359 my $ID;
360 chomp(my @lines = <$fh>);
361 for my $lines ( @lines ) {
362     if ( $lines =~ /ID=(.*)&Units/ ) {
363
364         $ID = $1;
365     }
366 }
367 my $cmd1 = 'perl find_energy5.plx < '.$ID.'_11.log';

```

```

366 system($cmd1);
367
368
369
370 my $file5 = "energy5.txt";
371 open (FH, "< $file5") or die "Can't open $file5 for read: $!";
372 my @lines5;
373 while (<FH>) {
374     push (@lines5, $_);
375 }
376 close FH or die "Cannot close $file5: $!";
377 print @lines5;
378 #read lines from file into an array, so in this case reads ID numbers into an array
379 $lines5_ref = \@lines5;
380 $worksheet2->write_col( 4, 1, $lines5_ref );
381 #put the array into the excel workbook
382 }
383 }
384 close $fh;
385 #the above section of code extracts out the total energies from Gaussian input files and writes them
... into the appropriate column in excel.
386
387 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/find_zpe2.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/find_zpe2.plx') or die "Copy failed: $!";
388
389 open my $fh, '<', 'output.txt';
390
391 my $ID;
392 chomp(my @lines = <$fh>);
393 for my $lines ( @lines ) {
394     if ( $lines =~ /ID=(.*)&Units/ ) {
395
396         $ID = $1;
397
398 my $cmd1 = 'perl find_zpe2.plx < '.$ID.'_11.log';
399 system($cmd1);
400
401
402
403 my $file9 = "energy9.txt";
404 open (FH, "< $file9") or die "Can't open $file9 for read: $!";
405 my @lines9;
406 while (<FH>) {
407     push (@lines9, $_);
408 }
409 close FH or die "Cannot close $file9: $!";
410 print @lines9;
411 $lines9_ref = \@lines9;
412 $worksheet2->write_col( 4, 2, $lines9_ref );
413 }
414 }
415 close $fh;
416 #the above section of code extracts out the zero-point energies from Gaussian input files and writes them
... into the appropriate column in excel.
417
418 copy('/Volumes/MAIN/Dropbox/DMC/Research/Perl/find_thermal.plx',
... '/Volumes/MAIN/Dropbox/DMC/Research/RJR/',$formula.'/Workup/find_thermal.plx') or die "Copy failed: $!";
419
420 open my $fh, '<', 'output.txt';
421
422 my $ID;
423 chomp(my @lines = <$fh>);
424 for my $lines ( @lines ) {
425     if ( $lines =~ /ID=(.*)&Units/ ) {
426
427         $ID = $1;
428 my $cmd1 = 'perl find_thermal.plx < '.$ID.'_11.log';
429 system($cmd1);
430
431
432
433 my $file7 = "energy7.txt";
434 open (FH, "< $file7") or die "Can't open $file7 for read: $!";
435 my @lines7;

```

```

436 while (<FH>) {
437     push (@lines7, $_);
438 }
439 close FH or die "Cannot close $file7: $!";
440 print @lines7;
441 $lines7_ref = \@lines7;
442 $worksheet2->write_col( 4, 3, $lines7_ref );
443 }
444 }
445 close $fh;
446 #the above section of code extracts out the enthalpy corrections from Gaussian input files and writes
... them into the appropriate column in excel.
447 open my $ID_out, '<', 'ID.txt' or die($!);
448 my $end = 4;
449 $end++ while <$ID_out>;
450 close $ID_out;
451 print "Number of rounds:\n";
452 print $end, "\n";
453
454 $row=5;
455 $realend=($end)+1;
456
457 do{
458     $worksheet2->write( 'E'.$row.', 'B'.$row.'-C'.$row.' ' );
459     $worksheet2->write( 'F'.$row.', 'D'.$row.'-C'.$row.' ' );
460     $worksheet2->write( 'G'.$row.', 'H2-(S2*(M2-E'.$row.'-C'.$row.'))' );
461     $worksheet2->write( 'H'.$row.', 'G'.$row.'+(S2*(F'.$row.'))-R2', $format );
462     $row++;
463 }while($row < $realend);
464 #Loop that writes in formulas to calculate the total energy of molecule, the enthalpy corrections, heat
... of formation at 0K, and finally the heat of formation at 298K.
465

```

APPENDIX 7

FULL CODE OF FIND_ENERGY.PLX

```
1  #!/usr/local/bin/perl -w
2
3  # Modified from program written by Maciej Haranczyk, 2005
4
5
6  while($line = <STDIN> ) {
7      if($line =~ /SCF Done/) {
8          @entries = split(/\s+/, $line);
9          $energy = $entries[5];
10     }
11     else {
12         next;
13     }
14 }
15 printf STDOUT "%5.10f\n ", $energy;
16 open my $energies, '>>', 'energy.txt';
17 print $energies "$energy\n";
```

APPENDIX 8

FULL CODE OF FINE_ZPE2.PLX

```
1  #!/usr/local/bin/perl -w
2
3  # Modified from program written by Maciej Haranczyk, 2005
4
5
6
7  while($line = <STDIN> ) {
8      if($line =~ /Zero-point correction/) {
9          #print "testing\n";
10         #print "$line\n";
11         @entries = $line =~ /Zero-point correction=
12         #print "Entries\n";
13         #print "@entries\n";
14
15         #@entries = split(/\s+/, $line);
16         $energy9 = $entries[0];
17         # print "Energy9\n";
18         # print "$energy9\n";
19         }
20         else {
21             next;
22         }
23     }
24     #printf STDOUT "%5.10f\n ", $energy9;
25     open my $energies9, '>>', 'energy9.txt';
26     print $energies9 "$energy9\n";
```

APPENDIX 9

FULL CODE OF FIND_THERMAL.PLX

```
1 #!/usr/local/bin/perl -w
2
3 # Modified from program written by Maciej Haranczyk, 2005
4
5
6
7 while($line = <STDIN> ) {
8     if($line =~ /Thermal correction to Enthalpy/) {
9         @entries = split(/\s+/, $line);
10        $energy7 = $entries[5];
11    }
12    else {
13        next;
14    }
15 }
16 printf STDOUT "%5.10f\n ", $energy7;
17 open my $energies7, '>>', 'energy7.txt';
18 print $energies7 "$energy7\n";
```


APPENDIX 10

THEORETICAL RESULTS FROM CALCULATION PROGRAM

All energies have units of kcal/mol

IE: Ionization Energy

PA: Proton Affinity (Proton affinities are calculated for each neutral structure protonated at each oxygen)

ΔH_f : Enthalpy of Formation

Table A10.1. Theoretical Energy values for C₁₁H₁₂O₅

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	PA 5	ΔH_f
R273922	161.4	202.0	192.5	234.0	221.5	225.2	-1094.1
C35400192	178.1	220.3	200.4	198.1	218.6	212.6	-1126.0
U352933	181.0	208.4	220.9	215.9	194.2	213.8	-1125.1
U307542	170.3	206.8	222.0	199.0	209.0	217.3	-1122.1
C530596	163.6	206.7	206.4	206.5	212.1	226.3	-1103.5
C22955733	184.8	210.0	221.2	201.1	221.5	210.0	-1125.4
R306908	168.3	233.0	214.4	232.3	231.4	236.6	-1099.9
R306933	183.8	219.2	211.2	214.5	194.0	211.2	-1125.0
R306864	173.7	249.3	235.9	231.3	222.0	199.9	-1115.2
C39622532	201.4	223.3	215.5	202.9	213.0	222.6	-1075.2

Table A10.2. Theoretical Energy values for C₁₂H₁₄O₅

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	PA 5	ΔH_f
C20733942	164.3	240.0	220.0	206.0	206.0	205.5	-1154.5
R273977	174.8	229.5	218.0	219.2	197.3	226.3	-1183.3
C30218850	184.8	225.7	218.1	200.2	215.3	225.3	-1143.4
C33130039	165.0	212.7	211.5	212.7	214.9	226.1	-1156.1

Table A10.3. Theoretical Energy values for C₁₀H₁₀O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C131113	196.3	225.7	222.1	222.1	225.7	-955.2
C120616	203.7	192.4	213.7	213.7	213.7	-962.6
C1459934	204.6	193.4	212.8	212.8	216.2	-962.4
C326614	169.2	190.8	190.7	190.7	216.9	-939.7
C108587	184.6	192.5	213.0	213.0	209.7	-958.8
C881685	182.5	194.2	209.4	209.4	220.9	-939.0
C537984	167.8	198.1	198.0	198.0	212.0	-941.5

C569313	172.9	209.6	206.1	206.1	210.3	-931.2
C1135246	168.5	209.4	221.7	221.7	204.1	-936.3
C635676	186.7	194.3	210.1	210.1	210.1	-957.2
C1205910	177.7	192.2	210.9	210.9	213.9	-959.8
U352913	182.7	191.9	229.6	229.6	208.3	-941.2
U352848	184.2	216.4	195.0	195.0	216.4	-940.3
U352898	176.9	188.3	221.3	221.3	207.0	-939.4
U352920	182.6	185.6	229.1	229.1	204.3	-941.6
C2163124	182.0	178.8	222.3	222.3	216.7	-953.5
C2161855	186.2	180.2	180.3	180.3	229.6	-945.5
C2306334	194.8	219.7	243.2	243.2	228.2	-958.7
C24781235	190.5	189.5	210.6	210.6	194.6	-961.5
C24262666	190.9	188.6	207.1	207.1	218.8	-962.2
C537735	166.0	204.7	204.7	204.7	222.9	-936.0
T999919129	198.0	219.2	201.4	201.4	201.4	-969.8
C2815954	168.6	191.6	191.3	191.3	191.1	-944.9
T999917021	183.6	217.9	212.1	212.1	218.4	-955.0
C7325464	192.3	214.0	196.4	196.4	188.7	-967.0
C16475904	190.1	204.8	216.5	216.5	225.3	-953.1
T999926782	195.1	211.0	221.5	221.5	199.7	-971.8
C616751	187.8	215.4	209.6	209.6	209.6	-967.6
C3626946	172.0	199.5	199.5	199.5	226.2	-941.5
C6328445	193.9	201.1	210.6	210.6	214.9	-966.2

Table A10.4. Theoretical Energy values for C₁₀H₁₂O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C617050	172.1	203.1	221.4	205.8	205.8	-973.5
C2150381	168.6	208.9	207.8	203.7	223.5	-962.8
C90244	172.1	251.2	189.6	197.0	251.2	-951.1
C86817	170.2	212.4	212.3	210.0	226.6	-937.4
C2150370	169.8	199.7	221.5	199.3	200.7	-965.7
C2478388	169.4	205.7	206.6	206.6	230.5	-957.2
C4707475	174.2	218.4	233.5	224.7	206.3	-973.7
C2150416	173.7	217.7	228.7	217.7	196.7	-963.8
C13305141	171.6	219.1	219.0	216.3	201.5	-965.0
C3376598	178.3	214.8	221.6	212.5	211.6	-972.9
C4460860	121.3	299.1	222.9	222.7	299.1	-862.7
C944990	184.7	207.9	221.0	228.2	228.1	-958.2
C999213	185.6	204.2	214.3	194.3	225.6	-934.5
C2065272	175.1	218.3	225.6	240.0	218.4	-958.4
C2150427	162.5	209.6	230.1	226.7	218.9	-950.1
C3385340	187.5	194.3	213.9	198.6	226.7	-984.2
R69952	169.3	218.9	221.6	206.2	206.2	-960.5
C2103573	170.1	207.0	206.3	206.3	230.5	-938.0

C26699730	176.3	206.2	218.0	220.4	218.6	-971.5
C1421632	166.3	193.0	193.0	226.5	226.5	-983.7
C14309547	192.1	227.4	213.7	213.7	227.4	-946.4
C1135235	166.5	208.5	208.4	203.3	192.2	-974.2
C2807547	200.5	192.6	213.5	192.5	210.2	-939.1
C80018504	164.9	215.0	215.0	194.5	216.1	-976.6
C1758254	161.6	202.6	214.2	214.2	208.8	-969.1
C93403	163.3	212.9	213.8	216.7	195.1	-965.5
C4670104	170.4	204.2	200.1	227.7	232.0	-970.6
C830795	173.5	195.6	194.6	248.8	242.5	-942.4
C15964804	165.3	200.1	212.3	212.3	212.3	-965.7
C90539	164.6	213.8	213.8	216.1	228.6	-966.9

Table A10.5. Theoretical Energy values for C₁₁H₁₂O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C87345537	163.8	203.5	203.4	203.4	237.5	-978.0
C2316269	166.0	211.5	223.5	206.8	206.7	-996.3
C2309071	164.1	210.3	224.1	204.4	204.4	-997.5
C16909118	172.6	198.6	197.8	206.0	218.6	-1000.2
C16909094	164.9	197.1	196.1	228.7	217.0	-1000.2
C7345826	167.6	208.4	208.4	229.4	212.1	-993.0
U332839	171.2	212.0	218.2	211.5	199.5	-1000.8
C14737894	157.3	234.8	220.4	214.1	211.5	-987.9
U352929	179.3	190.2	231.4	196.1	208.0	-1011.3
U352908	173.8	224.9	190.3	199.1	208.6	-1009.9
U307539	184.0	204.6	211.1	209.9	215.9	-1026.1
U308342	190.3	222.5	202.8	212.5	216.0	-1025.4
C54771607	178.1	200.8	200.8	221.1	224.9	-1008.4
U352784	184.2	192.4	191.9	210.4	211.3	-1028.4
U352789	185.4	195.0	216.8	217.1	216.1	-1027.3
R411500	163.4	224.7	211.5	209.0	209.3	-992.3
C947579	177.2	236.6	225.9	225.6	213.6	-968.4
R88425	166.9	211.0	204.6	205.0	234.3	-972.2
C102374	168.3	191.5	181.6	231.6	210.8	-1010.9
C10538519	165.8	234.6	198.0	217.8	199.6	-992.4
C7461601	167.6	208.6	208.6	212.1	226.5	-993.4
C884333	184.2	215.9	195.9	204.4	188.8	-1035.7
C20116658	194.3	214.4	223.1	214.4	226.5	-1018.2
C23038611	197.3	196.3	212.5	195.6	216.8	-1030.8
C714534	197.7	230.1	213.1	236.5	217.4	-962.8

Table A10.6. Theoretical Energy values for C₁₁H₁₄O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C5826733	184.1	235.5	214.6	225.7	231.4	
C15964791	159.2	211.7	209.7	201.4	213.2	-1026.7
C2107702	162.1	211.5	211.2	202.7	191.6	-1033.8
C60563135	164.4	202.8	202.8	202.5	223.4	
C1136863	166.5	213.4	213.5	213.4	228.2	-1010.8
U332760	171.7	192.6	210.5	219.7	201.7	-1023.6
U333492	157.5	206.1	209.4	215.8	218.4	-1028.1
C3535248	167.5	203.2	211.2	204.1	224.4	-1033.3
U352535	189.9	215.6	190.4	192.6	231.4	-1033.1
C13909734	169.7	235.8	208.1	208.8	208.8	-1009.0
C6110378	168.6	226.2	226.2	234.9	197.5	-1023.4
R106753	170.3	210.3	192.8	200.6	213.2	-1033.7
C5650431	166.4	206.7	206.7	205.9	229.4	-1025.1
R88386	157.0	198.3	226.1	236.8	202.2	-1019.8
C10538495	159.6	228.5	204.1	195.4	205.8	-1037.9
C832586		253.1	253.6	199.1	253.1	-1010.3
C34874754		237.7	192.2	211.4	237.7	-1037.5
C6512329	165.6	204.5	203.1	229.6	211.4	-1031.2
C20587615	192.8	216.0	215.4	216.0	215.9	-1035.9
C3943779	166.9	207.7	206.6	223.9	205.6	-1035.1

Table A10.7. Theoretical Energy values for C₁₂H₁₂O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C29412622	151.4	201.8	221.0	293.3	308.5	-912.9
R284249	193.3	209.4	202.0	193.6	212.0	-1056.4
R504999	187.8	223.2	201.2	192.5	211.6	-1055.0
C4281407	164.5	206.4	232.9	205.1	203.7	-1039.0
C50551581	175.9	209.3	231.6	207.2	212.8	-1035.3
C50551569	176.3	197.7	212.2	208.8	225.7	-1036.6

Table A10.8. Theoretical Energy values for C₁₂H₁₄O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C484311	153.4	214.1	196.5	218.3	211.3	-1021.4
C523808	154.5	206.6	216.0	219.9	228.9	-1021.6
C636099	201.1	178.9	213.8	186.6	216.2	-1103.4
C5396645	163.8	211.4	211.4	207.7	207.7	-1054.5
C20803887	182.5	217.9	213.3	227.4	231.1	-1085.3
R290625	168.6	226.5	198.1	210.0	171.6	-1058.0
C4046020	164.5	200.0	200.0	211.2	229.5	-1072.5
U307541	180.6	221.0	211.8	216.9	216.9	-1095.4
C119450112	182.3	234.1	222.8	183.8	218.7	-1085.6

C131704	193.8	231.1	229.2	220.6	222.5	-1092.6
R275106	168.2	221.5	219.6	193.7	212.8	-1077.9
U355700	184.8	216.5	208.1	211.1	220.2	-1095.1
R298436	158.6	230.7	233.0	212.4	212.4	-1050.1
U355699	183.4	211.3	211.3	211.2	220.0	-1093.1
R298421	160.6	209.2	209.2	214.0	214.0	-1053.0
R84007	166.1	205.7	193.6	192.4	196.4	-1071.5
R399534	191.8	215.8	215.8	195.8	195.8	-1093.5
R334384	158.5	241.6	205.4	205.5	182.1	-1034.0
C71369192	190.6	229.4	247.2	247.2	235.8	-1088.6
R490850	180.4	192.4	192.7	210.8	209.5	-1096.3
C28153244	182.7	227.0	231.7	227.0	211.5	-1094.4
T999913563	179.2	214.5	211.3	213.7	211.3	-1107.0
T999917872	167.0	211.2	214.8	222.9	204.8	-1085.4
C101906	175.9	199.4	202.2	181.1	219.2	-1001.3
C36076252	185.2	199.6	211.2	220.2	211.2	-1090.9
U308750	170.0	191.5	226.7	218.7	216.0	-1072.7

Table A10.9. Theoretical Energy values for C₁₂H₁₆O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C27798738	159.5	210.9	211.2	194.6	214.4	-1094.7
R438243	152.8	208.0	209.6	226.0	219.4	-1098.9
U123085	159.3	194.1	211.9	212.4	239.2	-1077.1
C14174084	163.3	209.0	231.8	230.7	211.4	-1064.6
R235044	173.6	202.3	221.9	207.7	203.8	-1106.7
C61292908	164.9	196.9	212.4	209.6	209.6	-1104.2
C18066687	158.7	213.5	224.2	198.0	212.6	-1097.6

Table A10.10. Theoretical Energy values for C₁₃H₁₄O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C13620821	174.5	205.3	218.7	218.1	220.8	-1109.3
C33672945	172.6	227.0	239.6	228.5	239.3	-1097.9
C19598073	188.4	218.6	230.3	219.8	230.3	-1138.6
R109381	183.5	210.7	210.5	225.3	241.7	-1120.7
C19597967	191.7	214.5	217.0	214.4	219.0	-1137.0
C137833364	185.7	218.5	223.2	236.4	191.1	-1105.8

Table A10.11. Theoretical Energy values for C₁₃H₁₆O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
R290714	167.0	227.3	208.7	213.0	210.5	-1128.4
C83136	188.2	203.0	223.0	201.5	211.2	-1159.7
C34006763	192.7	218.2	235.1	219.6	229.8	-1152.6

C2689885	192.1	227.0	235.6	227.0	207.6	-1060.2
-----------------	-------	-------	-------	-------	-------	---------

Table A10.12. Theoretical Energy values for C₁₄H₁₈O₄

NIST ID Number	IE	PA 1	PA 2	PA 3	PA 4	ΔH_f
C3143064	199.9	217.1	196.7	217.1	196.8	-1237.7
C1962749	200.3	196.6	217.1	196.6	215.0	-1237.4
C607818	189.4	202.8	221.7	212.5	202.8	-1228.3
U352801	173.4	194.9	193.9	212.4	216.3	
U349653	182.3	221.6	232.7	211.4	216.0	-1231.5
C119450123	175.6	215.5	229.2	225.6	220.0	-1223.7
R329441	173.8	212.3	215.3	214.2	212.3	-1197.3
R329436	184.4	222.9	215.4	224.2	204.9	-1187.4

Table A10.13. Theoretical Energy values for C₁₀H₁₈O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
C531599	178.2	192.3	208.6	228.5	-807.1
R274272	175.2	201.8	226.1	193.8	-805.3
C90335	179.4	188.8	205.0	229.7	-818.1
C20280813	185.3	204.9	181.9	229.7	-811.1
C1131153	202.7	193.4	206.0	206.2	-822.5
U299234	189.7	210.7	210.7	227.8	-730.4
C2445810	177.8	200.5	207.9	226.4	-803.0

Table A10.14. Theoretical Energy values for C₁₂H₁₄O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
C97412232	165.1	206.6	204.1	227.4	-953.4
R432007	167.8	199.2	211.5	224.6	-961.4
R185523	192.4	224.9	219.8	224.7	-944.0
R185538	192.4	224.9	219.8	224.7	-944.0
U307439	167.8	200.9	220.6	200.6	-950.1
U307595	163.5	207.1	226.7	202.8	-960.9
U292250	179.9	200.9	220.4	198.4	-957.3
U299946	178.6	219.2	219.2	228.3	-949.8
R576553	169.6	232.7	212.1	195.9	-964.3
C6939759	187.8	207.8	221.8	217.0	-978.6
C38624474	170.7	226.1	234.9	203.2	-965.8
C91496581	178.5	231.6	220.2	204.1	-961.3
R323121	193.4	218.8	198.5	211.4	-981.9
C3572063	173.1	197.9	217.2	208.2	-975.2
C70637002	189.1	244.3	244.3	244.3	-955.5
C1501048	194.7	221.3	215.4	193.1	-977.6
C10595069	174.1	227.0	190.1	227.0	-958.1

C15001271	160.0	209.5	209.3	238.3	-940.4
C68555588	170.9	218.5	218.6	227.2	-964.5
C24370841	189.0	209.2	215.5	226.0	-978.1

Table A10.15. Theoretical Energy values for C₁₂H₁₆O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
C487116	158.2	219.5	209.4	219.5	-950.4
C87207	177.4	223.0	212.5	231.9	-998.5
C2883989	144.7	208.1	214.7	222.4	-958.2
C5273869	147.0	214.3	216.0	210.6	-954.5
C487127	152.6	220.1	220.1	220.0	-956.9
C2050080	177.7	212.2	231.6	222.7	-1000.0
C5353151	150.5	215.5	213.6	204.0	-952.8
C103606	177.8	198.3	191.4	212.3	-1001.9
C27739282	181.3	195.5	205.0	225.2	-1006.7
C6946356	181.1	195.7	204.7	225.1	
C4906256	179.3	224.1	203.8	201.0	-1006.5
C6963560	172.4	199.1	220.2	219.5	-1002.4
C20637085	170.6	193.1	210.2	201.9	-999.3
R409413	178.1	177.6	227.0	207.8	-987.8
U308042	168.0	220.1	200.8	200.5	-1000.2
C6555299	170.9	194.8	210.7	201.1	-998.9
U307986	168.2	199.1	216.7	200.5	-1000.5
C5273858	152.3	220.1	220.0	220.1	-956.8
C15872410	181.8	199.7	203.5	217.4	-1014.1
C33943269	169.4	225.0	204.4	216.8	-1000.4
C13183192	186.2	193.7	195.0	210.7	-1011.5
R235326	177.0	197.2	219.3	194.0	-1011.0
R235306	184.9	196.1	218.6	193.4	-1012.0
C159139421	180.5	201.4	197.5	221.5	-1005.5
C6521308	185.9	222.7	203.7	186.8	-1012.3
R540608	188.2	236.4	200.1	206.6	-1002.9
R421062	157.1	204.0	198.2	228.8	-970.9
R543038	173.5	192.5	220.9	220.9	-1001.8
R241400	150.0	218.9	218.9	218.9	-955.0
C13278278	187.7	214.2	193.8	221.8	-1011.7
C4547573	171.0	206.6	215.9	195.6	-1008.7
C1798045	172.6	209.5	185.0	210.5	-1001.8

Table A10.16. Theoretical Energy values for C₁₃H₁₆O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
C65405778	170.5	222.7	212.1	231.6	-1020.7
C644064	145.5	215.7	215.7	211.4	-1008.5

R576711	200.1	210.1	218.1	220.1	-1025.7
U307577	167.4	221.9	201.8	200.9	-1034.7
C204454035	176.1	227.3	206.7	198.6	-1027.8
U292572	174.9	227.7	221.9	221.9	-1023.1
U299192	171.3	196.5	209.1	227.0	-1029.9
R329481	163.7	229.0	208.8	206.6	-1008.8
U357380	180.9	198.9	220.5	200.8	-1026.2
R83934	164.6	193.4	193.2	213.2	-1024.6
R519263	149.4	220.3	220.2	220.2	-980.9
R519298	144.3	212.8	211.9	210.5	-980.1
C107274419	169.3	237.8	218.3	219.3	-1028.0
R292854	170.6	231.8	222.9	222.9	-1022.1
C620791	185.3	227.6	199.4	211.9	-1045.4
C43050288	171.1	218.5	201.6	200.4	-1038.9
C5524572	183.8	224.2	224.2	213.9	-1042.6
C7472437	193.4	223.6	202.8	190.9	-1052.3
C427496	191.1	205.4	209.2	219.1	-1041.4

Table A10.17. Theoretical Energy values for C₁₃H₁₈O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
R223259	193.4	224.5	220.8	202.2	-1005.0
C7070248	192.1	223.6	208.2	219.9	-1041.1
C27739293	180.9	205.2	195.9	225.4	-1072.9
R409908	170.9	211.4	229.8	224.9	-1060.8
R181818	175.1	199.2	223.8	226.6	-1059.1
R578826	178.5	225.6	205.4	201.5	-1076.9
C22708141	172.7	222.6	224.4	200.3	-1052.3
U299236	172.8	232.3	232.3	229.3	-1039.8
R540039	186.9	239.0	214.6	239.0	-1072.1
C959097871	178.5	219.3	228.7	219.3	-1064.9
R540871	174.3	212.3	191.4	225.5	-1069.2
C1142398	181.7	200.0	203.6	220.6	-1081.1
C116632930	161.3	211.1	211.0	225.9	-1048.0
T999927547	194.7	213.8	206.9	213.8	-4836.0
C5421045	188.8	217.1	208.1	208.1	-1070.2

Table A10.18. Theoretical Energy values for C₁₄H₁₈O₃

NIST ID Number	IE	PA 1	PA 2	PA 3	ΔH_f
U307705	167.4	201.9	219.1	200.7	-1107.3
U299075	169.5	205.2	222.7	214.4	-1085.3
R399529	189.2	198.5	218.8	222.9	-1102.8
C19833966	188.0	208.7	214.7	210.0	-1101.4
C7469832	171.1	200.8	209.9	202.1	

C62718632	169.2	201.1	212.2	224.1	-1106.9
C66147759	192.4	223.9	207.8	191.0	-1119.4

Table A10.19. Theoretical Energy values for C₉H₁₀O₂

NIST ID Number	IE	PA 1	PA 2	ΔH_f
C7786610	163.5	179.8	208.5	-665.2
C93890	201.5	218.0	218.0	-705.3
C140114	198.1	189.7	229.2	-694.1
C101417	192.4	194.8	217.6	-698.7
C100061	183.0	229.3	207.5	-688.2
C501520	193.1	205.4	205.4	-707.1
C140396	180.8	206.8	214.8	-703.1
C104621	199.3	188.4	209.3	-690.9
C122601	175.2	215.1	215.1	-654.1
C619045	196.7	205.6	223.4	-707.3
C586378	182.9	194.5	221.0	-686.7
C89714	196.2	178.7	218.6	-702.5
C499069	192.2	211.5	197.3	-714.9
C610720	190.6	215.7	199.6	-711.4
C632462	195.3	221.1	221.1	-703.0
C603792	186.5	220.3	220.3	-704.4
C579748	185.0	191.7	229.0	
C936516	192.4	220.4	219.1	-676.8
C1450722	179.8	237.5	156.6	-691.1
C122463	184.2	214.7	214.7	-702.7
C876028	184.2	181.5	228.9	-698.0
C619647	197.6	215.1	215.1	-712.0
C612191	190.3	219.3	205.2	-702.7
C619205	197.2	210.0	195.4	-711.6
C10031820	185.9	172.0	226.8	-686.6
C637274	187.1	194.0	211.9	-701.2
C875592	184.2	185.9	226.8	-696.1
C80463214	200.2	196.2	207.7	-685.8
C492375	196.5	213.2	197.1	-702.1
C613694	185.9	194.4	226.1	-684.4
C7216184	167.7	204.8	205.0	-675.3
C4647421	188.8	216.3	216.3	-687.4
C31776280	180.7	209.3	186.7	-649.3
R239180	157.5	191.5	195.8	-666.7
R519421	162.6	203.2	203.2	-671.0
C935922	203.4	216.6	183.1	-684.0
C621874	172.8	212.6	206.2	-680.6
U299375	196.4	205.7	218.5	-613.0
C4647432	184.5	212.5	212.5	-684.6

C610991	189.2	185.4	228.3	-693.5
C622479	186.8	192.6	213.5	-708.3
C70705736	184.9	203.4	219.4	-652.5
C5703264	174.5	199.6	208.4	-676.5
C60526423	181.4	208.1	213.1	-655.4
C1126610	167.5	197.0	196.7	-674.0
R256878	183.7	225.7	231.1	-687.5
C621363	187.2	196.2	192.2	-707.6
C70702	186.1	184.8	204.4	-694.5
C7782243	196.5	213.2	197.1	-702.1
C7782265	196.4	216.6	197.1	-702.1
C32723674	182.6	163.4	225.0	-685.8
C3604362	182.6	221.0	218.7	-645.1
C4079521	189.5	217.9	180.9	-678.4
C13426498	182.9	212.7	214.9	-650.0
C68347262	187.6	213.7	206.2	-665.6
C4024195	163.0	208.8	200.6	-660.7
C644360	189.6	189.2	212.3	-707.4
C1481932	172.4	201.1	222.6	-684.8
C22924158	184.5	196.6	218.1	-684.8

Table A10.20. Theoretical Energy values for C₉H₁₂O₂

NIST ID Number	IE	PA 1	PA 2	ΔH_f
C2785899	164.5	203.3	210.5	-709.7
C1125219	202.6	210.0	184.4	-717.4
C4179195	164.8	202.6	225.0	-707.2
C494995	160.5	183.8	212.9	-702.4
C13679566	170.1	172.9	213.2	-713.1
C5673074	170.3	204.2	192.7	-706.9
C24599584	158.5	205.2	209.4	-706.6
C1125888	190.8	218.3	199.7	-704.3
C4463336	162.6	212.6	212.6	-702.7
C702238	171.3	200.4	203.9	-706.8
C622082	189.0	189.4	189.6	-705.6
C2138489	169.8	200.0	200.0	-721.0
C770354	174.5	212.0	212.0	-712.7
C17600725	163.9	204.8	214.6	-704.1
C6180616	176.8	203.6	196.6	-706.9
C5020417	175.9	203.2	200.4	-707.1
C2903755	173.0	197.4	217.3	-644.3
C769857	162.9	163.6	246.1	-632.3
C700130	158.1	199.5	198.8	-722.7
R411882	190.8	201.5	221.7	-712.5
C1075043	184.6	213.5	213.5	-715.2
C5076722	160.1	207.5	205.1	-707.3

C4217667	187.1	214.4	240.4	
C7417187	174.5	197.7	215.5	-707.8
C4169044	172.9	214.6	214.6	-712.1
C38064903	164.0	191.0	175.5	-708.0
C41532814	177.4	200.7	168.3	-701.7
C25783453	171.5	227.6	173.7	-708.6
C16473113	194.9	219.6	212.2	-720.4
R109792	171.0	205.4	186.3	-683.1
B8001316	188.1	283.2	195.6	-662.8
C31681262	181.0	211.0	211.4	
R238607	166.5	194.1	218.6	-714.3
C6143299	190.4	195.4	227.8	-706.6
C825036	177.8	197.8	211.6	
C38476469	197.4	222.6	207.8	-713.9
C1570952	187.8	199.4	199.4	
C2563077	163.2	213.8	213.8	-713.4
C6214444	170.6	185.9	210.7	-710.8
C2785888	163.5	207.8	210.5	
C2138434	167.3	212.5	202.1	-721.8
C13513821		221.2	220.8	-710.1
C26828488	192.4	216.6	216.6	-700.1
C3319151		202.5	215.8	-709.1
C15149107	169.7	203.2	194.8	-707.5
C74896154	193.0	211.7	211.7	-716.5
C18979607	168.1	220.9	224.2	-721.8
C103548169	190.9	202.9	203.9	-707.0
C2534807	196.3	204.6	204.3	
C18979505	162.7	204.0	207.3	-714.3
B6006744	172.9	205.3	223.6	-692.9
C4033072	201.1	216.2	216.5	-608.0
U299119	178.5	214.8	214.8	-671.9
C6161860	171.4	217.4	206.6	-712.2
C4812208	166.6	215.5	215.5	-714.4

APPENDIX 11

FULL CODE OF EMAIL.PL

```
1  #!/usr/bin/perl
2  use strict;
3  use warnings;
4  use Email::Send;
5  use Email::Send::Gmail;
6  use Email::Simple::Creator;
7
8  my $formula = <$ARGV[0]>;
9  print "$formula\n";
10 my $email = <$ARGV[1]>;
11 print "$email\n";
12 #takes input from GUI and reads them in from example.pl
13
14 my $email2 = Email::Simple->create(
15     header => [
16         From    => 'david.crizer@gmail.com',
17         To      => ".$email.",
18         Subject => ".$formula." Done",
19     ],
20     body => 'The calculations for formula '.$formula.' are done!',
21 );
22 #creates the email to send and saves it as a variable.
23 my $sender = Email::Send->new(
24     { mailer      => 'Gmail',
25       mailer_args => [
26         username => 'finishedcalculation@gmail.com',
27         password => 'iontrap*',
28       ]
29     }
30 );
31 #Email address information where the email will be sent from
32 eval { $sender->send($email2) };
33 die "Error sending email: $@" if $@;
34 #sends the email
```

APPENDIX 12

FULL CODE OF MERGE_FILES.PL

```
1 use strict;
2 use warnings;
3
4 my $target = pop @ARGV;
5 my @sources = @ARGV;
6
7 die "Usage: $0 in in ... in out\n" if not @sources;
8
9 open my $out, '>>', $target or die "Could not open '$target' for appending\n";
10 foreach my $file (@sources) {
11     if (open my $in, '<', $file) {
12         while (my $line = <$in>) {
13             print $out $line;
14         }
15         close $in;
16     } else {
17         warn "Could not open '$file' for reading\n";
18     }
19 }
20 close $out;
21
22 print "done\n";
```

APPENDIX 13

FULL CODE OF RUN.SH

```
1 #!/bin/sh -x
2
3 `ls -l *.com| awk '{print $9}' |sed 's/.com//g' > file_list`
4
5 cat file_list|while read each_file
6 do
7
8 bsub -q week -n 4 -R "span[ptile=4]" -M 40 -W 120 g03 $each_file".com"
9
10 done
11 /bin/rm file_list
12
```

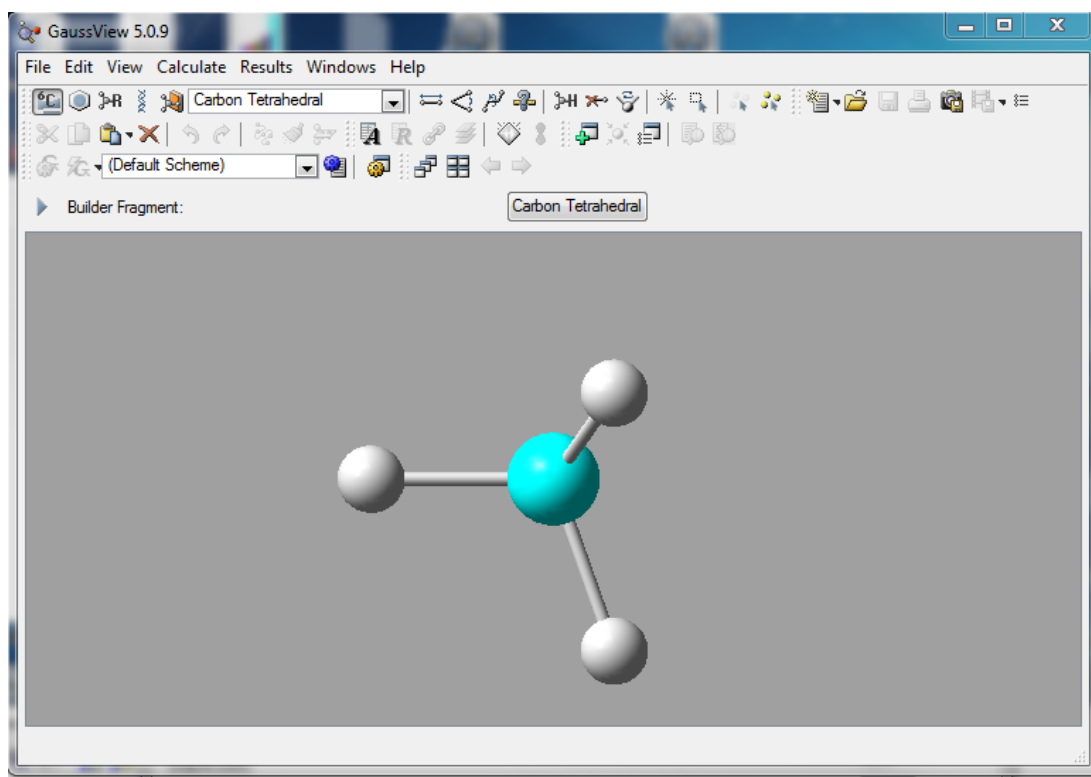
APPENDIX 14

GAUSSVIEW TUTORIAL

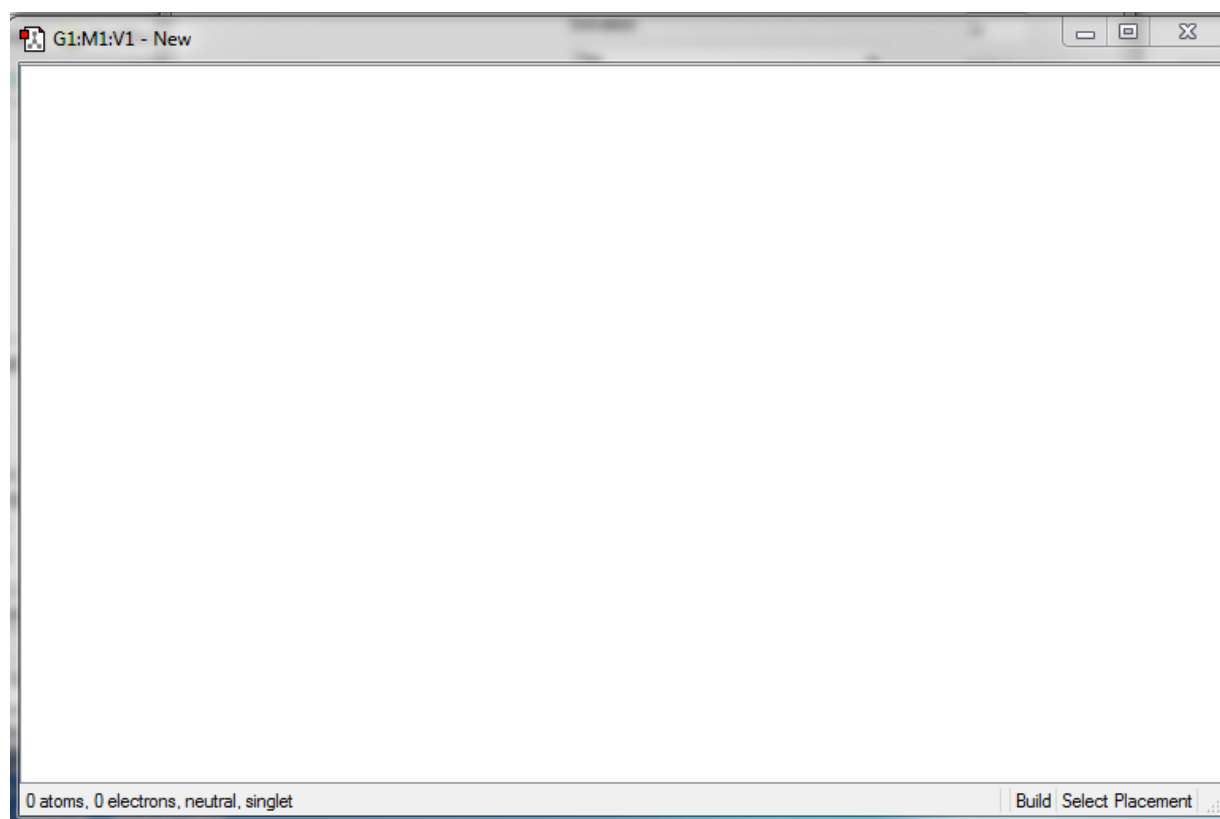
This appendix serves as an introduction/tutorial to the use of GaussView. The tutorial will go through how to make structures and setup input files to run calculations using Gaussian. The tutorial finishes by describing how certain modifications can be made to structures. This is important due to the fact that sometimes small changes in input geometry can help turn a failing calculation into a successful one. Going through this tutorial should allow a new user of GaussView to get a good handle on how to use the program. With this knowledge the user should be capable of using GaussView for any system of interest to them.

1. Open GaussView 5.0. This brings up two windows.

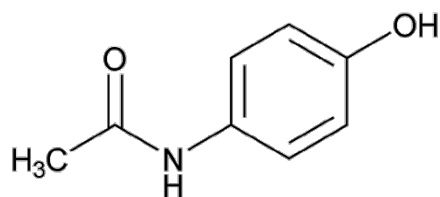
The “Control Window”



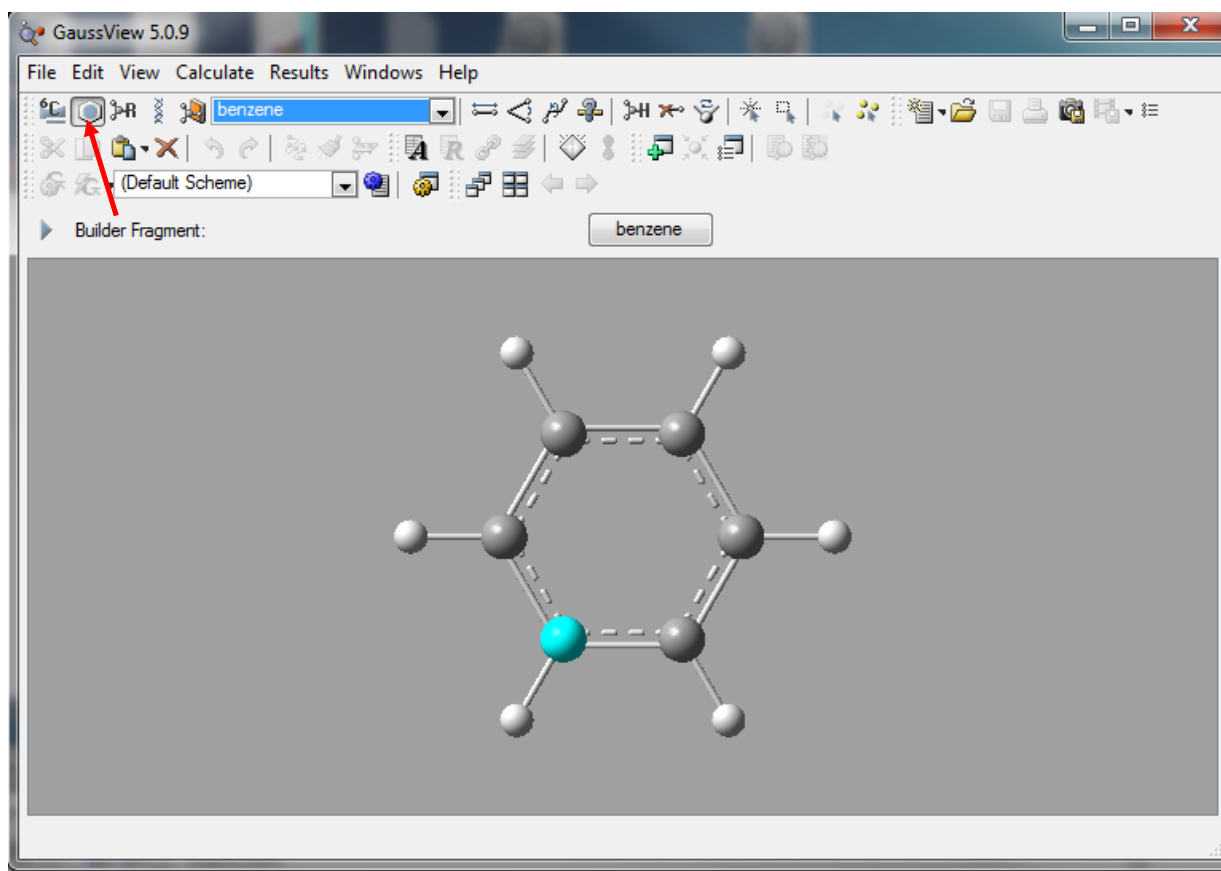
And the “Build Molecule Window”



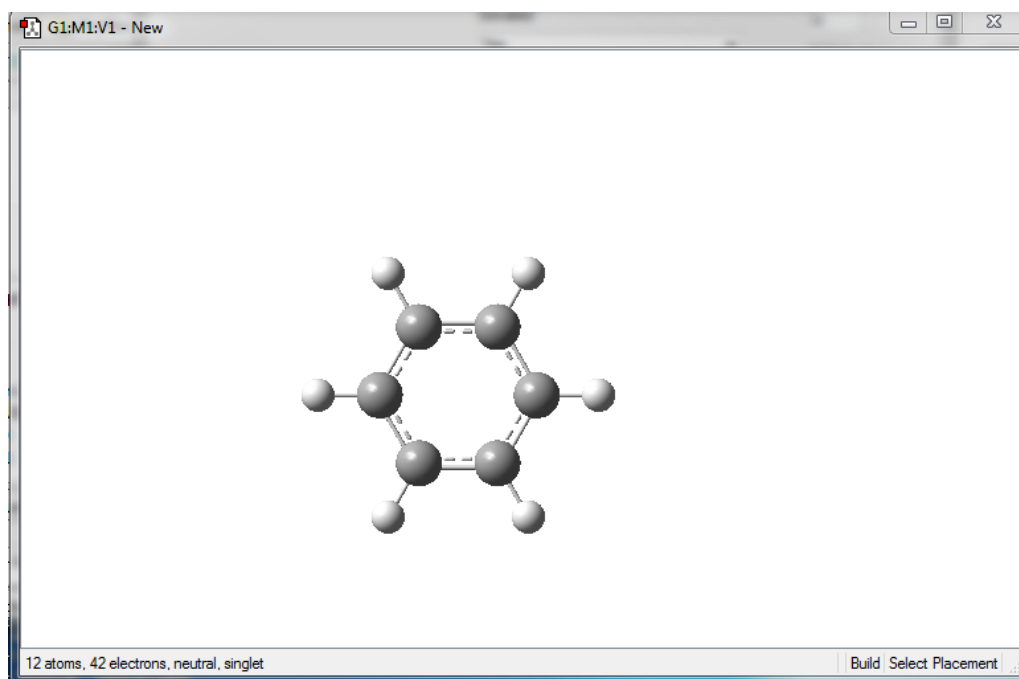
2. Let's build a 3D structure for acetaminophen



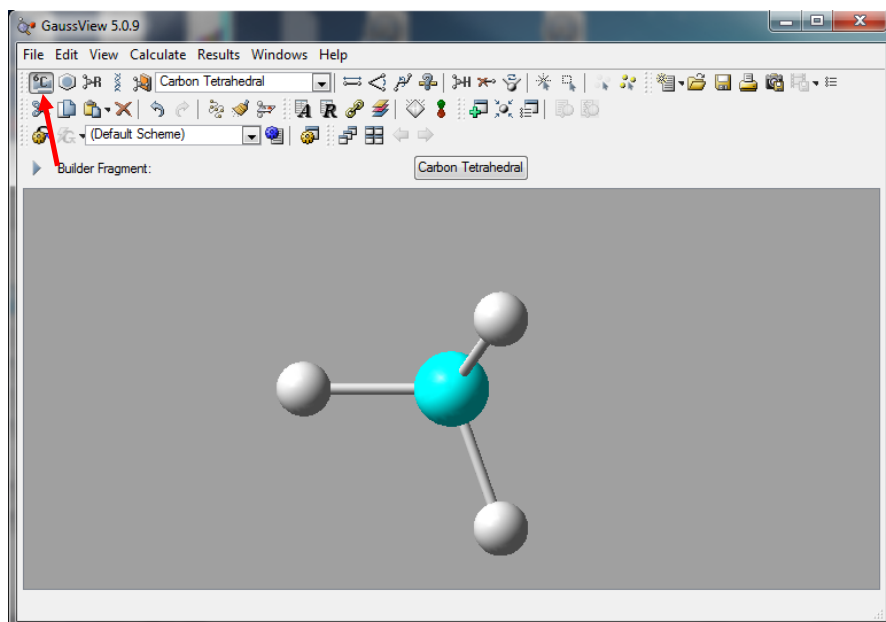
3. Select the ring fragments building tool



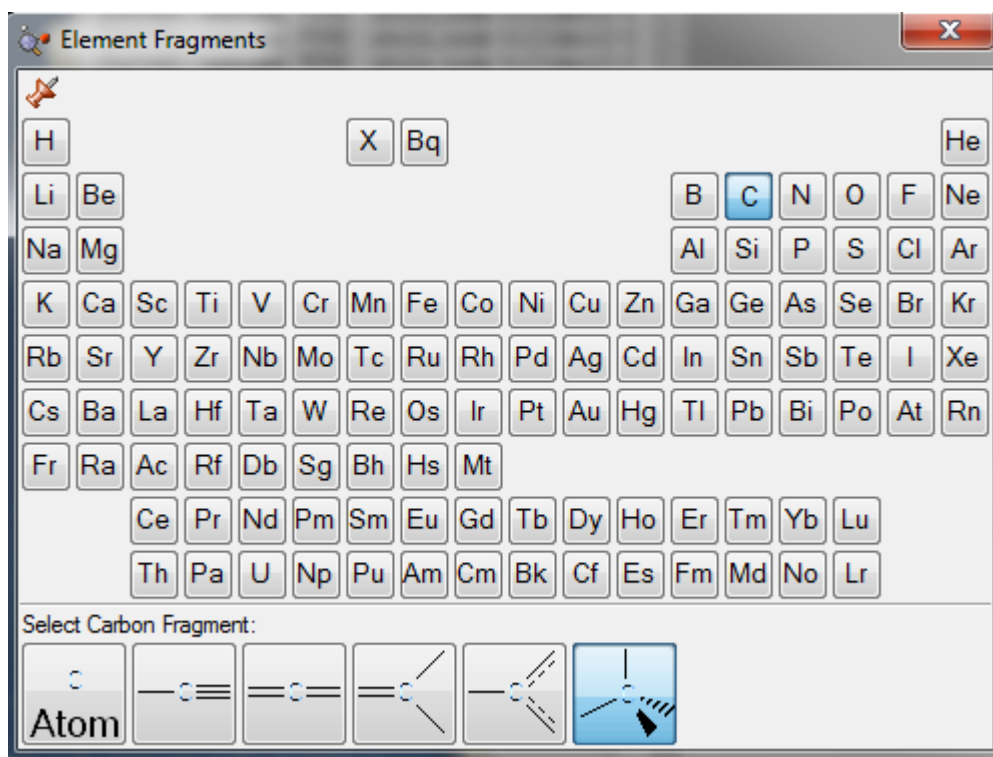
4. Click in the molecule building window and the benzene ring will show up



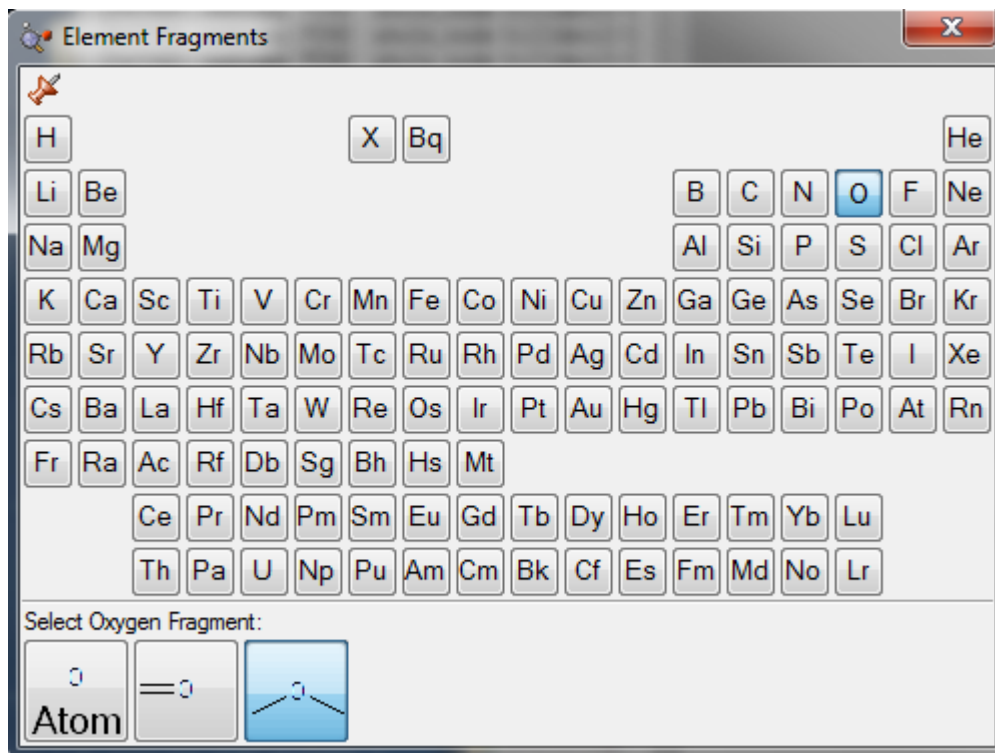
5. To add an oxygen atom click on the Element Fragment button



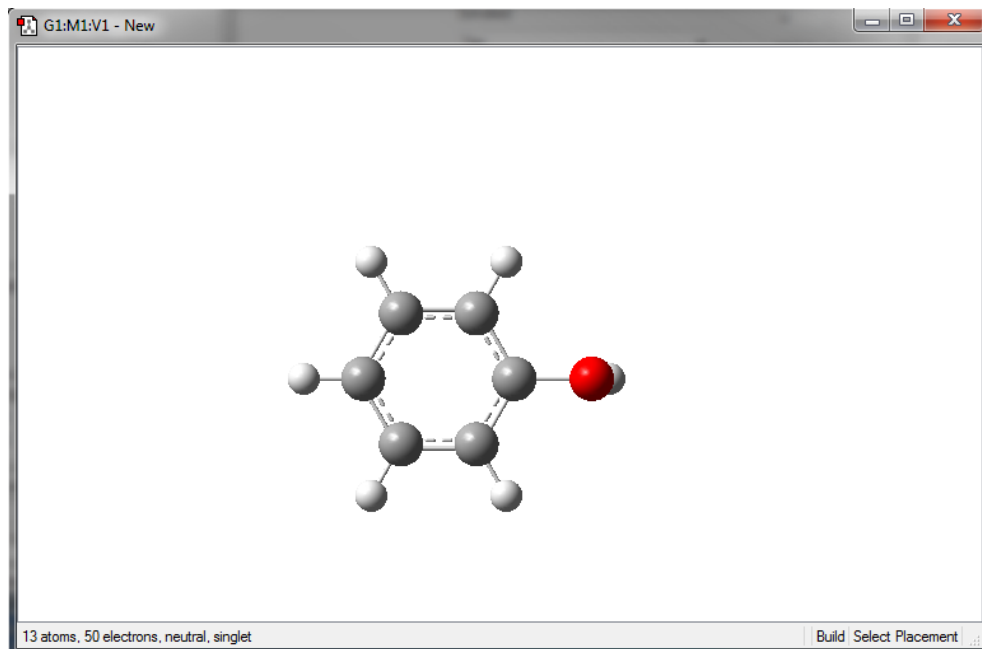
6. This brings up the “Element Fragments” window



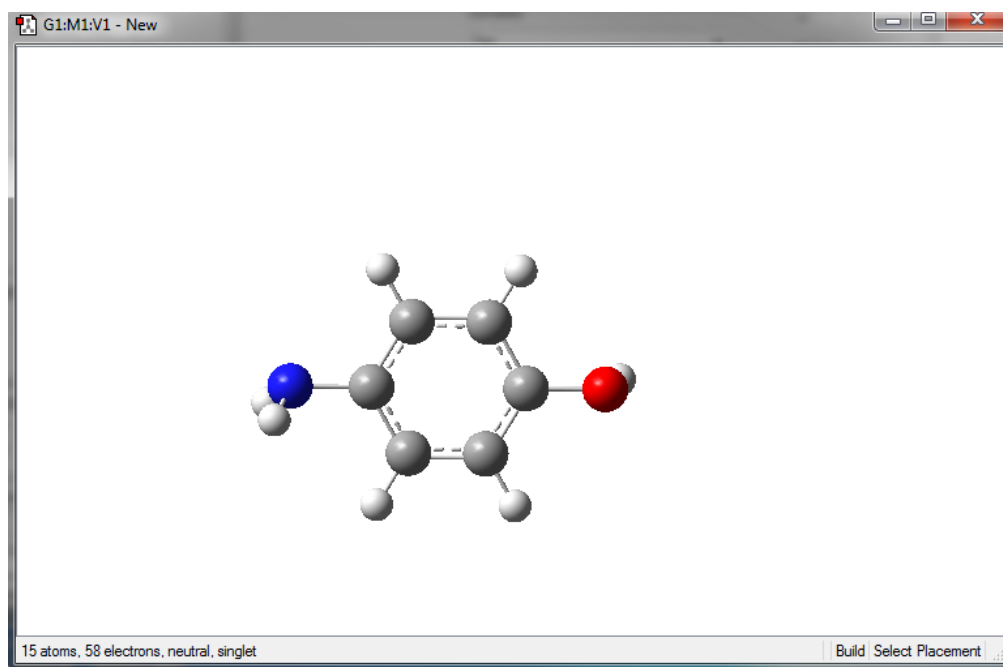
7. The user can select the atom of interest (in this case oxygen) and then select the type of oxygen fragment to add



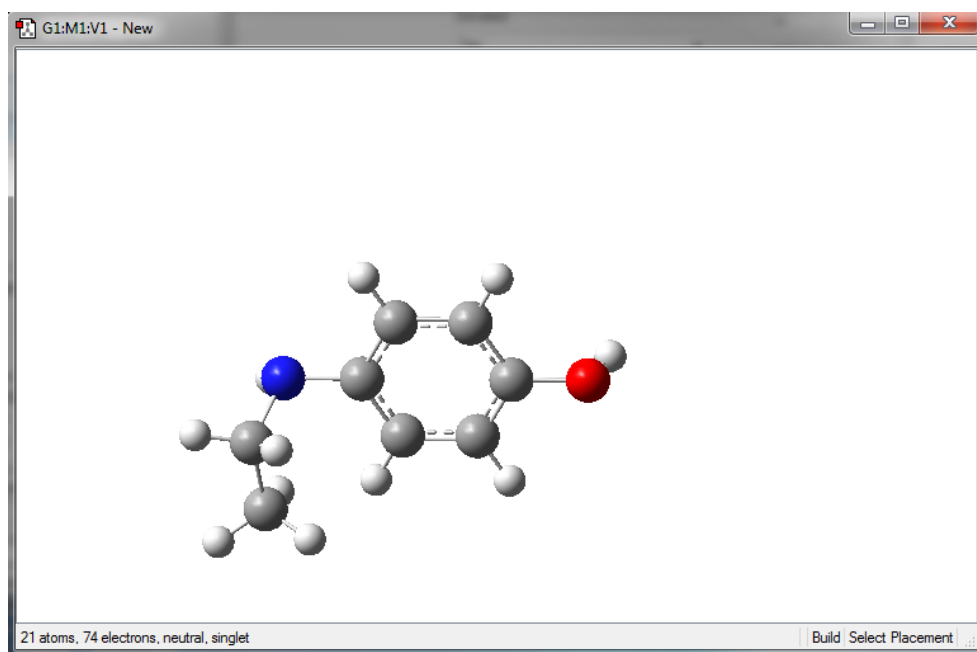
8. In the molecule building window you can then click where you want to place the element fragment that you chose.



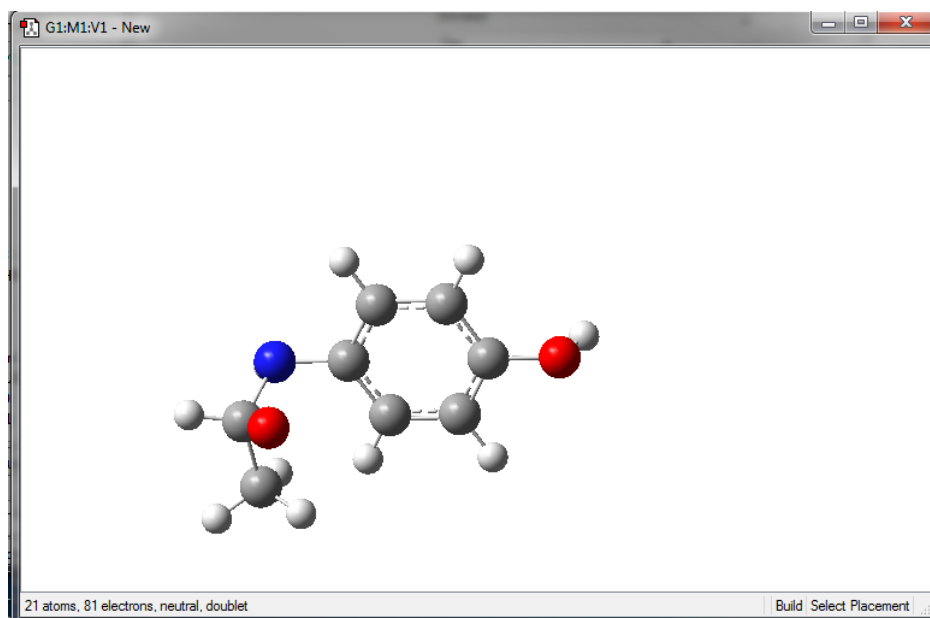
9. Do the same thing to add a NH_2 group to the other side of benzene ring



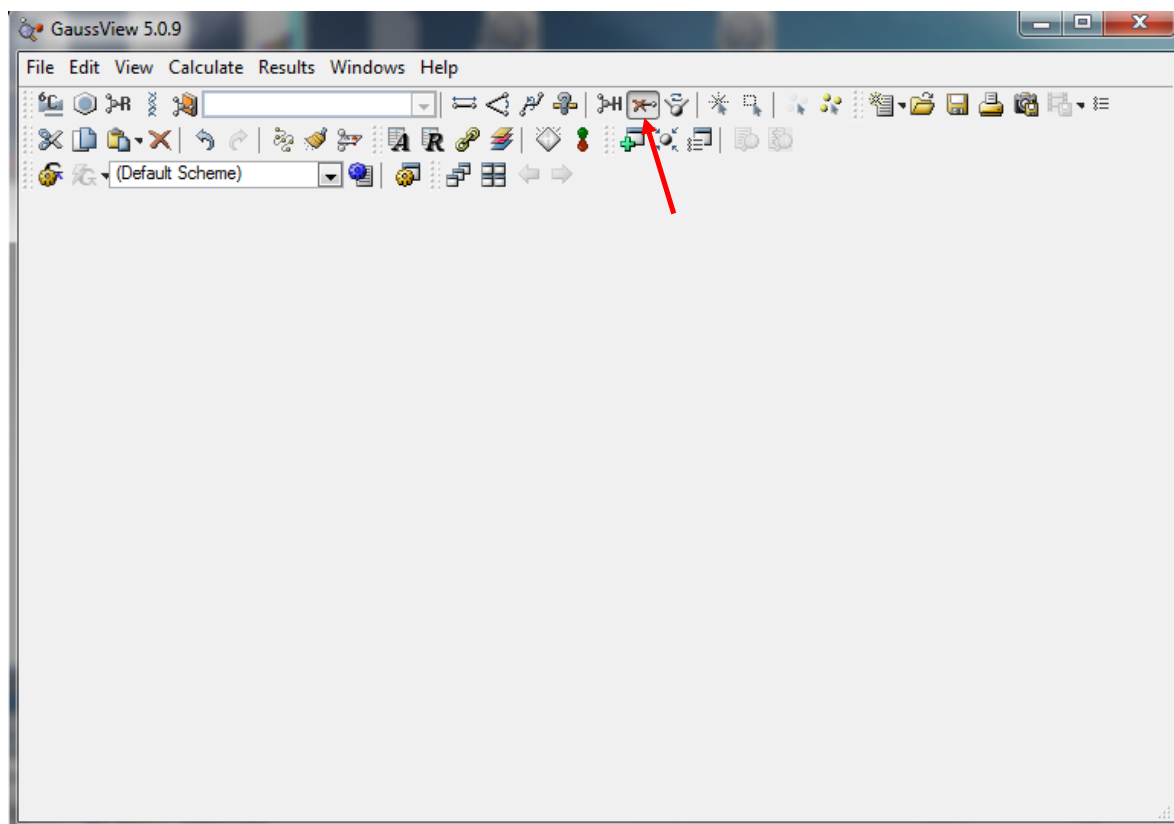
10. Now to add acetyl group we will first add two carbons



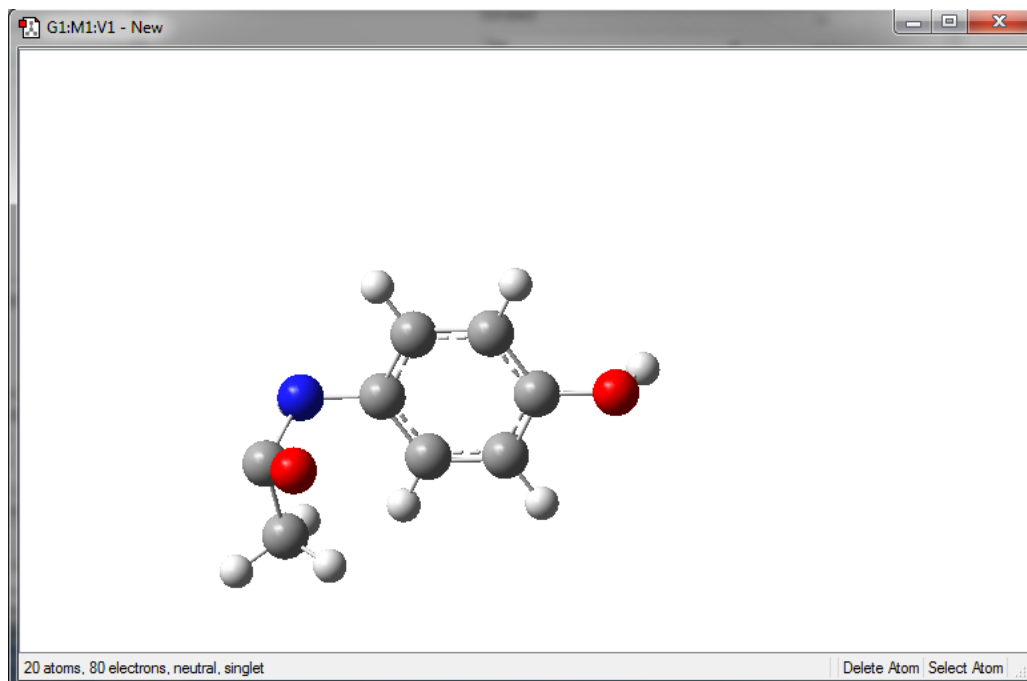
11. Convert one of the hydrogens on the carbon closest to nitrogen to an oxygen. This is done by selecting an oxygen atom in the element fragment window and then clicking on the hydrogen that you want to replace with the oxygen.



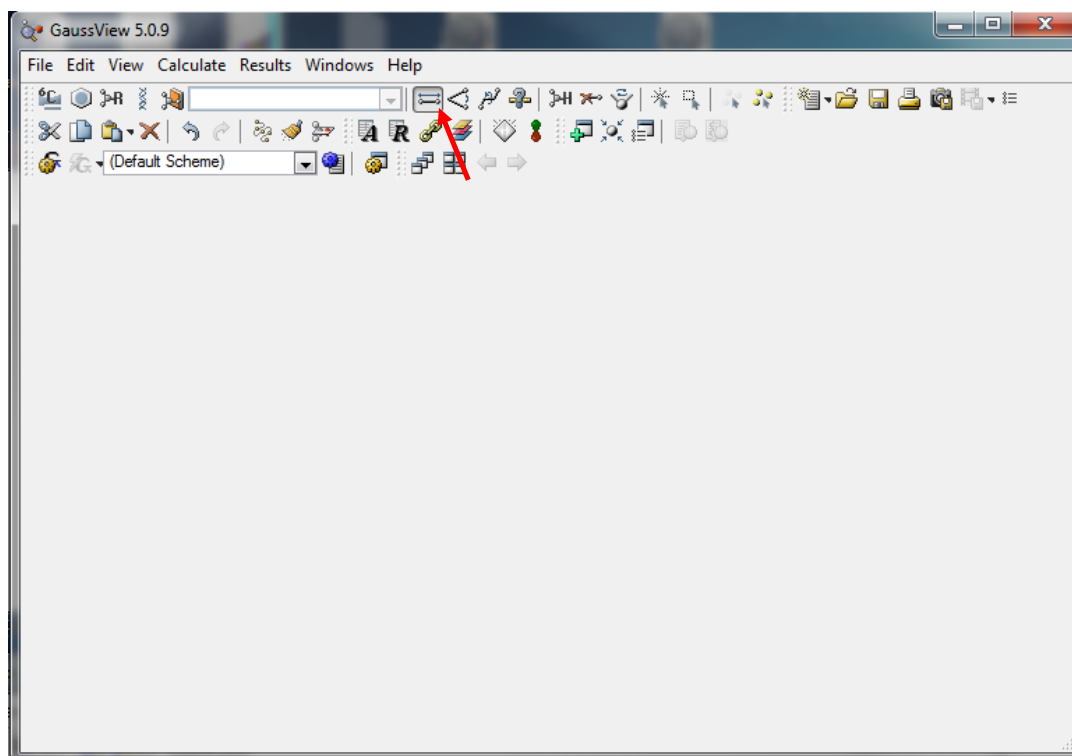
12. Now you need to delete the unneeded hydrogen with “delete atom function”. This is needed since the oxygen atom connects to carbon with a double bond.



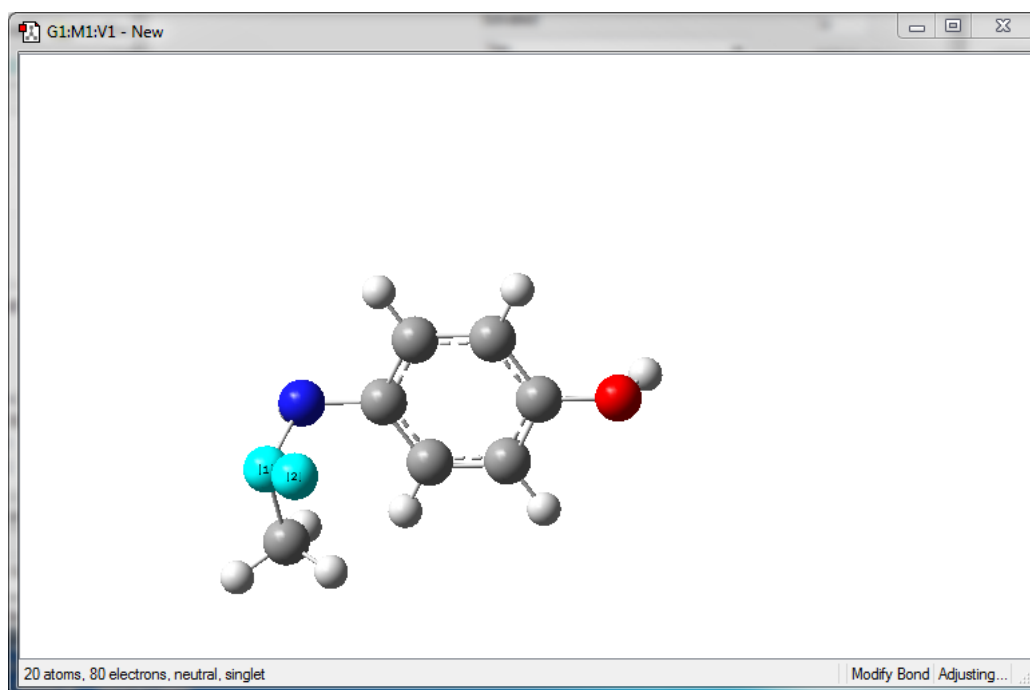
13. Once you select the “delete atom tool” you just have to click on the atom you want to delete and it will disappear.



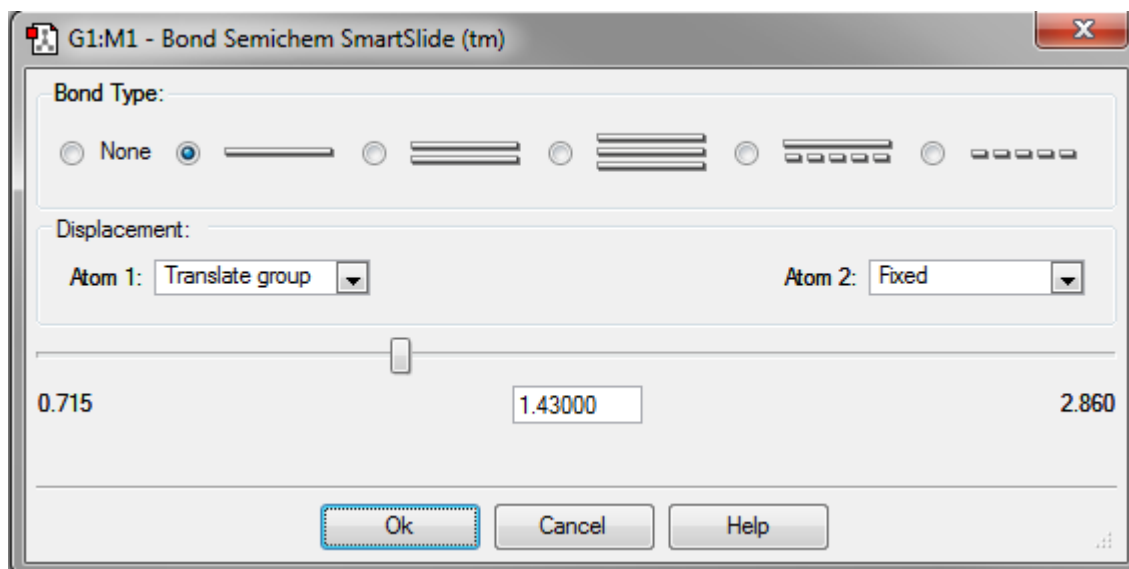
14. Change carbon oxygen bond to a double bound using “modify bond” function

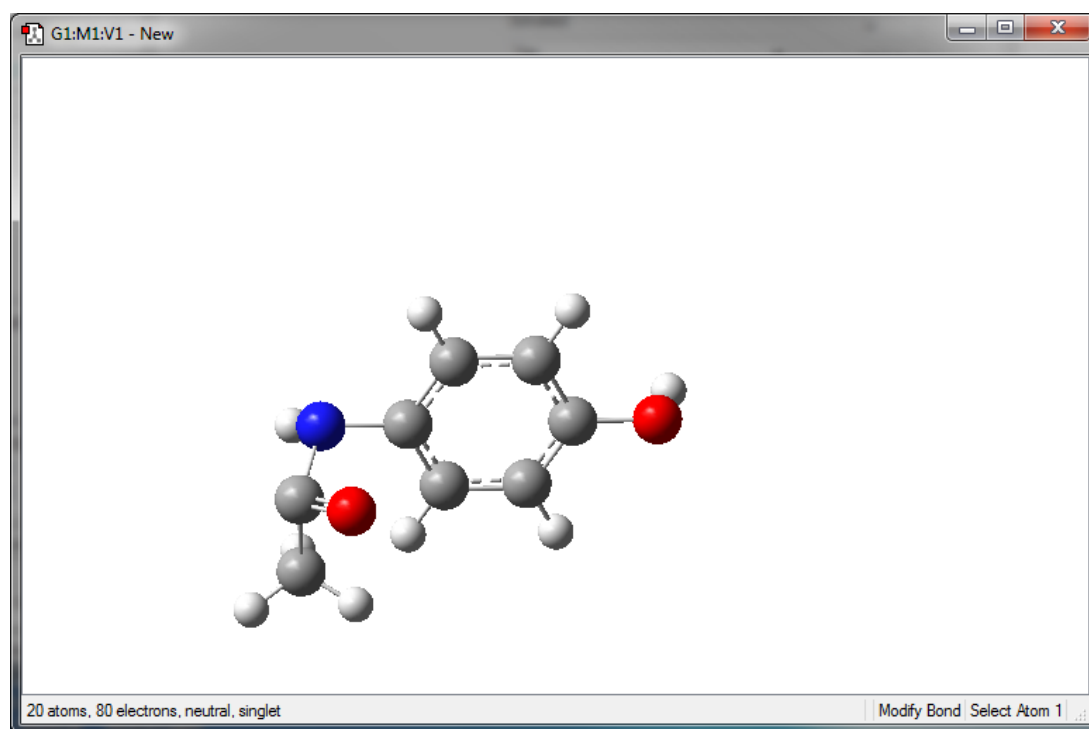
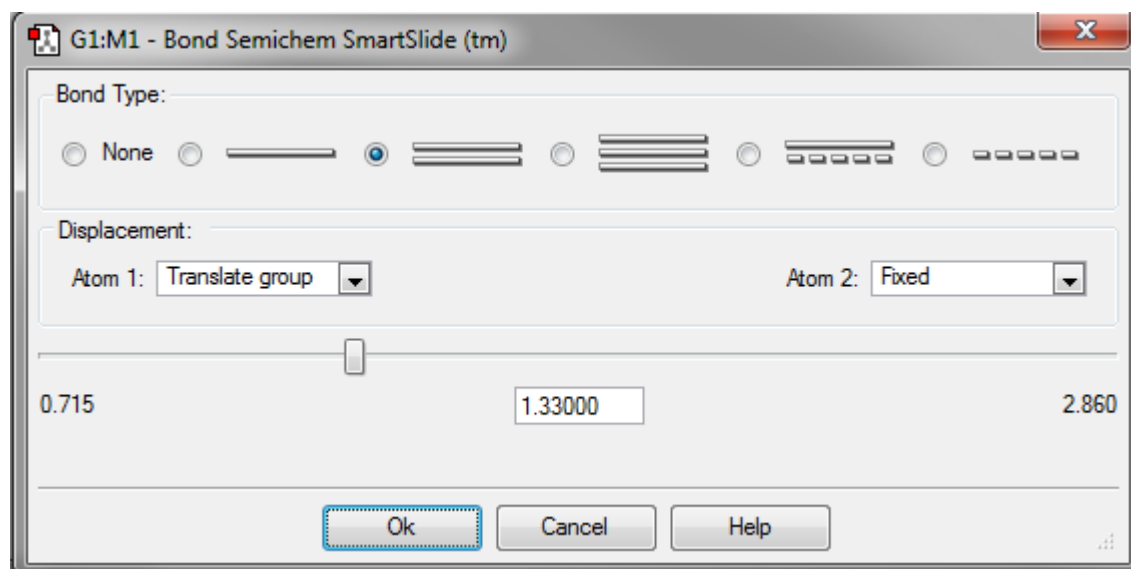


15. First you have to select the two atoms in the bond that you want to modify

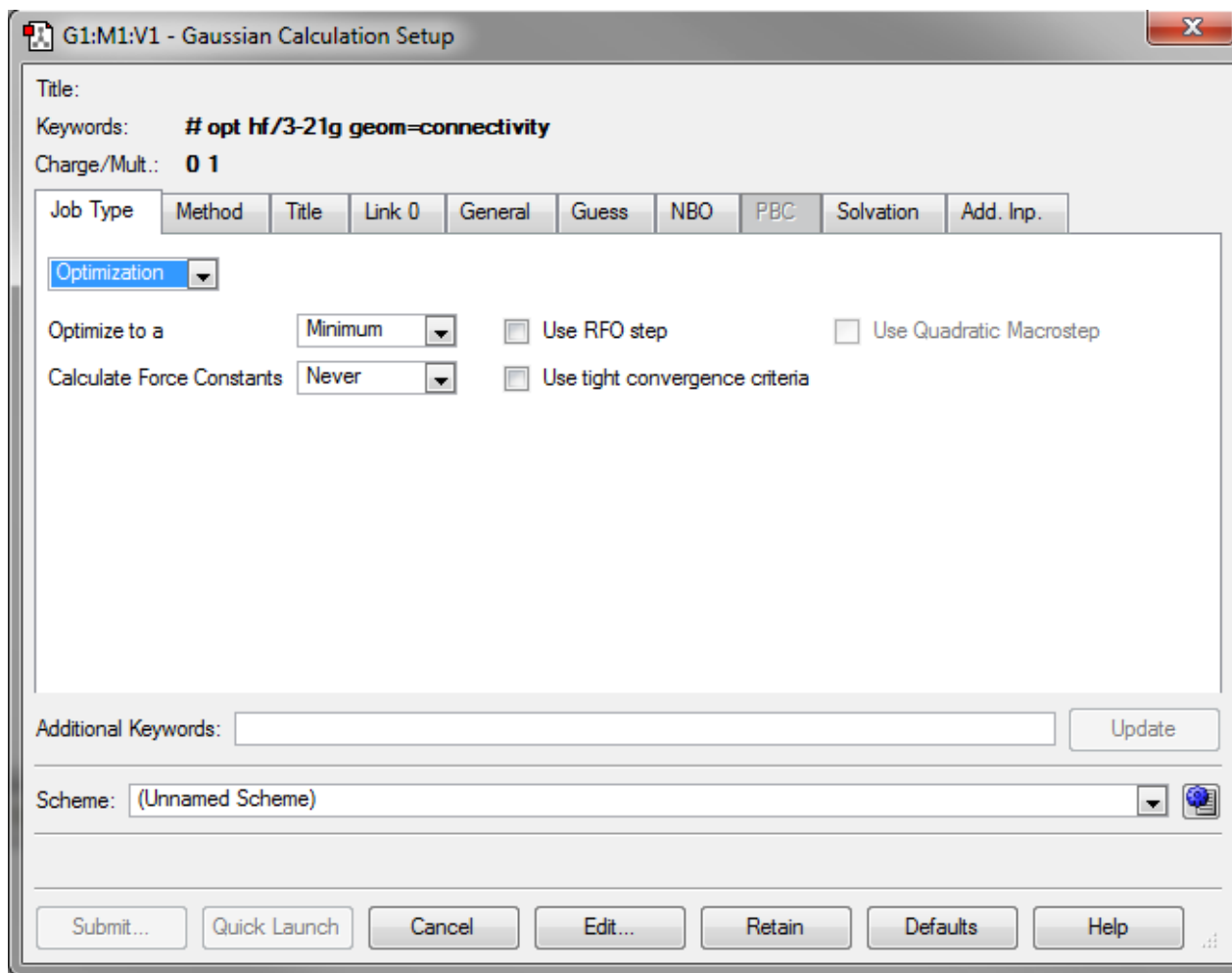


16. That brings up the bond modify window. In this case, change the bond type from single to double. You can then see that once you click “Ok” that there is now a double bond between the carbon and oxygen atoms.





17. Now the structure for acetaminophen has been successfully built. The next start is to prepare the Gaussian calculation input file. First Go to file→save to save the structure. Then choose Calculate→Gaussian Calculation Setup, which brings up the following window. The first tab in this window is the “Job Type” tab, this is where the user determines what type of calculation they want to do. A more through discussion on the types of calculations that you can do in Gaussian is included in Appendix 15.



18. The next tab is the “Method” tab. The method tab is where the user will choose the level of theory and the basis set that they want to use. Additionally, the user will enter the charge of the structure they made. The spin or multiplicity is automatically determined based upon the structure and charge that the user chooses. However, I suggest that the user always checks to make sure that this is correct.

The screenshot shows the 'G1:M1:V1 - Gaussian Calculation Setup' dialog box with the 'Method' tab selected. The 'Keywords' field contains '# opt hf/3-21g geom=connectivity' and 'Charge/Mult.' is set to '0 1'. The 'Method' section includes dropdown menus for 'Ground State', 'Hartree-Fock', and 'Default Spin'. The 'Basis Set' is set to '3-21G'. The 'Charge' is '0' and 'Spin' is 'Singlet'. There is a checkbox for 'Multilayer ONIOM Model'. At the bottom, there is an 'Additional Keywords' field, a 'Scheme' dropdown set to '(Unnamed Scheme)', and a row of buttons: 'Submit...', 'Quick Launch', 'Cancel', 'Edit...', 'Retain', 'Defaults', and 'Help'.

Title:

Keywords: **# opt hf/3-21g geom=connectivity**

Charge/Mult.: **0 1**

Job Type Method Title Link 0 General Guess NBO PBC Solvation Add. Inp.

☐ Multilayer ONIOM Model

Method: Ground State Hartree-Fock Default Spin

Basis Set: 3-21G

Charge: 0 Spin: Singlet

Additional Keywords: Update

Scheme: (Unnamed Scheme)

Submit... Quick Launch Cancel Edit... Retain Defaults Help

19. The next tab of interest is the “Link 0” tab. Here is where you supply information that tells the computer how much memory and how many processors to use. Also you can choose the name of your checkpoint file. Checkpoint files are useful in cases where the calculation didn’t complete. You can use the checkpoint file to restart the calculation where it stopped and not have to start back at the beginning. I typically choose a memory limit of 40 and use 4 processors. However, you may find that different values may suit your needs, but in general these values are pretty good places to start.

The screenshot shows the "G1:M1:V1 - Gaussian Calculation Setup" dialog box. The "Link 0" tab is selected. The "Title" field is empty. The "Keywords" field contains "# opt hf/3-21g geom=connectivity". The "Charge/Mult." field contains "0 1". The "Options" button is selected. The "Memory Limit" is set to "Specify..." with a dropdown arrow, the value "40" is entered, and the unit is "MW". The "Checkpoint File" is set to "Default name" with a dropdown arrow, the text "{parent file name}.chk" is entered, and the "Full Path" checkbox is checked. The "Read-write File" is set to "Don't save" with a dropdown arrow. The "Linda Workers" is set to "Don't use" with a dropdown arrow. The "Shared Processors" is set to "Specify..." with a dropdown arrow, and the value "4" is entered. The "Additional Keywords" field is empty. The "Scheme" is set to "(Unnamed Scheme)". The "Update" button is visible. At the bottom, there are buttons for "Submit...", "Quick Launch", "Cancel", "Edit...", "Retain", "Defaults", and "Help".

Title:

Keywords: **# opt hf/3-21g geom=connectivity**

Charge/Mult.: **0 1**

Job Type Method Title **Link 0** General Guess NBO PBC Solvation Add. Inp.

Options Edit

Memory Limit: Specify... 40 MW

Checkpoint File: Default name {parent file name}.chk Full Path

Read-write File: Don't save

Linda Workers: Don't use

Shared Processors: Specify... 4

Additional Keywords:

Scheme: (Unnamed Scheme)

Submit... Quick Launch Cancel Edit... Retain Defaults Help

20. The “NBO” tab is where the user can choose methods to calculate molecular orbitals and atomic charges.

The screenshot shows the 'G1:M1:V1 - Gaussian Calculation Setup' dialog box with the 'NBO' tab selected. The 'Keywords' field contains the text: `# opt hf/3-21g scrf=(cpcm,solvent=water) pop=nbo geom=connectivity`. The 'Charge/Mult.' field is set to `0 1`. The 'Type' dropdown is set to 'Full NBO' and the 'Checkpoint Save' dropdown is set to 'Don't save'. The 'Additional Keywords' field is empty. The 'Scheme' dropdown is set to '(Unnamed Scheme)'. The bottom of the dialog features buttons for 'Submit...', 'Quick Launch', 'Cancel', 'Edit...', 'Retain', 'Defaults', and 'Help'.

Title:

Keywords: **# opt hf/3-21g scrf=(cpcm,solvent=water) pop=nbo geom=connectivity**

Charge/Mult.: **0 1**

Job Type Method Title Link 0 General Guess **NBO** PBC Solvation Add. Inp.

Type: Full NBO

Checkpoint Save: Don't save

Additional Keywords: Update

Scheme: (Unnamed Scheme)

Submit... Quick Launch Cancel Edit... Retain Defaults Help

21. The “Solvation” tab is where users can choose to setup calculations that are done in solvents instead of in the gas-phase. There are a variety of models that the user can choose as well as a variety of solvents to choose from.

The screenshot shows the 'G1:M1:V1 - Gaussian Calculation Setup' dialog box with the 'Solvation' tab selected. The 'Title' field is empty. The 'Keywords' field contains the text '# opt hf/3-21g scrf=(cpcm,solvent=water) geom=connectivity'. The 'Charge/Mult.' field shows '0 1'. The 'Job Type' tab is selected in the top navigation bar. Below the tabs, the 'Model' dropdown is set to 'CPCM' and the 'Solvent' dropdown is set to 'Water'. There is an unchecked checkbox for 'Read additional input'. At the bottom, there is an 'Additional Keywords' text box, an 'Update' button, a 'Scheme' dropdown set to '(Unnamed Scheme)', and a row of buttons: 'Submit...', 'Quick Launch', 'Cancel', 'Edit...', 'Retain', 'Defaults', and 'Help'.

Title:

Keywords: **# opt hf/3-21g scrf=(cpcm,solvent=water) geom=connectivity**

Charge/Mult.: **0 1**

Job Type Method Title Link 0 General Guess NBO PBC Solvation Add. Inp.

Model: CPCM

Solvent: Water

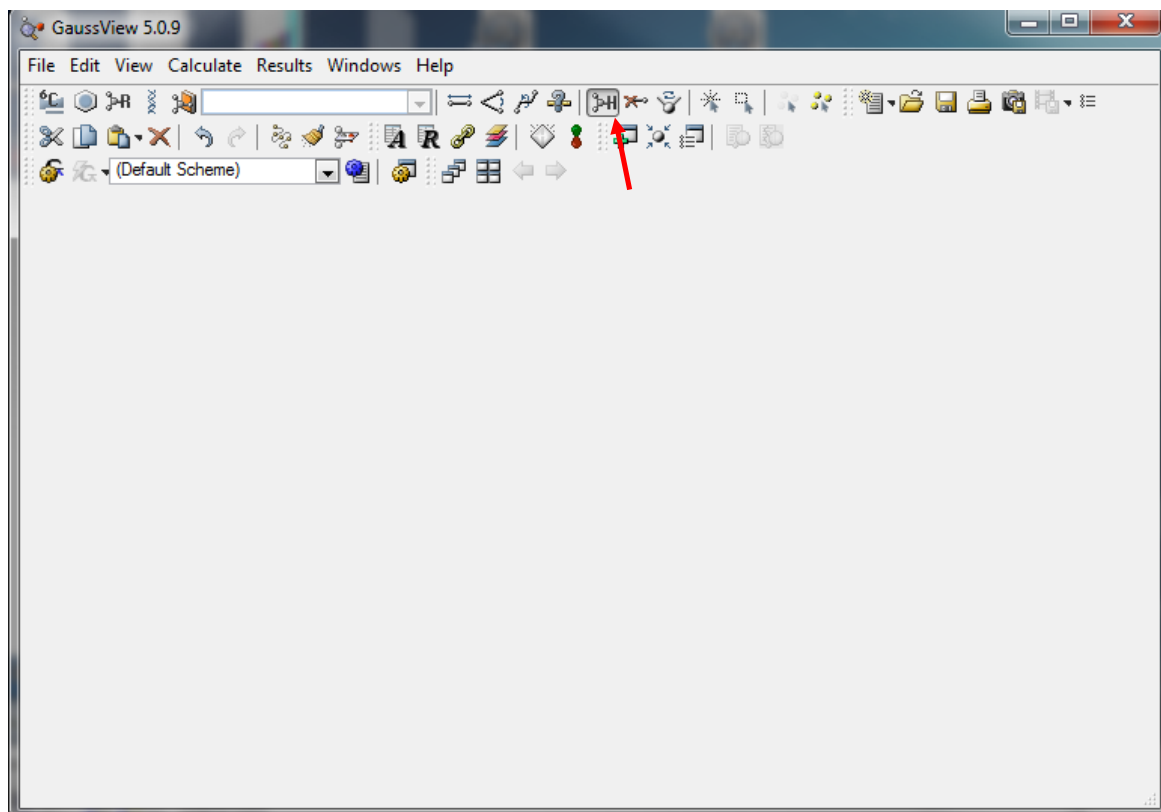
☐ Read additional input

Additional Keywords: Update

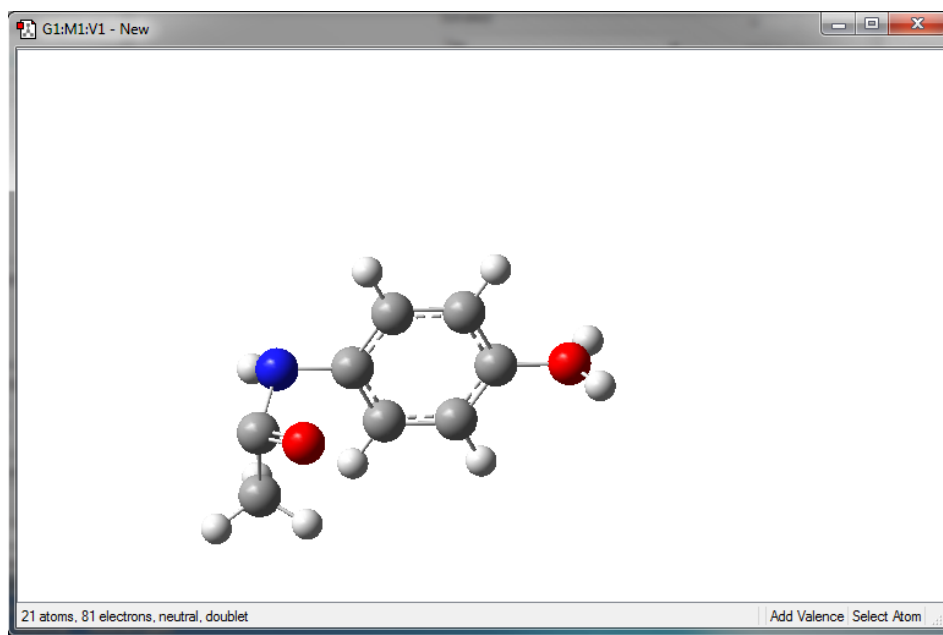
Scheme: (Unnamed Scheme)

Submit... Quick Launch Cancel Edit... Retain Defaults Help

22. If you need to make your structure protonated use the “add valence” button.



23. Once you have clicked on the “add valence” button you just have to click on the atom that you want to add the hydrogen to in the build molecule window. The example below shows an example for protonating the hydroxyl oxygen in acetaminophen.



24. Now to setup the calculation input file is essentially the same as previously discussed. The only difference comes in the “Method” tab where you need to change the charge from 0 to 1, this will automatically adjust the spin of the system.

The screenshot shows the 'G1:M1:V1 - Gaussian Calculation Setup' dialog box with the 'Method' tab selected. The 'Keywords' field contains '# opt hf/3-21g geom=connectivity'. The 'Charge/Mult.' field is set to '1 1'. The 'Method' section includes dropdown menus for 'Ground State', 'Hartree-Fock', and 'Default Spin'. The 'Basis Set' is set to '3-21G'. The 'Charge' is set to '1' and the 'Spin' is set to 'Singlet'. There is a checkbox for 'Multilayer ONIOM Model' which is unchecked. At the bottom, there is an 'Additional Keywords' field, a 'Scheme' dropdown set to '(Unnamed Scheme)', and a row of buttons: 'Submit...', 'Quick Launch', 'Cancel', 'Edit...', 'Retain', 'Defaults', and 'Help'.

Title:

Keywords: **# opt hf/3-21g geom=connectivity**

Charge/Mult.: **1 1**

Job Type Method Title Link 0 General Guess NBO PBC Solvation Add. Inp.

☐ Multilayer ONIOM Model

Method: Ground State Hartree-Fock Default Spin

Basis Set: 3-21G

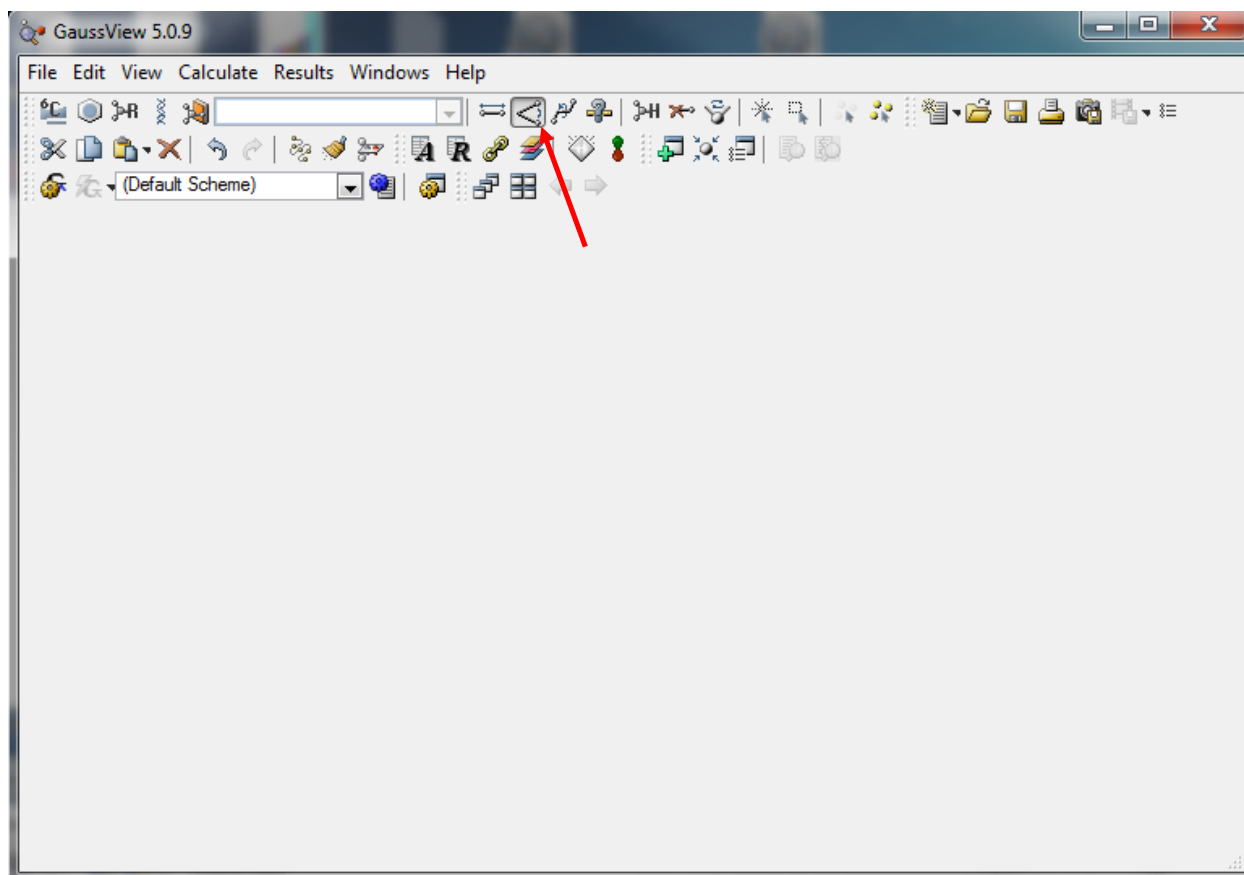
Charge: 1 Spin: Singlet

Additional Keywords: Update

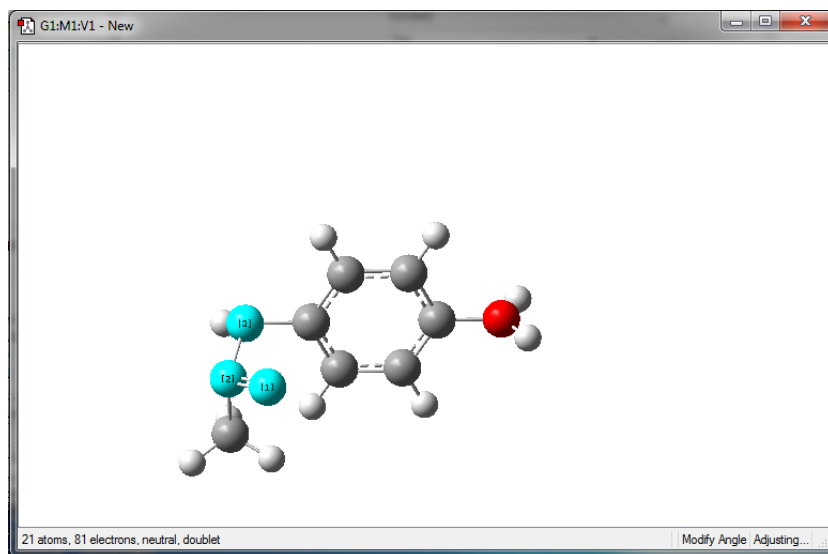
Scheme: (Unnamed Scheme)

Submit... Quick Launch Cancel Edit... Retain Defaults Help

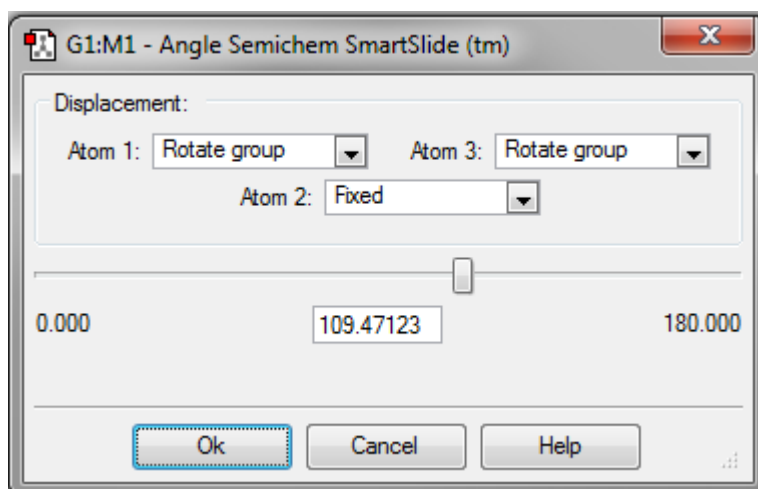
25. Sometimes you may find that calculations fail. It is often the case that small changes in input geometry can help make the calculation run and complete successfully. The first thing you can do is to modify angle. First choose the “modify angle” button.



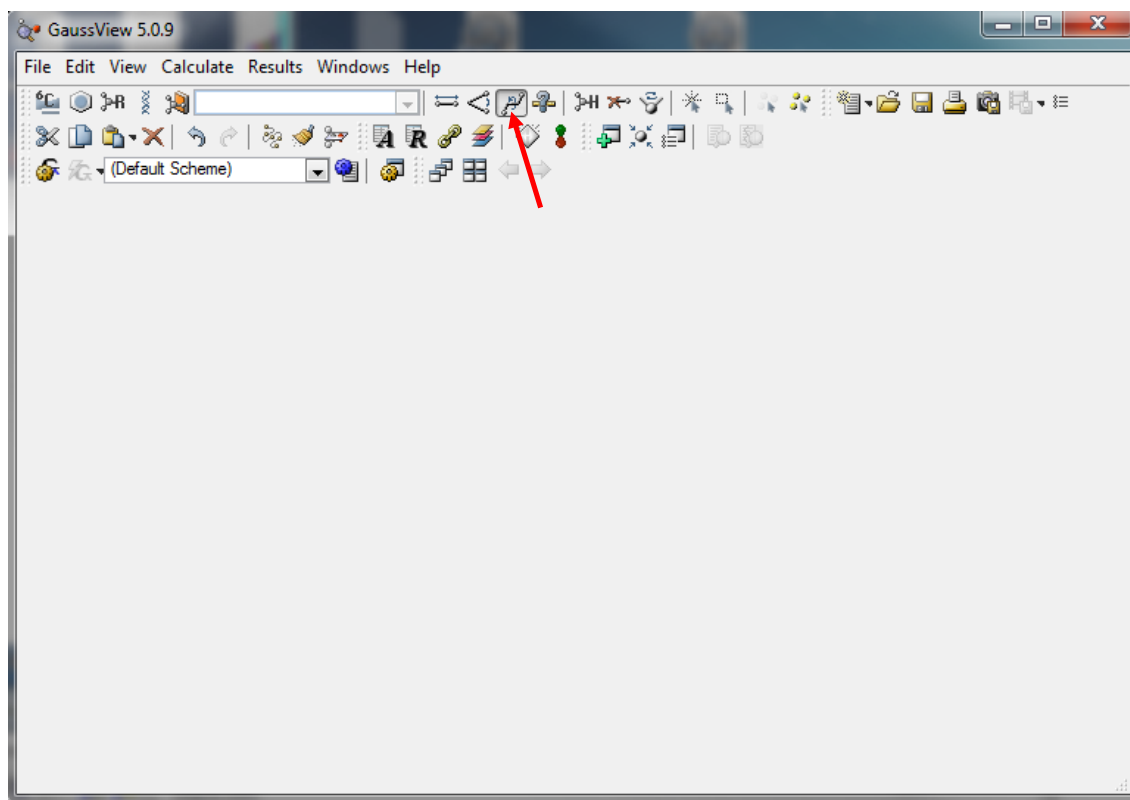
26. Choose the 3 atoms that you want to change the angle of



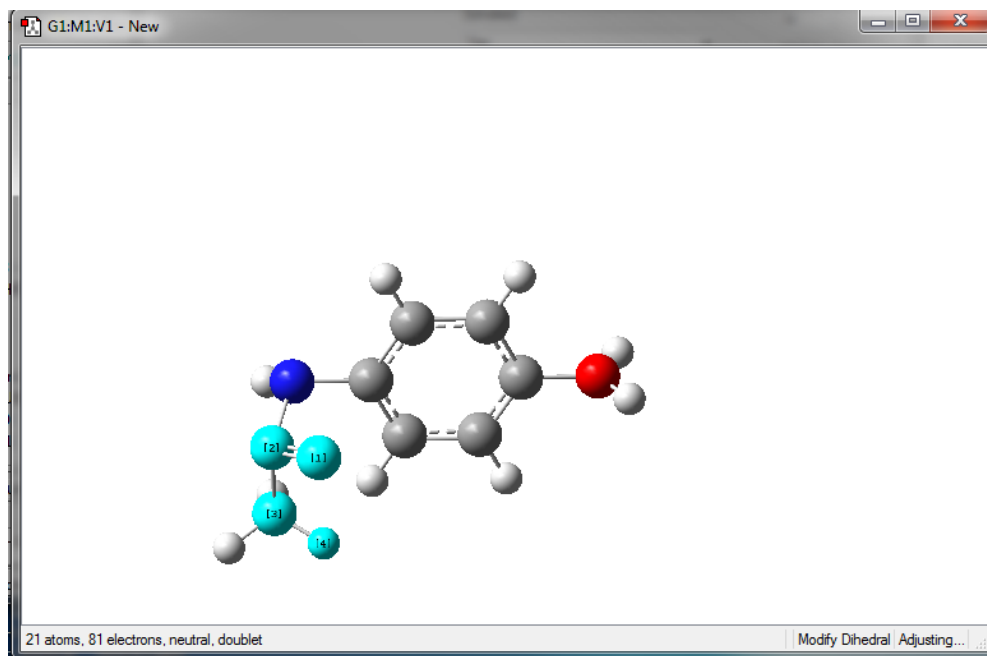
27. This brings up the modify angle window where the user can change the value of the angle



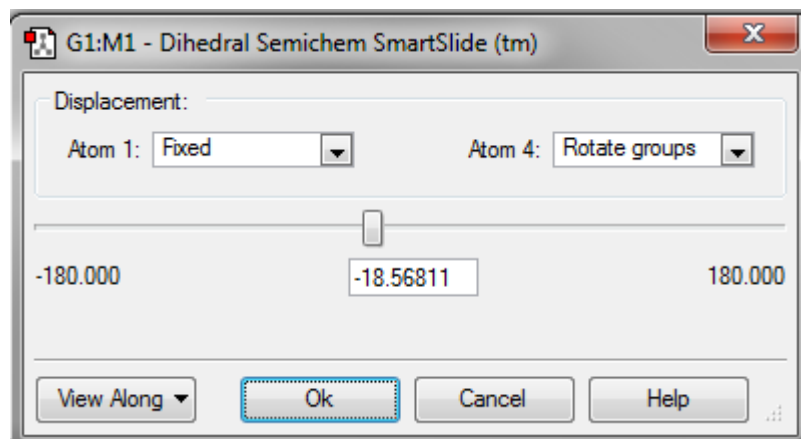
28. A similar modification that can be made is to manipulate the dihedral angle. Choose the “modify dihedral” button.



29. Choose the 4 atoms that make up the dihedral angle of interest



30. This brings up the modify dihedral angle window where the user can change the value of the dihedral angle



APPENDIX 15

OVERVIEW OF CALCULATION TYPES IN GAUSSIAN

A15.1 Introduction

This appendix serves as an overview of the types of calculations that can be done in the Gaussian software. General information about the various types of calculation will first be discussed. Finally, an overview of calculations that are particularly useful for the Glish Group will be presented.

A15.2 Calculation Types in Gaussian

The Gaussian program has the ability to run a variety of calculations. Some of these calculation types are not of great importance to the Glish Group and will not be discussed. This section serves as an overview on the basics on what the different types of calculations do and what information results from these calculations.

A15.2.1 Geometry Optimization

The first calculation type to be covered and the most commonly used in the Glish Group is geometry optimization. Keyword for calling for a geometry calculation to run is “Opt”. Geometry optimization is a procedure that attempts to find the configuration of minimum energy of the structure. The procedure calculates wave functions and energies at a starting geometry and then proceeds to search for a new geometry that is at a lower energy. This process repeats until the lowest energy geometry has been found. It is important to note that this process does not always find the global minimum. A geometry optimization calculation gives the atomic coordinates of the optimized structure, which is the most important piece of information that is produced from the calculation. The calculation also results in the calculation of optimized parameters such as atomic distances and angles. Other

important results from geometry calculations are HOMO/LUMO eigenvalues, Mulliken atomic charges, and dipole moments.

A15.2.2 Single Point Energy

A single point energy calculation calculates the energy, wave function, and other requested properties at a single fixed geometry. The keyword to start a single point energy calculation is “SP”. Single point energy calculations are particularly useful in situations where the geometry is optimized at more computer efficient theoretical models and basis sets then that optimized geometry is used to calculate energies with more accurate theoretical models and basis sets. The most important result from a single point energy is the single point energy at the fixed geometry. The calculation also results in information such as orbital symmetries, Mulliken atomic charges, and dipole moments.

A15.2.3 Frequency

The information obtained from running a frequency calculation can be of great use to the Glish lab. A frequency calculation is started using the keyword “Freq”. It is also important to note that the keyword “Opt Freq” will run a geometry optimization calculation followed by a frequency calculation. Frequency calculations determine the vibrations of the structure. Additionally, the results also include a thermochemical analysis. The same combination of theoretical model and basis has to be used for both the geometry optimization calculation and the frequency calculation. There is a useful check that the user can make to be sure that the calculation looks like it ran properly. First, the user can examine the number of vibrational modes determined and the number of vibrations should be equal to $3N-6$ or $3N-5$ where N is the number of atoms. Also there should be no imaginary frequencies unless the

frequency calculation is being run on an optimized transition state structure in which case there should be one imaginary frequency.

There is a variety of useful information that results from a frequency calculation. One of the most important pieces of information obtained from a frequency calculation are the infrared (IR) vibration frequencies and intensities. The calculation also gives a single point energy, harmonic frequencies, reduced masses, and constants. While not a default, frequency calculations can also give information about Raman frequencies and intensities. There is also a wealth of thermochemical data that comes from a frequency calculation. The calculation results in values for temperature, pressure, the isotopes used, and the molecular mass. It also results in information about entropy and constant volume molar heat capacity. Frequency calculations also give values for thermal energy, enthalpy, and free energy. An excellent reference on thermochemistry data and Gaussian calculation is the white paper by Ochterski. This includes a thorough overview of the thermochemical data output from a frequency calculation and how this information can be used to calculate a variety of important thermochemical values.

A15.2.4 Molecular Orbitals and Population Analysis

Population analysis is a mathematical method that partitions a wave function or electron density into charges on nuclei, bond orders, or other related information. Results from these calculations are not experimentally observable. The calculations can be particularly helpful in aiding in the prediction of molecular interaction sites. The keyword to run population analysis calculations is “Pop=X” where X can be None, Reg, Full, NBO and a variety of other methods.

A15.2.5 Solvation

All of the calculation types previously discussed have been run in the gas-phase. However, any of these calculation can be run in simulated solvent systems. The keyword “SCRF” will run the calculations in the solvent phase. This creates a solvent cavity around the structure and performs whatever calculation type that is desired. There are a variety of solvation methods available in Gaussian. Additionally, there are multiple solvents that can be used.

A15.3 Calculations of Importance to the Glish Group

This section serves to discuss how to use some of the previously discussed calculation types to obtain information that has been found to be of particular use to the Glish Group. Included is discussion on relative energies, ion energetic values, thermochemistry, IR frequencies, and transition states.

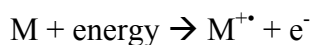
A15.3.1 Relative Energies

A relative energy is a comparison of energies of different structures. This gives a basic idea as to which structures are most likely to be observed. It is particularly useful for a quick comparison of many structures. Relative energy is calculated by just subtracting energies from one another.

$$\text{Relative Energy} = E(\text{structure 1}) - E(\text{structure 2})$$

A15.3.2 Ion Energetics

Ion energetic values of interest to the Glish Group are ionization energy, electron affinity, and proton affinity. Ionization energy is the amount of energy that is required to remove the most loosely bound electron, the valence electron, of an isolated gaseous molecule to form a cation.



Calculations need to be run on both the neutral molecule and the positively charged ion.

Ionization energy can then be calculated using the following equation.

$$\textit{Ionization Energy} = E(\textit{cation}) - E(\textit{neutral})$$

There are two different types of ionization energies that can be calculated, vertical or adiabatic. Vertical ionization energy is calculated using the same geometry for both the neutral molecule and the cation. To do this the neutral molecule would be optimized and then this optimized geometry is used for single point energy calculation of the cation.

Adiabatic ionization energy is calculated using optimized geometries for both the neutral molecule and the cation.

Electron affinity is the amount of energy that is released when an electron is added to a neutral species in the gaseous state to form a negative ion.



Again two calculations are needed to determine the electron affinity. The energy of the neutral molecule and the energy of the negatively charged ion. Electron affinity is calculated using the following equation.

$$\textit{Electron Affinity} = E(\textit{neutral}) - E(\textit{anion})$$

Again the consideration of whether the value for adiabatic or vertical electron affinity is desired.

Proton affinity is the energy that is released when a proton is added to a species.



Again two calculations are needed. Energies are needed for both the neutral molecule and the positively charged protonated ion. The proton affinity is calculated using the following equation.

$$\text{Proton Affinity} = E(\text{neutral}) - E(\text{protonated})$$

A15.3.3 Thermochemistry

As mentioned previously there is a lot of thermochemical information that comes from a frequency calculation. The most important information for the Glush Group from a Gaussian output file from a frequency calculation is shown in Figure A15.1. Listed beside each line is the thermochemical equations and energy values that relate to the thermochemical values that results from a Gaussian frequency calculation.

Zero-point correction=	0.317977 (Hartree/Particle) ZPE
Thermal correction to Energy=	0.338330 $E_{\text{tot}} = E_t + E_r + E_v + E_e$
Thermal correction to Enthalpy=	0.339274 $H_{\text{corr}} = E_{\text{tot}} + k_b T$
Thermal correction to Gibbs Free Energy=	0.267547 $G_{\text{corr}} = H_{\text{corr}} + TS_{\text{tot}}$
Sum of electronic and zero-point Energies=	-1008.997916 $\epsilon_0 + \epsilon_{\text{ZPE}}$
Sum of electronic and thermal Energies=	-1008.977563 $\epsilon_0 + E_{\text{tot}}$
Sum of electronic and thermal Enthalpies=	-1008.976619 $\epsilon_0 + H_{\text{corr}}$
Sum of electronic and thermal Free Energies=	-1009.048346 $\epsilon_0 + G_{\text{corr}}$

Figure A15.1. Gaussian thermochemistry output

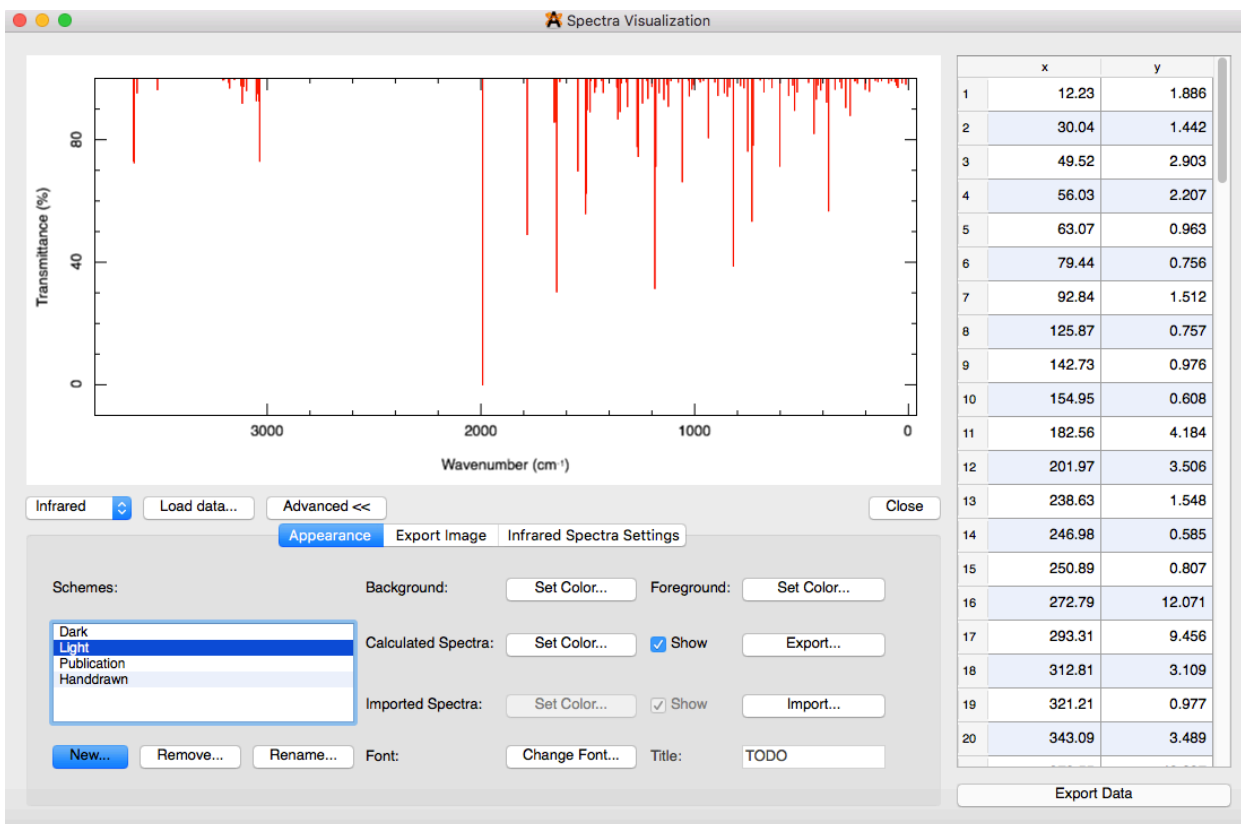
The Gaussian white paper by Ochterski is a great resource for using these values to calculate important thermochemical values. Therefore, I will not discuss these here. The white paper includes information about how to calculate enthalpies and free energies of reactions, reaction rates, and enthalpies and free energies of formation.

A15.3.4 IR Frequencies

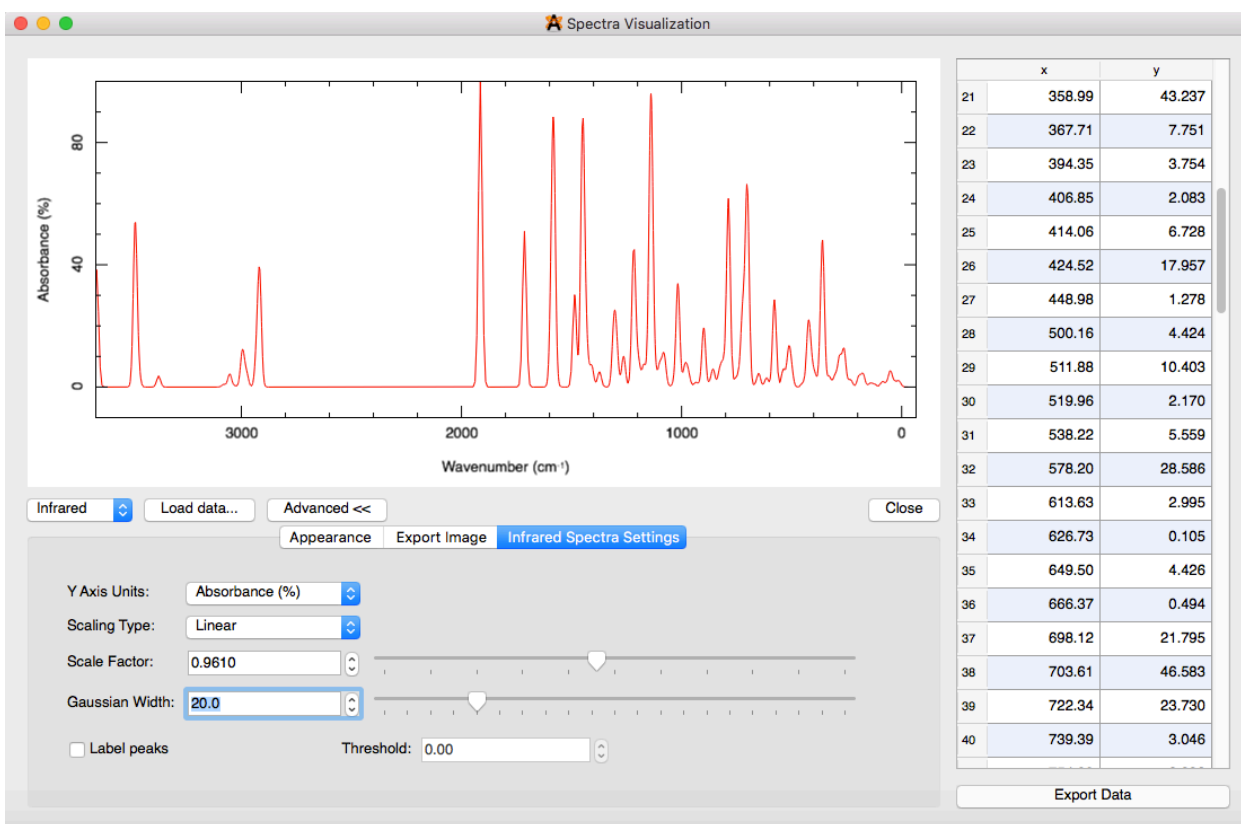
Calculated IR frequencies are needed for comparison to IR spectroscopy experiments. The comparison of theoretical and experimental data allows for the determination of what the ion structure looks like. It is of importance to note that theoretical IR frequencies must be scaled depending on the theoretical model and basis set used for the calculation. An extensive table of IR frequencies scaling factors be found on the NIST website at “<http://cccbdb.nist.gov/vibscalejust.asp>”. The most commonly used model in the ion

spectroscopy field is B3LYP/6-31G(d,p) which requires a scaling factor of 0.961. I have found the Avogadro program very useful for looking at theoretical IR spectra and therefore below is a short introduction on how to do this.

First go to File → Open to open the Gaussian output file of interest. Then go to Extensions → Spectra which opens the following window.



The spectrum can be manipulated to be more useful for the Glish Group by clicking on the “Infrared Spectra Settings” tab. In this tab, change “Y Axis Units” to Absorbance, enter the desired scaling factor, and change the Gaussian width. The Gaussian width will depend on the experimental setup, but typically for ion spectroscopy data used in the Glish Group a value of 20 is appropriate. Below is the Avogadro window with the appropriate manipulations made.



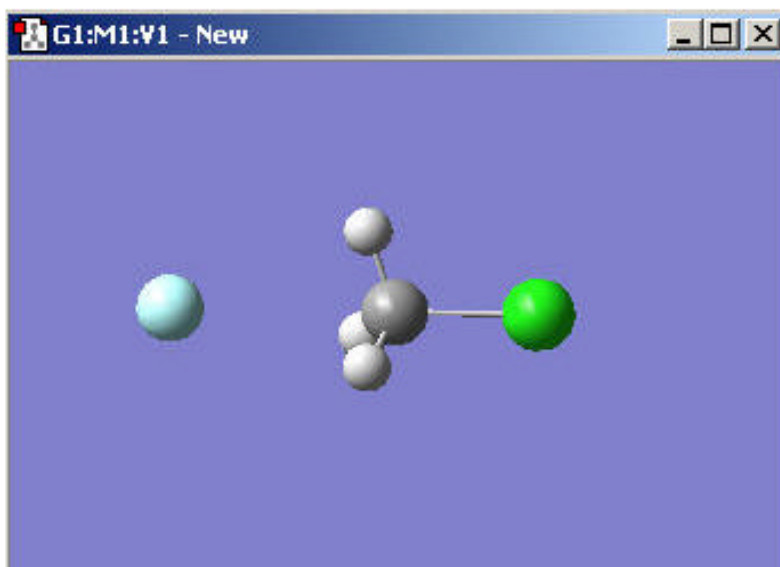
This data can now be exported to be used for comparison to experimental data.

A15.3.5 Transition States

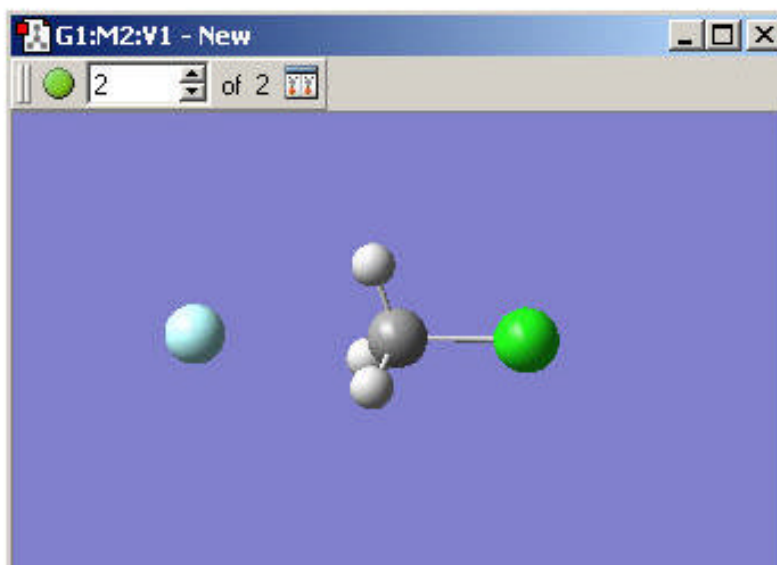
A transition state is a particular configuration along the reaction coordinate. It is defined as the state corresponding to the highest potential energy along the reaction coordinate. Determining an optimized transition state can lead to the determination of activation energy which helps in determining what different products are likely to form. To determine a transition state structure requires a series of calculations. These include geometry optimizations of both reactants and products, optimization of a transition state structure, and then a confirmation of the transition state structure. Calculations to optimize a transition state structure are typically fairly difficult calculations. There are two main methods to optimized transition state structures: QST2 and QST3. A QST2 calculation requires both reactant and product structures, while a QST3 calculation requires an initial

guess to the transition state structure in addition to the reactant and product structures.

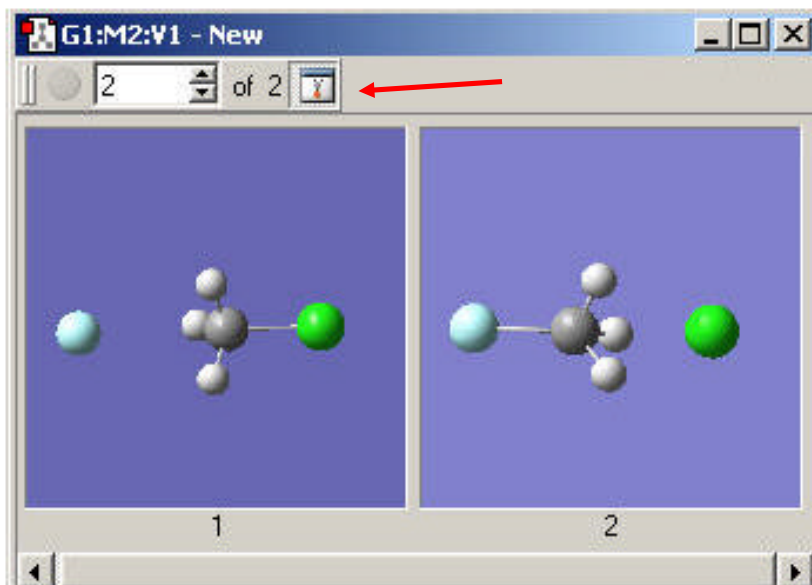
Below is a quick discussion on how to setup a transition state optimization calculation. The example is for a simple S_N2 type reaction. The first step is to make the reactant structure.



Next choose Edit → Copy and then choose Edit → Paste → Add to molecule group which will bring up the following window.



Now this structure has to be manipulated into the product structure. You can then view both the reactant and product structure in one window. This is done by clicking on the button identified by the red arrow in the figure below.



Below is shown how to setup the transition state calculation.

The screenshot shows a window titled "G1:M2:V1 - Gaussian Calculation Setup". The "Title:" field is empty. The "Keywords:" field contains "# opt=qst2 hf/3-21g geom=connectivity". The "Charge/Mult.:" field contains "0 2". Below these fields are several tabs: "Job Type", "Method", "Title", "Link 0", "General", "Guess", "NBO", "PBC", and "Solvation". The "Optimization" tab is selected. Under this tab, there are two dropdown menus: "Optimize to a" set to "TS (QST2)" and "Calculate Force Constants" set to "Never". There are also two checkboxes: "Use GDIIS" and "Use tight convergence criteria", both of which are unchecked. At the bottom, there is an "Additional Keywords:" field and an "Update" button. The bottom of the window features a row of buttons: "Submit...", "Cancel", "Edit...", "Retain", "Defaults", and "Help".

Once a transition state has been determined it has to be confirmed. The first step in this process is to do a frequency calculation. The calculation should show that there is one imaginary frequency. Next an intrinsic reaction coordinate (IRC) calculation is done to further confirm the transition state. The following websites describe the input parameters for an IRC calculation and give an example of how to do an IRC calculation.

- <http://www.cup.uni-muenchen.de/ch/compchem/geom/irc1.html>
- <http://www.cup.uni-muenchen.de/ch/compchem/geom/irc5.html>

APPENDIX 16

SIMULATED ANNEALING IN AMBER

A16.1. Introduction

Simulated annealing is a probabilistic technique that approximates the global optimum of a function. These simulated annealing procedures can be used to find a global minimum structure. While it is common to do geometry optimization calculations in Gaussian the results of these calculations are not guaranteed to give a global minimum often times only the local minimum is found. Figure A16.1 gives a good pictorial overview of this problem. Geometry optimizations done using Gaussian will often get stuck in local minima and therefore it is necessary to do a more thorough search of the energy landscape to be sure to find the global minimum.

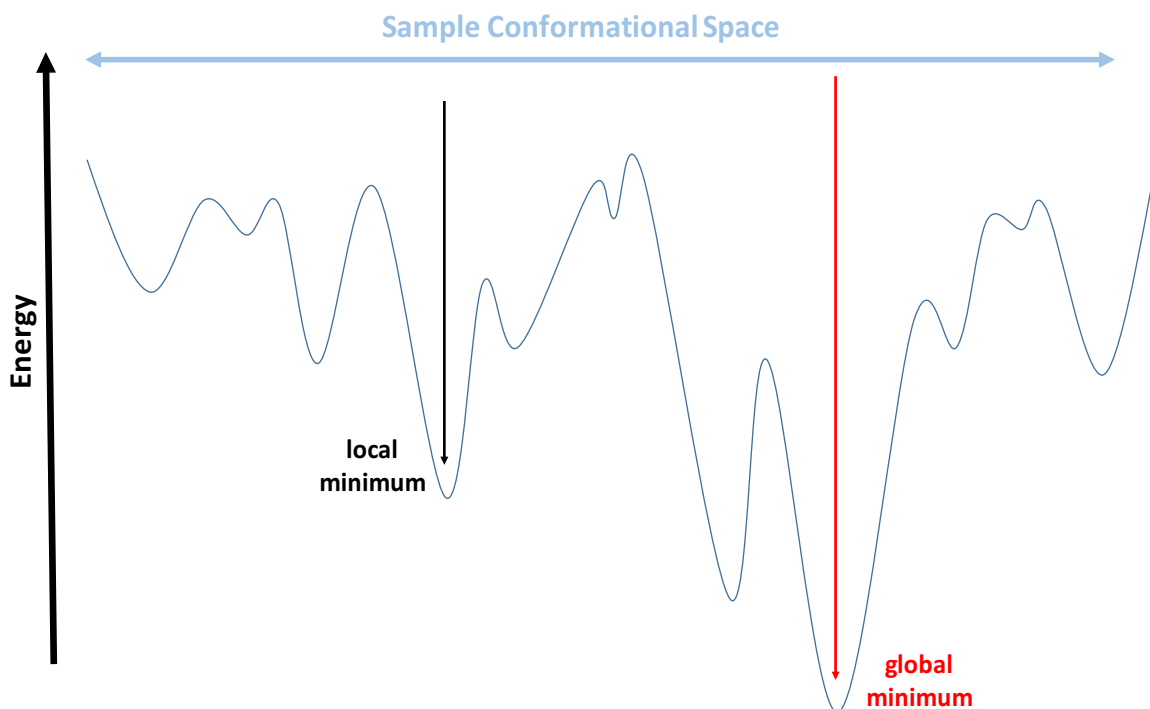


Figure A16.1. Schematic of a potential energy surface

Simulated annealing is a method that is used to be sure that a thorough search of the energy landscape is accomplished. This is achieved by taking a starting structure and subjecting it to dynamics for approximately 30 picoseconds (ps) at a temperature of 800 K. The structure is then cooled to 50 K over a time period of choice. At this point the structure's geometry is optimized by minimizing its energy. This structure is the first candidate structure. The next step is to take this first structure and use it as a input to the same simulated annealing sequence just discussed. A representation of this process is shown in Figure A16.2.

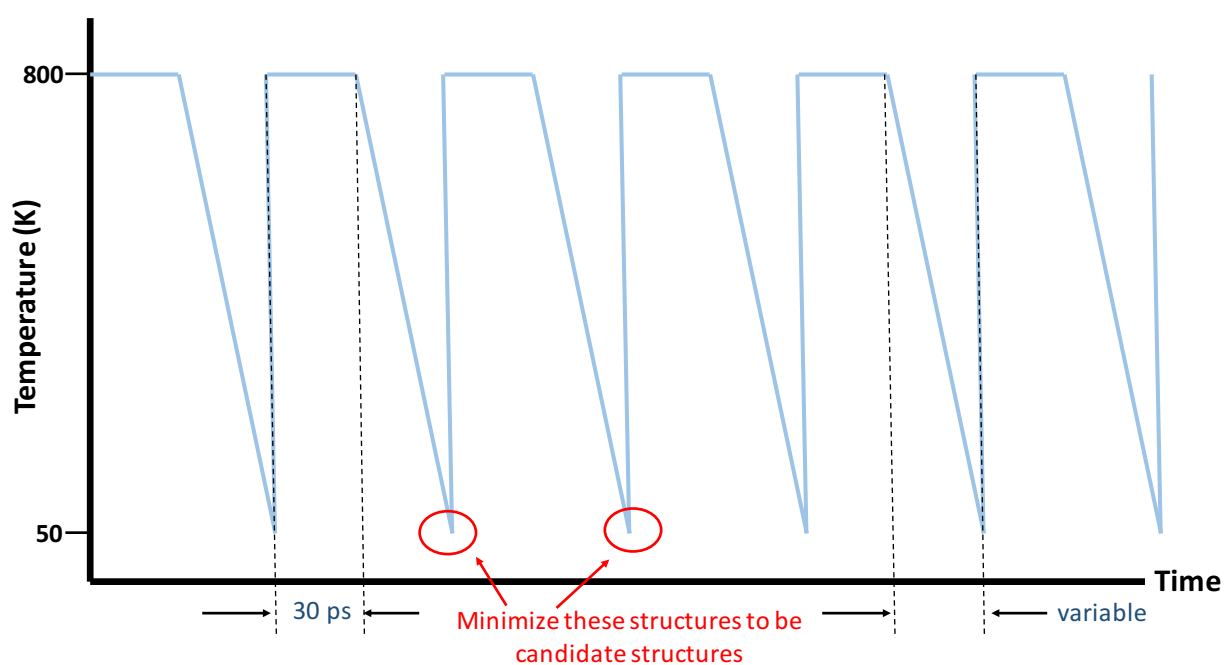


Figure A16.2. Simulated annealing process

The simulated annealing sequence is repeated depending on how many candidate structures are desired. Typical simulated annealing procedures usually generate around 100 candidate structures. Calculations for the simulated annealing process are done using molecular mechanics and can be done in many different programs. The rest of this appendix will cover some basics of molecular mechanics calculations and how to use the AMBER program to do simulated annealing calculations.

A16.2. Molecular Mechanics Basics

Molecular mechanics uses Newtonian mechanics to model molecular systems. The potential energy of the system is calculated using force fields. There are a huge variety of force fields available. A discussion on different available force fields is not included in this work, but the force field that is used in the tutorial section below should serve any purpose for the Glish group. In molecular mechanics calculations, every atom in a structure is considered to be a single particle. Each of these particles then is assigned a radius, polarizability, and net charge. Bonded interactions are treated like springs. As previously mentioned a force field is used to calculate a potential energy and the potential energy is made up of energies from covalent and noncovalent interactions.

$$Energy = E_{covalent} + E_{noncovalent}$$

$$E_{covalent} = E_{bond} + E_{angle} + E_{dihedral}$$

$$E_{noncovalent} = E_{electrostatic} + E_{van\ der\ Waals}$$

These equations are a general description of the form of a force field, but this is dependent on the calculation program that is used. The specifics of the force field used in the AMBER program will be discussed below in addition to a basic tutorial on how use the AMBER program for simulated annealing calculations.

A16.3. Simulated Annealing in AMBER

AMBER stands for Assisted Model Building with Energy Refinement and is a commonly used program for molecular mechanics and dynamics calculations. There are many subprograms inside the AMBER program, but only two of these are of interest to this application. The first of these is LEaP which is used for preparing input files for the simulation programs. The other subprogram of interest is SANDER (Simulated Annealing

with NMR-Derived Energy Restraints) which is the central simulation program and allows for energy minimization and molecular dynamics with a wide range of options.

A16.3.1 Setting up AMBER to run on KillDevil

The AMBER program must be added to the KillDevil environment to be used. To do this a specific compiler has to be added to the environment in addition to the AMBER program. The commands to add the correct compiler and the AMBER program is shown below:

```
ipm add mvapich_intel
```

```
ipm add amber
```

The above commands add the compiler and program to the environment just for the current session. To add these things to the environment every time you log onto KillDevil use the following commands instead.

```
ipm initadd mvapich_intel
```

```
ipm initadd amber
```

Next you have to make sure that the compiler and program was added to the environment correctly. Below is the command to check that they are added correctly. Make sure that you see “amber” and “mvapich_intel” listed when the following command is executed.

```
module list
```

Now the AMBER program is ready to be used to prepare input files and run calculations.

A16.3.2 LEaP

The LEaP subprogram is used to prepare input files for AMBER calculations. There are two forms of the LEap program: *xleap* and *tLeap*. The only difference between these two is that *xleap* has a graphical user interface. I suggest using *xleap* since it makes it easier to

check that your structure of interest is loaded in correctly. First, start the *xleap* program on the KillDevil server using the following command.

```
bsub -Ip xleap
```

Now that the *xleap* subprogram is running, the first step is to add the force field you want to use. This is done by using the following command.

```
source leaprc.ff99
```

Next load the PDB of the structure of interest into *xleap* using the following command.

```
name = loadpdb file.pdb
```

The parameter “name” will be whatever you want to name the structure in *xleap* and “file.pdb” will be the filename of whatever structure you are loading into *xleap*. Now if you are using *xleap* you can check to make sure that the PDB file for the structure was loaded in correctly. The command to do this is as follows.

```
edit name
```

Once you have checked that the structure loaded in correctly the next step is to generate the needed topology and coordinate files to use for calculations. The command to do this is as follows where “name” will be whatever you chose to name your structure above.

```
saveamberparm name name.prmtop name.inpcrd
```

The needed input files are now prepared to be used for calculations and now the SANDER program can be setup to run calculations.

AX3.3 SANDER

The first step in using SANDER is to prepare files to run minimization and simulated annealing calculations. These two scripts are called *min.in* and *sim_anneal.in* and are shown in Figures A16.3 and A16.4.

```

1 Minimization to 0K
2 &cntrl
3     imin=1, maxcyc=1000, ncyc=500,
4     cut=999., rgbmax=999., igb=1, ntb=0,
5     ntp=100,
6 /
7 ~

```

Figure A16.3. AMBER Minimization script

```

1 15 ps simulated annealing protocol
2 &cntrl
3     nstlim=50000, dt=0.001,
4     scee=1.2, cut=999.0, rgbmax=999.0,
5     ntp=500, ntw=500, ntwe=0, ntwx=500, pencut=0.1,
6     ipnity=1, ntc=2, ntf=2, ntt=3,
7     vlimit=10, ntb=0, imin=0,
8     nmropt=1, igb=6, gamma_ln=1, saltcon=0.,
9 /
10 &ewald
11     eedmeth=5,
12 /
13 #
14 #Simple simulated annealing program
15 #
16 #from steps 0 to 1000: raise target temperature 0 --> 800K
17 #from steps 1000 to 3000: leave at 800K
18 #from steps 3000 to 15000: re-cool to low temperature of 50K
19 #
20     &wt type'TEMPO', istep1=0, istep2=10000, value1=0.,
21         value2=800., /
22     &wt type'TEMPO', istep1=10001, istep2=40000, value1=800.,
23         value2=800., /
24     &wt type'TEMPO', istep1=40001, istep2=50000, value1=800.,
25         value2=50., /
26     &wt type='END' /
27 ~

```

Figure A16.4. AMBER Simulated annealing script

To run SANDER you have to create a LSF run script. This is a file that tells the KillDevil server what calculations to run. I wrote a Perl script named *sim_anneal_setup.pl* that writes this run script for you. To use this Perl script, in a command shell moved to the directory

where the program is located. First use the following command to allow you to execute the program.

```
chmod a+x sim_anneal_setup.pl
```

Now you can run the script that writes the LSF run script. The command to do this is as follows:

```
./sim_anneal_setup.pl name number
```

Where “name” is what you name the AMBER input files in LEaP and “number” is the number of rounds of simulated annealing that you want to do. This will give you an load sharing facility (LSF) run script that is named “amber.KillDevil.lsf” and an excerpt of this can be found in Figure A16.5.

```
1  #!/bin/csh -f
2  #BSUB -J Amber
3  #BSUB -q week
4  #BSUB -n 4
5  #BSUB -o log.%J
6  #BSUB -a mpichp4
7  set output = Amber.out
8
9  mpirun.lsf sander -O -i min.in -p name.prmtop -c name.inpcrd -r min0.rst -o min0.out -x
... min0.mdcrd
10 mpirun.lsf sander -O -i sim_anneal.in -p name.prmtop -c min0.rst -r sim_anneal1.rst -o
... sim_anneal1.out -x sim_anneal1.mdcrd
11 mpirun.lsf sander -O -i min.in -p name.prmtop -c sim_anneal1.rst -r min1.rst -o min1.out
... -x min1.mdcrd
12
13 mpirun.lsf sander -O -i sim_anneal.in -p name.prmtop -c min1.rst -r sim_anneal2.rst -o
... sim_anneal2.out -x sim_anneal2.mdcrd
14 mpirun.lsf sander -O -i min.in -p name.prmtop -c sim_anneal2.rst -r min2.rst -o min2.out
... -x min2.mdcrd
15
16 mpirun.lsf sander -O -i sim_anneal.in -p name.prmtop -c min2.rst -r sim_anneal3.rst -o
... sim_anneal3.out -x sim_anneal3.mdcrd
17 mpirun.lsf sander -O -i min.in -p name.prmtop -c sim_anneal3.rst -r min3.rst -o min3.out
... -x min3.mdcrd
18
19 mpirun.lsf sander -O -i sim_anneal.in -p name.prmtop -c min3.rst -r sim_anneal4.rst -o
... sim_anneal4.out -x sim_anneal4.mdcrd
20 mpirun.lsf sander -O -i min.in -p name.prmtop -c sim_anneal4.rst -r min4.rst -o min4.out
... -x min4.mdcrd
```

Figure A16.5. LSF run script

Now access KillDevil and change the directory to your netscr directory, “/netscr/onyen”.

Then use the following command to execute the LSF runscript and do the simulated annealing calculations.

```
bsub < amber.KillDevil.lsf
```

Running this script will give you an output of the number of minimized structures that corresponds to the number of times you chose to go through the simulated annealing process. For example, if you chose to have 100 simulated annealing steps then you will have energy minimized structures named min1.rst to min100.rst. You can then convert these coordinate files to a more useful format like a PDB file using the following command on the KillDevil server.

```
ambpdb -p name.prmtop < min1.rst > min1.pdb -aatm
```

APPENDIX 17

THEORETICAL COLLISION CROSS SECTION CALCULATIONS

A17.1 Introduction

Experimental collision cross sections are determined using ion mobility experiments and are dependent on ion shape. These experimental cross sections are due to collisions between ions and buffer gas. Due to this scattering process it is unsure how geometry coordinates from theoretical structures can be converted into collision cross sections. Early studies on ion mobility theory showed that collision cross sections are related to the momentum transfer collision integrals. These collision integrals are very difficult to calculate and therefore need to be approximated.

For ions that are of interest to the Glish lab there are two methods to approximate collision integrals. The first method is known as the projection approximation method and is useful for ions with less than 200 atoms. A projection approximation can basically be thought of as the shadow of an ion's structure. An advantage of projection approximation calculations is that they are computationally inexpensive. For ions containing 200 to 1000 atoms the preferred approximation to determine theoretical cross sections uses a trajectory calculation of 12-6-4 potential. The downside of trajectory calculations are that they are computationally expensive.

There are two programs that are commonly used to calculate theoretical collision cross sections. The first program uses a projection approximation calculation and was developed by the Bowers lab and is called Sigma. Mobcal, developed by the Jarrold lab, is the other commonly used program and uses trajectory calculations. The rest of this appendix will focus on how to use these two programs to determine theoretical collision cross sections.

A17.2 Sigma

The sigma program is obtained by contacting the Bowers Lab. The person to contact to get the software is Thomas Wytttenbach. His email address is wyttenbach@chem.ucsb.edu. The software is also in the Glish Lab groupspace. The software is in a zipped folder titled “sigma”. In the folder are seven files. The source file, a README file, a coordinate file for C₆₀ (fullerene), and four parameter files. The program is run in a Unix environment. The first step is to compile the source code. You need to use the GNU compiler to compile the source code correctly. The GNU compiler can be obtained by going to the following website: <https://gcc.gnu.org/>. The command to compile the source code is shown below.

```
g77 -O -o sigma_aug02.x sigma_aug02.f
```

The next step is to make sure that everything compiled correctly. To do this you use the files included in the sigma folder to calculate the collisional cross section for C₆₀. The command to do this step is found below.

```
sigma_aug02.x -i C60.mol2 -p parameters.dat
```

The result from this calculation should have a value of approximately 65.1717. Now the sigma program can be used to calculate collisional cross sections for any structure you want. The structure does need to be in PDB format. Below is an example command to calculate the collisional cross section for a 10 residue peptide. Replace “min3.pdb” with the name of your structure file.

```
sigma_aug02.x -i min3.pdb -f pdb -h sca.par
```

The sigma program is really easy to use and gives results very fast. One thing to keep in mind is that the program is written to be used in cases where helium is the buffer gas. So if

the experimental data was collected using a different buffer gas sigma will not give useful results.

A17.3 Mobcal

There is a copy of the mobcal program in Glish Lab group space. It is located in a folder called “Mobcal”. This folder contains six files: the source code for mobcal using helium, source code for mobcal using nitrogen, a parameter file, an input file for choline, an output file for the calculation done in helium, and an output file for the calculation done in nitrogen. This program is again used in a Unix environment. The first step is to compile the source code using the GNU compiler. The command below is the general format to use to compile either source code (helium or nitrogen).

```
g77 -o <file name of the compiled code> <file name of the source code>
```

So to compile the nitrogen source code the command would be the following.

```
g77 -o mn mobcal.N2.f
```

To compile the helium source code the command would be the following.

```
g77 -o mhe mobcal.He.f
```

To run the now executable code, use the command below (use mn or mhe for nitrogen and helium respectively).

```
./mn
```

This will pull information from the “mobcal.in” file to use in the calculation. This file contains three lines: the name of the input file, the name for the output file, and a random number seed. Use the “mobcal.in” file included in the folder in group space to test to see if the program is working correctly. The “Mobcal” folder in group space has output files for the calculations done in both helium and nitrogen gas. To use the program for any structure that

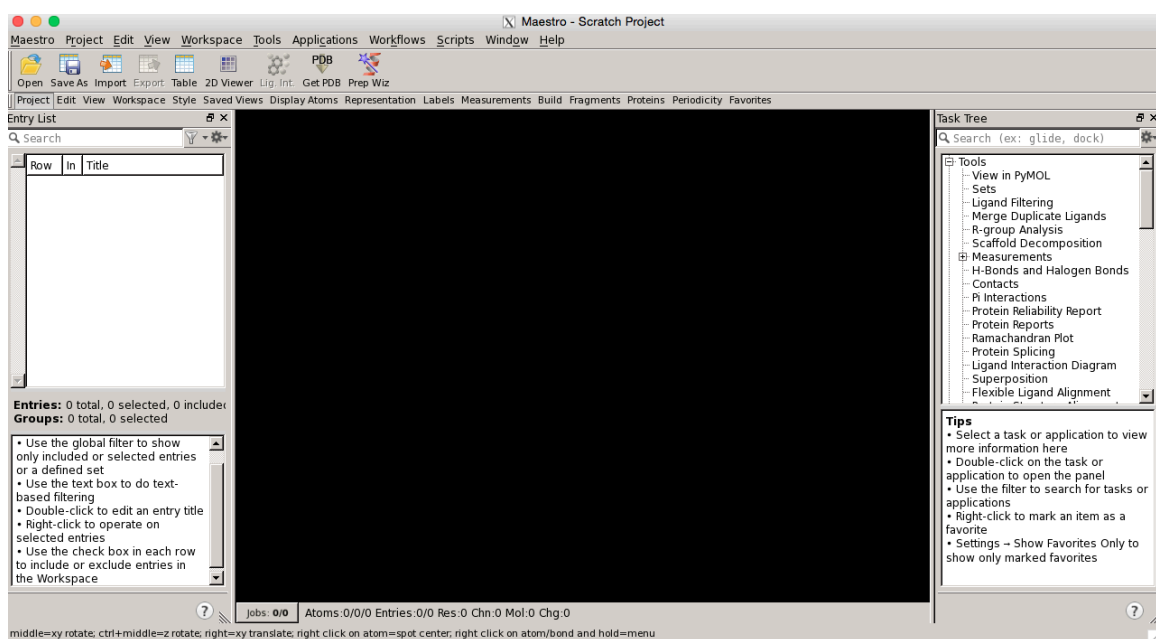
you want you just need to put the structure file in the same folder as the mobcal program and change the “mobcal.in” file to have the desired names of input file and output file.

The mobcal program uses the trajectory method and because of this the calculation is computationally expensive. So be prepared to wait a while to get a result out from the calculation. There is a major advantage to using the mobcal program and that is the fact that the calculation of collisional cross section can be done in nitrogen gas. This is particularly important for the Glish Lab for future possibility of calculating collisional cross sections to compare to experimental collisional cross sections determined using the ion mobility device from Excellims.

APPENDIX 18

TORSIONAL SAMPLING TUTORIAL

1. Download Avogadro at: http://avogadro.cc/wiki/Main_Page
2. Use Avogadro to draw peptide of interest
 - a. Go Build -> Insert -> Peptide...
 - b. Choose sequence of peptide from N-terminus to C-terminus
 - c. Choose how N- and C- terminus are terminated, most likely for these peptides NH_3^+ for N-terminus and CO_2H for C-terminus to give a +1 charge overall
 - d. Select Insert Peptide
 - e. Go File -> Save as... (save as filetype PDB(*.pdb))
3. Now move PDB file from computer you are working on to KillDevil in your scratch space (/netscr/onyen)
4. Log into KillDevil using a suitable X-terminal (X-windows or X11) by entering:
ssh -Y onyen@killdevil.unc.edu
5. This step only needs to be done the first time you do this: in the X-terminal enter:
module add schrodinger
enter: module initadd schrodinger
enter: module list (to make sure schrodinger added correctly to your environment)
6. In X-terminal
enter: cd /netscr/onyen
enter: mkdir -p .schrodinger/tmp (this only needs to be done first time you do this.)
7. In xterm enter: bsub -q day -Ip maestro
Which should open up a window that looks like this:

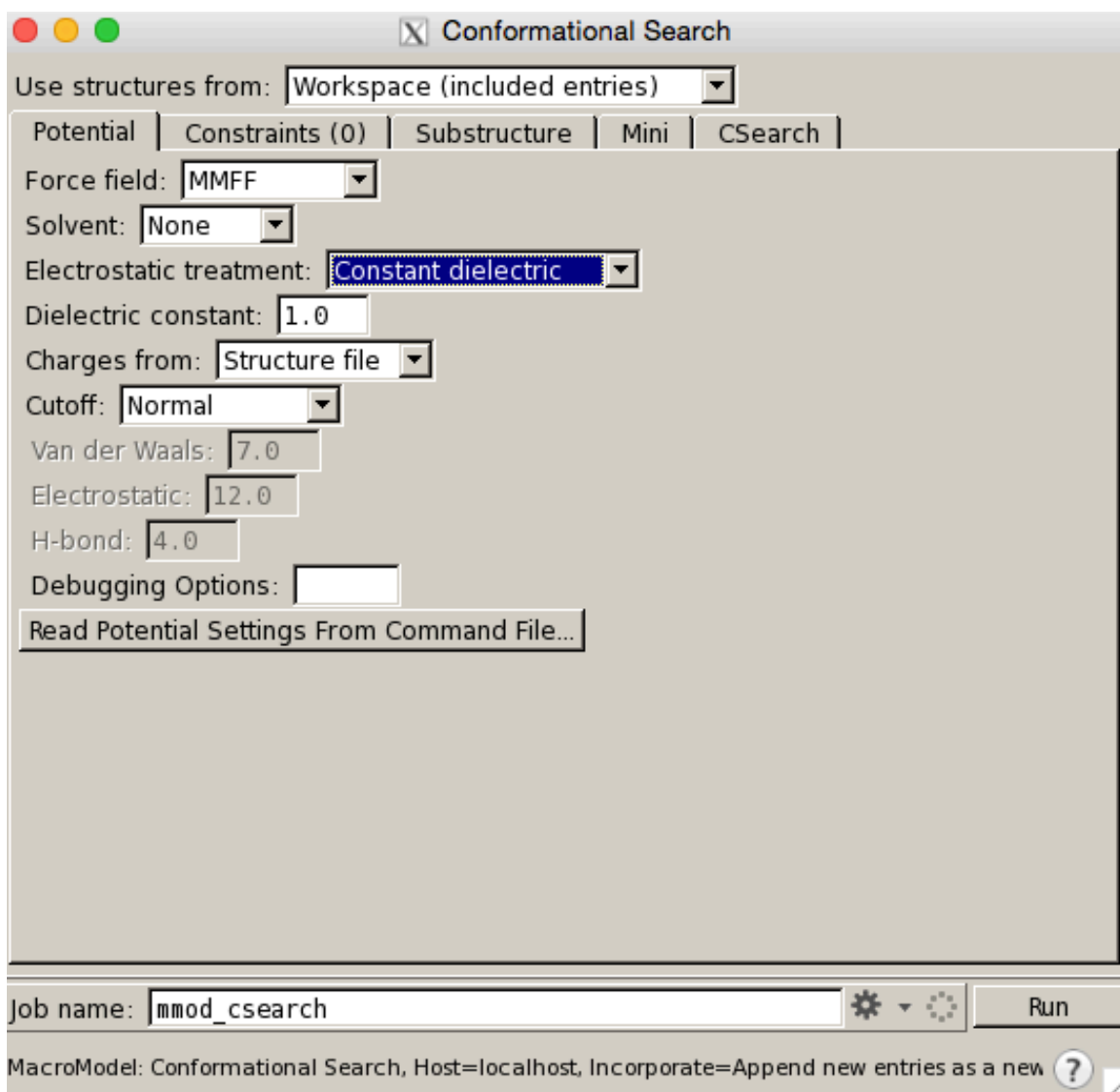


8. Select Import and choose the PDB file that you want to run conformational search on

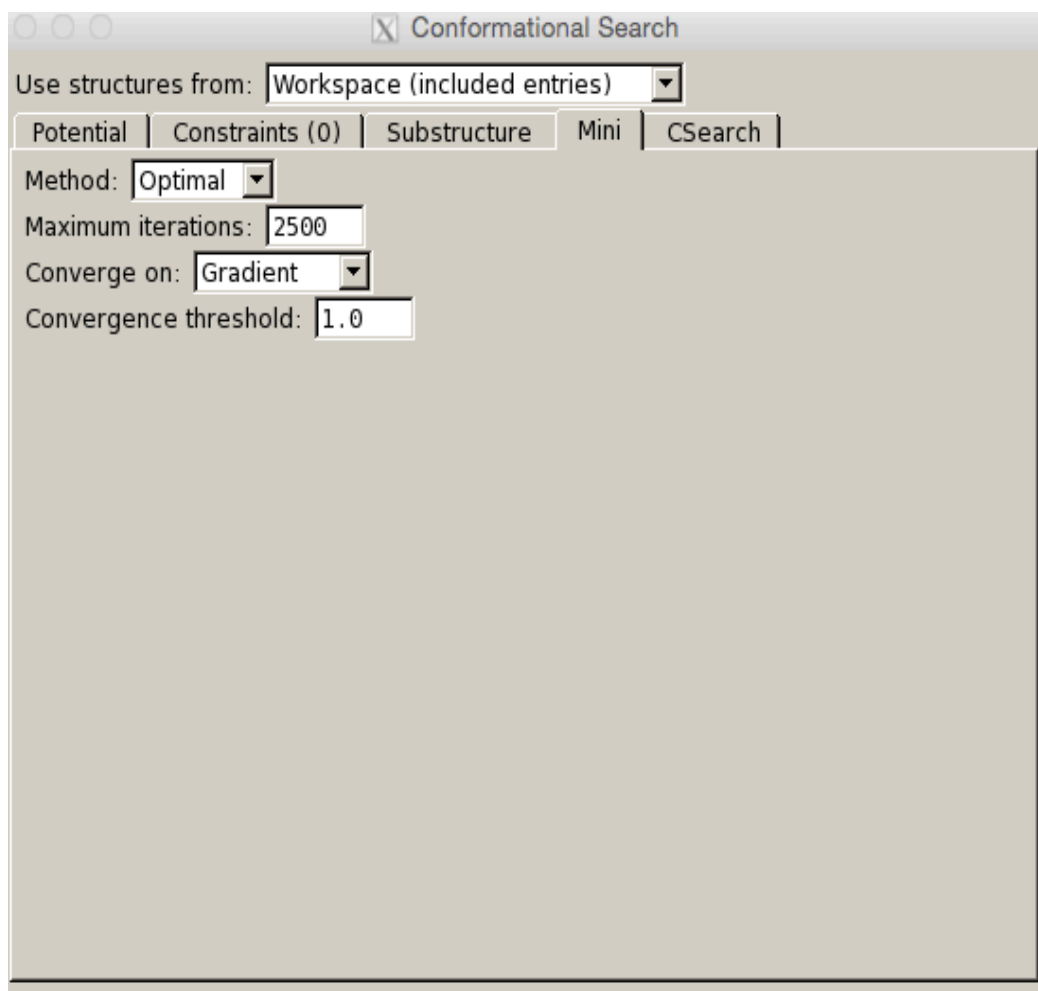
9. Go to Applications -> Macromodel -> Conformation Search

10. In the Potential Tab :

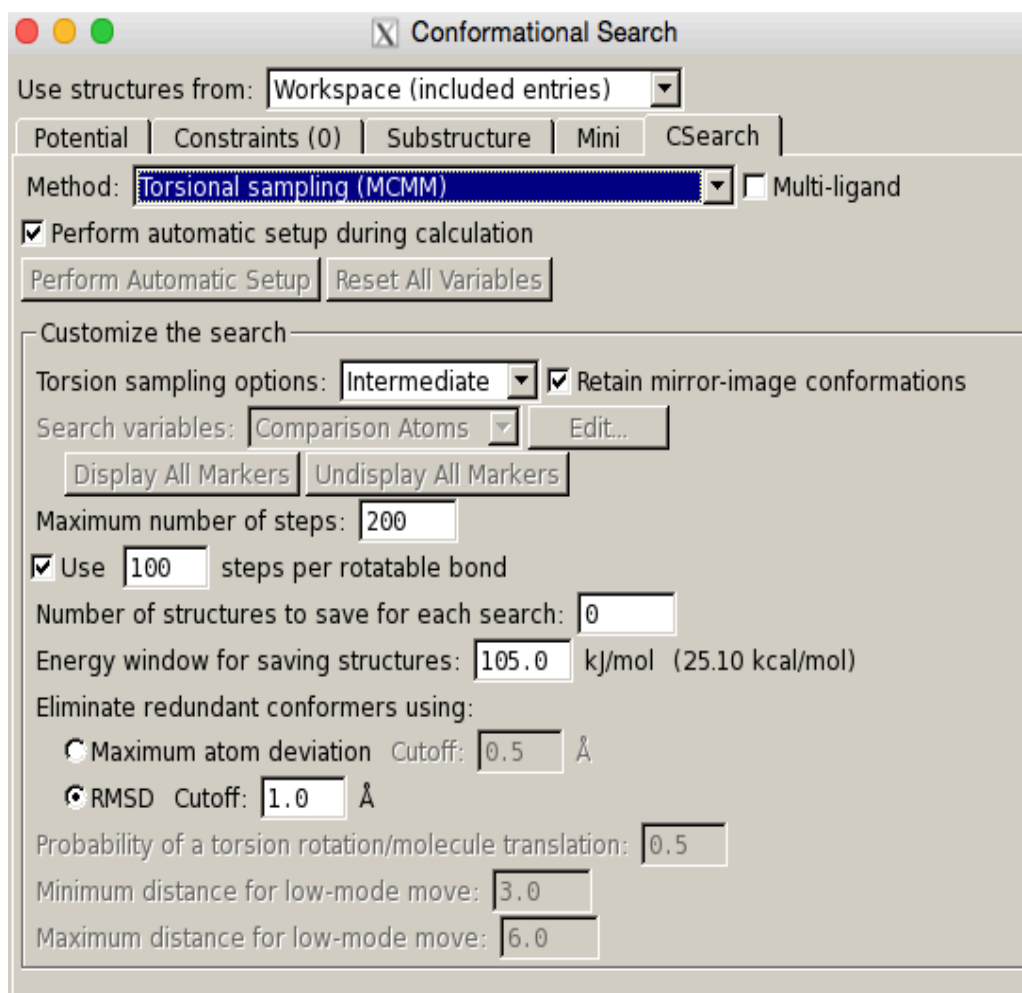
- Force Field → MMFF
- Solvent → None
- Charges from → Structure file



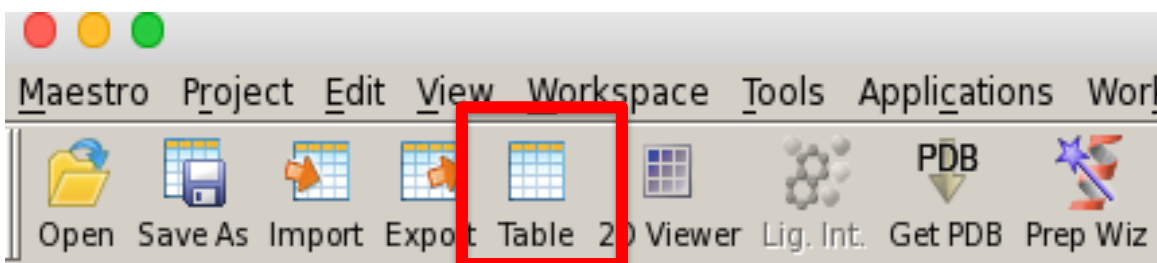
11. In the Mini Tab:



12. In the CSearch Tab:



13. Make sure to fill in job name with whatever you want to name your file.
14. Click Run
15. You can monitor your job by going to Applications → Monitor Jobs...
16. After job is finished go to Scripts → Project Table → Convert MacroModel Energies to kcal/mol
17. Click on table in main window to open up a table with conformation number and energies



18. You can export this table of information by going to Table → Export → Spreadsheet
19. You can then export all of the generated conformations by going to Table → Export → Structures... give a file name and setup as seen below.

