

Kinetic Analysis and Modelling in Heterogeneous
Catalysis

Cardiff University



Christian Reece

August 15, 2017

Abstract

A combination of Temporal Analysis of Products, Temperature Programmed Reduction, and Density Functional theory techniques have been used to perform kinetic analysis on data from heterogeneous catalysis experiments. A new method of data filtering has been developed for Temporal Analysis of Products, and has been applied to a system of 4 Pt–Mo₂C, and the current methodology has been expanded upon to calculate rate coefficients for the oxidation of CO to CO₂ via the Boudard reaction. From the kinetic constants it appears that a phase change occurs in the material at approximately 200°C.

The current theory for analysing Temperature Programmed Reduction has been applied in a new methodology which is able to perform the deconvolution of thermograms with high accuracy, while also calculating the kinetic parameters related to the reduction processes. This new methodology has been applied to a system of CeO₂ calcined at 400, 500 and 600°C and the strengths and limitations of the methodology are explored. From the deconvolution procedure it was found that there are three distinct reduction processes occurring on the CeO₂ and that a phase change occurs between 400 and 500°C.

Finally Density Functional Theory combined with classical dynamics has been used to explore the mechanism of the hydrogenation of Levulinic Acid to gamma-Valerolactone over a CuZrO₂ catalyst. It was found that the Levulinic Acid is more likely to hydrogenate then cyclise, and from using molecular dynamics simulations it was shown that the solvent H₂O plays a very important role in the cyclisation of

the hydrogenated intermediate.

List of Publications

Chapter 3

- Gas phase stabiliser-free production of hydrogen peroxide using supported gold–palladium catalysts. *Chemical Science* 7, no. 9 (2016): 5833-5837.
- Laplace Domain Filtering of TAP Data *In Preparation*

Chapter 4

- A Numerical Approach to the Extraction of Physical Data from Temperature Programmed Reduction Experiments. *In preparation*

Chapter 5

- Identification of the catalytically active component of Cu–Zr–O catalyst for the hydrogenation of levulinic acid to γ -valerolactone. *Green Chemistry* 19, no. 1 (2017): 225-236.
- The conversion of levulinic acid into γ -valerolactone using Cu–ZrO₂ catalysts. *Catalysis Science & Technology* 6, no. 15 (2016): 6022-6030.

- The effect of the addition of Ni to Cu-ZrO₂ catalysts for the hydrogenation of levulinic acid *Applied Catalysis A: General Under Review*

Papers not explicitly mentioned in this thesis.

- Structure Sensitivity in Catalytic Hydrogenation at Platinum Surfaces Measured by Shell-Isolated Nanoparticle Enhanced Raman Spectroscopy (SHINERS). *ACS Catalysis* 6, no. 3 (2016): 1822-1832.
- Modelling analysis of the structure and porosity of covalent triazine-based frameworks. *Physical Chemistry Chemical Physics* 17, no. 2 (2015): 817-823.

Acknowledgements

First I would like to acknowledge my supervisor Dr. David J. Willock for all of the help and guidance he has given me over the course of my PhD program. Thanks to him I have been able to develop the skills required of me to excel at my research, and without his guidance none of this work would be possible. I would also like to mention my mentor Professor Peter J. Knowles for all of the useful feedback he has provided throughout the various stages of my PhD.

I would also like to express my gratitude to Professor Graham Hutchings, Dr. Ewa Nowicka, Dr. Sarwat Iqbal, Dr. Jonathan Bartley, Professor Stan Golunski, Daniel Jones and all of the members of the Cardiff Catalysis Institute for their help working as part of the NOVACAM and After the Gold Rush projects. I would also like to thank Dr. Satoshi Isikawa for the valuable discussion and support as part of the NOVACAM project.

I would like to thank the larger members of the Temporal Analysis of Products community, Dr. Evgeniy Redekop, Dr. Lucun Wang, and the Belfast group for useful discussions on theory and experimental technique. In particular I would like to thank Dr. Rebecca Fushimi for providing the gateway to accessing the larger community and for helping me to launch my career in TAP.

I wish to also acknowledge all of my colleagues in the Willock group at Cardiff from the last three years: Dr Soon Wen Hoh, Dr Adam Thetford, Dr. Liam Thomas, Dr. Carlo Buono, and Leticia Ruiz. Also I would like to acknowledge all of the other

members of the computational group 55 Park Place, Matthew Turner, Dr. Shaun Mutter, Dr Jamie A. Platts and Joshua Black.

I would like to thank HPC Wales, ARCCA, and ARCHER (formerly HECToR) for providing the computing time required for the computational simulations.

Finally I would like to thank all of my friends and family for supporting me throughout the last few years.

Contents

	Page
1 Introduction	1
1.1 Temperature Programmed Reduction	4
1.2 Temporal Analysis of Products	9
1.3 Hydrogenation of Levulinic Acid	18
1.3.1 Computational studies of biomass derived molecules	21
1.3.2 Modelling the LA system	25
2 Methodology	33
2.1 MATLAB Programming Environment	33
2.1.1 Generation of a Graphical User Interface in MATLAB	36
2.1.2 Regression Analysis in MATLAB	37
2.2 Temperature Programmed Reactions	39
2.2.1 Basics of Temperature Programmed Experiments	40
2.2.2 The Kinetic Model	42

2.2.3	The Arrhenius Parameters	44
2.2.4	The Kinetic Constants	51
2.3	Temporal Analysis of Products	56
2.3.1	Basics of the TAP experiment	57
2.3.2	Modelling the standard diffusion curve	58
2.3.3	Confirming Knudsen Diffusion	62
2.3.4	Modelling the Catalyst Zone	64
2.3.5	Understanding the Kinetic Equation	66
2.3.6	Thin Zone Reactor Model	68
2.3.7	Exit Flow as Moments	71
2.3.8	The Basic Kinetic Coefficients	74
2.4	Density Functional Theory	78
2.4.1	Understanding Ab-Initio codes	78
2.4.2	Moving on from Hartree Fock	81
2.4.3	Kohn and Sham and the Origins of DFT	82
2.4.4	The Exchange Correlation Energy	84
2.4.5	Calculating the Total Energy of a System	85
2.4.6	Vienna Ab-initio Simulation Package	86
2.4.7	Long Range Dispersion Corrections	91
2.4.8	Bulk and Surface Calculations	93

2.4.9	Transition State Searching	95
2.5	Molecular Dynamics	97
2.5.1	Describing Atomistic Interactions	98
2.5.2	Solving as a Function of Time	100
2.5.3	Accounting for Temperature and Pressure	102
2.5.4	DL_POLY Simulation Package	103
2.5.5	Ab Initio Molecular Dynamics	103
3	Development and Application of TAP Numerical Analysis	110
3.1	Development of Analysis Software	110
3.1.1	Importing User Data	111
3.1.2	Processing the Users Data	113
3.1.3	Developing the Front-End GUI	115
3.2	Application of TAP Data Analysis	116
3.2.1	Analysis of Experimental Response	116
3.2.2	Baseline Prediction and Tail Expansion	118
3.2.3	Correcting for Noise	121
3.2.4	Laplace Fourier Filtering	123
3.2.5	Confirmation of the LFF Regression Technique	130
3.2.6	Moment Analysis and Calculation of Kinetics	133

3.2.7	Kinetic Analysis of Experimental Data	138
3.3	Summary	142
4	Development and Application of TPR Numerical Analysis	145
4.1	Application of TPR Data Analysis	145
4.2	Baseline Correction	145
4.3	Deconvolution of TPR thermograms	147
4.4	Calculation of Kinetic Parameters	151
4.5	Further Analysis	155
4.6	Application to Real Data	159
4.7	Understanding the Reduction Process	168
4.8	Summary	171
5	Computer Simulation Results	173
5.1	Hydrogenation of Levulinic Acid	173
5.2	Understanding the Catalyst Material	174
5.3	The Reaction Mechanism	182
5.4	Inclusion of Copper	189
5.5	Calculation of Energy Barriers	196
5.6	Dynamic Transition State Sampling	201
5.7	Summary	205

6	General Conclusions	210
6.1	Future Work	215
7	Appendix	217

List of Figures

	Page
1 Introduction	1
1.1 Example of catalytic mechanisms a) Scheme for catalytic hydrogenation of ethene b) Energy profile for uncatalysed and catalysed process	1
1.2 Number of citations regarding catalytic mechanisms	2
Temperature Programmed Reduction	4
1.3 Basic schematic of a TPR experiment.	5
1.4 Thermogram for the reduction of CeO_2 as a function of temperature at three heating rates: 5 K min^{-1} , 10 K min^{-1} , 15 K min^{-1}	6
1.5 Parameters calculated from regression algorithm taken from [16] . . .	7
1.6 The <i>CCI-TPR</i> main graphical user interface with overlaying activation energy subplot	8
Temporal Analysis of Products	9
1.7 Comparison between MBS and TAP experiments[18]	9

1.8	Locations of TAP reactor systems	10
1.9	TAP reactor schematic a) TAP-3 reactor at Idaho National Laboratory b) Schematic of a TAP-3 system	10
1.10	Cutaway of the reactor and the pulse valve manifold.	11
1.11	Standard TAP experimental responses a) Pulse intensity v time for a TAP pulse b) Exit flow v time for a TAP pulse	12
1.12	Visualisation of the principle behind <i>Chemical Calculus</i> a) Short term pulse response v time b) Long term pulse response v time	12
1.13	Knudsen exit flow comparisons a) Simulated v experimental Knudsen exit flow b) Knudsen v adsorption/desorption exit flow	14
1.14	Pulse by pulse conversion of CO v total CO conversion	16
1.15	Apparent transformation constant for CO v total CO conversion . . .	17
1.16	CO oxidation experiment repeated at three temperatures, with over- laying activation energies	17
	Hydrogenation of Levulinic Acid	18
1.17	Periodic table with " <i>unsustainable</i> " elements highlighted in red. . . .	19
1.18	Skeletal structures for Levulinic Acid and γ -Valerolactone	19
1.19	Table for conversion of LA and ML to gVL in both methanol and water taken from [53]	20
1.20	Catalytic hydrogenation of LA in presence of water and methanol taken from [53]	21

1.21	Proposed reaction pathways for reaction of neutral glycerol taken from [54]	22
1.22	Potential energy plot for the reaction of neutral glycerol taken from [54]	22
1.23	Skeletal structures of α - (left) and β - (right) D-glucopyranose	23
1.24	Microsolvation of α - (left) and β - (right) D-glucopyranose, taken from [57]	24
1.25	Reaction paths in eV for the hydrogenation of acetone on Ru(0001) in the absence (left) and in the presence of a water molecule (right). Alkyl path is the dashed line, and alkoxy path is the solid like. Taken from [58]	25
1.26	Two different methods of calculating the δE taken from [58]	26
1.27	Energetic spans for various metals for the hydrogenation of acetone taken from [58]	27
2	Methodology	33
	MATLAB Programming Environment	33
2.1	The GUIDE layout editor	37
2.2	Flow of variables in CCI-TPR and CCI-TAP	37
	Temperature Programmed Reactions	39
2.3	Hypothetical reduction mechanisms, taken from [8]	42
2.4	Master plots for kinetic models in table 2.2 a) Master plots for the $y(\alpha)$ function b) Master plots for the $z(\alpha)$ function	53

2.5	Flow chart for assigning a kinetic model with greater than and less than signs swapped for the RO(n model)	54
	Temporal Analysis of Products	56
2.6	Knudsen exit flow comparisons a) Standard dimensionless diffusion curve b) Standard dimensional diffusion curve	62
2.7	Integrands of the various moments for simulated Knudsen exit flow .	72
2.8	Simulated exit flow for a reversible adsorption/reaction mechanism using the kinetic model and 3 coefficient rate terms	75
2.9	Reversible Adsorption and Reversible reaction mechanisms with rate constants	77
	Density Functional Theory	78
2.10	Simplified Hartree-Fock method taken from [36]	80
2.11	LDA and GGA approximations using a simplified 2D model a) Example of LDA approximation b) Example of GGA approximation . .	85
2.12	Flow chart of DFT optimisation in VASP	87
2.13	Schematic showing how to define a unit cell simplified to two dimensions	88
2.14	Treatment of solids using Periodic Boundary Conditions [49]	89
2.15	Convergence in plane wave energies for a Zr_2O_4 unit cell	90
2.16	Convergence in k -point cutoffs for a Zr_2O_4 unit cell	91
2.17	Murnaghan fit for t- ZrO_2 unit cell	94
2.18	Interpolated reaction for conversion of LA to GVL	95

2.19	Graphical representation of NEB method a) Contour plot of reaction pathway in NEB calculation b) Optimisation of energy in NEB calculation	96
2.20	Graphical representation of Umbrella Sampling method a) Energy with added potential well (well is exaggerated for clarity) b) Umbrella Sampling v NEB calculation	97
	Molecular Dynamics	97
2.21	Cutoff potentials use in MD simulations a) Graphic showing cutoff in 2D periodic image b) Lennard-Jones nonbonding interaction potential with cutoff	100
3	MATLAB + Temporal Analysis of Products	110
	Development of Analysis Software	110
3.1	Example plots of raw data sets a) Raw TAP experimental data b) Raw TPR experimental data	113
	Application of TAP Data Analysis	116
3.2	Pulsing of 2% H ₂ in air over AuPd–TiO ₂ at 60°C. Red line – mass spec signal at M/Z 2 (H ₂), black line – mass spec signal at M/Z 34 (H ₂ O ₂), blue line – "baseline" mass spec signal taken at M/Z 99.	118
3.3	Selection of baseline cutoff using CCI-TAP	119
3.4	Tail expansion using double exponential function	121
3.5	Comparison of second moment for experimental response and Savitzky-Golay smoothed response	122

3.6	Comparison of second moment for experimental response and LFF method	123
3.7	Comparison of time and frequency domains for an example TAP exit flow curve a) An example of exit flow intensity v time for a TAP experiment b) FFT of example exit flow curve, with real and imaginary components plotted as intensity v frequency	125
3.8	Intensity v Frequency for TAP experimental response with <i>physical</i> noise removed	126
3.9	Second order polynomial v explicit rate model calculated numerically in frequency domain a) Three coefficient model from Taylor Expansion b) Explicit kinetic model for adsorption/desorption/reaction . . .	127
3.10	LFF four coefficient v explicit rate model calculated numerically in frequency domain a) Generalised kinetic model used in LFF method b) Explicit kinetic model for adsorption/desorption/reaction	128
3.11	Intensity v Frequency for TAP experimental and simulated responses	129
3.12	Intensity v time for TAP experimental and baseline-shifted simulated responses	129
3.13	Residuals v time for the TAP experimental and simulated response .	130
3.14	Inert, reactant and product curves simulated using TZTR exit flow equations, and a reversible adsorption/reaction mechanism a) Original simulated curves b) Original curves with added noise	131
3.15	Simulated curves + noise compared with the curve calculated from the LFF regression technique a) Simulated reactant b) Simulated product	132

3.16	Simulated curves + noise compared with the curve calculated from the SG smoothing a) Simulated reactant b) Simulated product	132
3.17	Zeroth moments taken from experimental data for 4 Pt–Mo ₂ C a) Calculated M_0 from experimental data b) Calculated M_0 normalised to inlet (inert) pulse size.	140
3.18	Calculated basic kinetic coefficients for the reactants on the original and the LFF corrected data. a) Calculated r_0 coefficients b) Calculated r_1 coefficients c) Calculated r_2 coefficients	140
3.19	Moment integrands v time for the first three moments for the reactant pulse.	141
3.20	Calculated basic kinetic coefficients for the products on the original and the LFF corrected data. a) Calculated r_0 coefficients b) Calculated r_1 coefficients c) Calculated r_2 coefficients	142
4	Temperature Programmed Reduction	145
	Application of TPR Data Analysis	145
4.1	Baseline correction in CCI-TPR software	146
4.2	Baseline correction of TPR data a) Raw TPR signal b) Baseline corrected TPR signal	146
4.3	Simulated TPR thermogram using a three process kinetic model . . .	148
4.4	Testing of deconvolution procedure a) Simulated TPR thermogram at 5 K min ⁻¹ with added random noise b) Calculated activation energy profile	149
4.5	$dE_a/d\alpha$ profile calculated for the simulated TPR thermogram	150

4.6	Comparison of simulated + noise three peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles	153
4.7	Comparison of simulated + noise two peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles	153
4.8	Comparison of simulated + noise one peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles	154
4.9	$dE_a/d\alpha$ profile with deconvolved processes overlayed	155
4.10	Master plots calculated using Malek's procedure a) Master $z(\alpha)$ plot for deconvolved curves b) Master $y(\alpha)$ plot for deconvolved curves . .	156
4.11	CeO ₂ catalyst calcined at 400°C baseline corrected data, activation energy, and $dE_a/d\alpha$ profiles respectively.	160
4.12	Experimental (Exp.) and simulated two process (Sim.) $d\alpha/dT$ and activation energy profiles overlaid for CeO ₂ calcined at 400°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	161
4.13	Experimental (Exp.) and simulated three process (Sim.) $d\alpha/dT$ and activation energy profiles overlaid for CeO ₂ calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	161
4.14	a) CeO ₂ catalyst calcined at 500°C baseline corrected data b) activation energy c) $d\alpha/dT$ profiles respectively.	163

4.15	Experimental (Exp.) and simulated two process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO ₂ calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	164
4.16	Experimental (Exp.) and simulated three process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO ₂ calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	164
4.17	a) CeO ₂ catalyst calcined at 600°C baseline corrected data b) activation energy c) $d\alpha/dT$ profiles respectively	166
4.18	Experimental (Exp.) and simulated four process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO ₂ calcined at 600°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	167
4.19	Experimental (Exp.) and simulated two process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO ₂ calcined at 600°C zoomed on second peak a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.	167

5 Computer Simulation Results 173

Hydrogenation of Levulinic Acid	173
---	-----

5.1	Standard unit cells used in energy calculations a) Zr ₅₄ O ₁₀₈ unit cell, Zr - Grey and O - Red b) Cu ₁₀₈ O ₁₀₈ unit cell, Cu - Cyan and O - Red	176
-----	--	-----

5.2	Optimisation of Cu doped ZrO_2 a) Initial $CuZr_{53}O_{107}$ unit cell, Zr - Grey, O - Red and Cu - Cyan b) Optimised $CuZr_{53}O_{107}$ unit cell, Zr - Grey, O - Red and Cu - Cyan	177
5.3	Simulated physiochemical properties for standard ZrO_2 and CuO unit cells normalised a) Lattice parameter c as a function of Cu doping % b) Mixing energy as a function of Cu doping %	178
5.4	gVL yield as a function of $Cu / (Cu + Zr)$ determined by ICP. Open square - Og preparation method, Closed square - Me preparation method taken from[1]	179
5.5	gVL yield normalised by BET surface area taken from[12]	180
5.6	Energies of the various nanoclusters normalised to cluster size taken from [13]	180
5.7	13, 38, and 75 atom nanocluster shapes	181
5.8	Various sites available on 38 and 75 atom nanoclusters	181
5.9	Possible mechanisms for conversion of LA to gVL (1-3)	182
5.10	Gas phase reaction mechanism for conversion of LA to gVL with most likely mechanism shown in red.	183
5.11	Liquid phase reaction mechanism for conversion of LA to gVL with errors bars showing the standard deviation of the energies over the course of the simulation.	186
5.12	Schematic of interaction between the molecules and water	187

5.13	Radial distribution functions for the interaction between the oxygen atoms on the molecule, and the hydrogen atoms on the water. a) Both enol systems superimposed on top of each other b) Hydrogenated intermediate	188
5.14	Radial distribution functions for the interaction between the oxygen atoms on the molecule a) Both enol systems superimposed on top of each other b) Hydrogenated intermediate.	189
5.15	Mechanism for adsorption and dissociation of a H ₂ molecule on the (100), (111), (110), and <i>exposed</i> sites on the Cu ₇₅ nanocluster, <i>exposed</i> site is pictured.	191
5.16	Mechanism for hydrogenation of LA to gVL over (100) site on Cu ₇₅ .	192
5.17	Mechanism for hydrogenation of LA to gVL over (111) site on Cu ₇₅ .	193
5.18	Geometry of (110) site on Cu ₇₅	194
5.19	Mechanism for hydrogenation of LA to gVL over (110) site on Cu ₇₅ .	195
5.20	Mechanism for hydrogenation of LA to gVL over <i>exposed</i> site on Cu ₇₅	196
5.21	Example of interpolated images between start and end point (hydrogenated intermediate to gVL) as generated by <code>inter_vasp</code>	197
5.22	Hydrogenated intermediate with labelled atoms.	198
5.23	Results from NEB calculations for ring closing of hydrogenated intermediate to for gVL a) Reaction coordinates R1 and R2 (5.22) as a function of NEB run b) Energy as a function of reaction coordinate R1.	198
5.24	Unit cells for both the bulk and surface used in the NEB calculations a) Cu 1x1x1 bulk unit cell b) 6 Layer (110) surface cut from bulk unit cell	199

5.25	Mechanism for dissociation of hydrogen over Cu (110) surface slab . .	200
5.26	Transfer of hydrogen in 1,3-pentadiene complex with reaction coordi- nate labelled	201
5.27	Example of potential walls for neighbouring images emphasised for clarity	202
5.28	Results from Umbrella sampling and NEB methods for determining reaction pathways a) Energies from calculations b) Minimum pathways	202
5.29	Umbrella sampling for ring closing of hydrogenated intermediate . . .	203
5.30	Lowest energy configuration of the reaction intermediate, with hydro- gen bonding displayed in blue	204
5.31	Comparison of the intramolecular and intermolecular ring closing mechanisms and the effect of the H ₂ O solvent.	205
6	General Conclusions	210
7	Appendix	217
7.1	Three coefficient solutions for various reaction mechanisms	218
7.2	Binding modes for the Levulinic Acid a) 1 ketone group b) 2 carbonyl in carboxyl group c) 3 alcohol in carboxyl group.	230
7.3	Binding modes for the hydrogenated intermediate a) 1 carbonyl in carboxyl group b) 2 alcohol in carboxyl group c) 3 alcohol group. . .	231
7.4	Binding modes for the GVL a) 1 carbonyl group b) 2 through ring. .	231
7.5	Binding modes for the H ₂ a) 1 molecular b) 2 dissociated.	231

1 | Introduction

Catalysis is arguably one of the most important advancements in the industrial process in the modern history. It is often stated that catalysis is involved either directly or indirectly with over 90% of all chemically produced products,[1] and can be directly linked to the explosion in the human population since the early 1900's.[2] With catalysis being so central to almost all aspects of life it is understandable why catalytic research has gained such a large following in recent years.

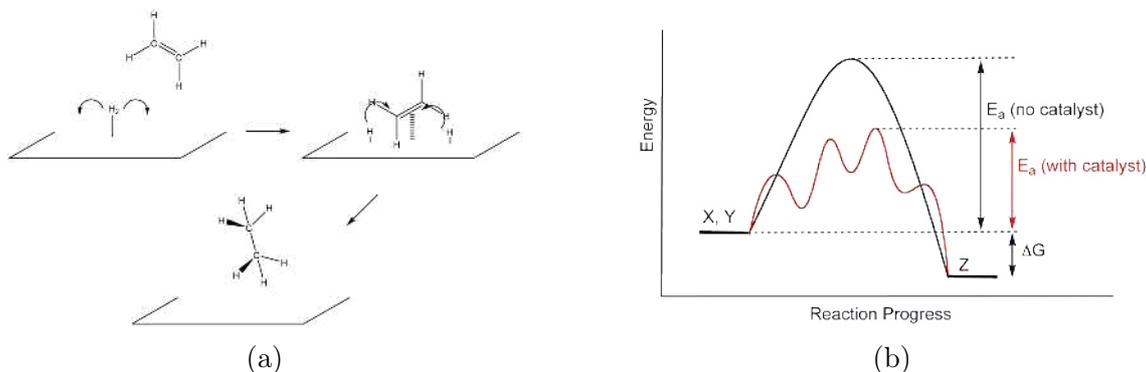


Figure 1.1: Example of catalytic mechanisms a) Scheme for catalytic hydrogenation of ethene b) Energy profile for uncatalysed and catalysed process

Catalysis as a whole can be divided into two subsections: heterogeneous catalysis, where the catalyst is in a separate phase to the reactants, or homogeneous catalysis where the catalyst is in the same phase. For the purpose of this thesis focus will be purely on the former. When one thinks about a heterogeneous catalyst the standard image that comes to mind is of a reactant (or reactants) binding to the surface of the catalyst, and subsequently undergoing a chemical change to form the desired products. This gives us the standard expression: *A catalyst is a substance that*

increases the rate of a chemical reaction by providing an alternate route of lower activation energy, without itself being consumed in the process subsequently leading us to the basic idea of a catalyst as shown in figure 1.1. This idea of a catalyst as a simple anchoring point for the reaction to occur is outdated, and attempting to understand the nuances of how a catalyst and the reactant interact with each other is becoming more popular, (figure 1.2) this has lead a shift in chemists mindsets from *what* catalysts work to *how* catalysts work.

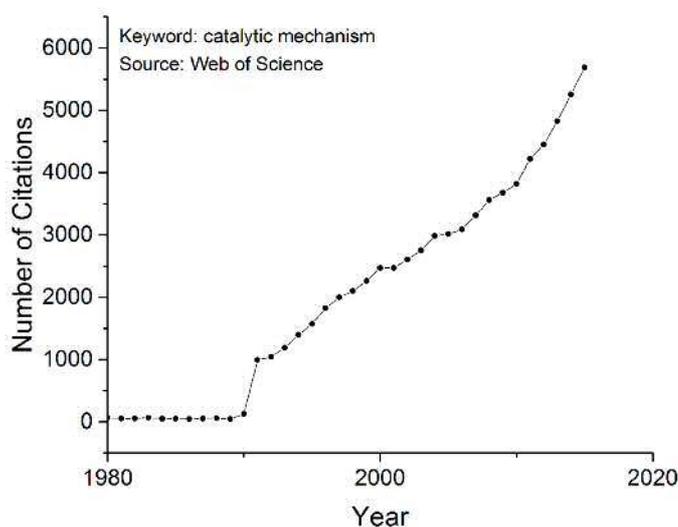


Figure 1.2: Number of citations regarding catalytic mechanisms

In order to understand the fine details of how a catalyst works, one of the first things catalytic chemists will look at is it's surface structure. This is performed by a number of standard experiments (*e.g.* STM, XRD, XANES, XAFS, IR) which we have no need to discuss in detail in this thesis. By studying the surface structure of the catalyst they would hope to extrapolate between the initial surface structure of the catalyst, to the structure after the reaction, and use that information to guess the catalytic mechanism. By taking only the start and end point of the catalytic system we can fall into the trap of considering the catalyst as a simple, sometimes static, image (figure 1.1a). It is becoming increasingly more apparent that when we consider a catalyst system we need to think of it as being dynamic, instead of our static model, [3] but unfortunately attempting to capture that dynamics is difficult.

There have been some bounds made in *in-situ* experimental techniques [4, 5] for studying catalysts but they require specialist equipment, which tends to be very expensive and not readily available, meaning they are not commonplace. In order to counteract this we attempt to show in this thesis that actually understanding *how* a catalyst works can be performed by gaining more information on the *kinetics* of the reaction, instead of trying to directly capture the dynamics of the catalytic system.

Reaction (or chemical) kinetics is the study of the rate at which chemical processes occur. Our understanding of reaction kinetics has changed drastically over the last century, with the latest iteration being focused mainly on the mathematics which can be used to describe how chemical reactions occur.[6] It can be assumed that if one has all of the kinetic information for each of the reaction steps, they can use that information to fully understand the catalytic system. As kinetics is based on the rate of change of one substance to another, it means that the dynamics of the system is preserved. It is this ability to preserve the dynamics of the system that sets kinetic analysis above other methods when it comes to understanding catalysts.

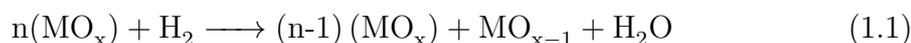
The most basic method of calculating reaction rates is to measure the change of concentration of the reactants and products as a function of time. This means that any experimental method by which you can measure the concentration of a chemical can be used to calculate kinetics. If one spends enough time studying the reaction kinetics of a particular system or experiment it is possible to start to develop a mathematical model to understand - and in some cases simulate - their results. The problem with using mathematical models is that they tend to be very complex, meaning that they can be difficult and time consuming to solve, which results in kinetic analysis being underused. The main aim of this research has been to streamline this process of performing the kinetic analysis and to try and develop it further. The particular experiments that we are interested in are Temperature Programmed Reduction (TPR), and Temporal Analysis of Products (TAP).

Alongside the kinetic analysis of experimental data, computational simulations are often used to understand reaction kinetics, as they can be used to model chemical reactions in a level of detail not possible by current experimental means. In this thesis the particular reaction of interest has been the hydrogenation of Levulinic acid to gamma-Valerolactone. The aim has been to model each the intermediate steps of the reaction, and by using dynamic simulations, attempt to calculate the most likely pathway for the reaction to occur.

1.1 Temperature Programmed Reduction

Temperature programmed reduction is a very simple experiment which is commonplace in almost all catalytic and chemical laboratories. The setup for the reactor is very simple (figure 1.3) and consists of flowing a reducing gas (typically H₂) over the surface of the catalyst (typically a metal oxide), at a set flow rate, and measuring the uptake of the gas as a function of time by recording the entrance and exit readings of the reactor. The reactor is heated at a constant rate, and as such TPR is considered a non-isothermal technique. Due to this linear heating, instead of plotting the thermogram as a function of time, it is often plotted as a function of temperature instead.

A TPR experiment can be expressed using the chemical equation 1.1:



Where n is the number of moles of the metal oxide (MO_x) present in the catalyst sample. Throughout the TPR thermogram the adsorption of H₂ will increase as the temperature is increased. As the reduction process is able to overcome its activation energy barrier, the adsorption of H₂ will rise, until it reaches a peak and then

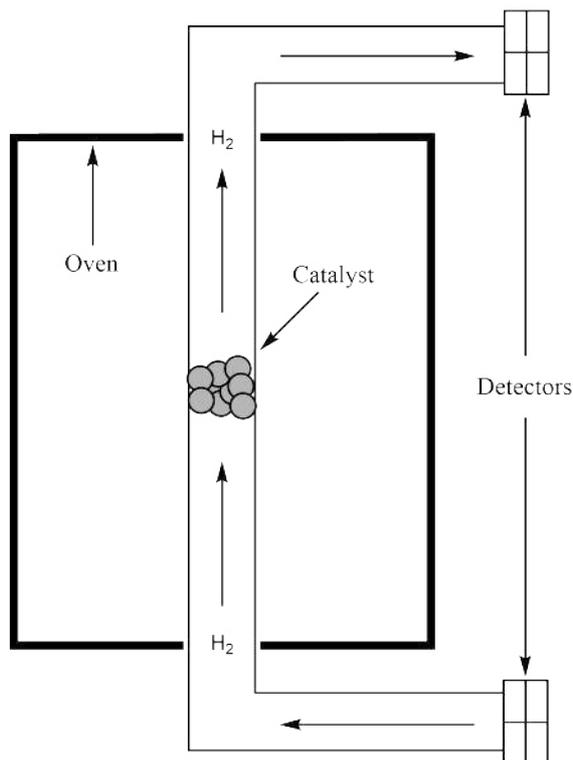


Figure 1.3: Basic schematic of a TPR experiment.

decays back to the baseline as the material converts from the metal oxide (MO_x) to its metallic or reduced state (MO_{x-1}). A standard TPR thermogram can be seen in figure 1.4 which shows this rise and fall as a function of temperature.

As the TPR thermogram rises and falls back to the baseline we can assign a variable α using equation 1.2 which we define as the degree of reduction. Where T_m is the temperature at which the degree of reduction (α) is being measured, T_0 and T_f being the start and final temperatures of the thermogram respectively, and finally c_{H_2} being the concentration of hydrogen being absorbed.

$$\alpha(T_m) = \frac{\int_{T_0}^{T_m} c_{H_2} dT}{\int_{T_0}^{T_f} c_{H_2} dT} \quad (1.2)$$

Using equation 1.2 to calculate α for the TPR thermogram one can start applying current kinetic analysis.

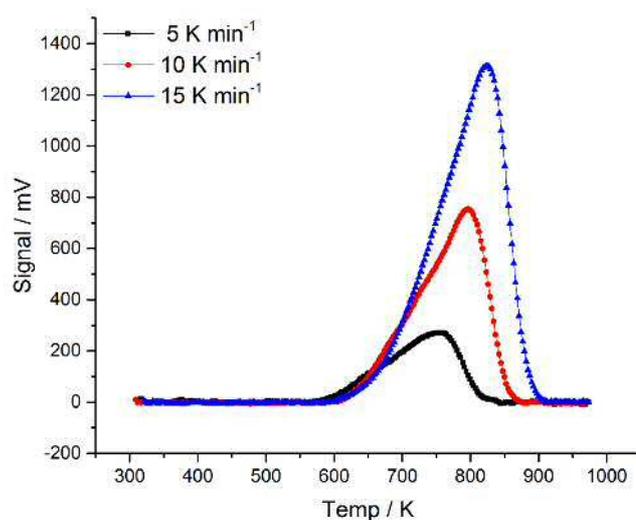


Figure 1.4: Thermogram for the reduction of CeO_2 as a function of temperature at three heating rates: 5 K min^{-1} , 10 K min^{-1} , 15 K min^{-1}

The idea for non-isothermal kinetic analysis first originated in the late 1950's as a result of the development of various thermal based analytical methods. It was only then that technology had reached a point where accurate thermal and adsorption readings could be taken. After the results of these experiments were studied in depth it was theorised that these temperature programmed thermograms, during which some process is occurring (be it oxidation, reduction or desorption), contain kinetic information, and mathematical models were developed to explain these processes.[7–10] Over the last 40 to 50 years these models have barely changed, with the majority still in use to this day,[11] but unfortunately their application has been sparse at best.

The majority of TPR analysis, particularly in catalysis, is rudimentary at best. If one was to perform a sampling of the current literature they would see that for the majority of papers which contain temperature programmed experiments they simply study the location of the peaks and comment on any shifts that occur,[12] or they look at quantitative data *e.g.* the quantity of H_2 absorbed during a TPR thermogram.[13] When the kinetic analysis is seen in the literature it is often in a paper which in which the kinetic modelling, rather than the actual catalytic system,

is the main focus of the paper.[14, 15]The few papers in which the kinetic modelling has been applied to describe a system can be used to gain deep insight into the behaviour of a catalyst. One of the model papers for this would be the 2004 paper by Munteanu *et al.*[16] In this paper they use the kinetic modelling - in a similar method as is used in this thesis - in order to describe the influence gold has on the reduction behaviour of a complex catalyst (Au–V₂O₅/CeO₂).

In the paper they took varying forms of the Au–V₂O₅/CeO₂) catalyst and performed a TPR on each sample. They then used the kinetic constants calculated from the regression algorithm to understand the structure of the surface, and how the gold influences the reduction behaviour.

The main parameters of the TPR processes characteristic to the samples without gold							
Sample	T_{\max} (K)	Percentage of the peak area	Activation energy (kJ mol ⁻¹)	k	Pre-exponential factor (m ³ mol ⁻¹ s ⁻¹)	Sestak–Berggren coefficients	
						m	n
C	689	22	127.0	0.0	1.8×10^7	0.0	1.33
	774	78	131.2	0.0	3.0×10^6	0.0	0.66
4 VC	675	13	155.1	0.0	6.0×10^9	0.0	1.0
	773	62.6	145.4	0.0	3.0×10^7	0.0	1.33
	794	24.4	193.0	0.0	3.0×10^{10}	0.0	0.66
8 VC	722	6	140.0	0.0	6.0×10^7	0.0	1.0
	807	64	156.3	0.0	6.0×10^7	0.0	1.33
	824	30	175.5	0.0	6.0×10^8	0.0	0.9

The main parameters of the TPR processes characteristic to the samples with gold, prepared by the DP method							
Sample	T_{\max} (K)	Percentage of the peak area	Activation energy (kJ mol ⁻¹)	k	Pre-exponential factor (m ³ mol ⁻¹ s ⁻¹)	Sestak–Berggren coefficients	
						m	n
AC-DP	392	100	71.0	0.11	6.0×10^7	0.8	0.8
4 VAC-DP	377	6	90.2	0.04	6.0×10^{10}	0.7	1.3
	392	38	96.9	0.17	4.8×10^{10}	0.3	1.3
	428	56	82.7	0.04	6.0×10^7	0.15	1.3
8 VAC-DP	399	7	82.7	0.08	6.0×10^8	0.5	1.3
	416	23	102.8	0.28	3.6×10^{10}	0.25	1.3
	456	70	80.6	0.07	6.6×10^6	0.07	1.3

Figure 1.5: Parameters calculated from regression algorithm taken from [16]

It was found that when gold was included in the sample, the activation energy was found to be highly dependent on the degree of reduction, this, along with other parameters, showed that the gold was highly involved in the reduction process, and that the reduction process starts initially on the grain boundary with the gold particles, and then spreads out over the catalyst surface. It was also found that the gold changed the bulk characteristics of the material, facilitating the diffusion of O atoms through the lattice. The paper shows that the kinetic theory can be used

to gain deep understanding of the catalyst material, which is invaluable in catalyst design.

The work outlined in this thesis is involved in attempting to automate the kinetic analysis of TPR experiments using the methods outlined in the literature. The kinetic analysis has been programmed into a "front-end" focused software, which is based around a central graphical user interface named *CCI-TPR* shown in figure 1.6. Alongside automating the current analysis of TPR thermograms a new more robust methodology for calculating the various kinetic parameters from experimental TPR data has been developed. Using this new method the accuracy of the calculated kinetic parameters is increased greatly, meaning that the user can gain a deeper insight into the surface reduction of their catalytic material. Users can apply this new information to their catalytic system helping them create more efficient and better catalysts. With TPR being such a commonplace experiment to perform it

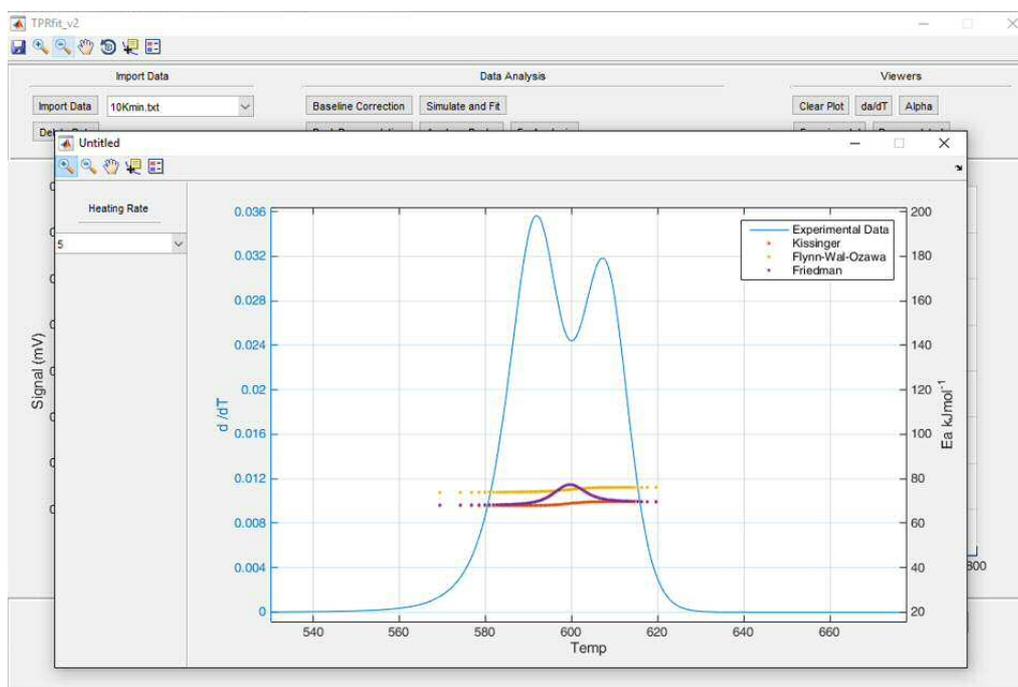


Figure 1.6: The *CCI-TPR* main graphical user interface with overlaying activation energy subplot

would be in the best interest of catalytic chemists to use these newly developed tools in order to greatly increase the quality and quantity of their data.

1.2 Temporal Analysis of Products

Temporal analysis of Products is a more recent tool for gas phase heterogeneous catalysis. It was first developed by John Gleaves in 1988[17] as an advance on molecular beam scattering (MBS) experiments (see figure 1.7). Whereas in molecular beam experiments a pulse (or beam) of gas is sent at a single crystal target and the scattering recorded by a mass spectrometer, the TAP experiment consists of sending a pulse of gas through a catalyst sample and the flow of gas out the end of the reactor is detected instead.

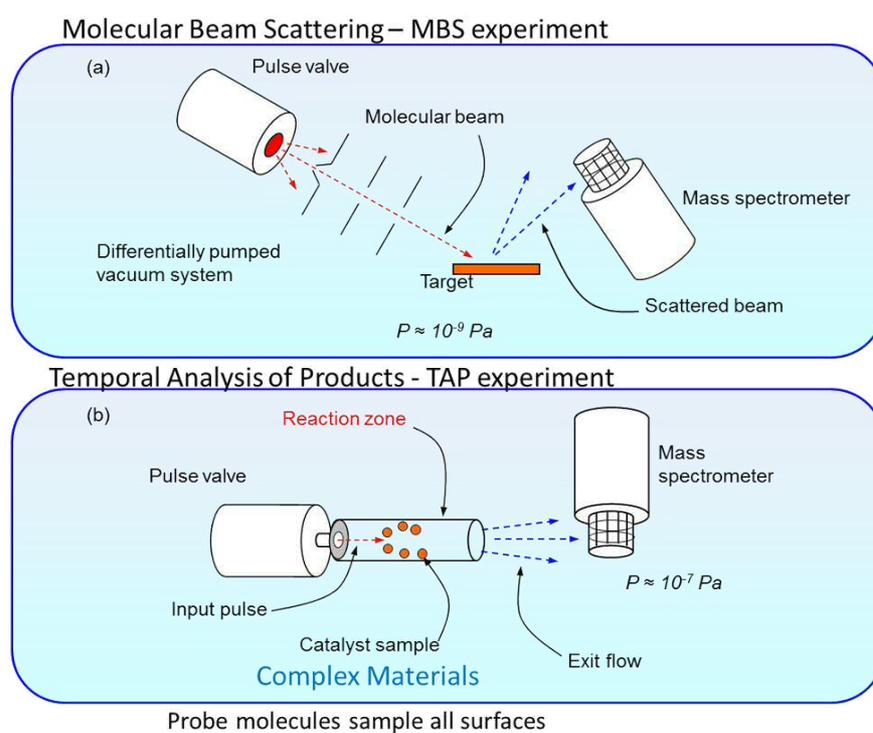


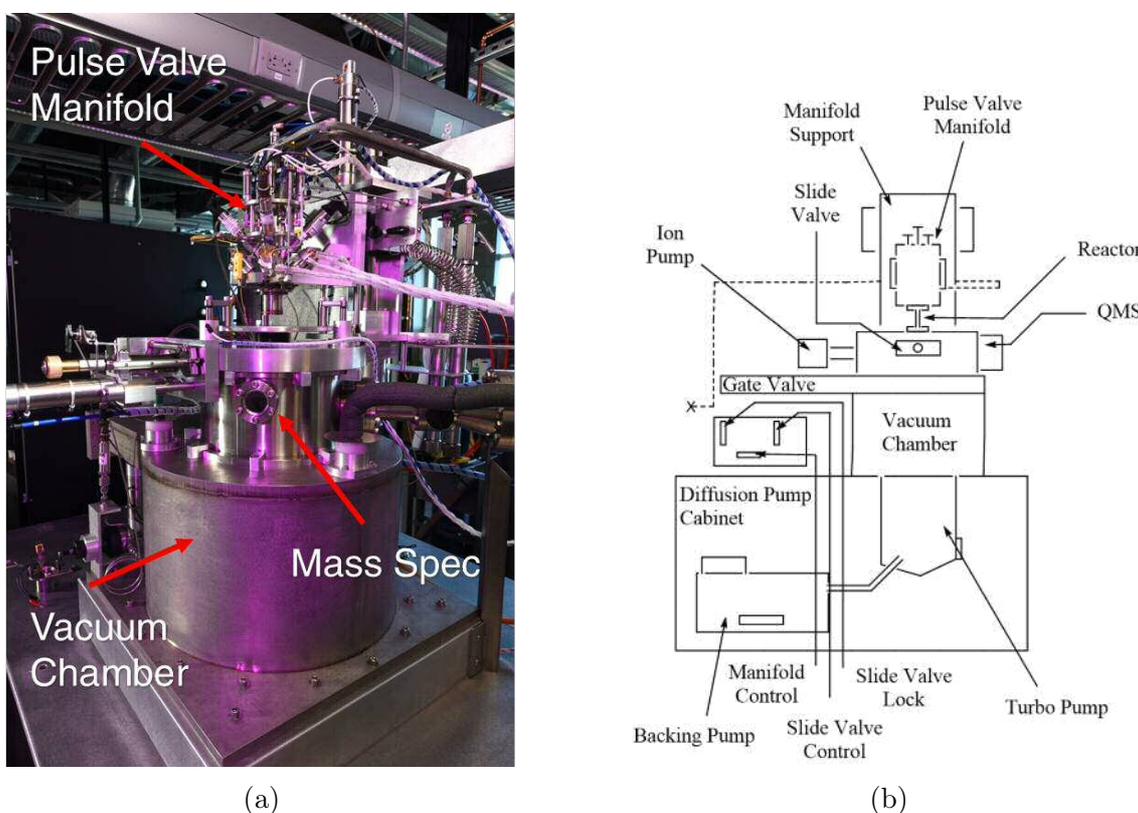
Figure 1.7: Comparison between MBS and TAP experiments[18]

Unlike TPR, TAP is a highly specialised technique, requiring complex machinery and a dedicated analyst. Due to this large barrier to entry TAP is considered a niche technique, with there being approximately 20 reactors worldwide (figure 1.8).

The TAP reactor itself is a highly complicated piece of machinery (figure 1.9) which contains multiple moving parts. For the sake of simplicity it is easy to break the reactor down into the four main components.



Figure 1.8: Locations of TAP reactor systems

Figure 1.9: TAP reactor schematic a) TAP-3 reactor at Idaho National Laboratory
b) Schematic of a TAP-3 system

First is the ultra-high vacuum chamber, this is backed out using high throughput vacuum pump such as a turbo molecular pump, or an oil diffusion based pump down to approximately 10^{-7} mbar. The second component is the quadrupole mass spectrometer (QMS), this is housed inside the vacuum chamber and is used to detect the exit flow of the gas out of the reactor. Sat on top of the vacuum chamber, just

above the QMS, is the reactor. The reactor is approximately 4 cm in length and it is where the catalyst is housed. The final component is the pulse valve manifold. Inside the manifold the pulse valves are housed, and it is these valves which control the amount of gas entering and exiting the system. The manifold is the only part of the main system that is not under ultra high vacuum conditions.

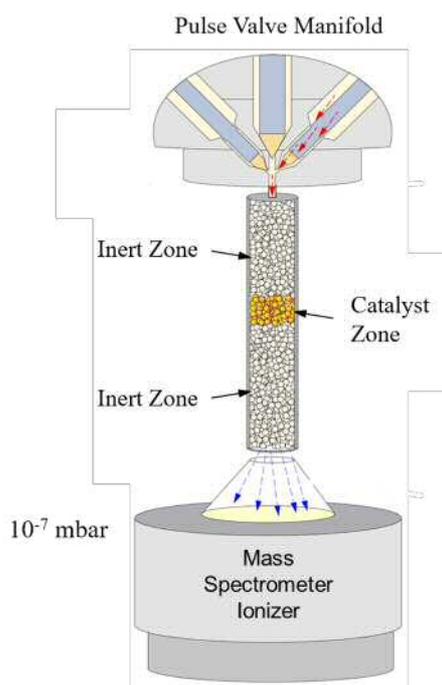


Figure 1.10: Cutaway of the reactor and the pulse valve manifold.

As can be seen in figure 1.10 the reactor is packed with an inert (usually SiC) and in the centre is a thin layer of the catalyst. When one performs a TAP experiment a voltage is sent to one of the pulse valves causing it to retract, then due to the pressure differential between the gas in the pulse valve and the vacuum chamber the gas flows from the pulse valve, over the surface of the catalyst, and into the vacuum chamber to be collected by the QMS. The current is turned off and the pulse retracts and the reactor is brought back to ultra high vacuum conditions. An example of a TAP response is shown in figure 1.11b and as it can be seen in figure 1.11a the actual time the valve is open is very small (approximately $300 \mu\text{s}$), this means that the size of the pulse is also very small (approximately 10^{-10} moles).

Due to this small pulse size we can state that the each individual pulse will

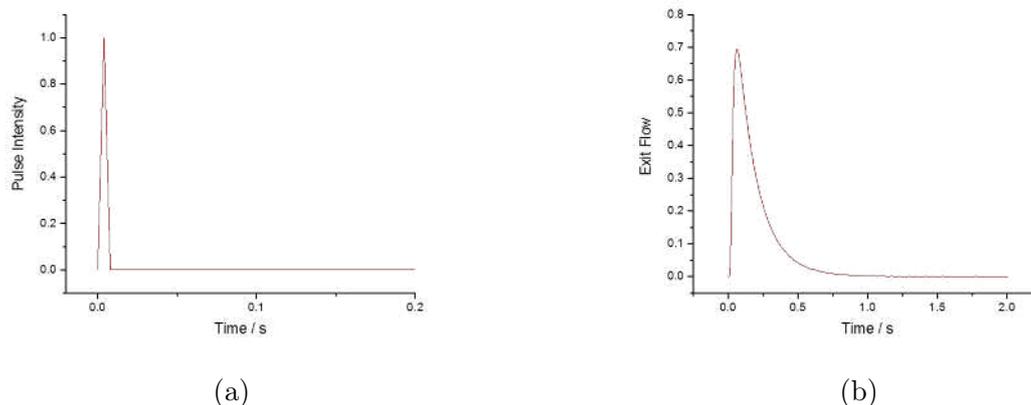


Figure 1.11: Standard TAP experimental responses a) Pulse intensity v time for a TAP pulse b) Exit flow v time for a TAP pulse

not significantly change the structure of the catalyst, meaning that each pulse is a snapshot of the catalyst at that particular state. This means that we classify TAP as a transient technique, with one of the major bonuses of TAP being that we can study structure far from equilibrium. While being able to study structures far from equilibrium is a novel method, in real life it has little application as almost all major industrial processes are at steady state conditions (at equilibrium). While for most reactors this would be a problem, where TAP is able to excel is that by following a principle coined as *Chemical Calculus*[19] one can start to induce change in the catalyst, and eventually study it in its steady state conditions.

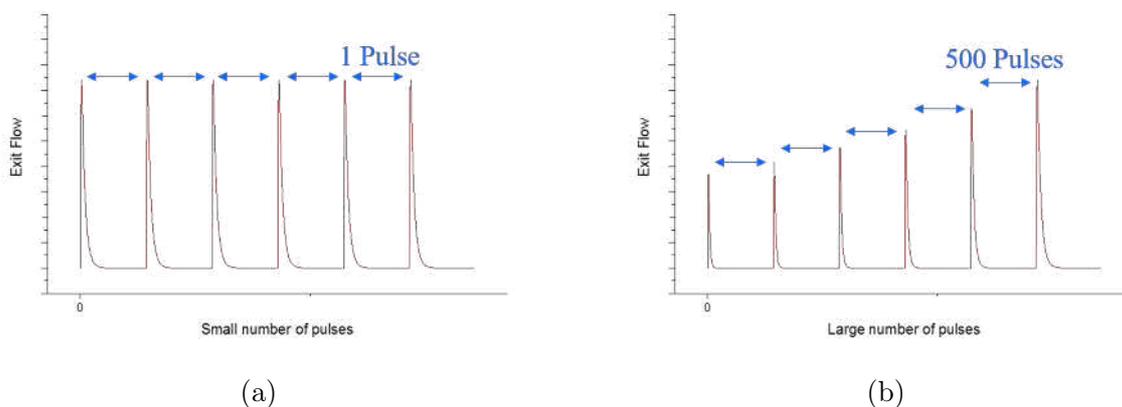


Figure 1.12: Visualisation of the principle behind *Chemical Calculus* a) Short term pulse response v time b) Long term pulse response v time

The theory behind *Chemical Calculus* is that even though an individual pulse

will not change the state of the catalyst significantly, we can state that a long train of sequential pulses will start to induce change in the catalyst surface. This means that by performing this sequential pulsing, one can get the catalyst from its initial state to its equilibrium state, and as we consider each pulse a snapshot of the catalyst, this means we view the catalyst at every single point during its evolution. By viewing the catalyst as it evolves, it is possible to capture the dynamics of an evolving catalyst system which gains invaluable insight into how the material functions. Alongside watching the catalyst evolve, it is possible to disturb the catalyst by sending a series of pulses, and then view the catalyst as it relaxes back to its initial state, gaining insight into how the material functions.

In order to understand how to use these TAP pulse responses, one first needs to understand the underlying principles of a TAP experiment. Due to the high-throughput vacuum and small pulse size, the diffusion through the TAP is classified as Knudsen Diffusion. This special case of diffusion occurs when the length of the between the particles is considered to be much smaller than the *mean free path* of the gas molecules involved, and as such gas-gas interactions are removed.

$$D = \frac{\epsilon}{t_r} \frac{d_i}{3} \sqrt{\frac{8RT}{\pi M_w}} \quad d_i = \frac{2\epsilon}{3(1-\epsilon)} d_p \quad (1.3)$$

Equation 1.3[20] represents the calculation of coefficient of *Knudsen Diffusivity* and is represented by D . The other coefficients are the fractional voidage of the reactor bed ϵ , the particle diameter d_p , the molecular weight of the gas M_w , and finally the bed tortuosity t_r which describes the "straightness" of the reactor. As can be seen, for a consistent reactor packed with the same amount of catalyst and inert, the average speed through the porous media is defined by a Maxwell-Boltzman distribution at a given temperature T . The diffusion is defined by the interaction between the gas and the porous medium rather than the molecule-molecule collision

frequency, and as such it can be said that the transport of the gas through the reactor is well defined as seen in figure 1.13a.

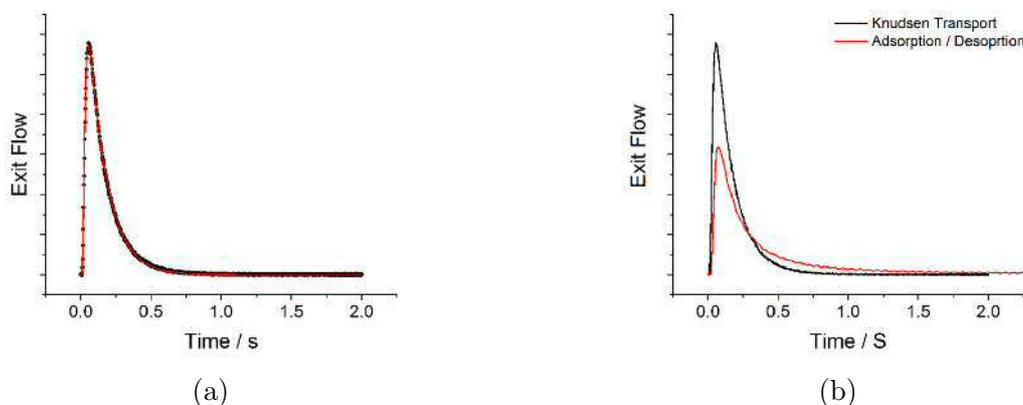


Figure 1.13: Knudsen exit flow comparisons a) Simulated v experimental Knudsen exit flow b) Knudsen v adsorption/desorption exit flow

From figure 1.13b it can be seen that the exit flow purely defined from Knudsen transport is different to the exit flow of the adsorption/desorption case. In the latter case, the transport of the species through the reactor is still governed by Knudsen diffusion, but is also effected by the interaction between the surface of the particles in the reactor. As long as the reactor is packed with an inert and the catalyst it can be said that by "removing" the well defined Knudsen diffusion, the exit flow is related purely to the interaction between the gas and the catalyst. It is using this principle that the kinetic theory of the TAP reactor experiments was developed. Spearheaded by the work of Yablonsky[21] an unprecedented amount of detail can be calculated from the exit flow response curves, [18] including parameters such as:

- Reaction rate constants
- Activation energies
- Surface concentrations
- Surface residence times
- Deconvolution of complex microkinetic modelling

For the purposes of this thesis the parameters of interest are the reaction rate constants.

When reviewing the current literature on TAP it comes apparent very quickly that the majority of the work is performed by a small community of approximately 15-20 researchers. This small community is due to the large barrier to entry to join the TAP community in the high price and specialised nature of the TAP reactor. The researches working with TAP tend to be broken into two groups, those who are dedicated to developing the theory behind the analysis of TAP, [21–29] and those who are most interested in the experimental application of TAP.[30–35]

A good paper which highlights the full capabilities of TAP is the 2008 paper by Shekhtman *et. al.*[36] This combines a robust experimental analysis of TAP data and utilises the more complex kinetic analysis in order to characterise a 2 %Pt/CeO₂ catalyst. The paper studies the effect of using various pretreatments (O₂, H₂, CO₂, and H₂O) by studying the effect on the consumption of CO as it is pulsed over the catalyst. The experimental procedure was to pretreat the 2 mg of catalyst at atmospheric pressure using four combinations of gases:

- 20% O₂ for 1 hour
- 20% O₂ for 1 hour, followed by H₂ at 300°C for 1 hour, followed by 10% H₂O flow at 300°C for 1 hour.
- 20% O₂ for 1 hour, followed by H₂ at 300°C for 1 hour, followed by 100% CO₂ flow at 300°C for 1 hour.
- 20% O₂ for 1 hour, followed by H₂ at 300°C for 1 hour

Then insert the catalyst into the TAP reactor, pump out the reactor until it was at the correct conditions for Knudsen diffusion (10^{-6} Torr), and then perform the standard multi pulse experiment using CO. During the experiment the pulse-by-pulse conversion of CO was recorded as a function of total amount of CO converted.

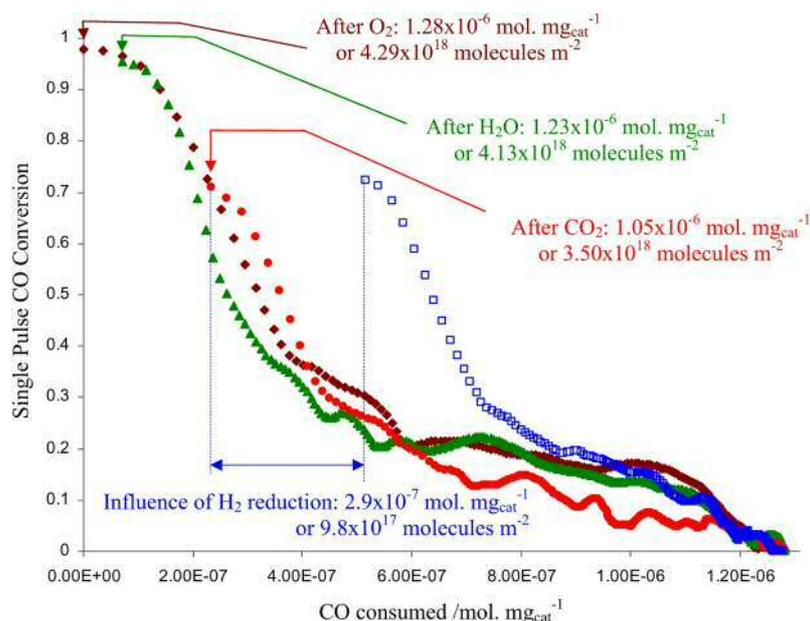


Figure 1.14: Pulse by pulse conversion of CO v total CO conversion

In figure 1.14 the conversion of CO has been normalised so that all of the pulse trains end at the same point (considered full reduction of the catalyst surface). By plotting the data in such a way it can be seen that CO consumption profiles for catalysts pretreated with an oxidising agent (*i.e.* O_2 , H_2O , CO_2) all follow the same pattern, meaning that each oxidising agent consists of a different starting point on the general curve shape. The starting points of the curve clearly give insight into the active sites for oxidation of the catalyst. As O_2 is the strongest oxidising agent, followed by H_2O , then finally CO_2 , by studying the difference between their starting points the number and strength of the active sites on the surface of the 2%Pt/CeO₂ catalyst can be estimated. Alone this information is very useful for understanding a catalyst but the next step was to use kinetic modelling to calculate an apparent transformation constant for CO.

Figure 1.15 clearly shows a change in the *transformation constant*, which is a term used to describe an apparent rate constant for conversion, as a function of CO consumed (and hence the number of active sites left on the catalyst), therefore it can be stated there are two different types of active sites present. The initial high activity sites, followed by low activity sites. By repeating the experiment at multiple

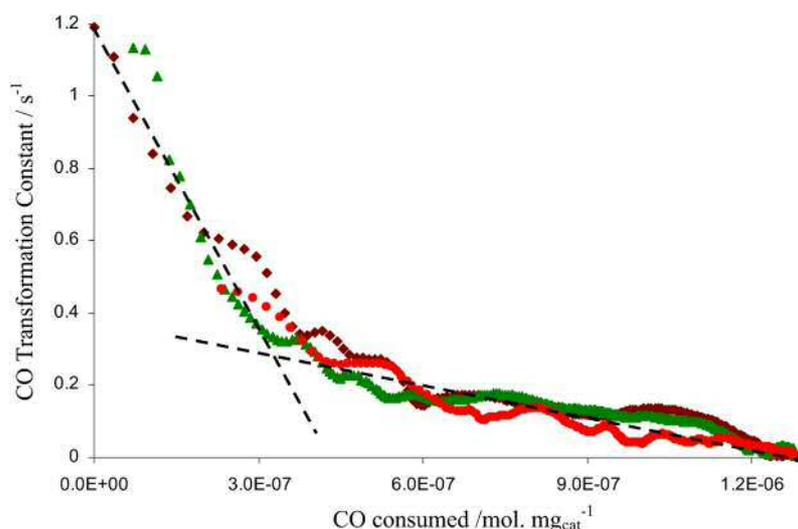


Figure 1.15: Apparent transformation constant for CO v total CO conversion

temperatures an Arrhenius plot can be generated, giving the activation energy of each of the catalytic sites 1.16.

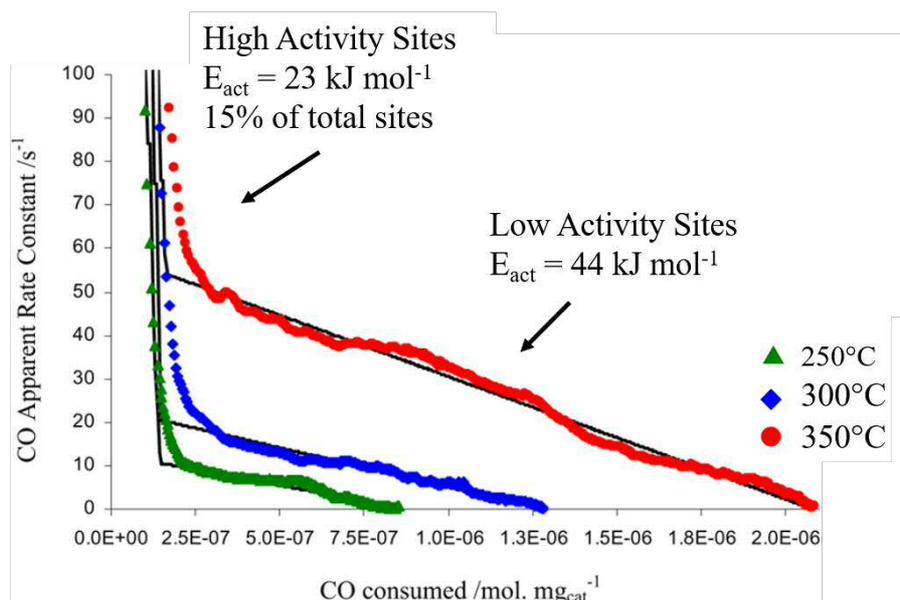


Figure 1.16: CO oxidation experiment repeated at three temperatures, with overlaying activation energies

From the paper it can be concluded that by using the TAP reactor a large amount of useful kinetic information can be gathered on a catalyst by using the kinetic analysis outlined in the literature. The major issue for TAP is that there is no simple way of performing this analysis and it is usually done by hand, meaning it is very time consuming. One of the major projects of this thesis has been to

automate this kinetic analysis by implementing it into a user-friendly graphical user interface called *CCI-TAP*. By using the combination of the TAP reactor and the *CCI-TAP* software gathering this detailed kinetic information can be performed rapidly and with little to no knowledge of the underlying kinetics.

Although TAP has been around for almost 30 years, it is still considered to be an emerging technique. This is in part due to the small TAP community and also the huge barrier to entry when it comes to TAP, both in working with the reactor and understanding the theory. By creating the *CCI-TAP* software, and by developing a new streamlined experimental methodology for TAP data analysis, it is hoped that at least one of those barriers can be lowered making TAP a more accessible resource for catalytic research.

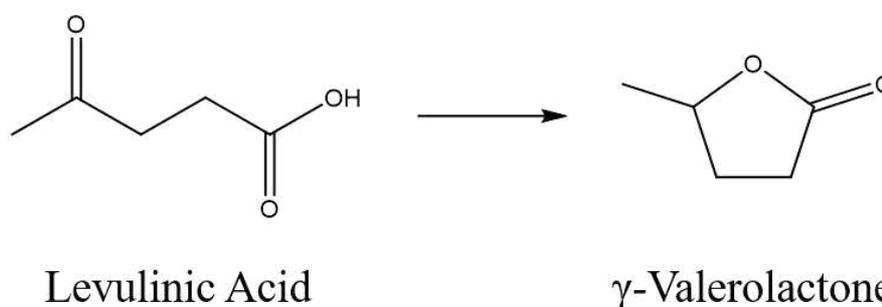
1.3 Hydrogenation of Levulinic Acid

The final section of this thesis is devoted to work which was performed as part of a larger European Union (EU) project titled NOVACAM, which stands for: *Novel and Cheap Abundant Materials for catalytic biomass conversion*. The aim of the NOVACAM project was to find catalysts which use materials which had been defined as "*unsustainable*" by the EU[37] (figure 1.17) and attempting to recreate their reactivity and selectivity using sustainable analogues. Biomass conversion was the main goal of the NOVACAM project, with three subsections, Chemocatalytic Glycolysis, Aqueous Phase Reforming, and The Valeric Platform.

The goal of the Valeric Platform was the convert cellulose to γ -Valerolactone (gVL), and the particular reaction of interest for this work was the hydrogenation of Levulinic Acid (LA) to gVL, shown in figure 1.18. This reaction was chosen as LA is a readily available byproduct from various cellulose processing techniques [38, 39], and gVL was chosen as a target molecule as it has its uses as a food and fuel additive, and as also a potential green solvent. [40, 41]

hydrogen 1 H 1.008																	helium 2 He 4.003	
lithium 3 Li 6.941	beryllium 4 Be 9.012											boron 5 B 10.811	carbon 6 C 12.011	nitrogen 7 N 14.007	oxygen 8 O 15.999	fluorine 9 F 18.998	neon 10 Ne 20.180	
sodium 11 Na 22.990	magnesium 12 Mg 24.305											aluminum 13 Al 26.982	silicon 14 Si 28.086	phosphorus 15 P 30.974	sulfur 16 S 32.065	chlorine 17 Cl 35.453	argon 18 Ar 39.948	
potassium 19 K 39.098	calcium 20 Ca 40.078	scandium 21 Sc 44.956	titanium 22 Ti 47.887	vanadium 23 V 50.942	chromium 24 Cr 51.996	manganese 25 Mn 54.938	iron 26 Fe 55.845	cobalt 27 Co 58.933	nickel 28 Ni 58.693	copper 29 Cu 63.546	zinc 30 Zn 65.39	gallium 31 Ga 69.723	germanium 32 Ge 72.63	arsenic 33 As 74.922	selecnium 34 Se 78.96	bromine 35 Br 79.904	krypton 36 Kr 83.80	
rubidium 37 Rb 85.468	strontium 38 Sr 87.62	yttrium 39 Y 88.906	zirconium 40 Zr 91.224	niobium 41 Nb 92.906	molybdenum 42 Mo 95.94	technetium 43 Tc [98]	ruthenium 44 Ru 101.07	rhodium 45 Rh 102.91	palladium 46 Pd 106.42	silver 47 Ag 107.87	cadmium 48 Cd 112.41	indium 49 In 114.82	tin 50 Sn 118.71	antimony 51 Sb 121.76	tellurium 52 Te 127.60	iodine 53 I 126.90	xenon 54 Xe 131.29	
cesium 55 Cs 132.91	barium 56 Ba 137.33	* 57-70	lanthanum 57 La 138.91	hafnium 72 Hf 178.49	tantalum 73 Ta 180.95	wolfram 74 W 183.84	reuterium 75 Re 186.21	osmium 76 Os 190.23	iridium 77 Ir 192.22	platinum 78 Pt 195.08	gold 79 Au 196.97	mercury 80 Hg 200.59	thallium 81 Tl 204.38	lead 82 Pb 207.2	bismuth 83 Bi 208.98	polonium 84 Po [209]	astatine 85 At [210]	radon 86 Rn [222]
francium 87 Fr [223]	radium 88 Ra [226]	* 89-102	actinium 89 Ac [227]	rutherfordium 103 Rf [261]	bohrium 104 Bh [264]	hassium 105 Hs [265]	meitnerium 106 Mt [266]	darmstadtium 107 Ds [271]	roentgenium 108 Rg [272]	copernicium 109 Cn [285]	nihonium 110 Nh [286]	flerovium 111 Fl [289]	tennessine 112 Ts [289]	unbinilium 113 Uuq [288]				unseptennium 114 Uuq [288]
		* Lanthanide series	lanthanum 57 La 138.91	cerium 58 Ce 140.12	praseodymium 59 Pr 140.91	neodymium 60 Nd 144.24	promethium 61 Pm [145]	samarium 62 Sm 150.36	europium 63 Eu 151.96	gadolinium 64 Gd 157.25	terbium 65 Tb 158.93	dysprosium 66 Dy 162.50	holmium 67 Ho 164.93	erbium 68 Er 167.26	thulium 69 Tm 168.93	ytterbium 70 Yb 173.05		
		** Actinide series	actinium 89 Ac [227]	thorium 90 Th 232.04	protactinium 91 Pa 231.04	uranium 92 U 238.03	neptunium 93 Np [237]	plutonium 94 Pu [244]	americium 95 Am [243]	curium 96 Cm [247]	berkelium 97 Bk [247]	californium 98 Cf [251]	esboium 99 Es [252]	fremium 100 Fm [257]	mendelevium 101 Md [258]	nobelium 102 No [259]		

Figure 1.17: Periodic table with "unsustainable" elements highlighted in red.

Figure 1.18: Skeletal structures for Levulinic Acid and γ -Valerolactone

The literature surrounding biomass conversion as a whole is rich with systems with supported metal catalysts suspended in an aqueous solution ((e.g. H_2O)[42–47] which are shown to be highly active for the reforming of biomass, but the cost of these bio-derived molecules is held high due to the large cost of the metals used in these catalysts. When it comes to the conversion of LA to gVL there is a long history of using supported metal catalysts, with papers from the 1930's[48] and the 1940's[49] presenting high levels of conversion and selectivity using platinum oxide and chromium-copper based catalysts respectively, although under undesirable conditions. In more recent times other metals have been studied such as Ru[50], Pt[51], and Re,[52] but these all carry the same trend of using expensive unsustainable metals. Eventually it was shown that using Cu, a more sustainable metal, it was possible to perform the hydrogenation reaction[53]. The two systems presented

were Cu–ZrO₂ and Cu–Al₂O₃, and the reaction was performed in both water and methanol, using LA and the methylated form of LA, Methyl Levulinate (ML).

Catalysts	Substrate	Solvent	Conversion, %	Selectivity, %			Metal leaching (ppm)
				GVL	Me-LA	4-hydroxy Me-LA	
Cu–ZrO ₂	Levulinic acid	Water	100	>99.9	0.01	0.01	34
	Levulinic acid	Methanol	100	90	2	8	2
	Methyl levulinate	Methanol	95	92	SM	8	ND
	Methyl levulinate ^b	Methanol	81	79	SM	21	ND
Cu–Al ₂ O ₃	Levulinic acid	Water	100	>99.9	0.01	0.01	174
	Levulinic acid	Methanol	100	86	4	10	31
	Methyl levulinate	Methanol	93	88	SM	12	ND

^a Reaction conditions: LA, MeLA, (5% w/w); temperature, 473 K; H₂ pressure, 500 psi; catalyst, 0.5 g; reaction time, 5 h. ^b Reaction conditions: methyl LA, (20% w/w); temperature, 473 K; H₂ pressure, 500 psi; catalyst, 0.5 g; reaction time, 5 h.

Figure 1.19: Table for conversion of LA and ML to gVL in both methanol and water taken from [53]

Figure 1.19 shows that when the reaction is performed in water at 200°C and at 35 bar, the selectivity to gVL is almost 100%, and after 5 hours the conversion also reached 100%, when the reaction was repeated in methanol, the selectivity dropped as there was conversion from LA to ML, and other side products were seen. It was also seen that when the reaction was performed in water there was a high level of leaching of the Cu metal from the catalyst, which is undesirable. The conversion of the ML in methanol was found to be lower than the LA, but with a slightly higher selectivity, the main benefit to performing the reaction in methanol was the the level of leeching was massively reduced.

A reaction scheme was also proposed (figure 1.20), with the LA in water being hydrogenated and then cyclises by means of loss of water to form the gVL, and when in methanol a transestrification step occurs, forming the ML, which then is hydrogenated and cyclises by means of loss of methanol. The Cu–ZrO₂ catalyst discussed in the paper formed the basis on which the catalyst that was analysed as part of this thesis was developed. In order to understand the catalyst and the reaction mechanism, a computational studied was performed simultaneously with the design of the catalyst material.

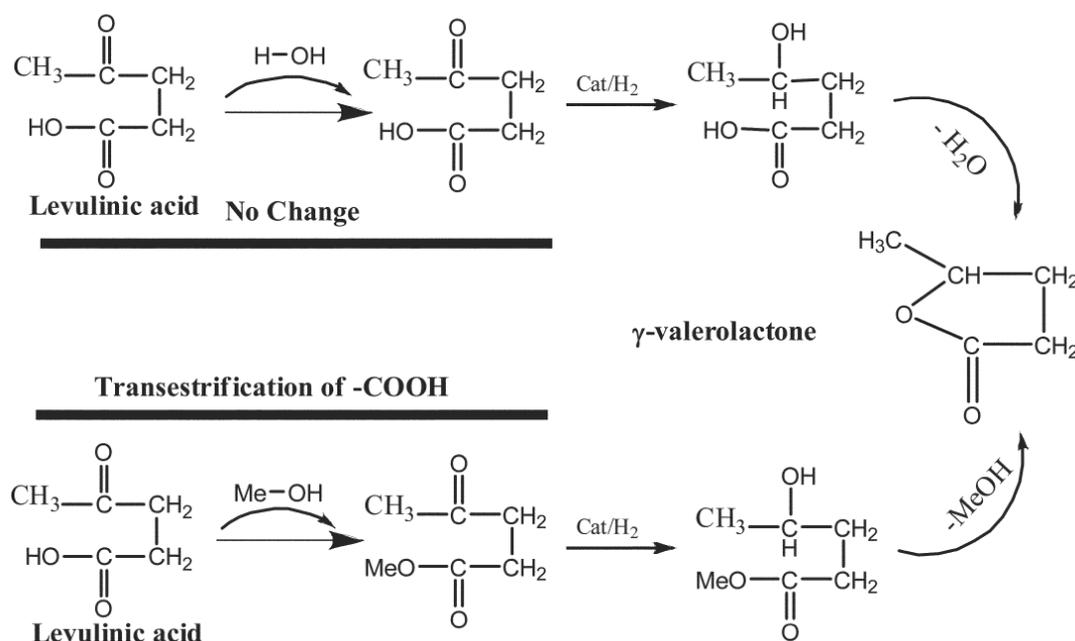


Figure 1.20: Catalytic hydrogenation of LA in presence of water and methanol taken from [53]

1.3.1 Computational studies of biomass derived molecules

Computational simulations are often used for understanding the conversion of biomass, and biomass derived molecules. There are a number of possible methods and techniques available in computational chemistry that can be applied to bio molecule systems, and they have been outlined in the subsequent chapter. This section is dedicated to outlining the techniques currently used in the analysis of biomass derived molecular systems, and their benefits and shortfalls.

The first method discussed is the most simple and therefore commonly used method of analysing reactions and reaction pathways - simple *gas-phase* calculations. This is when specific molecular structures are optimised using the desired level of theory, and the energies reported. To perform this analysis, first a system is selected, and a desired property outlined, for example calculating the barriers to the dehydration of glycerol.[54] Using a combination of experimental data (*e.g.* mass spectrometry), transition state searching tools, and estimation a reaction mechanism can be developed.

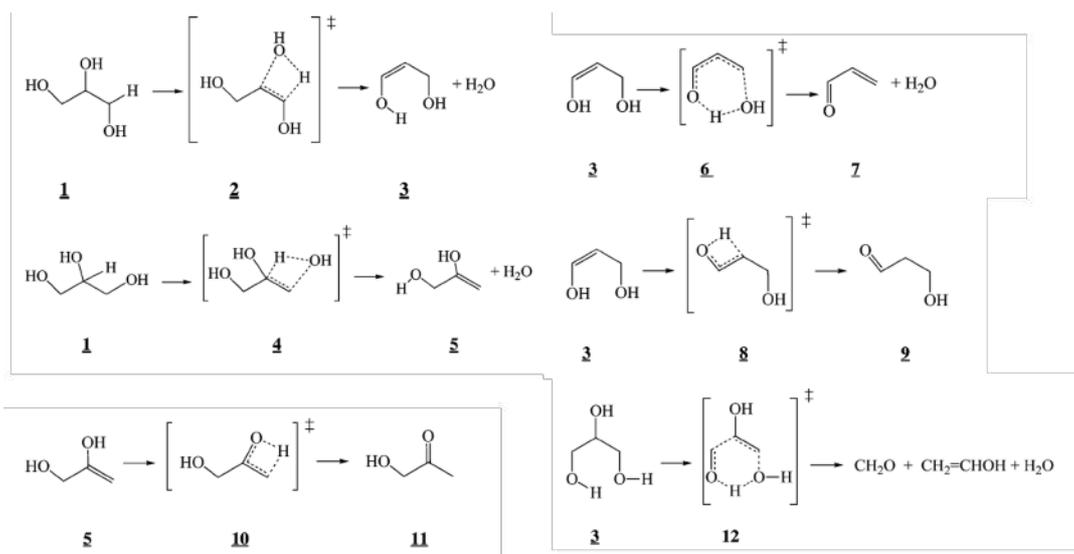


Figure 1.21: Proposed reaction pathways for reaction of neutral glycerol taken from [54]

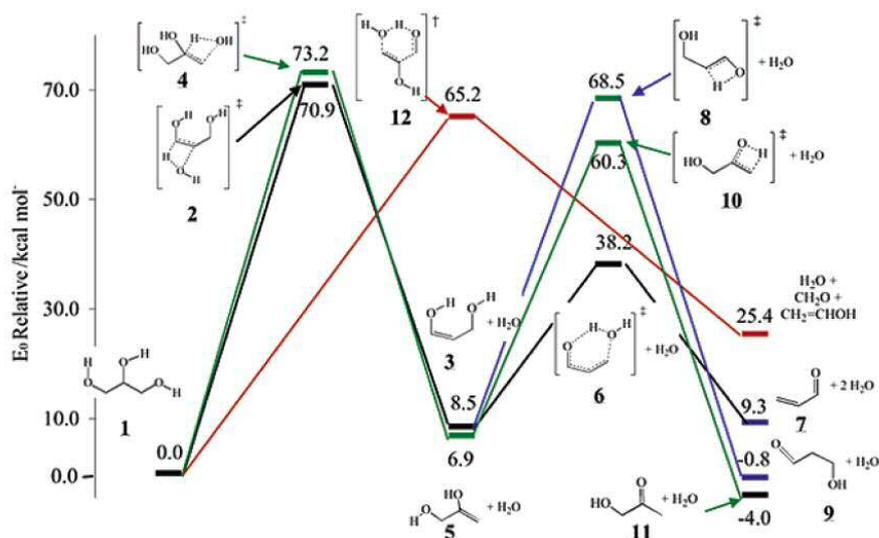


Figure 1.22: Potential energy plot for the reaction of neutral glycerol taken from [54]

Once the various molecules involved in the reaction mechanism have been set up, the structure is optimised and the energy, in this particular case using the Gaussian 03 code with a B3LYP/6-311G-(d,p) basis set. This allows a reaction profile (figure 1.22) to be developed. This reaction profile can be then be used to understand the energetics of the reaction and is very useful in understanding which pathway is more likely.

While this method can be very useful, in particular when performing gas phase

reactions, it can be found that when the reaction is performed in an aqueous medium these calculations fall short.

One example of how these reactions fall short is when the energies of the various isomers of α - and β -D-glucopyranose and 1,5-anhydro-D-glucitol were calculated[55]. In this particular case, the structures various isomers were known and their proportions when found in solution were known from experiments, and the theory was used to try and understand why these proportions occur.

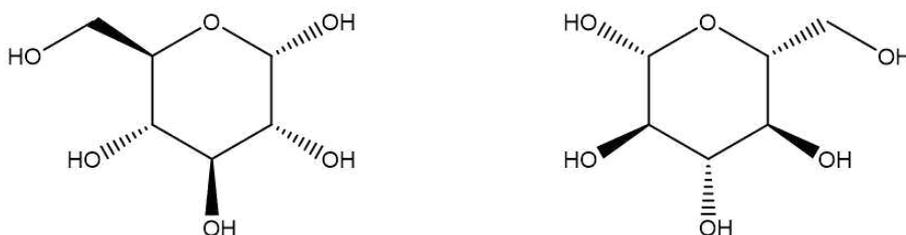


Figure 1.23: Skeletal structures of α - (left) and β - (right) D-glucopyranose

The calculations were performed using the Parallel Quantum Solutions software, using a B3LYP/6-31+G basis set. The various structures were optimised and the free energies of each of the conformations were calculated at $T = 298$ K. Using the ratio of the free energies, the expected ratio of α to β isomers was calculated. It was found that the predicted ratio of the α to β conformations was 63 : 37, whereas the experimental value was found to be 36 : 64. This drastic shift in the calculated and experimental values indicated that there were major problems with the model used to simulate the isomers. As these calculations were performed on gas phase molecules, and the experimental measurements were performed in the aqueous phase, it became clear that the interactions with the solvent (in this case water) were having a large effect on the energies of these isomers, and therefore the gas phase calculations (which contain no solvent interaction) are not an accurate representation of the system.

In order to counteract the errors relating to the interaction with the solvent, there are two main methods that can be undertaken, the first is to apply an implicit solvation model, in which the molecule is placed in a homogeneous polarisable con-

tinuum, the second is an explicit solvation model, in which the solvent is explicitly modelled as part of the simulation. Both methods have been shown to accurately recreate experimental systems[56] It is generally considered that a full explicit solvation model is the best analogue for the real experimental system, but as the number of atoms in the simulation is greatly increased, and as it often requires molecular dynamics, this can lead to a drastic increase in computational cost. The implicit solvation model is a good compromise for including the effect of solvation of the system, but the accuracy of the system is heavily based on the accuracy of the solvation model, which can sometimes be difficult to get correct.

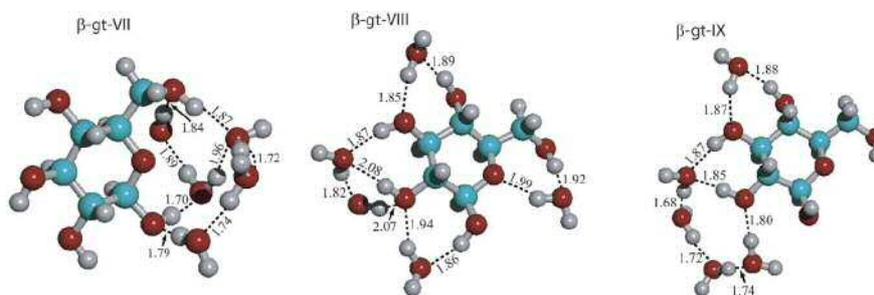


Figure 1.24: Microsolvation of α - (left) and β - (right) D-glucopyranose, taken from [57]

Regarding the isomers of α - and β -D-glucopyranose a further paper was published[57] in which the solvent effects were accounted for using an explicit solvation model. The explicit solvation model used in the paper is more accurately described as a *microsolvation* model, as the full solvent has not been modelled, only a small contingency of 5 water molecules (figure 1.24). For systems where only the first order interactions with the solvent are important this is often accurate enough to replicate experimental results, and it was found in this paper that instead of the 63 : 37 ratio of α to β seen with the gas phase molecules, the in the presence of water was 32 : 68, recreating the experimentally calculated ratio of 36 : 64.

While full system replication (explicit solvent, catalyst surface, reactant), is the end goal of computational simulations, unfortunately computing power has not reached a point at which it is viable for anything other than the simplest of systems. Therefore usually a combination of gas phase and liquid phase calculations

are performed.

1.3.2 Modelling the LA system

There have been a few studies on understanding the mechanism for LA to gVL conversion, but the basis for this work was a paper which studied the influence of water on the hydrogenation reaction of LA to gVL over a Ru catalyst[58]. Similar to the paper which studied the glucopyranose, a microsolvation model was used for modelling the interactions with the solvent, as it was stated that an implicit model does not grasp the effect of hydrogen bonding well. In order to model the reaction, acetone was used as an analogue for the LA as it was more simple to work with, and it was found that they had similar trends on metal and solvent changes. In order to include the interaction with the Ru metal, a slab of Ru(0001) was used as an analogue of the metal surface.

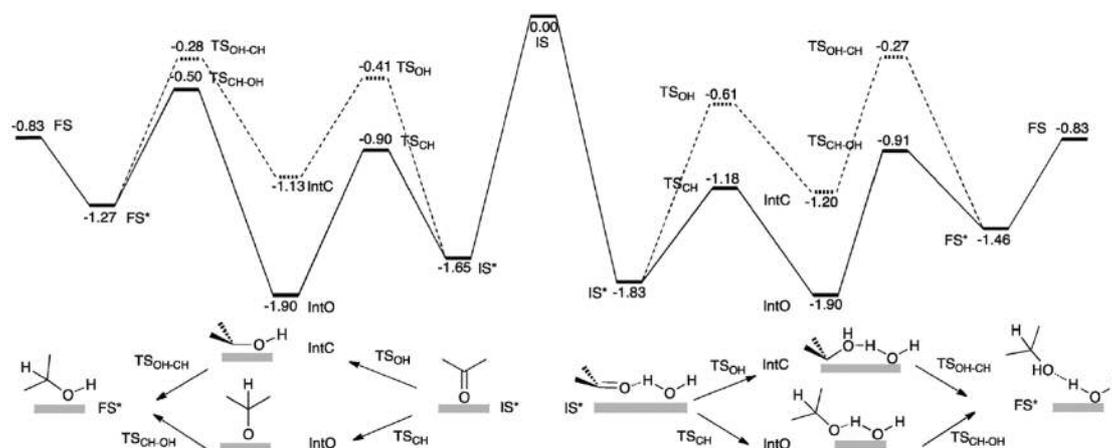


Figure 1.25: Reaction paths in eV for the hydrogenation of acetone on Ru(0001) in the absence (left) and in the presence of a water molecule (right). Alkyl path is the dashed line, and alkoxy path is the solid like. Taken from [58]

In order to gauge the efficiency of the catalyst, the energetic span δE was used as a comparison. The energetic span is defined as the difference in energy between the transition state (TDTS) and reaction intermediate (TDI):

$$\delta E = E_{TDTS} - E_{TDI} : \text{If TDTS appears after I}$$

$$\delta E = E_{TDTS} - E_{TDI} + \Delta G_r : \text{If TDTS appears before TDI}$$

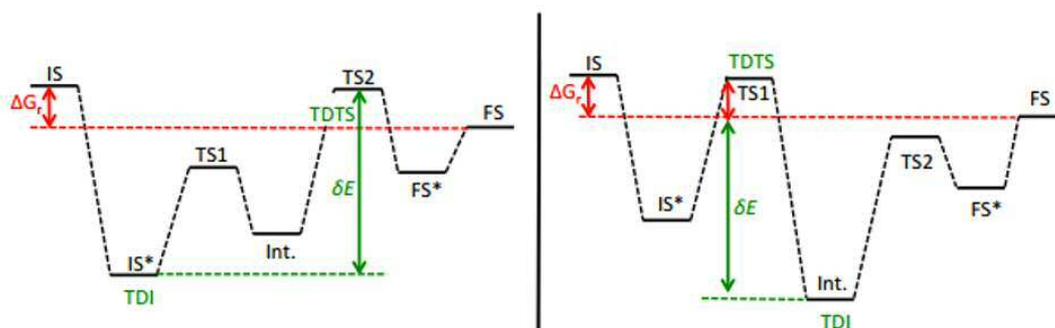


Figure 1.26: Two different methods of calculating the δE taken from [58]

It was found in figure 1.25 that when there was no water present, the alkyl route was more favourable with an energetic span of 1.37eV, but when water was included in the model, the alkoxy route became more favourable with an energetic span of 0.99eV. This demonstrates that in this particular reaction that including water has a drastic effect on the calculated reaction mechanism. It was also stated that when the solvation model was increased in complexity to include 3 water molecules, and then 11 water molecules energetic span was unchanged (0.99eV for 1 water molecule, 1.03eV for 11 water molecules). Afterwards their technique was repeated using a number of different metals (see figure 1.27) and it was found that Cu would also be highly active for this reaction, with a energetic span of 0.87eV.

Using the information gathered on the literature relating to modelling of aqueous systems, it was found that for the system being studied as part of this thesis (LA to gVL using Cu–ZrO₂) the inclusion of the solvent molecules in any model would be very important.

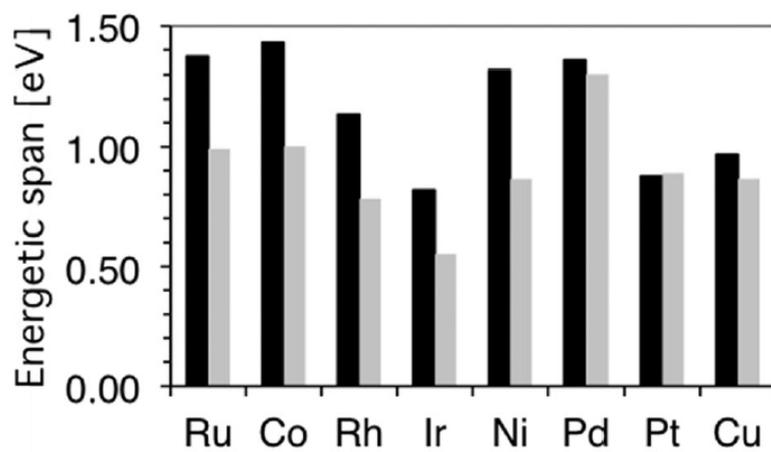


Figure 1.27: Energetic spans for various metals for the hydrogenation of acetone taken from [58]

Bibliography

- (1) J. G. de Vries and S. D. Jackson, *Catalysis Science & Technology*, 2012, **2**, 2009–2009.
- (2) V. Smil, *Nature*, 1999, **400**, 415–415.
- (3) N. Fischer, H. G. Manyar and A. Roldan, *Chemical Communications*, 2016, **52**, 8335–8341.
- (4) J. Hanson and P. Norby, *In-situ Characterization of Heterogeneous Catalysts*, 2013, 121–146.
- (5) T. Carstens, A. Ispas, N. Borisenko, R. Atkin, A. Bund and F. Endres, *Electrochimica Acta*, 2016, **197**, 374–387.
- (6) A. Gorban and G. Yablonsky, *Mathematical Modelling of Natural Phenomena*, 2015, **10**, 1–5.
- (7) J. Šestak, *Talanta*, 1966, **13**, 567–579.
- (8) J. H. Flynn and L. A. Wall, *Journal of Polymer Science Part B: Polymer Letters*, 1966, **4**, 323–328.
- (9) J. Šestak, *Silikaty (Prague)*, 1967, **11**, 153–90.
- (10) J. Šestak, *Thermal analysis: their measurements and theoretical thermal analysis. Part D, Thermophysical properties of solids*, Elsevier, 1984, vol. 12.
- (11) S. Vyazovkin, A. K. Burnham, J. M. Criado, L. A. Perez-Maqueda, C. Popescu and N. Sbirrazzuoli, *Thermochimica Acta*, 2011, **520**, 1–19.

- (12) S. Ponce, M. Pena and J. Fierro, *Applied Catalysis B: Environmental*, 2000, **24**, 193–205.
- (13) K. Zhou, X. Wang, X. Sun, Q. Peng and Y. Li, *Journal of Catalysis*, 2005, **229**, 206–212.
- (14) J. M. Kanervo and A. O. I. Krause, *The Journal of Physical Chemistry B*, 2001, **105**, 9778–9784.
- (15) B. Jankovic, B. Adnadevic and S. Mentus, *Chemical Engineering Science*, 2008, **63**, 567–575.
- (16) G. Munteanu, L. Ilieva, R. Nedyalkova and D. Andreeva, *Applied Catalysis A: General*, 2004, **277**, 31–40.
- (17) J. Gleaves, J. Ebner and T. Kuechler, *Catalysis Reviews Science and Engineering*, 1988, **30**, 49–116.
- (18) J. T. Gleaves, G. Yablonsky, X. Zheng, R. Fushimi and P. L. Mills, *Journal of Molecular Catalysis A: Chemical*, 2010, **315**, 108–134.
- (19) G. Marin and G. S. Yablonsky, *Kinetics of chemical reactions*, John Wiley & Sons, 2011.
- (20) J. R. Welty, C. E. Wicks, G. Rorrer and R. E. Wilson, *Fundamentals of momentum, heat, and mass transfer*, John Wiley & Sons, 2009.
- (21) G. Yablonsky, M. Olea and G. Marin, *Journal of Catalysis*, 2003, **216**, 120–134.
- (22) E. A. Redekop, G. S. Yablonsky, D. Constales, P. A. Ramachandran, C. Pherigo and J. T. Gleaves, *Chemical engineering science*, 2011, **66**, 6441–6452.
- (23) S. O. Shekhtman, G. S. Yablonsky, J. T. Gleaves and R. Fushimi, *Chemical engineering science*, 2003, **58**, 4843–4859.
- (24) S. O. Shekhtman, G. S. Yablonsky, J. T. Gleaves and R. R. Fushimi, *Chemical engineering science*, 2004, **59**, 5493–5500.

- (25) G. Marin, R. van Santen, J. T. Gleaves, G. S. Yablonskii, P. Phanawadee and Y. Schuurman, *Applied Catalysis A: General*, 1997, **160**, 55–88.
- (26) D. Constales, G. Yablonsky, G. Marin and J. Gleaves, *Chemical Engineering Science*, 2001, **56**, 1913–1923.
- (27) D. Constales, S. Shekhtman, G. Yablonsky, G. Marin and J. Gleaves, *Chemical Engineering Science*, 2006, **61**, 1878–1891.
- (28) D. Constales, G. Yablonsky, G. Marin and J. Gleaves, *Chemical Engineering Science*, 2001, **56**, 133–149.
- (29) D. Constales, G. Yablonsky, G. Marin and J. Gleaves, *Chemical Engineering Science*, 2004, **59**, 3725–3736.
- (30) E. Kondratenko, O. Buyevskaya and M. Baerns, *Journal of Molecular Catalysis A: Chemical*, 2000, **158**, 199–208.
- (31) O. Buyevskaya, M. Rothaemel, H. Zanthoff and M. Baerns, *Journal of Catalysis*, 1994, **146**, 346–357.
- (32) R. Fushimi, J. T. Gleaves, G. Yablonsky, A. Gaffney, M. Clark and S. Han, *Catalysis Today*, 2007, **121**, 170–186.
- (33) R. Fushimi, S. O. Shekhtman, A. Gaffney, S. Han, G. S. Yablonsky and J. T. Gleaves, *Industrial & engineering chemistry research*, 2005, **44**, 6310–6319.
- (34) U. Rodemerck, B. Kubias, H.-W. Zanthoff, G.-U. Wolf and M. Baerns, *Applied Catalysis A: General*, 1997, **153**, 217–231.
- (35) K. Alexopoulos, M.-F. Reyniers and G. B. Marin, *Journal of Catalysis*, 2012, **289**, 127–139.
- (36) S. Shekhtman, A. Goguet, R. Burch, C. Hardacre and N. Maguire, *Journal of Catalysis*, 2008, **253**, 303–311.
- (37) E.-E. Commission et al., *Report of the ad-hoc working group on defining critical raw materials. Brussels*, 2010.
- (38) M. J. Climent, A. Corma and S. Iborra, *Green chemistry*, 2011, **13**, 520–540.

- (39) G. W. Huber, S. Iborra and A. Corma, *Chemical reviews*, 2006, **106**, 4044–4098.
- (40) J.-P. Lange, J. Z. Vestering and R. J. Haan, *Chemical Communications*, 2007, 3488–3490.
- (41) I. T. Horváth, H. Mehdi, V. Fábos, L. Boda and L. T. Mika, *Green Chemistry*, 2008, **10**, 238–242.
- (42) J. Shabaker, G. Huber and J. Dumesic, *Journal of Catalysis*, 2004, **222**, 180–191.
- (43) J. Shabaker, G. Huber, R. Davda, R. Cortright and J. Dumesic, *Catalysis Letters*, 2003, **88**, 1–8.
- (44) G. W. Huber, J. W. Shabaker, S. T. Evans and J. A. Dumesic, *Applied Catalysis B: Environmental*, 2006, **62**, 226–235.
- (45) T. Van Haasterecht, C. Ludding, K. De Jong and J. Bitter, *Journal of catalysis*, 2014, **319**, 27–35.
- (46) Z. Wu, S. Ge, C. Ren, M. Zhang, A. Yip and C. Xu, *Green Chemistry*, 2012, **14**, 3336–3343.
- (47) Y. Wang, S. Van de Vyver, K. K. Sharma and Y. Román-Leshkov, *Green Chemistry*, 2014, **16**, 719–726.
- (48) H. Schuette and R. W. Thomas, *Journal of the American Chemical Society*, 1930, **52**, 3010–3012.
- (49) R. V. Christian Jr, H. D. Brown and R. Hixon, *Journal of the American Chemical Society*, 1947, **69**, 1961–1963.
- (50) Z.-p. Yan, L. Lin and S. Liu, *Energy & Fuels*, 2009, **23**, 3853–3858.
- (51) J.-P. Lange, R. Price, P. M. Ayoub, J. Louis, L. Petrus, L. Clarke and H. Gosselink, *Angewandte Chemie International Edition*, 2010, **49**, 4479–4483.
- (52) D. J. Braden, C. A. Henao, J. Heltzel, C. C. Maravelias and J. A. Dumesic, *Green chemistry*, 2011, **13**, 1755–1765.

- (53) A. M. Hengne and C. V. Rode, *Green Chemistry*, 2012, **14**, 1064–1072.
- (54) M. R. Nimlos, S. J. Blanksby, X. Qian, M. E. Himmel and D. K. Johnson, *The Journal of Physical Chemistry A*, 2006, **110**, 6145–6156.
- (55) M. Appell, G. Strati, J. Willett and F. Momany, *Carbohydrate research*, 2004, **339**, 537–551.
- (56) R. Skyner, J. McDonagh, C. Groom, T. van Mourik and J. Mitchell, *Physical Chemistry Chemical Physics*, 2015, **17**, 6174–6191.
- (57) F. A. Momany, M. Appell, J. Willett and W. B. Bosma, *Carbohydrate research*, 2005, **340**, 1638–1655.
- (58) C. Michel, J. Zaffran, A. M. Ruppert, J. Matras-Michalska, M. Jędrzejczyk, J. Grams and P. Sautet, *Chemical Communications*, 2014, **50**, 12450–12453.

2 | Methodology

In this section the programming environment in which all of the analysis software was built will be discussed, then the underlying theory behind Temperature Programmed and Temporal Analysis of Products reactions, and an explanation of the basics of Density Functional Theory will follow. These three topics will be essential for understanding the further work being discussed in this thesis.

2.1 MATLAB Programming Environment

In order to develop software for data analysis, the first, and sometimes most difficult choice, is to decide which programming language to use, and for this project the multi-paradigm numerical computing environment MATLAB[1] was chosen. MATLAB is considered to be a high level programming language, which means that there is a large degree of abstraction away from the internal computing code. This means that the language is much more user friendly, providing many benefits to using MATLAB over other more common programming environments (*e.g.* Python, C, FORTRAN). Alongside the standard MATLAB package a number extra toolboxes were used throughout the code.

- Global Optimisation
- Curve Fitting

- Signal Analysis
- Optimisation
- Application Compiler

The MATLAB code is built around the MATLAB scripting language. It is highly interactive, meaning it is very easy to perform simple mathematics for example:

```
1 >> x = 3;
2 >> y = 4;
3 >> x + y
```

would output:

```
1 ans =
2
3     7
```

where MATLAB really excels is in its handling of vectors and matrices, through simple commands it is easy to generate a matrix of elements and quickly perform any function on that matrix:

```
1 >> A = [16 3 2 13; 5 10 11 8; 9 6 7 12; 4 15 14 1]
2
3 A =
4
5     16     3     2    13
6     5    10    11     8
7     9     6     7    12
8     4    15    14     1
9
10 >> sqrt(A)
11
12 ans =
13
14     4.0000     1.7321     1.4142     3.6056
15     2.2361     3.1623     3.3166     2.8284
16     3.0000     2.4495     2.6458     3.4641
17     2.0000     3.8730     3.7417     1.0000
```

Finally the data which has been generated can be easily saved to a structure.

```
1 >> sStructData = struct(...
2     'peak', peak, ...
3     'data', data, ...
4     'time', time, ...
```

```

5     'peakno', peakno, ...
6     'tpulse', tpulse, ...
7     'delay', delay, ...
8     'number', number, ...
9     'averagepeak', averagepeak, ...
10    'timesingle', timesingle, ...
11    MW, MW, ...
12    'T', T);
13
14 >> sStructData.MW
15
16 ans =
17
18     40

```

Using these methods, it is very simple to generate functions and pass around variables, the code below provides an example of solving a differential equation numerically, and how a function is structured. The actual use of the function will be discussed further on in this thesis in section 2.2.

```

1 function [F, alphTPR] = TPR_function(A, Ea, m, n, g, Temp, beta)
2
3 % Creates time variable
4 betas = beta/60;
5 T     = Temp;
6 t     = Temp./betas;
7
8 % First alpha value cannot be 0 to avoid NaN
9 alphTPR = zeros(1, length(T));
10 F       = zeros(1, length(T));
11 alphTPR(1) = 0.00001;
12 dt      = diff(t);
13
14 % Calculates alpha numerically
15 for i = 2:length(T)
16     % Include Ea correction
17     Eag = (Ea*(1+alphTPR(i-1)*g));
18     % Performs calculation using SB equation
19     alphTPR(i) = alphTPR(i-1)+((A)*exp(-Eag/(8.314*T(i)))*...
20         ((alphTPR(i-1)^m)*((1-alphTPR(i-1))^n))*dt(i-1));
21     if alphTPR(i) >= 0.99999
22         alphTPR(i:end) = 1;
23         break
24     end
25 end
26
27 % Calculates dalpha by dt
28 F(2:end) = diff(alphTPR)./diff(T);
29 end

```

It is this ability to quickly write code in a simple and human readable manner which facilitated the development of the algorithms required for analysis of TPR and TAP experiments.

2.1.1 Generation of a Graphical User Interface in MATLAB

For the software to have a larger impact there was a shift to a more "front-end" focused development. This means that the mathematics is hidden away from the user, and the majority of it is performed by simple button presses on a GUI. The reasoning behind this was that then anyone, with any level of knowledge in kinetic modelling could pick up the software and perform their kinetic analysis with ease. To facilitate the creation of this front-end focused GUI the MATLAB, Graphical User Interface Development Environment (GUIDE) was used. GUIDE is a tool used to design user interfaces for custom applications, it provides a graphical editor for creating a GUI as seen in figure 2.1, and then automatically generates the required MATLAB code for constructing the user interface. Once the interface has been created it is simple to slot in the required functions for the various interactive sections of the user interface.

The graphical user interface designed by GUIDE is used as the heart of the software. It is the space where all of the variables and structures are stored for use in the primary and secondary functions. The flow between the central GUI and the various functions is outlined in figure 2.2. Once the interface has been set up properly and the functions finalised, it is then compiled using the Application compiler toolbox in MATLAB. The compiled software is then distributed to the experimental groups, allowing them to drastically increase the quality of their own research by performing the kinetic analysis outlined in the next sections.

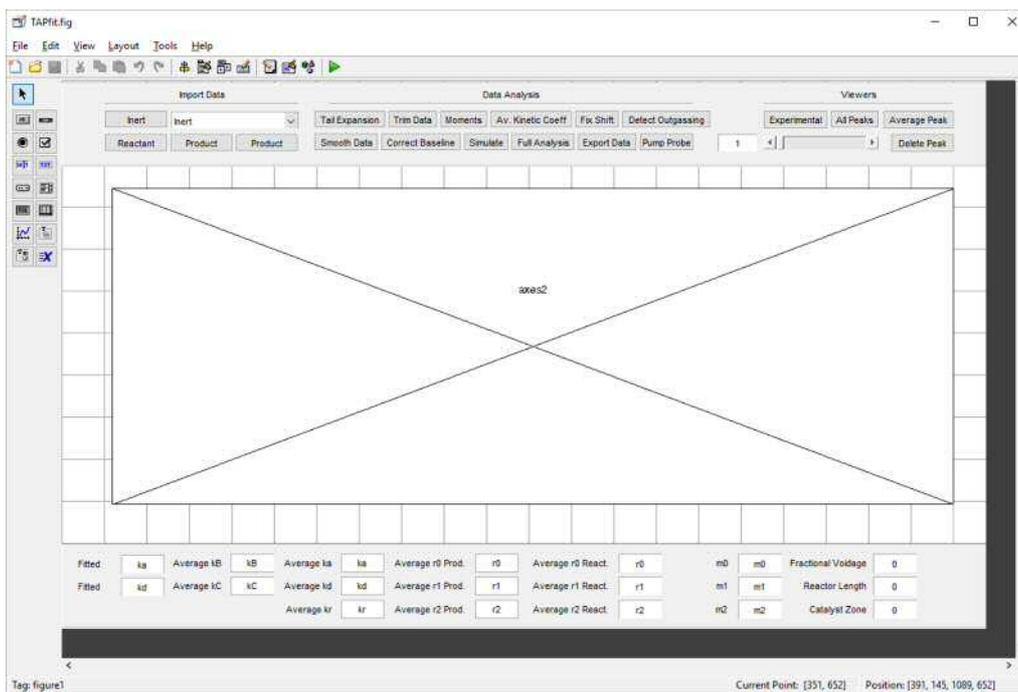


Figure 2.1: The GUIDE layout editor

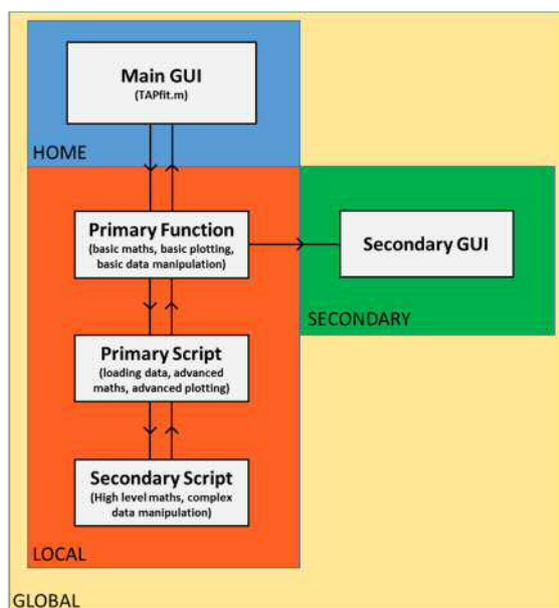


Figure 2.2: Flow of variables in CCI-TPR and CCI-TAP

2.1.2 Regression Analysis in MATLAB

When performing the kinetic modelling of TPR experiments, it is quite common to use a regression method in order to calculate the various kinetic parameters. In order to facilitate this a number of scripts were implemented into the CCI-TAP and

CCI-TPR programs. These scripts work by first setting up the variables, then using pre-built functions in the MATLAB global optimisation and optimisation toolboxes in order to vary the kinetic parameters until a minimum, as defined by some function, is reached.

To calculate the minima where the simulated and experimental responses overlapped, a sum of vectors method was used:

$$\overrightarrow{AB} = \left| \sqrt{(x_{exp} - x_{sim})^2 + (y_{exp} - y_{sim})^2} \right| \quad (2.1)$$

in which x is the value of the simulated and experimental parameters on the x axis (*i.e.* time and temperature), and y is the simulated variable (*e.g.* TAP exit flow). The magnitude of the vectors is then summed over the total number of elements in the simulated and experimental arrays:

$$S = \sum_{i=0}^n \overrightarrow{AB}_i \quad (2.2)$$

where n is the is an index for the element in the array AB , and S is the sum of the array, and is the variable to be minimised by the function. To minimise the variable S the pre-built MATLAB *fmincon* is used. The function *fmincon* is a nonlinear constrained multi-variable function solver, this means it can take an input matrix of variables, and it will vary them within some pre-defined constraints in order to minimise the value of S . The algorithm used to minimise the value of S is called an Interior Point Algorithm as described in [2]. While *fmincon* is a very quick algorithm for performing minimisation of a function, it is not a global solver, this means it can easily fall into the trap of a local minima, giving results which may not truly reflect the kinetics of the experimental response. To counteract this a global optimisation algorithm[3] using the *GlobalSearch* is wrapped around the *fmincon* function, to ensure that a global solution has been reached. The *GlobalSearch* algorithm works by first running the *fmincon* function using the initial input parameters. Once a

minimum has been reached, it uses a scatter search algorithm[4] in order to generate a number of trial points within the bounds set in the initial *fmincon* function. The *GlobalSearch* function then evaluates the function value at each of the trial points, and then takes the function with the best scores and re-runs *fmincon* at that specific point, and then removes those generated points from the algorithm. Once the *fmincon* function converges using the trial point a vector is generated between the two function values (for the initial guess and the trial guess), it then uses this vector in order to decide whether or not the function is in a local or global minima. This cycle is then repeated until the conditions for a global minima are set. An example of how to setup the global optimisation search is shown below.

```

1 ...
2 for i = 1:numvars
3     % Creates input variables matrix
4     vars(i,:) = [A(i),Ea(i),m(i),n(i),g(i),p(i)];
5     % Creates lower and upper bounds matrices
6     LB(i,:) = [0.0,0.2,0,0.3,0.0,0.0];
7     UB(i,:) = [3.5,12,1,3.0,0.5,1.0];
8 end
9
10 % Defines the problems structure for use in GlobalSearch
11 problem = createOptimProblem('fmincon',...
12     'objective',@(vars) SumSquareEa(vars,AA(1,:),TT,Temp,GF,B),...
13     'x0',vars,'lb',LB,'ub',UB,'Aeq',[],'Aineq',[],'beq',[],'bineq',[],...
14     'options',...
15     optimoptions(@fmincon,'Display','iter','MaxIter',500));
16
17 % Creates the GlobalSearch object
18 gs = GlobalSearch('Display','iter');
19
20 % Runs the GlobalSearch object using the parameters defined using problem
21 [variables,~] = run(gs,problem);
22 ...

```

2.2 Temperature Programmed Reactions

A temperature programmed reaction is considered to be any experiment during which the temperature is controlled by some external factor and monitored as a function of the reaction. The particular experiment of interest in this thesis is

Temperature Programmed Reduction. During a TPR experiment the temperature is heated at a linear heating rate and a reducing gas is flowed over the surface of the material of interest as outlined in section 1.1. This section will be used to outline the underlying theory behind TPR experiments, and how it can be used to help explain the kinetics of complex reduction reactions.

2.2.1 Basics of Temperature Programmed Experiments

Kinetics is defined as the measurement of the rate of change of a process. In TPR experiments this is defined using:

$$\frac{d\alpha}{dt} = k(T) f(\alpha) h(P) \quad (2.3)$$

Equation 2.3 relates the rate of change of the degree of conversion, α , as a function of time t . The degree of conversion is a dimensionless variable which ranges from 0 to 1 and is defined as degree of the process being measured (with 0 being before it has started and 1 being at the end). The rate of change of α as a function of t is related to three functions: $k(T)$ which is some function of Temperature, $f(\alpha)$ which is some function of degree of reduction, and finally $h(P)$ which is a function of the pressure of the reducing gas. In order to simplify the model the majority of kinetic analysis of TPR experiments removes the effect of pressure ($h(P)$). As pressure is considered to be a function of surface coverage of the gas, and as the reducing gas is in excess it can be removed from the equation, although there have been some attempts at including it[5–7]. By removing the effect of pressure this leaves just two variables T and α giving

$$\frac{d\alpha}{dt} = k(T) f(\alpha) \quad (2.4)$$

The first function of interest is $k(T)$, and it is considered that $\frac{d\alpha}{dt}$ has an Arrhenius dependence on temperature for TPR experiments:

$$k(T) = A \exp\left(\frac{-E_a}{RT}\right) \quad (2.5)$$

where E_a is the activation energy of the process, R is the universal gas constant, and T is the temperature. By combining equation 2.4 and equation 2.5 we end up with:

$$\frac{d\alpha}{dt} = A \exp\left(\frac{-E_a}{RT}\right) f(\alpha) \quad (2.6)$$

which is the kinetic equation used to describe TPR thermograms. In TPR experiments the sample is heated at a linear rate, instead of considering α as a function of time, it is considered as a function of temperature instead by including the heating rate which is defined using the variable β :

$$\beta = \frac{dT}{dt} \quad (2.7)$$

this can then be combined with equation 2.3 to give

$$\frac{d\alpha}{dt} = \beta \frac{d\alpha}{dT} = A \exp\left(\frac{-E_a}{RT}\right) f(\alpha) \quad (2.8)$$

and it is using this form of the kinetic equation that the TPR analysis is performed, as the majority of experimental data is reported as function of temperature.

2.2.2 The Kinetic Model

The effect on the rate of degree of reduction as a function of temperature has been outlined, and the next step is to try and understand the function $f(\alpha)$. We define this as the *kinetic model* and it is used to describe how the reduction rate is influenced by the degree of reduction. There are multiple methods by which materials can reduce, and figure 2.3 gives a visual representation of these methods, for example 3D interface shrinkage in which the reduced material envelops the oxidised material forming a spherical outer layer.

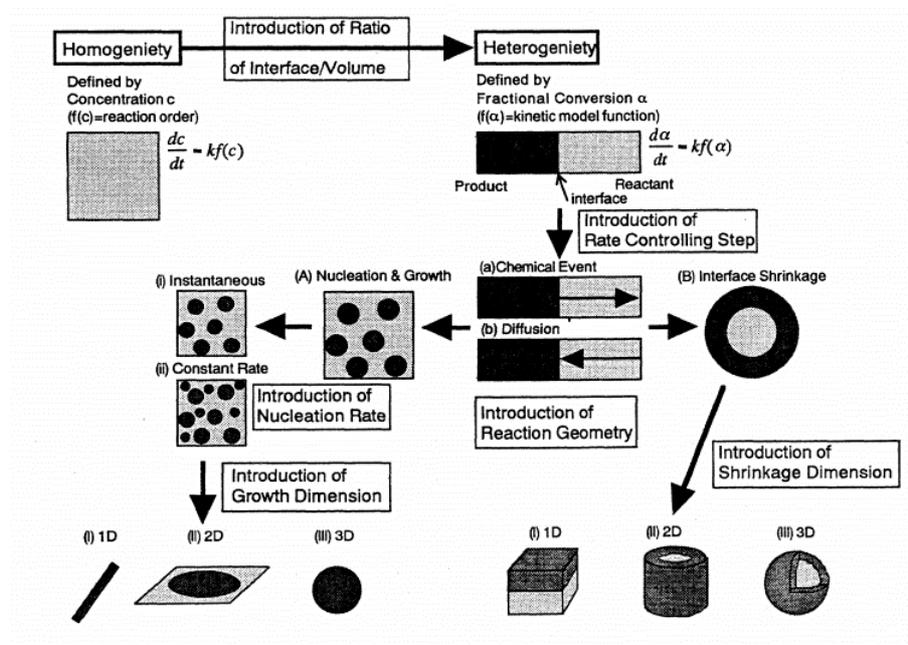


Figure 2.3: Hypothetical reduction mechanisms, taken from [8]

By understanding these different processes of reduction, one can then attempt to model it as a function of α , which gives us multiple reaction models as shown in table 2.1.

These reaction models can be broken down into three basic types: /accelerating, decelerating and autocatalytic (sigmoidal), and are separated by comparing the dependence of α as a function of t or T . The first type of model, accelerating, is characterised by a process in which $d\alpha/dt$ increases as a function of α and reaches

Table 2.1: Different models used for describing solid state reactions with corresponding kinetic models

Reaction Model	$f(\alpha)$
Random Nucleation	$(1 - \alpha)$
Generalised n^{th} order	$(1 - \alpha)^n$
Generalised Avrami-Erofeyev	$n(1 - \alpha) (-\ln(1 - \alpha))^{(n-1)/n}$
Contracting Area	$(1 - \alpha)^{1/2}$
Contracting Volume	$(1 - \alpha)^{1/3}$
One dimensional diffusion	$\alpha/2$
Two dimensional diffusion	$-\ln(1 - \alpha) - 1$
Generalised Sestak Berggren	$\alpha^m (1 - \alpha)^n [-\ln(1 - \alpha)]^p$

a maximum when $\alpha = 0$. The second model, *decelerating*, is considered to be the opposite of the first, during which $d\alpha/dt$ decreases as a function of α with a maximum at $\alpha = 1$. The final model, autocatalytic, is when $d\alpha/dt$ reaches a maximum at $0 < \alpha < 1$. The problem with the majority of these models is that they are derived from first principles, and tend to only work for a single type of reaction mechanism. This means that if you would like to try and perform a curve fitting to some experimental data, you would first need to assign a reaction mechanism to your kinetic model. In order to perform a curve fitting without any prior knowledge of the kinetic model it is possible to use a generalised model, such as the one developed by Sestak and Berggren. [9]

$$\alpha^m (1 - \alpha)^n [-\ln(1 - \alpha)]^p \quad (2.9)$$

By varying the parameters m , n , and p it is possible to perform a fit to almost all reaction model types, encompassing the accelerating, decelerating and autocatalytic reduction mechanisms. The majority of the time the Sestak Berggren is used in its truncated form with $p = 0$ giving:

$$\alpha^m (1 - \alpha)^n \quad (2.10)$$

In this form the Sestak Berggren takes on an autocatalytic shape, but it is possible

to recreate similar shapes to the accelerating reaction type by setting $n = 0$, and decelerating by setting $m = 0$. By using a generalised model this gives more freedom when performing the regression analysis as a pre-selected mechanism does not need to be enforced, but the drawback is that by being a general model there is no *real* meaning to the kinetic parameters.

2.2.3 The Arrhenius Parameters

Alongside $f(\alpha)$ there are two extra parameters in the kinetic equation (2.8) A and E_a . One of the first relationships noted when studying the Arrhenius parameters is that A and E_a are mutually correlated. This correlation arises by first taking the integral form of equation 2.8

$$g(\alpha) = \int_0^\alpha \frac{d\alpha}{f(\alpha)} = \frac{A}{\beta} \int_0^T \exp\left(\frac{-E_a}{RT}\right) dT \quad (2.11)$$

and then also differentiating equation 2.8 with respect to time[10]:

$$\frac{d^2\alpha}{dt^2} = \left[\frac{\beta}{T\pi(x)} \right]^2 f(\alpha) g(\alpha) [f'(\alpha) g(\alpha) + x\pi(x)] \quad (2.12)$$

where x is E_a/RT and $\pi(x)$ is the Arrhenius temperature integral:

$$\pi(x) = \int_0^T \exp\left(\frac{-E_a}{RT}\right) dT \quad (2.13)$$

If we set equation 2.12 to zero then we can find the equation representing the peak of the reduction curve:

$$-f'(\alpha_p)g(\alpha_p) = x_p\pi(x_p) \quad (2.14)$$

With subscript p indicating that it is the value at the peak of the reduction curve.

From substitution of equation 2.11 into equation 2.14 we end up with:

$$-f'(\alpha_p) A \exp\left(\frac{-E_{ap}}{RT_p}\right) = \frac{\beta R x_p^2}{E_a} \quad (2.15)$$

which can then be simplified into its logarithmic form:

$$\ln(A) = a + bE_a \quad (2.16)$$

with:

$$b = \frac{1}{RT_p} \quad a = \ln\left[\frac{-\beta x_p}{T_p f'(\alpha)}\right] \quad (2.17)$$

Equation 2.16 clearly shows that any shift in the activation energy E_a can be accounted for by a shift in the value of A . Therefore that if one wants to perform a regression analysis on a single TPR thermogram it is almost impossible to separate the values of A and E_a , another method is required for calculating the Arrhenius parameters correctly. The first step to trying to deconvolute the Arrhenius parameters is to understand the ideas behind the isoconversional principle. The isoconversional principle states that the reaction rate at a constant degree of conversion is purely a function of the temperature. This is demonstrated by taking the logarithmic form of the derivative of equation 2.4

$$\left[\frac{\delta \ln(d\alpha/dt)}{\delta T^{-1}}\right]_{\alpha} = \left[\frac{\delta \ln k(T)}{\delta T^{-1}}\right]_{\alpha} + \left[\frac{\delta \ln f(\alpha)}{\delta T^{-1}}\right]_{\alpha} \quad (2.18)$$

where the subscript α indicates that the values are taken at a constant degree of conversion. As α is constant, the second term in equation 2.18 is considered to be zero. This then gives

$$\left[\frac{\delta \ln(d\alpha/dt)}{\delta T^{-1}} \right]_{\alpha} = \left[\frac{\delta \ln k(T)}{\delta T^{-1}} \right]_{\alpha} \quad (2.19)$$

It can be seen from equation 2.19 that the temperature dependence of the rate of conversion under isoconversional conditions can be used to calculate the activation energy without any prior knowledge of the kinetic model $f(\alpha)$. Although there is no kinetic model involved in the direct calculation of the activation energy, the results are still a function of the kinetic model which is encapsulated by the $d\alpha/dT$ function, meaning that for the isoconversional method to be valid the experimental thermogram must follow some kinetic model. To get the dependence of the rate of conversion as a function of temperature the most simple method is to perform the reduction reaction at multiple heating rates (β). By using a combination of at least 3 heating rates, the linear relationship between rate of conversion and temperature can be exploited to calculate the activation energy.

The applications of the isoconversional method to calculate activation energy can be split into two categories: integral and differential methods. The integral based methods use equation 2.11 and then applies the isoconversional principle. Equation 2.11 does not have an analytical solution, this means that a number of approximations need to be made in order to evaluate the temperature integral in 2.11. The general idea for the integral methods is based on the Kissinger-Akihara-Sunose (KAS) method [11] in which they split equation 2.11 by separation of variables.

$$g(\alpha) = \int_0^{\alpha} \frac{d\alpha}{f(\alpha)} = \frac{A}{\beta} \int_0^T \exp\left(\frac{-E_a}{RT}\right) dT = \frac{AE_a}{\beta R} \int_{x_{\alpha}}^{\infty} \frac{\exp(-x)}{x^2} dx \quad (2.20)$$

It is stated that as long as one assumes that $x \gg 1$ then the right most integral can be approximated[12] using:

$$\int_{x_{\alpha}}^{\infty} \frac{\exp(-x)}{x^2} dx \cong \frac{\exp(-x)}{x^2} \quad (2.21)$$

From application of equation 2.21 into 2.20 and taking the logarithm this obtains:

$$\ln \left(\frac{\beta}{T_\alpha^B} \right) = \ln \left(\frac{AE_a}{R} \right) + \ln \left(\frac{1}{\beta x_\alpha^2} \right) - x \quad (2.22)$$

then as long as α is kept constant then equation 2.22 can be expressed as:

$$\ln \left(\frac{\beta}{T_\alpha^B} \right) = -C \left(\frac{E_{a\alpha}}{RT_\alpha} \right) + D \quad (2.23)$$

with B , C , and D being various constants which come from the various derivations of the temperature integral in equation 2.21 and assumptions of the kinetic model $f(\alpha)$. The two approximations most commonly used are the Flynn-Wall-Ozawa [13] and the Kissinger [14], which use the values $B = 0$, $C = 1.052$, and $B = 2$ and $C = 1$ respectively. In order to calculate the activation energy, a linear plot of $\ln T_\alpha^B / \beta v^{1/T_\alpha} C$ will give a gradient of E_a/R , when performed at isoconversional conditions. Alongside the linear integral methods it is also possible to apply a nonlinear approach developed by Vyazovkin *et. al*[15], which is considered to be more accurate than the previous linear methods. Using the integral kinetic equation (2.11) it can be assumed from the isoconversional principle that:

$$\frac{A_1}{\beta_1} \pi(x_1) = \frac{A_2}{\beta_2} \pi(x_2) = \dots = \frac{A_n}{\beta_n} \pi(x_n) \quad (2.24)$$

in which the subscript value for n indicates reactions run at different heating rates. It then follows that as long as equation 2.24 is true that:

$$\sum_{i=1}^n \sum_{j \neq i}^n \frac{\beta_j \pi(x_i)}{\beta_i \pi(x_j)} = n(n-1) \quad (2.25)$$

where i and $j = 1 \dots n$ with n being the number of thermograms performed at different heating rates. If one has the correct values for T_α and $E_{a\alpha}$ used in the function $\pi(x)$ (equation 2.13) at isoconversional conditions then from rearrangement of equation

2.25 we should get the relationship:

$$\left| n(n-1) - \sum_{i=1}^n \sum_{j \neq i}^n \frac{\beta_j \pi(x_i)}{\beta_i \pi(x_j)} \right| = \Phi \quad (2.26)$$

where Φ is expected to be zero, but will usually have a small finite value. If the activation energy is not known, then by recording the experimental values for T_α and varying the value of $E_{a\alpha}$ until a minimum value of Φ is reached one can calculate the activation energy of the reduction process. A minimum value of Φ is used instead of zero as quite often there are some experimental errors in the recorded thermogram (usually from the reading of T), but as long as Φ is at a minimum value it can be assumed that the calculated $E_{a\alpha}$ is an accurate representation of the real $E_{a\alpha}$. Now as it can be expected, if there is a complex thermogram with multiple reduction processes occurring, the value of E_a will change as a function of degree of reduction. To counteract this the integral $\pi(x)$ from equation 2.13 is calculated in steps of $d\alpha$.

$$\pi(x_\alpha) = \int_{T_\alpha - \Delta\alpha}^{T_\alpha} \exp\left(\frac{-E_{a\alpha}}{RT}\right) dT \quad (2.27)$$

When it comes to minimising the value of Φ using equation 2.26 calculating the integral in equation 2.27 is computationally expensive, and as such the common method is to approximate the integral using a 4th order polynomial.

$$\pi(x_\alpha) = \frac{\exp(-x_\alpha)}{x_\alpha} \left(\frac{x_\alpha^3 + 18x_\alpha^2 + 86x_\alpha + 96}{x_\alpha^4 + 20x_\alpha^3 + 120x_\alpha^2 + 240x_\alpha + 120} \right) \quad (2.28)$$

Equation 2.28 was developed by Senum and Yang[16], and is known as the Senum Yang approximation. It has been shown that this polynomial predicts the integral equation 2.27 with a very high degree of accuracy, even at low values of x [17], and although there are higher degree polynomials available the increase in accuracy is negligible. When equation 2.28 is used in the minimisation of Φ from equation 2.26 there is a drastic increase in computation speed which increases the viability of the

nonlinear Vyazovkin method, and as the method is considered to be more accurate than the previous linear methods it is considered to be the gold standard for integral based activation energy calculation methods.

The second method for calculating the activation energy is the differential method, with the most popular differential method being that of Friedman. [18] The Friedman method takes the direct logarithm of the kinetic equation 2.8 giving:

$$\ln \left[\left(\frac{d\alpha}{dT} \right)^\beta \right] = \ln[f(\alpha)A_\alpha] - \left(\frac{E_{a\alpha}}{RT_\alpha} \right) \quad (2.29)$$

by performing a linear plot, similar to the kind mentioned in the integral methods, one can calculate the activation energy at each degree of reduction. The main benefit of using a differential method over a integral method is that no assumptions are made when performing the calculation of $E_{a\alpha}$ and as such the value of $E_{a\alpha}$ that is calculated is considered to be the *real* value. The reason why integral methods are still used in conjunction with differential methods is that the differential methods are highly sensitive to any experimental error in the recording of the temperature programmed experiment, as they use the value of $d\alpha/dT$. In the majority of temperature programmed experiments this is not directly recorded in the experiment and needs to be calculated manually in the post experimental analysis, during which the numerical differentiation introduces noise. Alongside the numerical differentiation errors there is also the problem of determining the baseline. It is quite common in experimental data for the baseline to not be well defined which means the selection of baseline correction tools has a large effect on the value of $E_{a\alpha}$. In order to counteract this it is possible to employ noise correction algorithms to attempt to smooth out any noise generated in the $d\alpha/dT$ curve, but then it can become possible to see activation energy as a function of the noise correction function rather than the actual *real* processes occurring.

In practice it is the best methodology to use a combination of analysis methods

(integral, differential, linear, and nonlinear) in order to calculate the activation energy. While the results will most likely not overlap with each other, if the values are within a small range it would indicate that the calculated values for activation energy are a good estimate. It is also important to use integral alongside differential methods. As differential methods include the effect of the kinetic model through the $d\alpha/dT$ term it is important to study this effect, but as mentioned previously they are highly sensitive to noise, so using an integral method such as the Vyazovkin's nonlinear approach combined with the differential method will validate that the calculate activation energies from the differential method are accurate as the integral methods do not include a kinetic model and are much less sensitive to noise.

It has been stated multiple times that when one has a complex reduction profile from a TPR thermogram (as is often the case in catalytic systems) that there are multiple reduction processes occurring. When the the total reduction profile has been recorded and converted into a $d\alpha/dT$ profile, it can be considered that the actual $d\alpha/dT$ can be represented using:

$$\frac{d\alpha}{dT} = \sum_{i=1}^n \left[\frac{A_i}{\beta} f(\alpha_i) \exp\left(\frac{-E_{ai}}{RT}\right) \right] \cdot P_i \quad (2.30)$$

where subscript i is an index for the i^{th} process and n is the total number of processes to be considered. This function is similar to the standard kinetic model equation 2.8 with one extra variable P_i which is the integral of the i^{th} peak divided by the total integral, which acts as a proportionality constant for the specific process. This means that the value of E_a as calculated from the previous methods will change as the process which dominates the reduction profile changes. Alongside this change from one reduction process to another, it also needs to be considered the E_a can also change during a single reduction process as a function of α_i . In order to attempt to model this change in E_a as a function of α_i the following relationship was proposed by Munteanu *et. al.*[19]

$$E_{a(\alpha)} = E_{a(0)} \cdot (1 + \gamma\alpha) \quad (2.31)$$

Here a constant scaling factor, represented by the parameters γ is introduced. It can be seen that the change in activation energy as a function of α is considered to be linear with respect to the degree of reduction. When combining equation 2.30 with equation 2.31 this gives us the final form the kinetic equation for a temperature programmed thermogram.

$$\frac{d\alpha}{dT} = \sum_{i=1}^n \left[\frac{A_i}{\beta} f(\alpha_i) \exp \left(\frac{-E_{a(0)i} \cdot (1 + \gamma\alpha)}{RT} \right) \right] \cdot P_i \quad (2.32)$$

While the parameter γ will have an effect on the calculated activation energy, and the subsequent shape of the temperature programmed profile (usually in the form of a shoulder on the peak) the value tends to be very small ($\gamma < 0.1$) which means that while it does has an effect on the temperature programmed profile, it's effect is not so significant that it causes errors when evaluating the kinetic data.

2.2.4 The Kinetic Constants

Now that we have shown how it is possible to calculate the Arrhenius parameters, next up is how the kinetic constants, and subsequently the reaction model, are calculated. The reaction model is represented by the function $f(\alpha)$ in the kinetic equation 2.32, and the two most popular methods of calculating which model to use is by performing a regression fit using equation 2.32, or via a methodology commonly known as Malek's procedure[20]. The regression analysis uses the methodology outlined in section 2.1.2 to compare a simulated thermogram profile using equation 2.32 and then uses the regression algorithm to vary the values for A_i , E_{ai} , γ_i , P_i , and the kinetic constants used in the various reaction models in table 2.1 until a minima is reached and the final variables are used to estimate the reaction model.

The second method, Malek's procedure, is slightly more complex and uses so called *master plots* in order to evaluate which kinetic model describes the reduction process and the kinetic constants calculated.

Malek's procedure is based on calculating two *master plots* $y(\alpha)$ and $z(\alpha)$ [21, 22]. The first *master plot* is $y(\alpha)$ which is calculated using:

$$y(\alpha) = \left[\left(\frac{d\alpha}{dT} \right) \beta \right] \exp \left(\frac{E_a}{RT} \right) \quad (2.33)$$

which as can be seen is proportional to the $d\alpha/dT$ function, and as such the function $y(\alpha)$ can be plotted as a function of α and as such gives the shape of the function $f(\alpha)$. This means that for a given kinetic model, the shape of the $y(\alpha)$ function will be unique for that specific kinetic model, and hence if one knows the shape of $y(\alpha)$ for the various kinetic models before hand a comparison can be made. The downside to this technique is that the shape of the $y(\alpha)$ function is also highly dependant on the activation energy, meaning that an accurate value for the activation energy is needed in order for the comparison to the *master* $y(\alpha)$ plots. The second function $z(\alpha)$ can be expressed by:

$$z(\alpha) = \pi(x) \left[\left(\frac{d\alpha}{dT} \right) \beta \right] \frac{T}{\beta} \quad (2.34)$$

The importance of the $z(\alpha)$ function arises from the fact that if one takes the standard kinetic equation 2.8 and the integral form equation 2.20 and combines them together then the following equation for reaction rate is obtained:

$$\left[\left(\frac{d\alpha}{dT} \right) \beta \right] = \left[\frac{\beta}{T\pi(x)} \right] f(\alpha)g(\alpha) \quad (2.35)$$

then by rearranging equation 2.34 the following relationship is established:

$$z(\alpha) = \pi(x) \left[\left(\frac{d\alpha}{dT} \right) \beta \right] \frac{T}{\beta} = f(\alpha)g(\alpha) \quad (2.36)$$

then differentiating equation 2.36 with respect to α gives:

$$\frac{dz(\alpha)}{d\alpha} = \frac{df(\alpha)}{d\alpha}g(\alpha) + 1 \quad (2.37)$$

and then if we set $dz(\alpha)/d\alpha$ to zero, indicating that we are at the peak value of $z(\alpha)$

$$-\frac{df(\alpha)}{d\alpha}g(\alpha_p) = 1 \quad (2.38)$$

with the variable α_p being the value of α at the peak of the $z(\alpha)$ function. As equation 2.38 is constant at the value α_p for a given reaction model, the maxima of the $z(\alpha)$ function can be compared to a master $z(\alpha)$ plot, alongside the $y(\alpha)$ function, in order to determine the kinetic model.

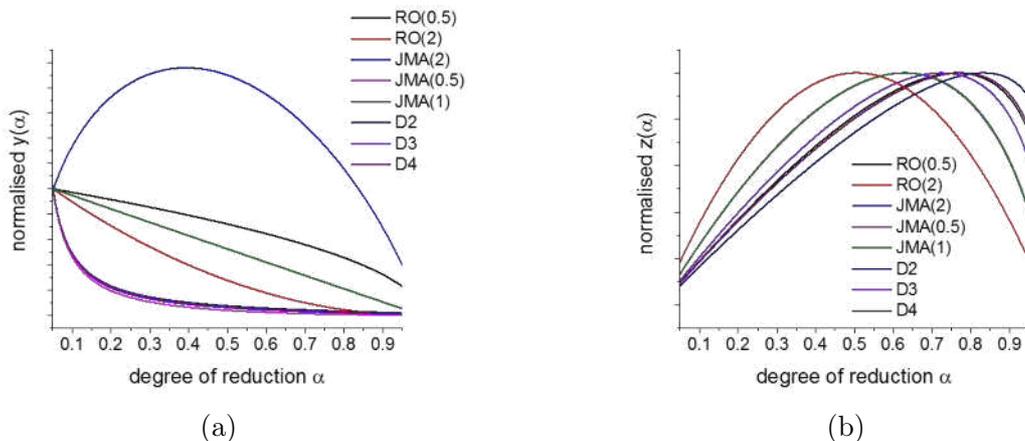


Figure 2.4: Master plots for kinetic models in table 2.2 a) Master plots for the $y(\alpha)$ function b) Master plots for the $z(\alpha)$ function

Using a combination of these two plots it is possible to get an accurate estimate of the reduction process occurring on the surface of the catalyst. In the original paper which outlines Malek's procedure[20] he uses the kinetic models outlined in table 2.2 and the master plots shown in figure 2.4 in order to calculate the kinetic model.

Table 2.2: General kinetic models used in Malek's Procedure

Reaction Model	Symbol	$f(\alpha)$
Reaction Order	RO(n)	$(1 - \alpha)^n$
Sestak-Berggren	SB(m, n)	$\alpha^m(1 - \alpha)^n$
Johnson-Mehl-Avrami	JMA(n)	$n(1 - \alpha)[- \ln(1 - \alpha)]^{1-(1/n)}$
n D-reaction	R(n)	$(1 - \alpha)^{1/n}$
2D-Diffusion	D2	$1/[- \ln(1 - \alpha)]$
Jander equation	D3	$3/2(1 - \alpha)^{2/3}/[1 - (1 - \alpha)^{2/3}]$
Ginstline-Brounshtein	D4	$3/2[(1 - \alpha)^{-1/3} - 1]$

It was also noted in the paper by Malek that the α_p value for the $z(\alpha)$ functions is insensitive to value of activation energy used in the function, with a variance of only 1% noted[20]. The master plots shown in figure 2.4 contain all of the kinetic information about a reduction process, and hence will be unique for each model. The process to estimate which model is best applied to the system is outlined in a flow chart shown in figure 2.5 which is taken from Malek's paper[20] (*note*: signs for greater than and less than are swapped for the RO(n) model due to an error in the paper), and all that is required is the maxima of $z(\alpha)$ and $y(\alpha)$, and the shape of the $y(\alpha)$ plot.

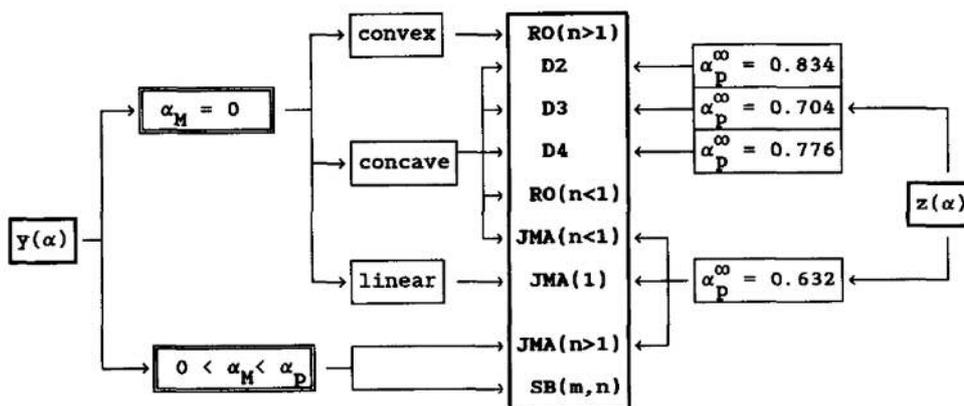


Figure 2.5: Flow chart for assigning a kinetic model with greater than and less than signs swapped for the RO(n) model)

Using the flow chart shown in figure 2.5 it can be seen that for certain models only (*e.g.* SB(m, n), JMA(n) and RO(n)) only a general idea of the model can assigned using the flow chart. In order to calculate the actual values for the kinetic

constants some further calculations are required. To calculate the kinetic constant for the RO(n) model the relationship established by Gorbachev[23] can be used.

$$\alpha_p = 1 - \left[1 + \frac{1-n}{n} x_p \pi(x_p) \right]^{1/(n-1)} \quad (2.39)$$

The kinetic constant n can then be solved by a simple iterative process. For the JMA(n) model it is slightly more complex, as there are two different types of master plots for JMA($n > 1$) and JMA($n < 1$). To solve when values of $n > 1$ the kinetic constant can be calculated using the maxima of the $y(\alpha)$ function

$$n = \frac{1}{1 + \ln(1 - \alpha_m)} \quad (2.40)$$

with α_m being the maxima of the $y(\alpha)$ function. With a kinetic constant of $n < 1$ then there are two possible ways of calculating n . The first is the Satava method[16]

$$\ln[-\ln(1 - \alpha)] = C - \frac{nE}{RT} \quad (2.41)$$

which uses a linear plot of $\ln[-\ln(1 - \alpha)]$ v $1/T$ to calculate n . The second method uses the direct relationship

$$n = \frac{1 - x_p \pi(x_p)}{\ln(1 - \alpha_p) + 1} \quad (2.42)$$

It tends to be best practice to use a combination of the two methods in order to get an accurate estimate of the n parameter. The final kinetic model to solve is the SB(m, n) model, which uses:

$$\frac{m}{n} = \frac{\alpha_m}{(1 - \alpha_m)} \quad (2.43)$$

The ratio of the kinetic constants is simple to calculate by using the maxima of

the $y(\alpha)$ curve, deconvolution of the parameters can be performed by a simple rearrangement of equation 2.3 where $f(\alpha)$ is the SB(m,n) function (table 2.2).

$$\ln \left[\left(\frac{d\alpha}{dt} \right) \exp \left(\frac{E}{RT} \right) \right] = \ln A + n \ln[\alpha^p(1 - \alpha)] \quad (2.44)$$

By using a simple linear plot the kinetic constant n can be calculated, then inputting it into equation 2.43 the kinetic constant m can be calculated.

It is shown in this section that by using well established theory one can calculate a large amount of information, from a temperature programmed thermogram. This simple and commonly used experiment has the potential to provide in depth analysis of the reduction mechanism of a material, which is of particular interest in catalysis, but unfortunately while the theory is robust and sound the application of the theory is minimal at best when compared to the quantity of temperature programmed experiments. In the next chapter we develop a new methodology for performing this analysis using the CCI-TPR software in order to allow the deconvolution complex temperature programmed profiles, and subsequently perform the analysis outlined in this section.

2.3 Temporal Analysis of Products

The general ideas behind the TAP experiment are outlined in section 1.2, but in order to fully appreciate the complexities of TAP we need to first delve deeper into the theory behind the experiment. TAP is designed around the idea of using the standard diffusion through the packed bed reactor as measuring stick by comparing the exit flow responses for the reactants and products, and using the differences to calculate various kinetic parameters. The basic theory behind TAP is outlined in the 1997 paper by John Gleaves.[24]

2.3.1 Basics of the TAP experiment

The experimental data obtained from a TAP experiment can be simplified as the flow of gas out of the reactor as a function of time. As the flow is detected using a mass spectrometer, the data has a very high time resolution and a high level of accuracy, which means that even fine chemistry can be picked up. If one knows the amount of gas that enters the reactor, and the flow of gas that is picked up at the end of the reactor the normalised flow can be calculated, which is used in the majority of TAP analysis. Another way to view the TAP exit flow is as a probability density

$$\rho = \frac{F_A}{N_{pA}} \quad (2.45)$$

where ρ is the *temporal probability density*, F_A is the exit flow of gas A measured at the outlet of the reactor, and finally N_{pA} is the total number of moles of A released from the flow valve. By differentiating ρ with respect to time, the probability of finding a molecule at the reactor exit for a specific time interval can be expressed

$$P_t = \int_0^t \rho \, dt \quad (2.46)$$

where P_t is the probability of finding a molecule between the interval $0 \rightarrow t$. If t is set as ∞ then P_t at this most basic level can give some information on the reaction mechanism. If $P_\infty = 1$ then all of the gas sent through the reactor is recorded at the exit - indicating either no reaction or a reversible adsorption/diffusion type reaction occurring. If $P_\infty < 1$ then it can be assumed that some of the gas has reacted or adsorbed to the surface of the catalyst, therefore conversion can be defined as:

$$X = 1 - P_\infty \quad (2.47)$$

While this idea of conversion works well in theory, in practice there are multiple other factors that need to be considered, such as: mass spec sensitivity, fragmentation patterns, and inherent system and electrical noise. This simple idea of measuring conversion can be taken further in order to understand the kinetics of the processes occurring on the surface of the catalyst.

The final aim of running the TAP experiment is to study the exit flow from the reactor and attempt to assign a kinetic model to a process, and then be able to calculate the kinetic constants. Before we can start to model and predict a complex catalytic process, it is easier to start with a simple *one-zone* model, which was first described in the original TAP paper[25]. In order for the *one-zone* model to be applied the following assumptions need to be made:

- The catalyst bed is completely uniform along the reactor bed
- There is no radial concentration gradient in the catalyst bed
- The temperature in the catalyst bed is completely uniform
- The diffusion process through the catalyst bed is constant for all gases

The final assumption is held by applying the conditions required for Knudsen exit flow (see section 1.2) meaning that the diffusion through the reactor is purely a function of mass and temperature.

2.3.2 Modelling the standard diffusion curve

When modelling Knudsen exit flow it can be considered that the flow of gas is related to the flux of gas through the following relationship:

$$\varepsilon_b \frac{\delta C_A}{\delta t} = D_{eA} \frac{\delta^2 C_A}{\delta z^2} \quad (2.48)$$

where ε_b is the fractional voidage of the packed bed reactor, C_A is the concentration of gas A , t is the time, D_{eA} is the Knudsen diffusivity of a particular gas, and finally z is the axial coordinate along the reactor bed. We can then apply some boundary conditions to equation 2.48.

Initial Condition:

$$0 \leq z \leq L \quad t = 0 \quad C_A = \delta_z \frac{N_{pA}}{\varepsilon_b A} \quad (2.49)$$

Boundary Conditions:

$$z = 0 \quad \frac{\delta C_A}{\delta z} = 0 \quad (2.50)$$

$$z = L \quad C_A = 0 \quad (2.51)$$

where L is the length of the reactor in cm, N_{pA} is the number of moles of gas A , and A is the cross sectional area of the reactor. The initial condition states that at $t = 0$, at the entrance of the reactor the concentration of the gas is represented by a delta function. The boundary conditions represented by equation 2.50 and equation 2.51 specify that at the reactor entrance, when the pulse valve is closed there is zero flux of the gas, and that at the reactor exit there is essentially zero concentration (due to ultra-high vacuum conditions at the reactor exit). The flow of gas out of the exit of the reactor (as detected by the mass spectrometer) can be described using:

$$F_A = -AD_{eA} \left. \frac{\delta C_A}{\delta z} \right|_{z=L} \quad (2.52)$$

where F_A is the exit flow (mols^{-1}), and the flux can be represented by:

$$Flux_A = \frac{F_A}{A} \quad (2.53)$$

To solve the differential equation for exit flow, it is simpler to use dimensionless parameters instead:

$$\xi = \frac{z}{L} \quad (2.54)$$

$$\bar{C}_A = \frac{C_A}{N_{pA}/V_{void}} = \frac{C_A}{N_{pA}/\varepsilon_b AL} \quad (2.55)$$

$$\tau = \frac{tD_{eA}}{\varepsilon_b L^2} \quad (2.56)$$

$$\bar{F}_A = F_A \frac{\varepsilon_b L^2}{N_{pA} D_{eA}} \quad (2.57)$$

where ξ is the dimensionless axial coordinate, \bar{C}_A is the dimensionless concentration, \bar{F}_A is the dimensionless exit flow, and τ is the dimensionless time. This means we can then solve the differential equation the same as previously (equations 2.49 to 2.51) but this time using dimensionless conditions.

Initial Condition:

$$0 \leq \xi \leq 1 \quad \tau = 0 \quad \bar{C}_A = \delta_\xi \quad (2.58)$$

Boundary Conditions:

$$\xi = 0 \quad \frac{\delta \bar{C}_A}{\delta \xi} = 0 \quad (2.59)$$

$$\xi = 1 \quad \bar{C}_A = 0 \quad (2.60)$$

The dimensionless exit concentration can be solved by separation of variables giving:

$$\bar{C}_A(\xi, \tau) = 2 \sum_{n=0}^{\infty} \cos((n + 0.5)\pi\xi) \exp(-(n + 0.5)^2\pi^2\tau) \quad (2.61)$$

and the dimensionless exit flow:

$$\bar{F}_A(\xi, \tau) = \pi \sum_{n=0}^{\infty} (2n + 1) \sin((n + 0.5)\pi\xi) \exp(-(n + 0.5)^2\pi^2\tau) \quad (2.62)$$

if we only consider the exit flow at the end of the reactor $\xi = 1$ then equation 2.62 can be expressed as:

$$\bar{F}_A = \pi \sum_{n=0}^{\infty} (-1)^n (2n + 1) \exp(-(n + 0.5)^2\pi^2\tau) \quad (2.63)$$

Equation 2.63 is called the standard diffusion curve, and is constant for a Knudsen diffusional response, when no reaction is occurring. In reality, when we measure the exit flow from the reactor what we really measure is a dimensional form of the equation:

$$\frac{F_A}{N_{pA}} = \frac{D_{eA}}{\varepsilon_b L^2} \sum_{n=0}^{\infty} (-1)^n (2n + 1) \exp\left(- (n + 0.5)^2 \pi^2 \frac{t D_{eA}}{\varepsilon_b L^2}\right) \quad (2.64)$$

The difference between the dimensional and dimensionless curve can be seen in figure 2.6. The dimensionless curve has all factors relating to the molecular weight, and the temperature of the gas removed, and can be considered a master curve for any non-interacting gas diffusing through the reactor - which is used for checking to see if Knudsen diffusion is present in the reactor. The standard diffusion curve has applications not only in confirming that the reaction is performed under the Knudsen regime, but also in calculating various transport properties of the reactor which come into use in later analysis.

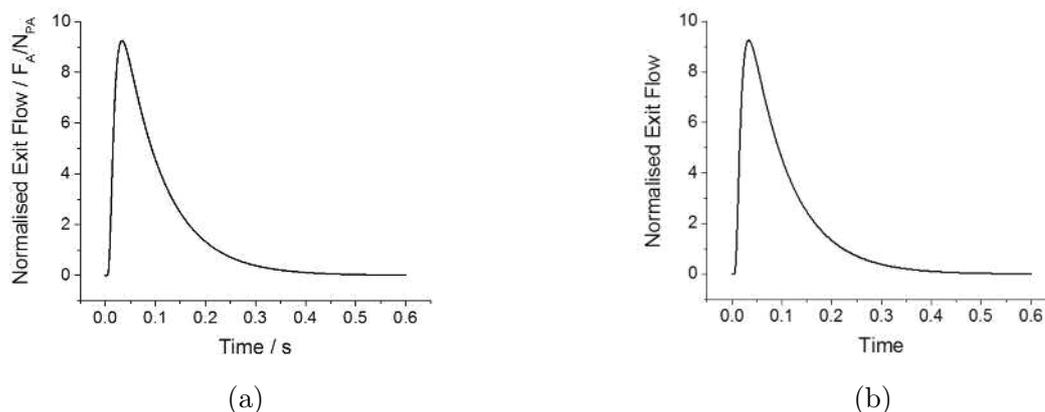


Figure 2.6: Knudsen exit flow comparisons a) Standard dimensionless diffusion curve
b) Standard dimensional diffusion curve

2.3.3 Confirming Knudsen Diffusion

One of the first steps when performing a TAP experiment is to confirm that the reaction is being performed under standard conditions. This can be done in one of two ways: The first is to use a regression analysis using equation 2.64 by varying the parameters for D_{ea} , ε_b , and L and seeing if the simulated curve overlaps with the experiment. In the case of Knudsen diffusion the fit between experimental and simulated should be close to perfect, as even a slight deviation between the two can indicate that Knudsen conditions are not met. The second method is to analyse the diffusional response curve itself. For the dimensionless diffusion curve there is a maximum when:

$$\tau_p = \frac{1}{6} \quad (2.65)$$

where the subscript p indicates at the peak of the exit flow curve. When the value for τ from equation 2.65 is substituted into equation 2.63, we end up with:

$$\bar{F}_{A,p} = 1.85 \quad (2.66)$$

which gives the following relationship:

$$\bar{F}_{A,p} \cdot \tau_p = 0.31 \quad (2.67)$$

As this is a dimensionless equation this relationship should hold true for any gas. As mentioned previously, experimentally we don't actually measure dimensionless exit flow, we measure the dimensional normalised exit flow (equation 2.64) instead. If one then takes the dimensional forms of equations 2.65 to 2.67 then the following relationships can be expressed as true:

$$t_p = \frac{1}{6} \frac{\varepsilon L^2}{D_{eA}} \quad (2.68)$$

$$H_p = 1.85 \frac{\varepsilon L^2}{D_{eA}} \quad (2.69)$$

$$H_p \cdot t_p = 0.31 \quad (2.70)$$

What equation 2.70 shows is that if one takes the normalised exit flow response and takes the value of the peak maxima (H_p) and multiplies it by the time of the peak maxima (t_p), they should get a value of 0.31. This method is a very good indicator of whether or not the reaction has been performed under Knudsen conditions. While previously it was mentioned that when performing the curve fitting a near-perfect fit is required for Knudsen conditions to be met, when calculating the value of $H_p \cdot t_p$ anywhere between 0.29 and 0.33 is considered to be accurate enough to be under Knudsen conditions, as electrical and experimental noise is considered to have a large effect on its value. Using these various methods it is possible to predict and test for Knudsen conditions in the reactor if one considers the reactor to be *one-zone*.

2.3.4 Modelling the Catalyst Zone

In the TAP experiment there tends to be not just one single zone for the gas to pass through, but three zones: The first layer of the inert packing, the thin layer of catalyst, and finally the second layer of inert packing. For the case of an inert gas the standard diffusion curve discussed in the previous section is still applicable, as from the perspective of the inert gas, the whole reactor is considered to be the same (as long as the packing of the reactor is constant throughout). Where the modelling becomes more complex is when a reactant or product gas is considered, for the reactant gases there will be a large change in the standard flow through the reactor when it interacts with the catalyst, and for the product gas, there is no inlet pulse and hence the diffusion starts at the catalyst zone. Using equation 2.48 to model the flow of gas through the inert zones, the flow of gas through the catalyst zone can be expressed using:

$$\varepsilon_{cat} \frac{\delta C_A}{\delta t} = D_{cat} \frac{\delta^2 C_A}{\delta x^2} \pm R \quad (2.71)$$

Where R is some complex rate function, and the subscript cat indicates it is in the catalyst zone:

$$R = R_A(C_g(z, t), C_\theta(z, t), N_m) \quad (2.72)$$

Where C_g is the concentrations of the various gases, and C_θ is the concentration of the various dynamic surface concentrations. The solution to the rate equation is highly complex, but can be simplified by including two assumptions:

- The rates of the elementary reactions are not changed as a function of the change in the catalyst state in the reaction
- The rates of reaction are linear to small concentrations of gas or changes in

the catalyst

The first assumption stems from the fact the pulse size is very small, meaning that there is an insignificant change in the catalyst structure during an experimental pulse. The second assumption is a standard linear response theory assumption which has been proved to be valid for the majority of catalytic mechanisms.[26] From these two assumptions the following relationship can be established:

$$R = \sum_i k_i(C_{CS}, N_M)C_{g,i}(z, t) + \sum_j k_j(C_{CS}, N_M)C_{\theta,j}(z, t) \quad (2.73)$$

Where i and j are index processes relating to gaseous and surface reactions respectively, and k is the kinetic parameter for a given process at a specific catalyst state. We can then assign some boundary conditions to equation 2.71

Initial Condition at $t = 0$

$$C_g \Big|_{t=0} \quad (2.74)$$

The condition at the exit of the reactor is similar to equation 2.50

$$C_g \Big|_{z=L} = 0 \quad (2.75)$$

and finally at the inlet pulse

$$-AD_{in} \frac{\delta C_A}{\delta z} \Big|_{z=0} = 2N_{pA}\delta(t) \quad (2.76)$$

Before we can solve the equation for gas flow through the catalyst zone an explicit form of the rate term R is required.

2.3.5 Understanding the Kinetic Equation

As TAP is generally used for attempting to obtain information on reaction mechanisms, it is quite often the case that the mechanism for the reaction is not known beforehand, and as such a model-free method is required. By applying the assumptions onto equation 2.72 giving equation 2.73, it can then be combined with equation 2.71 to give:

$$\varepsilon_{cat} \frac{\delta C_A}{\delta t} = D_{cat} \frac{\delta^2 C_A}{\delta x^2} - k_a(C_{CS}, N_M)C_g + \sum_j k_j(C_{CS})C_{\theta,j} \quad (2.77)$$

To remove the time dependence of equation 2.77 it can be converted into the Laplace domain[27] giving:

$$\frac{d^2 \bar{C}_g}{dx^2} = \frac{s\varepsilon_{cat}}{D_{cat}} \pm \frac{k_a(C_{CS}\bar{C}_g + \sum_j k_j(C_{CS})\bar{C}_{\theta,j})}{D_{cat}} \quad (2.78)$$

with s being the Laplace variable, and \bar{C}_G and $\bar{C}_{\theta,j}$ indicating the Laplace versions of their various functions. In order to solve equation 2.78, the concentration of the various dynamic surface concentrations ($C_{\theta,j}$) should be solved as a function of the gaseous concentrations (C_g) and then eliminated. In the TAP reactor it is considered that the transport through the reactor occurs in the gaseous phase (diffusion on the surface of the catalyst is negligible), hence the rate of change of dynamic surface concentrations can be expressed as a function of time using:

$$\frac{dC_{\theta,i}}{dt} = \sum_j k_{i,j}(C_{CS})C_{\theta,j} + \sum_j k_{a,j}(C_{CS})C_{g,i} \quad (2.79)$$

with $k_{i,j}$ being the various kinetic constants as a function of catalyst state, this can then be expressed in the Laplace domain:

$$s\bar{C}_{\theta,i} = \sum_j k_{i,j}(C_{CS})\bar{C}_{\theta,j} + \sum_j k_{a,j}(C_{CS})\bar{C}_{g,i} \quad (2.80)$$

and then solved with respect to surface concentrations and generalised to give:

$$\bar{C}_{\theta,i}(z, s) = \sum_j \alpha_{i,j}(C_{CS}, s)\bar{C}_{g,j}(z, s) \quad (2.81)$$

where $\alpha_{i,j}$ contains the various coefficients that are a function of catalyst state and the Laplace variable. From the solution to equation 2.81 the rate term (equation 2.73) can be compressed into a simpler form:

$$R = \sum_j r^{i,j}(C_{CS}, s)\bar{C}_{g,j}(s, z) \quad (2.82)$$

This means that by applying equation 2.82 to equation 2.78 we get:

$$\frac{d^2\bar{C}_g}{dx^2} = \frac{s\varepsilon_{cat}\bar{C}_{g,i}(z, s) + \sum_j r^{i,j}(C_{CS}, s)\bar{C}_{g,j}(z, s)}{D_{cat}} \quad (2.83)$$

If one then considers a system in which only one gas is reacting with the catalyst at a time (as is usually the case in TAP) then we can express equation 2.83 as:

$$\frac{d^2\bar{C}_g}{dx^2} = \frac{s\varepsilon_{cat}\bar{C}_{g,i}(z, s) + r^i(C_{CS}, s)\bar{C}_{g,j}(z, s)}{D_{cat}} \quad (2.84)$$

The differential equation 2.84 can then be solved analytically, and combined with the analytical solutions for the flows in the diffusional zones (equation 2.48) in the Laplace domain, the exit flow for a three zone system can be solved in the Laplace domain as a general function of $r^{i,j}(C_{CS}, s)$. [27]

$$\frac{1}{\bar{F}_{exit}} = \left(\cosh \left(\frac{L_{cat}\sqrt{r^r(C_{CS}, s) + s\varepsilon_{cat}}}{\sqrt{D_{cat}^r}} \right) \cosh \left(\frac{2L_{in}\sqrt{\varepsilon_{in}}}{\sqrt{D_{in}^r}} \right) \right) \quad (2.85)$$

$$\begin{aligned} & \times \sinh \left(\frac{L_{cat} \sqrt{r^r(C_{CS}, s) + s\varepsilon_{cat}}}{\sqrt{D_{cat}^r}} \right) \\ & + \sinh \left(\frac{2L_{in} \sqrt{s\varepsilon_{in}} D_{cat}^r (r^r(C_{CS}, s) + s\varepsilon_{cat}) + D_{in}^r s\varepsilon_{in}}{\sqrt{D_{in}^r} 2\sqrt{D_{cat}^r (r^r(C_{CS}, s) + s\varepsilon_{cat}) D_{in}^r s\varepsilon_{in}}} \right) \end{aligned}$$

Equation 2.85 is the general solution for a reactant exit flow for a three zone reactor system and due to the large number of terms becomes non-trivial to apply to experimental data. It is a function the rate term ($r^r(C_{CS}, s)$), the fractional voidage ε , the Knudsen Diffusivity D , and the length of various zones L , with the subscript *cat* and *in* indicating the catalyst and inert zones respectively.

2.3.6 Thin Zone Reactor Model

In order to simplify the kinetics of the reactor a Thin Zone TAP Reactor (TZTR) methodology is set up. The idea behind the TZTR setup is that the thickness of the catalyst zone is considered to be very small when compared to the total length of the reactor (and subsequent inert zones). This means that any deviation in the mean concentration of gas along the axial coordinate of the bed is considered to be small.

When the methodology for the TZTR was first published [28] it was used as a special case of a *three-zone* TAP reactor as explained previously. [24] The three zone model states that if there are three zones which can be expressed as the following:

$$C_{A,zone1} \Big|_{Z_1} = C_{A,zone2} \Big|_{Z_1} \quad (2.86)$$

$$C_{A,zone2} \Big|_{Z_2} = C_{A,zone3} \Big|_{Z_2} \quad (2.87)$$

$$-D_{eA,zone1} \left. \frac{\delta C_{A,zone1}}{\delta Z} \right|_{Z_1} = -D_{eA,zone2} \left. \frac{\delta C_{A,zone2}}{\delta Z} \right|_{Z_1} \quad (2.88)$$

$$-D_{eA,zone2} \left. \frac{\delta C_{A,zone2}}{\delta Z} \right|_{Z_2} = -D_{eA,zone3} \left. \frac{\delta C_{A,zone3}}{\delta Z} \right|_{Z_2} \quad (2.89)$$

Equations 2.86 to 2.89 represent that the concentration and the flux of the gas that leave one zone and enter the other are equal, meaning that the concentration (and flux) of the gas is continuous down the reactor bed, and should be modelled that way.

The way that the TZTR setup works is by considering the catalyst zone to be a boundary between two inert zones, rather than it's own individual zone:

$$C_{A,zone1}(z, t) \Big|_{z=L_{TZ}} = C_{A,zone2}(z, t) \Big|_{z=L_{TZ}} = C_{A,TZ} \quad (2.90)$$

where L_{TZ} is the location of the thin zone in the reactor (usually the middle), and $C_{A,TZ}$ is the average concentration in the thin zone. The next step is to consider the flows of the gas through their respective zones

$$F_{A,1}(z, t) \Big|_{z=L_{TZ}} = -AD \left. \frac{\delta C_{A,zone1}(z, t)}{\delta z} \right|_{z=L_{TZ}} \quad (2.91)$$

$$F_{A,2}(z, t) \Big|_{z=L_{TZ}} = -AD \left. \frac{\delta C_{A,zone2}(z, t)}{\delta z} \right|_{z=L_{TZ}} \quad (2.92)$$

It can then be considered that the difference between the exit of zone 1 and the entrance of zone 2 can be related to the rate of reaction occurring in the catalyst zone.

$$F_{A,1}(z, t) \Big|_{z=L_{TZ}} - F_{A,2}(z, t) \Big|_{z=L_{TZ}} = AL_{cat}R_{TZ} \quad (2.93)$$

where L_{cat} is the length of the thin catalyst zone (cm), and the function R_{TZ} is the same as the rate function from the *three-zone* model (equation 2.73). The function of the thin zone reactor can then be solved in a similar fashion to that of the *three-zone* model in the Laplace domain[29] giving:

$$\bar{F}_{exit,r} = \frac{1}{\cosh(\sqrt{2s\tau_{in}}) + \left(\frac{r^r(C_{CS}, s)\tau_{cat}}{\sqrt{2s\tau_{in}}} \right) \sinh(\sqrt{2s\tau_{in}})} \quad (2.94)$$

for the exit flow of the reactant, and:

$$\bar{F}_{exit,p} = \frac{\left(\frac{\sqrt{2}r^p(C_{CS}, s)\tau_{cat}}{\sqrt{s\tau_i n}} \right)}{\left(\frac{1}{\bar{F}_{exit,r}} \right) \times \cosh\left(\frac{\sqrt{2s\tau_{in}D_{in}^r}}{D_{in}^p} \right)} \quad (2.95)$$

for the exit flow of the product. With:

$$\tau_{in} = \frac{\varepsilon_{in}L_r^2}{2D_{in}} \quad \tau_{cat} = \frac{L_rL_{cat}}{2D_{in}} \quad (2.96)$$

Similar to the exit flow for the *three-zone* model, the exit flows for the TZTR model are functions of the rate term $r(C_{CS}, s)$ with the subscript r or p denoting for the reactants or products, the fractional voidage ε , and the diffusivity D and the length L of the catalyst and inert zones. Using the solutions for exit flow it is now almost possible to simulate the exit flow measured during a TAP experiment, the only variable missing is a real expression for the rate term. Before the rate term can be expressed we first need to understand how to perceive the exit flow as *moments*

2.3.7 Exit Flow as Moments

Analysing the *moments* of the exit flow from the reactor can be very useful in TAP analysis. Moments are defined as the integral of the exit flow multiplied by time raised to a power.

$$M_n = \int_0^{\infty} t^n F(t) dt \quad (2.97)$$

where n is the moment number (*e.g.* M_0 is the integral of the exit flow). The moments can also be related to the exit flow in the Laplace domain through the following relationship

$$M_n = (-1)^n \left. \frac{\delta^n \bar{F}(s)}{\delta s^n} \right|_{s=0} \quad (2.98)$$

The moments themselves do have a physio-chemical meaning, and represent the time dependence of the flow of gas. The zeroth moment (M_0) is defined as the total number of molecules in the exit flow response, thus the M_0 can be used to calculate the conversion if it is normalised to the intensity of the inlet and has the dimension of *moles*. The first moment (M_1) has the dimension *moles s* and is the number of molecules multiplied by the time. This means that M_1 can be used to determine the residence time of the gas in the reactor at a given point by normalising it to the zeroth moment. The residence time can be defined as the average amount of delay in the pulse as it travels through the reactor. The final moment that is commonly used is the second moment (M_2) which is the exit flow multiplied by time squared and has the dimension *moles s²*. The actual physical meaning of the second moment is slightly more complex as it relates to the relative amount of time the gas spends in the reactor when compared to the residence time, and is correlated to processes such as desorption. The integrand of the various moments can be seen in figure 2.7, and it can be clearly seen that the different moments tend to focus on separate parts of the

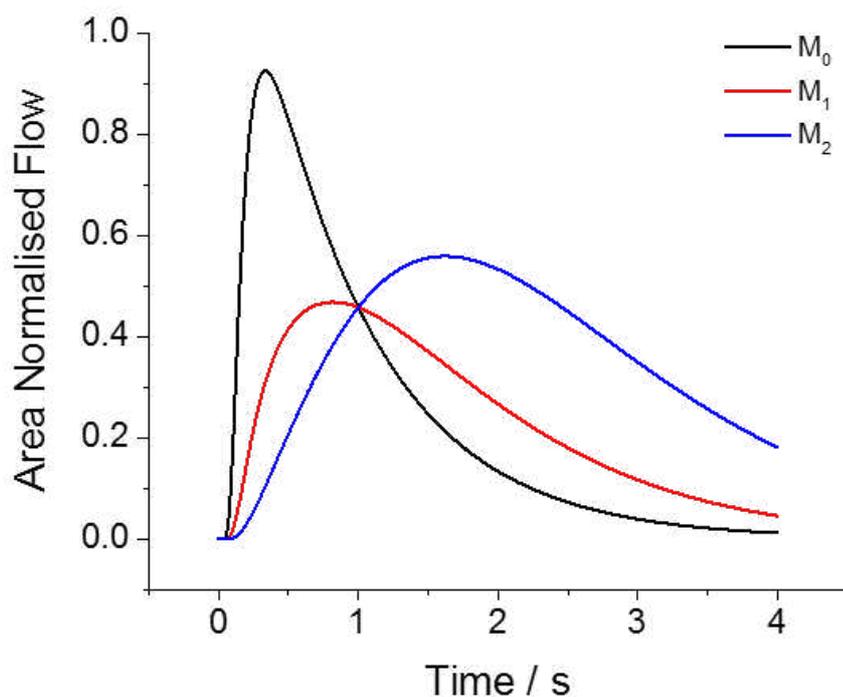


Figure 2.7: Integrands of the various moments for simulated Knudsen exit flow

curve, with the higher the degree moments being more influenced by slow reaction processes.

The relationship expressed in equation 2.98 is also useful when attempting to understand the rate term $r(C_{CS}, s)$. It can be stated that the nature of the function 2.98 at $s = 0$ suggests that the Laplace reactivity should be expanded in a Taylor series with respect to the Laplace variable s giving:

$$r(C_{CS}, s) = r_0(C_{CS}) + r_1(C_{CS})s + r_2(C_{CS})s^2 \quad (2.99)$$

the three coefficients r_0 , r_1 and r_2 are called the *basic kinetic coefficients* as a function of the catalyst state, are used to describe catalytic systems. Using these coefficients the equations for the moments in the TZTR setup have been solved as a function of the basic kinetic coefficients.[27]

For the reactants:

$$M_0^r = \frac{1}{1 + \tau_{cat} r_0^r} \quad (2.100)$$

$$\frac{M_1^r}{M_0^{R^2}} = \tau_{cat} r_1^r + \tau_{in} \left(1 + \frac{\tau_{cat} r_0^r}{3} \right) \quad (2.101)$$

$$\frac{M_2^r}{M_0^{R^2}} - 2 \frac{M_1^{r^2}}{M_0^{R^3}} = -2\tau_{cat} r_2^r + \frac{\tau_{in}}{3} \left(\frac{\tau_{in} \tau_{cat} r_0^r}{5} + 2\tau_{cat} r_1^r + \tau_{in} \right) \quad (2.102)$$

For the Products:

$$M_0^p = M_0^r \tau_{cat} r_0^p \quad (2.103)$$

$$\frac{M_1^p}{M_0^p} = \frac{r_1^p}{r_0^p} + \frac{\tau_{in}}{12} \left(8M_0^r + 3 + 9 \frac{D_{in}^r}{D_{in}^p} \right) + \tau_{cat} M_0^r r_0^p \quad (2.104)$$

$$\frac{M_2^p}{M_0^p} = 2 \frac{r_2^p}{r_0^p} + \frac{19 D_{in}^r{}^2 \tau_{in}{}^2}{16 D_{in}^p{}^2} \dots \quad (2.105)$$

$$+ \frac{D_{in}^r \tau_{in}{}^2}{8 D_{in}^p r_0^p} (r_0^p) ((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat}) - 12r_1^p) \dots$$

$$- \frac{r_1^p}{6r_0^p} ((3 + 8M_0^r) \tau_{in} \dots$$

$$+ 12M_0^r r_1^r \tau_{cat}) + \tau_{in}{}^2 \left(\frac{5}{48} + \frac{1}{45} M_0^r (23 + 40M_0^r) \right) \dots$$

$$+ \frac{1}{6} M_0^r (3 + 16M_0^r) r_1^r \tau_{in} \tau_{cat} + 2M_0^r \tau_{cat} (M_0^r r_1^{r^2} \tau_{cat} - r_2^r)$$

2.3.8 The Basic Kinetic Coefficients

The kinetic coefficients arise from the Taylor expansion of the rate term, and as such it can be considered that there are an infinite number of these coefficients, with increasing coefficients indicating an increasing accuracy. Due to the increased influence of noise with higher moments only the first three coefficients can be considered reliable when analysing and simulating TAP experimental data, as the coefficients are directly related to the moments of the exit flow, and in experimental data only up to the second moment can be calculated with any reliability. The benefit of using the kinetic coefficients over other methods is that they are considered to be model free, which means that there is no predefined mechanism when analysing the results, and when comparing the coefficients to a model-based method, the difference between the two curves is negligible (figure 2.8), and is only noticeable at small timescales ($t < 0.1$) which have little effect on the various moments.

The basic kinetic coefficients themselves have some physio-chemical meaning. The zeroth coefficient r_0 has the dimension s^{-1} which has been assigned as an apparent kinetic coefficient for the chemical reaction, similar to a rate constant. The first coefficient r_1 is dimensionless and is labelled "*intermediate-gas*" constant which relates the gas with the preceding intermediate gases. Finally the second constant r_2 has the dimension s and has been assigned to the apparent time delay that arises from reactions occurring on the surface of the catalyst. All three of these constants have been solved for exit flow, meaning that from the standard experimental response it is possible to calculate the basic kinetic coefficients, as long as the moments and the residence times in the inert and catalyst zones are known.[29]

For the reactants:

$$r_0^r = \frac{1 - M_0^r}{\tau_{cat} M_0^r} \quad (2.106)$$

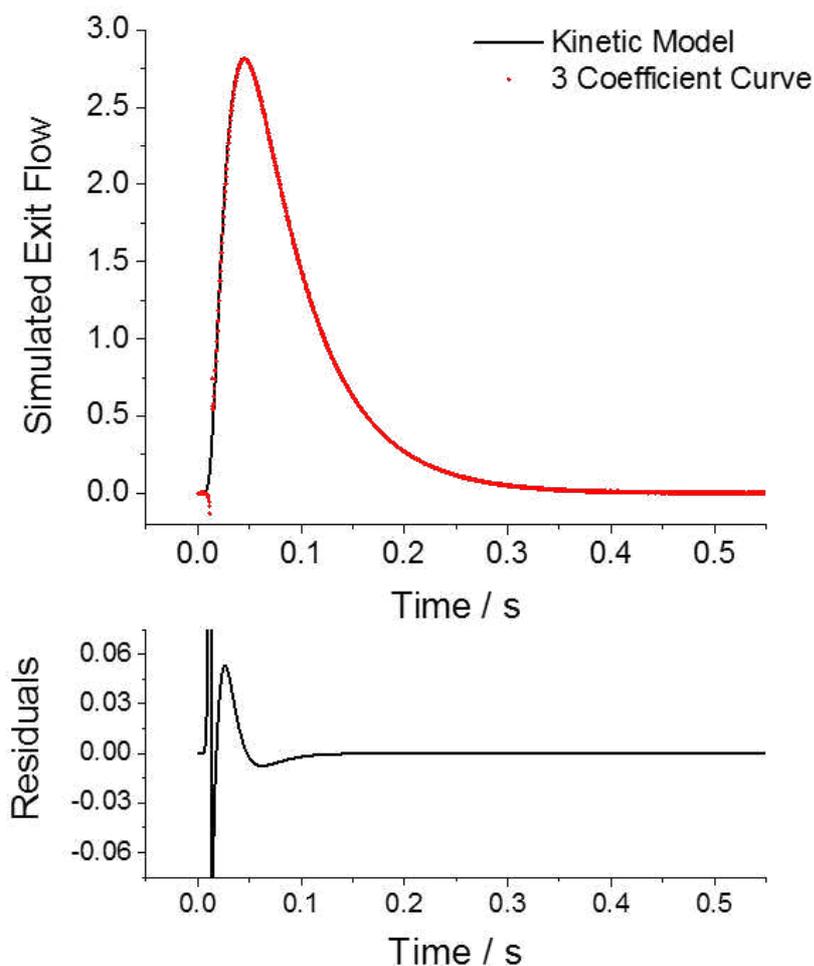


Figure 2.8: Simulated exit flow for a reversible adsorption/reaction mechanism using the kinetic model and 3 coefficient rate terms

$$r_1^r = \frac{M_1^r}{\tau_{cat} M_0^{r2}} - \tau_{in} \left(\frac{1}{\tau_{cat}} + \frac{r_0^r}{3} \right) \quad (2.107)$$

$$-r_2^r = \frac{M_2^r}{2\tau_{cat} M_0^{r2}} - \frac{M_1^{r2}}{\tau_{cat} M_0^{r3}} + \frac{\tau_{in}}{6} \left(\frac{\tau_{in} r_0^r}{5} + 2r_1^r + \frac{\tau_{in}}{\tau_{cat}} \right) \quad (2.108)$$

For the products:

$$r_0^p = \frac{M_0^p}{\tau_{cat} M_0^r} \quad (2.109)$$

$$-r_1^p = r_0^r \left[\frac{M_1^p}{M_0^p} - \frac{\tau_{in}}{12} \left(8M_0^r + 3 + 9 \frac{D_{in}^r}{D_{in}^p} \right) - \tau_{cat} M_0^r r_1^r \right] \quad (2.110)$$

$$\begin{aligned} r_2^p = & \frac{r_0^p}{2} \left[\frac{M_2^p}{M_0^p} - \frac{19D_{in}^r{}^2 \tau_{in}{}^2}{16D_{in}^p{}^2} - \frac{\tau_{in} D_{in}^r}{8D_{in}^p} \dots \right. \\ & \times \left((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat} - 12 \frac{r_1^p}{r_0^p} \right) \dots \\ & - \frac{1}{6} M_0^r (3 + 16M_0^r) r_1^r \tau_{in} \tau_{cat} - 2M_0^r \tau_{cat} (M_0^r r_1^r{}^2 \tau_{cat} - r_2^r) \dots \\ & + \frac{r_1^p}{6r_0^p} ((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat}) \dots \\ & \left. - \tau_{in}{}^2 \left(\frac{5}{48} + \frac{1}{45} M_0^r (23 + 40M_0^r) \right) \right] \quad (2.111) \end{aligned}$$

As it can be seen from equations 2.106 to 2.111 that solving the exit flow for the basic coefficients is a consecutive process. Due to this processes any errors in the calculation of the moments and coefficients is enhanced massively when one reaches the higher coefficients. As the accuracy decreases with increasing coefficient it is often that the zeroth coefficient is only calculated as the relative error is quite low, and it provides the most chemical information (apparent rate constant). If one is able to calculate the kinetic coefficients with accuracy then it is possible to apply a kinetic model, and then solve for the various rate constants in that model. The rate terms, and hence the coefficients have been solved for multiple mechanisms and are summarised in appendix 7.1 taken from [27]. The solution for a simple adsorption/desorption/reaction mechanism (figure 2.9) is shown below. It was shown that the various kinetic coefficients relate to the rate constants through the following equations:

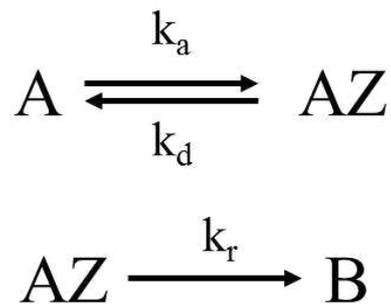


Figure 2.9: Reversible Adsorption and Reversible reaction mechanisms with rate constants

$$r_0 = \frac{k_a k_r}{k_d + k_r} \quad r_1^r = \frac{k_a k_d}{(k_d + k_r)^2} \quad \frac{r_2}{r_1} = \frac{1}{k_d + k_r} \quad (2.112)$$

It is then possible to solve for the rate constants through a simple rearrangement:

$$k_a = \frac{r_0(k_d + k_r)}{k_r} \quad k_r = \frac{1}{|r_2/r_1|} - k_d \quad k_d = \frac{1}{|r_2/r_1| \left(\frac{\frac{r_0}{|r_2/r_1|} + 1}{r_1} \frac{1}{|r_2/r_1|^2} \right)} \quad (2.113)$$

Using a combination of these methods it is possible to completely model and understand a large number of reaction mechanisms using the TAP reactor. The TAP reactor is a very powerful tool when considering the potential quality of the data it can generate, but as it can be seen the analysis of the data is highly complex and extremely difficult to perform without assistance from a piece of dedicated analysis software. The main method for analysing TAP data is to perform the moment analysis on the exit flow reposes, and using the moments to calculate the basic kinetic coefficients, and then if possible attempting to assign a reaction model and calculating the rate constants. Another method which we propose is to use the solutions for the exit flow equations for TZTR setup and perform a regression analysis using the same method for the TPR analysis, and use the outputted parameters to calculate

the basic kinetic coefficients / rate constants.

2.4 Density Functional Theory

This section will outline the underlying theory of Density Functional Theory, and how it has been applied to the work performed in this thesis. The simulation package used for these simulations is the Vienna Ab Initio Simulation Package (VASP) which is a plane-wave Density Functional Theory (DFT) code. Inside the VASP package there are various extensions and functions used alongside the standard DFT for example: DFT-D3, Nudged Elastic Band methods and Molecular Dynamics. The theory behind these various methods are well established in the literature and is outlined in various textbooks,[30–35] as such the majority of the derivations of various equations are left out for the sake of simplicity.

2.4.1 Understanding Ab-Initio codes

Ab-Initio (or *first-principles*) codes are interested in the nuclei and electrons of atoms, and use their interactions with each other to describe the properties of various systems. The mathematics behind these codes is based on the theory of quantum mechanics. The driving force behind quantum mechanical codes is the Schrödinger equation put forward by Erwin Schrödinger in 1926.

$$\hat{H}\Psi(r_i, r_j, t) = E\Psi(r_i, r_j, t) \quad (2.114)$$

Where \hat{H} is the Hamiltonian operator, Ψ is the wave function, E is the system energy, the parameters r_i and r_j are the coordinates of the electron and nucleus respectively, and finally t is time. In theory equation 2.114 can be solved for every atom in the atomic system and give a full representation of the interactions between them, but this is practically impossible for any system larger than a simple hydrogen

atom as the number of variables expands exponentially with increasing complexity. In order to simplify the problem the factor of time is removed from equation 2.114, which can be performed by only studying the ground state of electrons, meaning that the potential energy of the system is constant as a factor of time. To simplify the equation further, the nuclei of the system can be ignored as well. As the nuclei are far larger than the electrons it can be considered that any change in the position of the nucleus instantaneously effects the electrons, and as such the electrons are always in the ground state for any given position of the nuclei. This means that from the position of the electrons the nuclei are considered to be frozen in space, so instead of being a variable in the equation, the nuclei can be considered a constant parameter instead. This means the energy of the electrons and the nuclei can be decoupled, and this decoupling is known as the Born-Oppenheimer approximation.

$$E_{atom} = E_{nucleus} + E_{electron} \quad (2.115)$$

The Born-Oppenheimer equation shows that the energy of the atom is the sum of the nuclear and electronic energies. Applying these simplifications to equation 2.114 we end up with the *time-independent* Schrödinger equation

$$\hat{H}(r)\Psi(r) = E\Psi(r) \quad (2.116)$$

in which the Hamiltonian operator, and the wave function are now only functions of the electron coordinate r . The *time-independent* Schrödinger, equation 2.116, shows that all of the information relating to the energy of the system E is contained within the wave-function Ψ , which means that through selection of the correct Hamiltonian operator \hat{H} and wave-function Ψ the energy for the system can be calculated.

One of the first methods for calculating this energy is known as the Hartree-Fock (HF) approach, which attempts to solve equation 2.116 using the following assumptions:

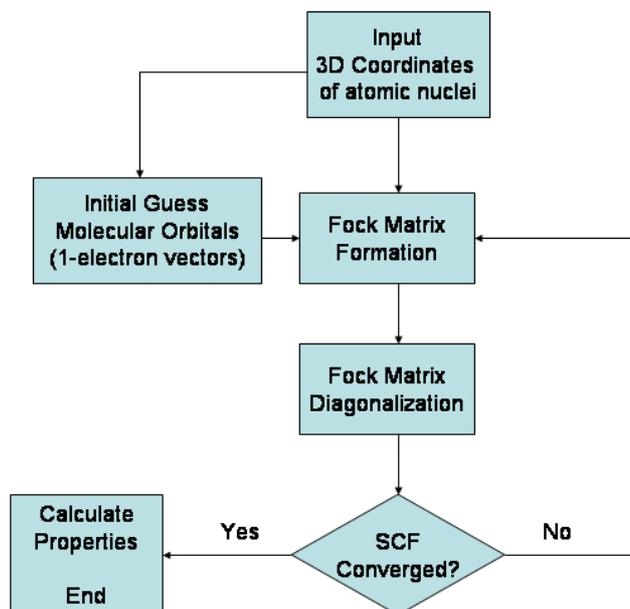


Figure 2.10: Simplified Hartree-Fock method taken from [36]

- Born-Oppenheimer approximation held true
- Relativistic effects are negligible
- The solution is a linear combination of finite orbitals
- Each energy is described by a single Slater determinant
- Correlation between electrons of opposite spin is zero
- Any trial wave function will always have an energy equal or greater than the true value

Using these assumptions it is possible to build an initial guess of the wave function, calculate the value of the function at that point, then then vary the values of the wave function until the value converges (see figure 2.10) essentially performing an optimisation of the wave function in order to calculate the true wave function, and hence the energy of the system.

2.4.2 Moving on from Hartree Fock

The Hartree-Fock method has been tested extensively over the years, and has proven to work well for very simple systems consisting of several atoms, however in practice this is not useful as several atoms can rarely be used to study properties of materials. The first breakthrough from the Hartree-Fock method was the 1964 paper by Hohenberg and Kohn [37] which provided two theories that it was possible to use the electron density instead of the wave function in order to calculate the energy of the interacting system. The first theory was that the ground state electron density of the system will provide a unique external potential (interaction between nuclei and electrons):

$$\rho(r) \iff U_{ext} \quad (2.117)$$

with $\rho(r)$ being the electron density, and U_{ext} the external potential. The internal potential (the electron kinetic energy and electron-electron interaction energy) is independent of the external energy (and subsequently the electron density) and is considered to be universal. The first theory can then state that as long as the system is at its ground state the electron density can be used to define the external potential (and vice versa). The internal energy of the system is universal and hence a universal functional should be used (although this is not explicitly known). Finally that different Hamiltonians will only differ by their external potentials, and therefore the electron density can be used to define the Hamiltonian, the wave function, the external potential, and therefore all of the properties of the system in its ground state. It is this idea that forms the basis for Density Functional Theory. The second theory ascertains to the deduction of the minimum energy of the system, and relates to the method of finding the ground state for a system through the *variational principle*. If we provide a system at a given external potential U_{ext} then by minimising the energy of the system by varying the electron density the ground

state electron density can be calculated, and hence ground state energy of the system can be calculated.

$$E[\rho(r)] = F[\rho(r)] + E_{ext}[\rho(r)] \geq E_{gs} \quad (2.118)$$

Where $E[\rho(r)]$ is the current energy of the system, $F[\rho(r)]$ is the internal energy, and $E_{ext}[\rho(r)]$ is the external energy all as a function of the current electron density. While this theory is very powerful it can be limited by the fact that while we know the ground state electron density exists we have no idea what the true value of the density is, so a system which is far from it's ground state electron density can take a long time to find the ground state.

2.4.3 Kohn and Sham and the Origins of DFT

Actually calculating the ground state electron density using the Hohenberg and Kohn methods turned out to be more difficult than originally thought due to the poorly written electronic kinetic energy as a function of electron density. Expanding on the Hohenberg and Kohn method, the Kohn-Sham[38] method was developed, which formed the modern Density Functional Theory still in use to this day. Kohn and Sham decided that given the principles outlined by Hohenberg and Kohn, a system could be built up using single electrons which removes the issue with the electronic kinetic energy. In order to fully understand how the Kohn-Sham method works we first need to consider the Hamiltonian from the n -electron system from the time-independent Schrödinger equation used in HF:

$$\hat{H} = -\frac{1}{2} \sum_{i=1}^n \nabla_i^2 - \sum_{I=1}^N \sum_{i=1}^n \frac{Z_I}{|r_i - r_j|} + \frac{1}{2} \sum_{i \neq j}^n \frac{1}{|r_i - r_j|} \quad (2.119)$$

where r_I and Z_I are the coordinates and the charges of the nuclei, and r_i and r_j are the coordinates of the electrons. The first term is the kinetic energy of the system,

the second is the external potential, and finally the last term is the Hartree potential (electron-electron interactions), which is summed over $i \neq j$ in order to remove any self interaction. This means that the energy of the system can be considered a simple summation of the individual energies:

$$E = E_{kin} + E_{ext} + E_H + E_x \quad (2.120)$$

with the four terms representing the kinetic, external, Hartree and exchange energies. Kohn and Sham then assumed that the electrons in their one-electron system were all non interacting and in their ground state. This meant that the energy of the system could be split into components consisting of interacting and non-interacting electrons, giving the standard energy term used in DFT.

$$E = E_{kin}^{non} + E_{ext} + E_H + E_{xc} \quad (2.121)$$

Where E_{kin}^{non} is the kinetic energy of the non interacting system, and the term E_{xc} is the exchange correlation energy of the system, which is made up of the correlation and exchange energies:

$$E_{xc} = E_x + E_c \quad (2.122)$$

$$E_c = E_{kin}^{int} + E_c^{int} \quad (2.123)$$

where E_x is the exchange energy, which relates to interactions between electrons with the same spin arising from the Pauli exclusion principle, and E_c is the correlation energy, which relates to the ability for electrons of opposite spins to occupy the same orbital and is made up of correlation due to electron-electron interactions and the kinetic energy of these interactions. When calculating the energy of the system using equation 2.121 the first three terms are elementary to calculate using the

Kohn-Sham equations which can be found in a multitude of textbooks, the final term, the exchange correlation energy contains all of the quantum effects of the system, and has no explicit solution, meaning it has to be approximated.

2.4.4 The Exchange Correlation Energy

In order to complete equation 2.121 the exchange correlation energy needs to be included. As the exchange correlation energy has no exact mathematical solution for an n -electron system, an approximate model is used. There are two main methods for performing the estimation of the exchange correlation energy, these are known as the Local Density Approximation (LDA) and the Generalised Gradient Approximation (GGA). The former splits the total electron density for the system into smaller uniform pieces (see figure 2.11a) which have a uniform electron density. As the electron density is constant across each of the pieces it is possible to calculate the exchange correlation energy for the system at that state, the total exchange correlation is then considered to be the sum of the individual exchange correlation energies. The LDA approximation is considered to be an outdated approach and has multiple issues such as over predicting binding between atoms causing incorrect lattice parameter estimation, and does not predict correct ground states for transition metals, and therefore the more modern GGA approach tends to be used instead. The GGA approach attempts to include the complex nature of the density function by removing the homogeneity of the LDA approximation by taking the gradient of the electron density at a given point giving a more realistic picture of the electron density (figure 2.11b). As the density functional is more accurate the subsequent exchange correlation functional is considered to be more accurate and therefore the results of the DFT calculation are considered to be more accurate. The GGA function is very well established in all modern codes (*e.g.* VASP) with the most popular approximations being the Perdew and Wang[39] functional known more commonly as PW91, and the Perdew, Burke, and Ernzerhof functional[40] known more com-

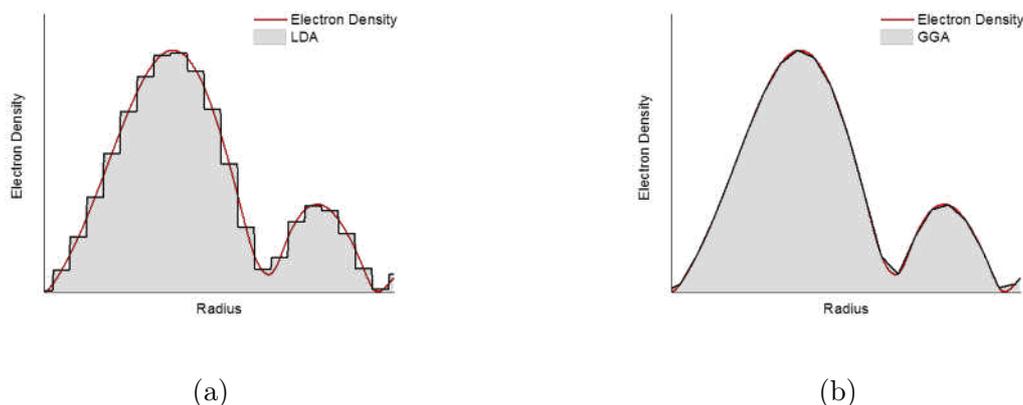


Figure 2.11: LDA and GGA approximations using a simplified 2D model a) Example of LDA approximation b) Example of GGA approximation

monly as PBE. The latter of the two is the most extensively used functional and has been shown to be accurate and efficient at calculating properties of the majority of systems, although it does have its own shortcomings (no inclusion of long range effects and errors in predicting binding energies for some complex organometallic systems) it is generally considered to be the accepted standard for DFT calculations and is used extensively throughout the work in this thesis.

There are many other functionals available, such as hybrid and meta-GGA, but while they often tend to counteract the shortcomings of the more popular functionals and are usually more accurate they are often far more computationally expensive, or designed with specific systems in mind. As such these functionals were not used in the work outlined in this thesis, and therefore can be ignored for the sake of brevity.

2.4.5 Calculating the Total Energy of a System

The most efficient method for calculating the ground state electron density of the structure is through an iterative method similar to the HF approach 2.10. It works by constructing an initial electron density based on the electronic configuration of the system (*e.g.* by combining the individual densities of the Cu atoms to form Cu metal) which provides an initial guess of the electron density of the system. The

next step involves evaluating the Kohn-Sham energy Hamiltonian:

$$\hat{H}_{ks}(\rho(r)) = E_{kin}^{non}(\rho(r)) + U_{ext}(\rho(r)) + U_H(\rho(r)) + U_{xc}(\rho(r)) \quad (2.124)$$

which takes a similar form to equation 2.121, once the Hamiltonian has been evaluated at a given density the Kohn-Sham equation is solved.

$$\hat{H}_{ks}\phi_i = E_i\phi_i \quad (2.125)$$

As it can be seen the KS equation is in a similar format to the time-in dependent Schrödinger equation from section 2.4.1. This means that solving the equation will give you a new KS orbital, which in turn gives you a new electron density and a new energy of the system. The process is then repeated until the energy converges within a preset limit (in a similar fashion to the HF optimiser). Once the system has converged the total energy of the system is then calculated and the resulting forces on the atoms can be used to optimise the position of the atoms.

2.4.6 Vienna Ab-initio Simulation Package

As mentioned previously VASP has been simulation package which was used for the various DFT calculations performed in this thesis. VASP has the ability pseudopotentials, or the Projector Wave Method (PAW) with a plane wave basis set.[41–47] The code has been used for simulations of materials, interactions between catalyst and substrates, and quantum mechanical Born-Oppenheimer molecular dynamics. The Kohn-Sham equations for calculating the ground state properties of a system are solved using an efficient iterative matrix diagonalisation method, with Pulay/Broyden charge density mixing. This leads to accurate calculations for transition metal systems, and by using the forces on the atoms allows for optimisation of the geometry of the system.

The PAW method in VASP is implemented in VASP in order to correctly simulate first-row transition metal elements with minimal effort, and also provides access to the full wave function when generating the density functional. In the PAW method the core electrons are considered to be frozen, and only the valence electrons are considered. The PBE functional has been used for all calculations performed in this thesis, the PAW functionals for PBE were taken from the VASP database.

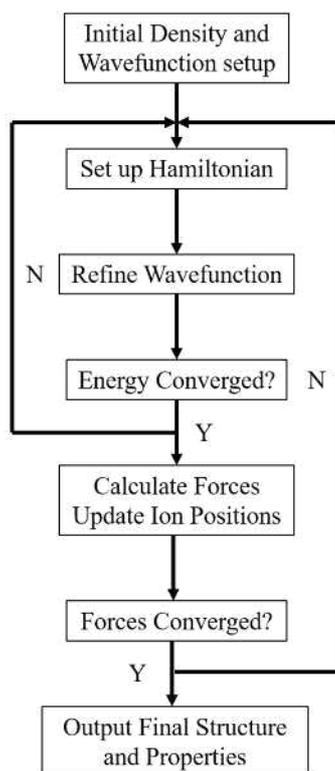


Figure 2.12: Flow chart of DFT optimisation in VASP

The optimisation of the density functional in VASP consists of two loops (figure 2.12). The density and wave function are optimised in the inner loop, and then the forces and atom positions are optimised in the outer loop. Inside the inner loop the method used to optimise the wave function is set by the *ALGO* tag. They can be optimised using a blocked algorithm (*ALGO = normal*), a residual minimisation scheme - direct inversion of the iterative subspace, known as RMM-DISS (*ALGO = Very Fast*), or a mixture of the two algorithms (*ALGO = Fast*). For all of the work performed in this thesis *ALGO = Fast* has been used.

For the optimisation of the outer loop (movement of ion positions), the *IBIRON*

tag is used to specify which method to use. For all of the calculations in this thesis $IBIRON = 2$ has been used, which uses a conjugate gradient method for the minimisation of the forces and atomistic positions, which is the recommended setting in VASP. The conjugate gradient method is a very common optimisation method for linear equations and is well established in the literature.[48]

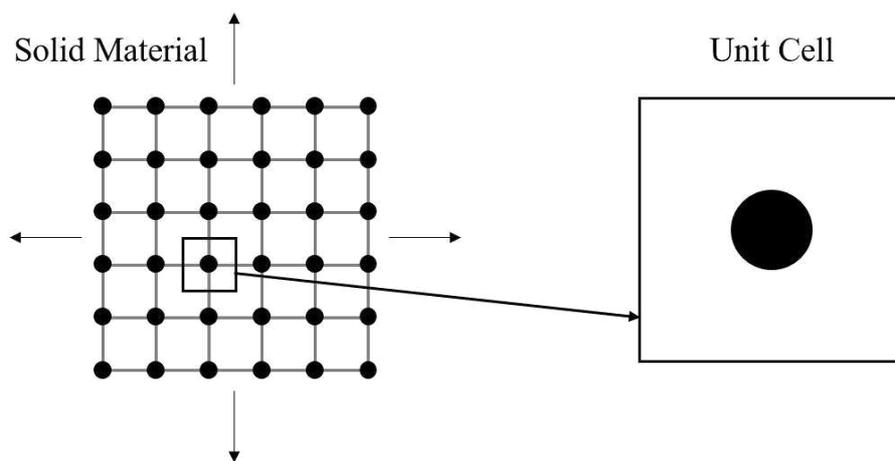


Figure 2.13: Schematic showing how to define a unit cell simplified to two dimensions

When studying a material it is not possible to simulate the almost infinite number of electrons required to replicate the material properties on the macroscopic scale, and hence a method of reducing the number of atoms in the system is required, and this is performed using Periodic Boundary Conditions. In order to replicate the material in three dimensions the unit cell needs to be defined (see figure 2.13), which is the simplest form of the 3D periodic structure, and the properties of the system are calculated using that unit cell. This reduces the total number of atoms used in the calculation. The actual calculation of these properties involves the complex transformation of the unit cell into reciprocal space and then into k -space. The process is described very well in *Computational Materials Science: An Introduction*[49]:

- The solid is reduced into a supercell consisting of several unit cells and is expanded to infinity by the Periodic Boundary Conditions
- The supercell is transformed to reciprocal space and is contained within a first Brillouin Zone

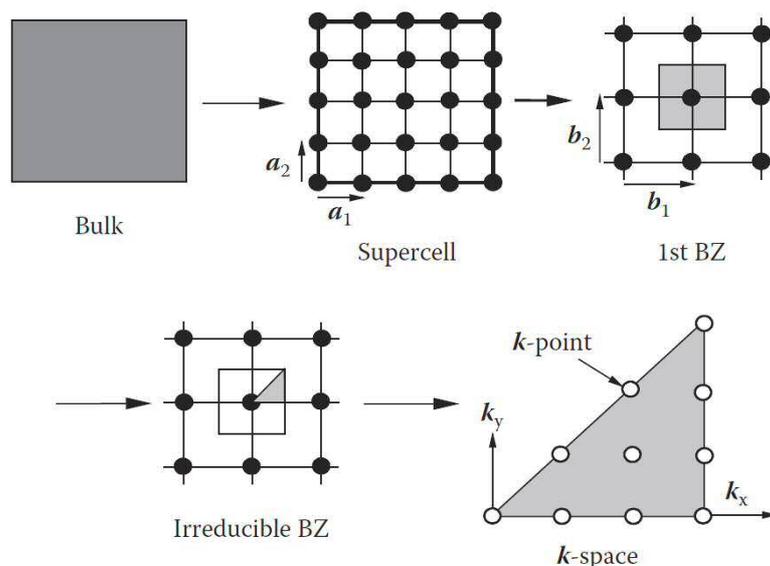


Figure 2.14: Treatment of solids using Periodic Boundary Conditions [49]

- The wave functions are mapped using two vectors k and G which are the wave vector and reciprocal lattice vector respectively.
- The first Brillouin Zone is then converted into its simplest form known as the irreducible Brillouin Zone
- The irreducible Brillouin Zone is then mapped using a grid of k -points and then by integration/summation/extrapolation of these points all of the properties of the infinite system can be obtained.

When replicating the true plane wave of the solid in reciprocal space, a periodic wave function and a plane wave function are summed together to generate a simulated wave function. When the number of plane waves summed to the periodic function rises to infinity then the simulated wave function can be said to be a true replication of the real wave function. Unfortunately, summing an infinite number of wave functions is impossible, and hence a cutoff for the energy of the plane waves is included in calculations with the *ENCUT* tag in VASP. It can be considered that the higher energy (and higher frequency) wave functions have little effect on the overall shape of the simulated function, and after a certain point the energy of the system will converge upon a certain value. Practically this is performed by a plane

wave cutoff test.

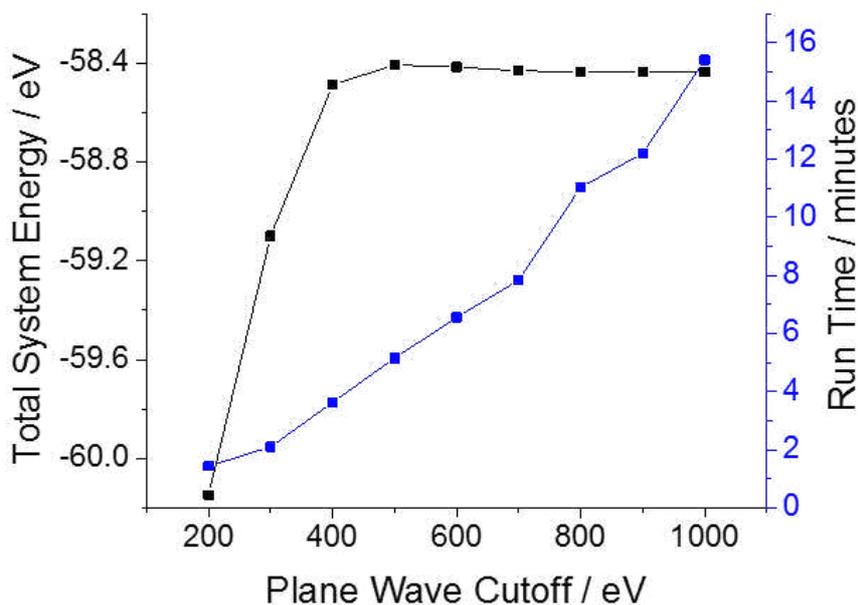


Figure 2.15: Convergence in plane wave energies for a Zr_2O_4 unit cell

As it can be seen in figure 2.15 the total energy of the system converges at > 500 eV but the increase in run time is still linear. Alongside setting the cut off energy for the plane wave, the total number of k -points to be used in the k -grid also needs to be set. This is performed via a similar method where instead of varying the plane wave cutoff energy, the size of the k -point grid is varied instead. The k -point grid is assigned using the *KPOINTS* file in VASP, and the total number of k -points per lattice vector is correlated to the size of that particular lattice vector (*i.e.* The larger the unit cell in the ab or c direction the less k -points required). A k -point expansion test is usually performed on the unit cell of the bulk structure being studied, so that the correct properties can be calculated. For larger systems using anything other than a single k -point in each direction becomes far too computationally expensive as the reciprocal space becomes very small and a single k -point is more than sufficient to describe the Brillouin Zone. As it can be seen in figure 2.16 setting the correct k -grid has a large effect on the computation time of the calculation, with the run time increasing exponentially with increasing k -grid density.

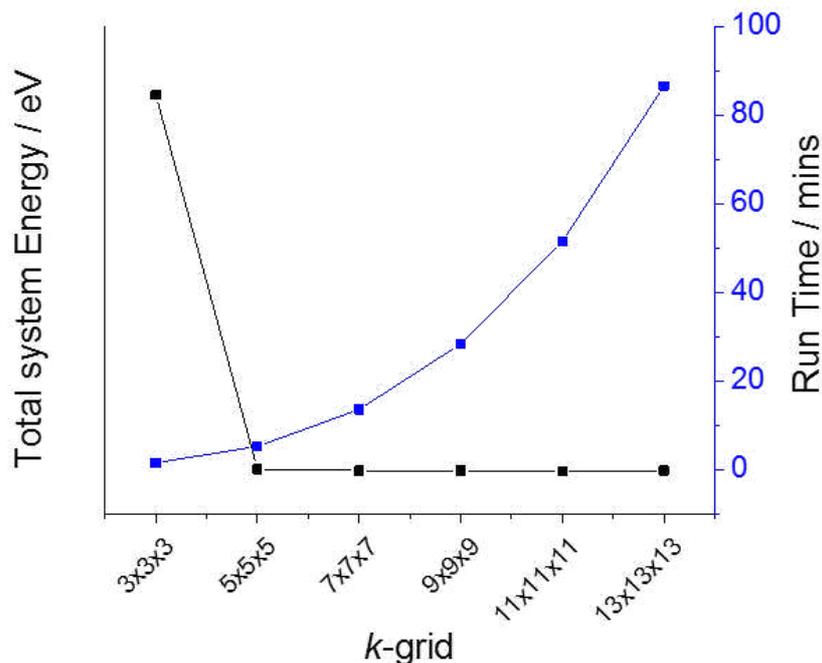


Figure 2.16: Convergence in k -point cutoffs for a Zr_2O_4 unit cell

The final step in setting up the calculation is setting up the partial occupancies of the orbitals, and this is performed using the *ISMEAR* tag in VASP. The calculations in this thesis all use either metallic smearing $ISMEAR = 1$ which uses the method of Mathfessel-Paxton in order to set the occupancies, or Gaussian smearing $ISMEAR = 0$ for semiconductors and metallic nanoparticles. The width of the smearing is defined by the *SIGMA* tag, with a standard setting of 0.2 eV used for metallic smearing, but in the case of Gaussian smearing a value of ≤ 0.05 is often required in order to get the correct occupancy of the orbitals.

2.4.7 Long Range Dispersion Corrections

Using the previous parameters discussed it is possible to perform a optimisation of a structure using VASP, but in order to have a more accurate representation of the structure sometimes extra corrections outside of the standard DFT calculation are required. One specific correction is for long range dispersion (van-der-Waals) forces which are dynamical correlations between fluctuating charge distributions. The

corrections are included in the calculation by simply summing up the standard Kohn-Sham energy calculated using DFT and the energy that arises from the correction:

$$E_{DFT_{D3}} = E_{DFT} + E_{D3} \quad (2.126)$$

Where E_{DFT-D3} is the Grimme D3 corrected system energy, E_{DFT} is the energy from the standard DFT calculation and finally E_{D3} is the energy that arises from the dispersion correction. There are multiple methods for perform dispersion correction calculations, but the one used in this work is the Grimme D3 correction[50]. The following expression is used for calculating the dispersion interaction:

$$E_{D3} = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_L \left(f_{d,6}(r_{ij,L}) \frac{C_{6ij}}{f_{ij,L}^6} + f_{d,8}(r_{ij,L}) \frac{C_{8ij}}{r_{ij,L}^8} \right) \quad (2.127)$$

$$f_{d,n} = \frac{s_n}{1 + 6 \left(\frac{r_{ij}}{s_{R,n} R_{0ij}} \right)^{-\alpha_n}} \quad (2.128)$$

$$R_{0ij} = \sqrt{\frac{C_{8ij}}{C_{6ij}}} \quad (2.129)$$

where i and j are atom indexes, s_6 , s_8 , and s_R are damping function parameters that depend on the choice of exchange functional, C_{6ij} and C_{8ij} are geometry dependant dispersion coefficients, N is the number of atoms in the system, and L is a translation number across unit cells. In VASP the D3 correction can be included by setting the tag `IVDW = 11` and including the following tags:

<code>VDW_RADIUS</code>	= Cutoff radius for pair interactions
<code>VDW_CNRRADIUS</code>	= Cutoff radius for calculation coordination number
<code>VDW_S6</code>	= Functional dependant damping parameter
<code>VDW_S8</code>	= Functional dependant damping parameter
<code>VDW_SR</code>	= Functional dependant damping parameter

The first two tags are system dependant, and for this work have been set at 15Å and 10Å respectively. The final three tags are functional dependant, and for the PBE functional are set as 1.0, 0.722 and 1.217 respectively as taken from the official DFT-D3 website.

2.4.8 Bulk and Surface Calculations

For the majority of catalytic research, the surface of a material is most primary interest, rather than it's bulk properties, but before you are able to calculate the properties of the surface, the properties of the bulk must be correctly set. The first step when accurately calculating the properties of the bulk is to select the correct k -grid and the correct *ENCUT* value using the procedures outlined in figure 2.15 and 2.16. Once the correct values have been selected, the ion positions and the shape and size of the unit cell need to be optimised. Optimising the ionic positions is elementary and can be performed via a simple VASP calculation, the optimisation of the unit cell is slightly more difficult. There are two methods of optimising the unit cell, the first is to manually change the size of the unit cell and record the bulk modulus, then use a Murnaghan[51] equation of state in order to calculate the optimal size of the unit cell, or the in-built VASP optimiser can be used. The latter was used for calculating the bulk properties in VASP, and it can be set by using the *ISIF=3* tag in the *INCAR* file when running a VASP calculation. The problem with running the optimisation using VASP is that as the size of the unit cell is fluctuating the Pulay stress needs to be accounted for, this can be accounted for by setting the *ENCUT* value at least 1.3× its converged value as stated in the VASP manual.

Table 2.3: Unit cell parameters for t-ZrO₂

Method	a	b	c	Volume / Å ³
Cell Optimisation	3.61	3.61	5.25	68.44
Murnaghan	3.57	3.57	5.16	66.42
Experimental	3.58	3.58	5.16	65.88

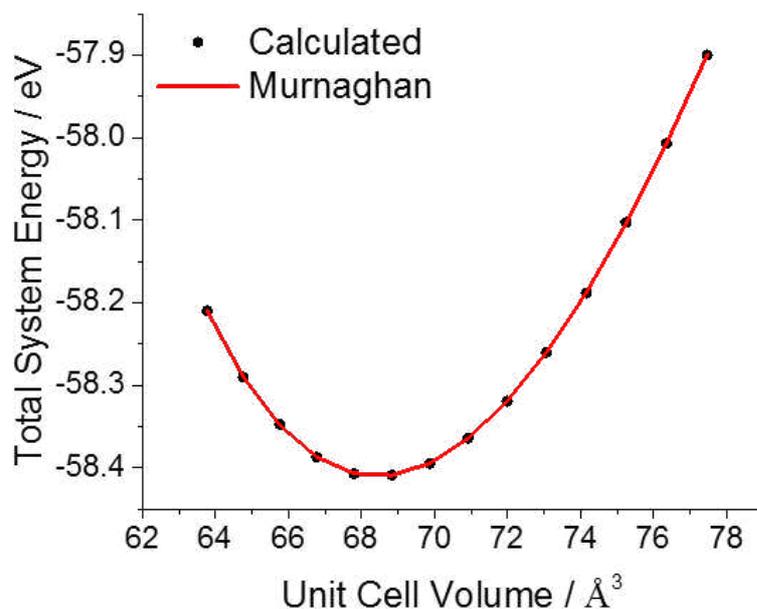


Figure 2.17: Murnaghan fit for t-ZrO₂ unit cell

It can be seen from 2.3 that both methods give an accurate prediction of the real lattice parameters (values taken from ICSD 9993[52]), but it can be seen that the Murnaghan fit solver was more accurate for estimating the lattice parameters overall when compared to the *ISIF* method. The downside to using the Murnaghan solver method is that you need multiple data points (figure 2.17 in order to fit the Murnaghan equation accurately for the system). This means that multiple calculations are required for the calculation of the bulk properties for a single system, for the *ISIF* method only a single optimisation is required meaning it is more efficient. For the work outlined in this thesis we decided to perform this trade off in accuracy for efficiency as a large number of bulk relaxations were required to be performed. Once the bulk has been optimised, the surface can be cut from the bulk unit-cell and used in further calculations.

2.4.9 Transition State Searching

When calculating a reaction mechanism, it is often important to understand how that reaction mechanism evolves, and any activation energy barriers that are present. To calculate the activation energy barrier for a certain process, the energy of the transition state is required. Unfortunately for the majority of processes the structure of the transition state is not known, and hence its energy can not be calculated. One method of attempting to find the structure of the transition state, and the minimum energy path for the reaction is the Nudged Elastic Band (NEB) method with climbing image modifier.[53] The NEB method works by taking the start and end point of a reaction mechanism (the two minima) and interpolating between them generating a number of images (figure 2.18).

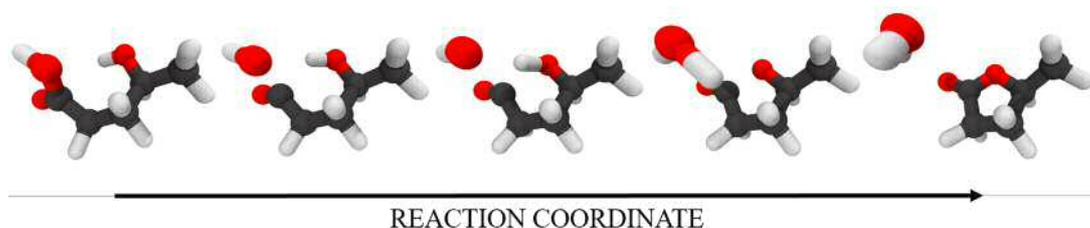


Figure 2.18: Interpolated reaction for conversion of LA to GVL

The images are connected to one another by a spring, so that when the structures are optimised they do not fall into either the start or end minima. This relaxation along the images is then repeated until a minima is reached, it is at this point that the minimum energy pathway is said to have been found. The climbing image modifier is applied in order to make sure that the highest energy image lies on the transition state.

The NEB method is implemented in the VASP package, but a third-party extension is available called VASP Transition State Tools (VTST), and is known to be more accurate and efficient for performing these types of calculations.[53–57] The two methods used to optimise the NEB calculation are the Quick Minima (QM) method, which is similar to the method implemented in the standard VASP code,

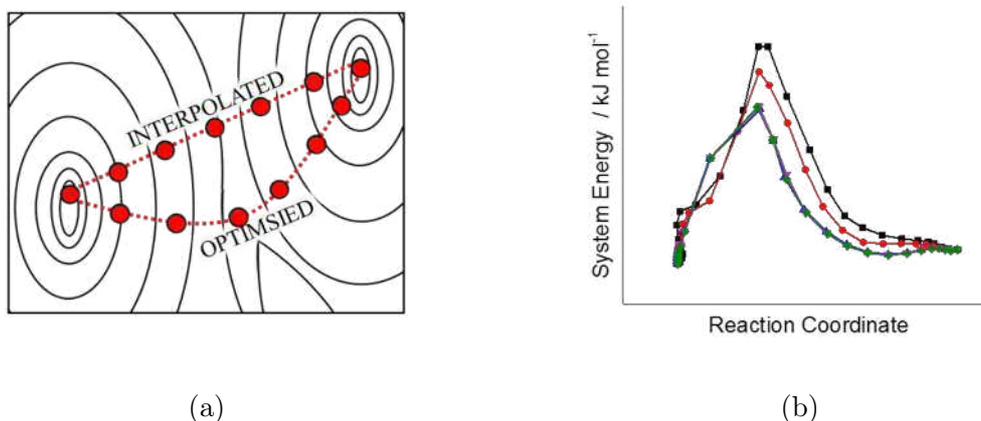


Figure 2.19: Graphical representation of NEB method a) Contour plot of reaction pathway in NEB calculation b) Optimisation of energy in NEB calculation

the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS), and the Conjugate Gradient method, the latter two are more accurate methods for minimising the forces on the ions. Example *INCAR* files can be seen in code snippets 7.1 to 7.3. The general methodology for performing the optimisation is to use the QM method in order to make sure the forces on the ions are at an acceptable starting point, and then use the more accurate optimisers to find the transition state and minimum energy pathway.

The shortfall of the NEB method is that it is using single image snapshots to try to replicate a dynamic system. While this is a fair analogy for simple systems, if there is a solvent or some other component with multiple degrees of freedom, this analogy tends to fall short. In order to capture the dynamics of the system, a method known as Umbrella Sampling (US) can be used. The US method works by taking the same start and end point as it used in the NEB calculation and interpolating the images, but instead of performing a relaxation of the ions, an ab-initio molecular dynamics calculation is performed at each step along the reaction pathway. In order to stop the system from entering non-physical states, or to stop it from relaxing down to the start and end images a potential wall is added to the system blocking it from entering certain energies (see figure 2.20a). By sampling the full reaction pathway it is possible to capture the dynamics of the evolving reaction system which

is more accurate for systems with a large number of degrees of freedom.

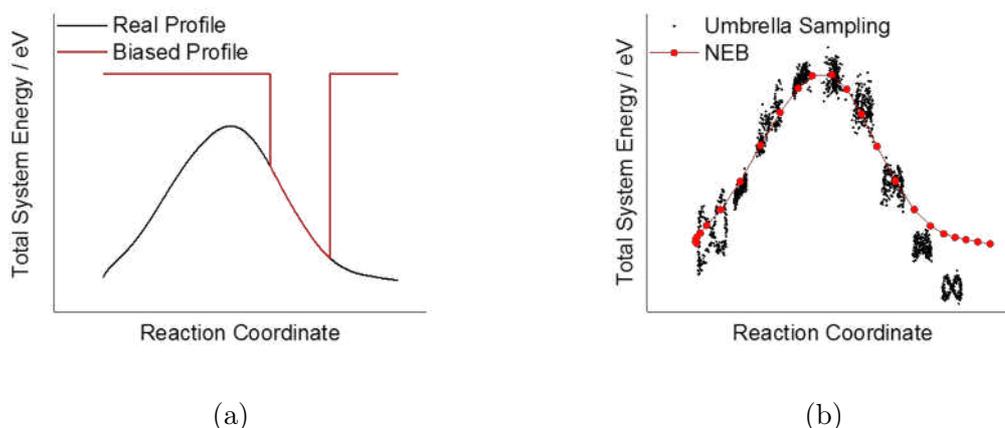


Figure 2.20: Graphical representation of Umbrella Sampling method a) Energy with added potential well (well is exaggerated for clarity) b) Umbrella Sampling v NEB calculation

From using a combination of these two methods of sampling the transition energy pathway an accurate representation of the real dynamic system can be generated.

2.5 Molecular Dynamics

Standard DFT calculations are great for calculating properties of given systems, and are able to calculate these properties to a high degree of accuracy. The issue with the majority of standard DFT work is that it is performed using static images at zero Kelvin, and as such are no dynamic or temperature effects on the system. In order to capture those effects Molecular Dynamics (MD) can be applied. The most popular for of molecular dynamics simulations use classical mechanics instead of quantum mechanics. Differing from DFT, which is mainly focused on the electrons, classical mechanics is only focused on the atom a whole, with the most popular form of using Newtons equations of motion to simulate atom-atom interactions as a function of time. Due to these simpler equations classical MD can be used to simulate systems with millions of atoms, compared to the few hundred of DFT. The world of classical mechanics is focused around the famous Newtonian equation:

$$F = ma \quad (2.130)$$

where F is the Force, m is the mass, and a is the acceleration. It is from this simple equation that it is possible to model how the system evolves as a function of time (backwards and forwards) giving us molecular dynamics.

2.5.1 Describing Atomistic Interactions

To describe how atoms interact with each other, potentials are used. A potential is a simple equation that describes the energy of a system as a function of distance between atoms. The simplest method is consider a system in which the total energy is constant, we can then state that that the force acting on the system is a function of the inter-atomic potential

$$F = -\frac{dU(r)}{dr} \quad (2.131)$$

Where U is the potential as a function of position, and r is the position of the atom. This means that calculating the forces on the atoms is elementary for a given potential and atom positions. The potential can have many forms, and there are many different potentials which have their various uses, for this work the OPLS_2005 force fields[58] in which the total energy of the system is calculated using:

$$E(r^N) = E_{bonds} + E_{angles} + E_{dihedrals} + E_{nonbonded} \quad (2.132)$$

The solution for the bond energy is a harmonic oscillator where:

$$E_{bonds} = \sum_{bonds} k_r (r_{ij} - r_0)^2 \quad (2.133)$$

with k_r being a force constant, r_{ij} is the current bond distance, and r_0 is a set bond

distance. This allows the bond to oscillate around a set point r_0 . For the bond angles a similar harmonic oscillator is used:

$$E_{angles} = \sum_{angles} \frac{k_\theta}{2} (\theta_{ij} - \theta_0)^2 \quad (2.134)$$

this follows the same relationship as equation 2.133, where k_θ is a force constant, and θ_{ij} and θ_0 are the bond angle and set bond angle respectively. The dihedral uses a triple cosine to describe the dihedral angle rotation

$$E_{dihedral} = \sum_{dihedral} = \left(\frac{1}{2}A_1[1 + \cos(\phi - \phi_1)] + \frac{1}{2}A_2[1 - \cos(2\phi - \phi_2)] \right. \\ \left. + \frac{1}{2}A_3[1 + \cos(3\phi - \phi_3)] + \frac{1}{2}A_4[1 - \cos(4\phi - \phi_4)] \right) \quad (2.135)$$

with A being the various force constants and ϕ being the angle measured in the dihedral. Finally we have the expression for non-bonded interactions, which uses a Lennard-Jones potential.

$$E_{nonbonded} = \sum_{i>j} 4\varepsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] \quad (2.136)$$

with

$$\sigma_{ij} = \frac{1}{2}(\sigma_i + \sigma_j) \quad \varepsilon_{ij} = \sqrt{\varepsilon_i \varepsilon_j} \quad \sigma = \frac{r_{min}}{2^{1/6}} \quad (2.137)$$

where equations 2.137 show the combining rules for the constants σ and ε which relate to the minimum energy bond angle and the depth of the potential well respectively.

When calculating the forces on the atoms using the potentials, it is important to

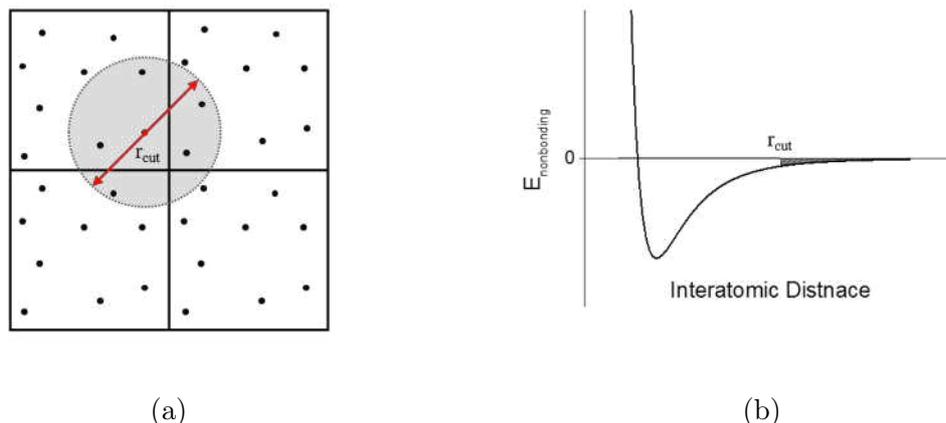


Figure 2.21: Cutoff potentials use in MD simulations a) Graphic showing cutoff in 2D periodic image b) Lennard-Jones nonbonding interaction potential with cutoff

include a cutoff distance for the potentials. The reasoning is twofold, first it helps cut down on computation time, and second in periodic systems the cutoff needs to be smaller than half of the unit cell size, as otherwise an atom could interact with another atom inside the box causing unphysical interactions (see figure 2.21b). Using a combination of these method for calculating the energy as a function of atomistic position, then allows the forces to be calculated on each atom in the system.

2.5.2 Solving as a Function of Time

Attempting to solve the differential equation 2.131 as a function of time (or dr) isn't computationally feasible, so MD employs a finite difference tactic, in which the positions of the atoms is solved at finite time steps by using a Taylor expansion such as:

$$r(t_0 + t) = r(t_0) + \nu(t_0)t + \frac{1}{2!}a(t)t^2 + \dots \quad (2.138)$$

where increasing the number of terms in the expansion increases the accuracy of the calculation of the new positions of the atoms. There are multiple methods for performing this Taylor expansion, with the most popular being the Velocity

Verlet, and Leapfrog algorithms.[59] The Leapfrog algorithm works by calculating the velocities first at a half time step, which are then used to calculate the positions at the full time step, so the velocities and positions are leaping over each other in steps of one time step, hence the name, the equations used in the Leapfrog algorithm are:

$$r(t_0 + dt) = r(t_0) + \nu \left(t_0 + \frac{1}{2} dt \right) dt \quad (2.139)$$

$$\nu \left(t_0 + \frac{1}{2} dt \right) = \nu \left(t_0 - \frac{1}{2} dt \right) + a(t_0)dt \quad (2.140)$$

where the velocities at t are predicted using:

$$\nu(t) = \frac{1}{2} \left[\nu \left(t_0 - \frac{1}{2} dt \right) + \nu \left(t_0 + \frac{1}{2} dt \right) \right] \quad (2.141)$$

In the Leapfrog algorithm the velocities and positions are explicitly calculated but can often fall into issues as the velocities and positions are never calculated at the same time. The Velocity Verlet algorithm can overcome this by calculating the velocities and positions using the following equations:

$$r(t_0 + dt) = r(t_0) + \nu(t_0)dt + \frac{1}{2}a(t_0)dt^2 \quad (2.142)$$

$$\nu(t_0 + dt) = \nu(t_0) + \frac{1}{2}[a(t_0) + a(t_0 + dt)]dt \quad (2.143)$$

As the equations are solved using Euler's method, the smaller the time step, the more accurate the calculation. This means there is a trade off between computational time and accuracy, for the MD runs used in this work a time step of 0.5 femtoseconds

is used which is small enough to provide accurate forces while being large enough to perform millions of steps in a standard calculation.

2.5.3 Accounting for Temperature and Pressure

Temperature and pressure are accounted for in a MD run by studying the velocities and forces acting on the atoms in the simulation. Assuming the system has no net momentum the temperature can be calculated using:

$$T(t) = \frac{\sum_{atoms} m v^2(t)}{k_B f} \quad (2.144)$$

where T is the temperature of the system at a time t , m is the mass of the atoms, k_B is the Boltzmann constant, and f is the number of degrees of freedom ($3N - 3$), where N is the number of atoms in the system.. The pressure in the system can be calculated by using the forces and ideal gas law equation.

$$PV = Nk_B T + \frac{1}{D} \left\langle \sum_i \sum_{j>i} r_{ij} \frac{d\phi}{dr} \Big|_{r_{ij}} \right\rangle \quad (2.145)$$

The simplest type simulation is a Micorcanonical ensemble (NVE) in which the number of atoms N , the volume V and the energy E of the system are kept constant, allowing temperature T and pressure P to fluctuate, and are calculated using equations 2.144 and 2.145. This type of ensemble is often used as it is the most efficient and provides an accurate representation for systems which are well defined. If the system is not well defined, then a Canonical ensemble (NVT) or a Isobaric-Isothermal ensemble (NPT) can be used to set the properties of the system. The NVT ensemble works by setting the number of atoms, volume and temperature as constant values, allowing for the system to relax to a certain temperature. The temperature is controlled by surrounding the simulation cell in a *heat-bath* which

can add or remove temperature from the system accordingly, and these methods of controlling temperature are known as thermostats. The most popular methods for controlling the temperature are the Nosé-Hoover[60], and Berendsen[61] thermostats. The NPT ensemble works by setting the number of atoms, temperature and pressure as constant values and uses a heat bath and a barostat in order to control the temperature and pressure. The most popular are the Hoover[62] and Berendesen barostats, with both scaling the size of the system so that the pressure reaches the set value.

2.5.4 DL_POLY Simulation Package

DL_POLY is an all purpose molecular dynamics simulation package. It can be completely customised to suit your system by selecting the correct algorithm for performing the calculation of the velocities, and by setting a thermostat to control the temperature and pressure of the system. DL_POLY uses input files called *CONTROL*, *FIELD*, and *CONFIG*. The first file is where the tags that decide the type of simulation required are included. The latter two are system dependant and contain the potentials (*FIELD*) and the atomistic positions and velocities (*CONFIG*). Examples of these input files are outlined in appendix 7.4 to 7.6.

2.5.5 Ab Initio Molecular Dynamics

All of the dynamics discussed so far have been based on Newton's equations of motion, while these can be useful for representing systems on the micro-scale, when one wants to study small local interactions between atoms, a more complex model is required. This is where Ab Initio MD (AIMD) comes into effect, this uses the same equations for calculating the movement of the atoms as the classical MD (time steps and integration algorithms), and it is the interaction between the atoms that is calculated using quantum mechanical methods. The first type of AIMD is

Born-Oppenheimer molecular dynamics (BOMD) which calculates the forces on the atoms via electronic minimisation using DFT as outlined previously, it then uses those forces in the classical dynamics equations to move the atoms forward in time.

$$L_{BO} = E_{kin} - E = \frac{1}{2} \sum_i m_i r_i^2 - E[\rho(r), r_i] \quad (2.146)$$

The Lagrangian describing the system derives from the kinetic energy (left term) from classical mechanics, and the electronic energy as calculated from DFT (right term).

This is a highly accurate approach for calculating the dynamics of a system as the "*potential*" being used to describe the system is the highly accurate DFT calculation, but the downside is that this is extremely computationally expensive, and is reserved for very small systems for a very small total simulation time. BOMD is available for use in the VASP package, along with the various thermostats and barostats mentioned in the previous sections. Alongside BOMD there is also the option of using Car-Parrinello Molecular Dynamics (CPMD), which is considered to be more efficient, but this is not available in VASP and is considered outside the scope of this thesis.

Bibliography

- (1) MATLAB, *version 8.5.0.197613 (R2015a)*, The MathWorks Inc., Natick, Massachusetts, 2015.
- (2) R. H. Byrd, J. C. Gilbert and J. Nocedal, *Mathematical Programming*, 2000, **89**, 149–185.
- (3) Z. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly and R. Martı, *INFORMS Journal on Computing*, 2007, **19**, 328–340.
- (4) F. Glover, *Lecture notes in computer science*, 1998, **1363**, 13–54.
- (5) J. Šestak, *Thermal analysis: their measurements and theoretical thermal analysis. Part D, Thermophysical properties of solids*, Elsevier, 1984, vol. 12.
- (6) J. Sestak, *Science of heat and thermophysical studies: a generalized approach to thermal analysis*, Gulf Professional Publishing, 2005.
- (7) S. Vyazovkin, *Recent Advances, Techniques and Applications*, 2008, **5**, 503.
- (8) N. Koga, J. Malek, J. Sestak and H. Tanaka, *Netsu Sokutei*, 1993, **20**, 210–223.
- (9) J. Šestak and G. Berggren, *Thermochimica Acta*, 1971, **3**, 1–12.
- (10) J. Malek and J. M. Criado, *Thermochimica acta*, 1990, **164**, 199–209.
- (11) H. E. Kissinger, *Journal of research of the National Bureau of Standards*, 1956, **57**, 217–221.
- (12) E. Mittemeijer, *Journal of Materials science*, 1992, **27**, 3977–3987.

- (13) J. H. Flynn and L. A. Wall, *J Res Nat Bur Stand*, 1966, **70**, 487–523.
- (14) T. Akahira and T. Sunose, *Res Rep Chiba Inst Technol (Sci Technol)*, 1971, **16**, 22–31.
- (15) S. Vyazovkin and D. Dollimore, *Journal of chemical information and computer sciences*, 1996, **36**, 42–45.
- (16) G. Senum and R. Yang, *Journal of thermal analysis*, 1977, **11**, 445–447.
- (17) L. Perez-Maqueda and J. Criado, *Journal of thermal analysis and calorimetry*, 2000, **60**, 909–915.
- (18) H. L. Friedman, *Journal of Polymer Science Part C: Polymer Symposia*, 1964, vol. 6, pp. 183–195.
- (19) G. Munteanu, L. Ilieva, R. Nedyalkova and D. Andreeva, *Applied Catalysis A: General*, 2004, **277**, 31–40.
- (20) J. Malek, *Thermochimica acta*, 1992, **200**, 257–269.
- (21) J. Criado, J. Malek and A. Ortega, *Thermochimica Acta*, 1989, **147**, 377–385.
- (22) J. Malek, *Thermochimica Acta*, 1989, **138**, 337–346.
- (23) V. Gorbachev, *Journal of Thermal Analysis and Calorimetry*, 1983, **27**, 151–154.
- (24) J. T. Gleaves, G. S. Yablonskii, P. Phanawadee and Y. Schuurman, *Applied Catalysis A: General*, 1997, **160**, 55–88.
- (25) J. Gleaves, J. Ebner and T. Kuechler, *Catalysis Reviews Science and Engineering*, 1988, **30**, 49–116.
- (26) G. Yablonskii, V. Bykov, A. Gorban and V. Elokhin, *Models of catalytic reactions. Comprehensive chemical kinetics*, 1991.
- (27) S. O. Shekhtman, *Interrogative kinetics: A new methodology for catalyst characterization*, 2003.
- (28) S. Shekhtman, G. Yablonsky, S. Chen and J. Gleaves, *Chemical Engineering Science*, 1999, **54**, 4371–4378.

- (29) S. Shekhtman, N. Maguire, A. Goguet, R. Burch and C. Hardacre, *Catalysis Today*, 2007, **121**, 255–260.
- (30) F. Jensen, *Introduction to computational chemistry*, John Wiley & Sons, 2013.
- (31) C. J. Cramer, *Essentials of computational chemistry: theories and models*, John Wiley & Sons, 2013.
- (32) C. Dykstra, G. Frenking, K. Kim and G. Scuseria, *Theory and Applications of Computational Chemistry: the first forty years*, Elsevier, 2011.
- (33) E. K. Gross and R. M. Dreizler, *Density functional theory*, Springer Science & Business Media, 2013, vol. 337.
- (34) W. Koch and M. C. Holthausen, *A chemist's guide to density functional theory*, John Wiley & Sons, 2015.
- (35) P. Balbuena and J. M. Seminario, *Molecular dynamics: from classical to quantum methods*, Elsevier, 1999, vol. 7.
- (36) W. Commons, *Greatly simplified algorithmic flowchart illustrating the Hartree–Fock method*, 2005, <https://upload.wikimedia.org/wikipedia/en/6/6d/Hartree-Fock.png>.
- (37) P. Hohenberg and W. Kohn, *Physical review*, 1964, **136**, B864.
- (38) W. Kohn and L. J. Sham, *Physical review*, 1965, **140**, A1133.
- (39) J. P. Perdew and Y. Wang, *Physical Review B*, 1992, **45**, 13244.
- (40) J. P. Perdew, K. Burke and M. Ernzerhof, *Physical review letters*, 1996, **77**, 3865.
- (41) G. Kresse and J. Hafner, *Physical Review B*, 1993, **47**, 558.
- (42) G. Kresse and J. Hafner, *Physical Review B*, 1994, **49**, 14251.
- (43) G. Kresse and J. Furthmüller, *Physical review B*, 1996, **54**, 11169.
- (44) G. Kresse and J. Furthmüller, *Computational Materials Science*, 1996, **6**, 15–50.

- (45) G. Kresse and D. Joubert, *Physical Review B*, 1999, **59**, 1758.
- (46) P. E. Blöchl, *Physical Review B*, 1994, **50**, 17953.
- (47) J. Perdew, K. Burke and M. Ernzerhof, *Errata:(1997) Phys Rev Lett*, 1996, **78**, 1396.
- (48) K. E. Atkinson, *An introduction to numerical analysis*, John Wiley & Sons, 2008.
- (49) J. G. Lee, *Computational materials science: an introduction*, Crc Press, 2011.
- (50) S. Grimme, J. Antony, S. Ehrlich and H. Krieg, *The Journal of chemical physics*, 2010, **132**, 154104.
- (51) F. Murnaghan, *Proceedings of the National Academy of Sciences*, 1944, **30**, 244–247.
- (52) G. Bergerhoff, R. Hundt, R. Sievers and I. Brown, *Journal of Chemical Information and Computer Sciences*, 1983, **23**, 66–69.
- (53) G. Henkelman, B. P. Uberuaga and H. Jonsson, *The Journal of chemical physics*, 2000, **113**, 9901–9904.
- (54) D. Sheppard and G. Henkelman, *Journal of computational chemistry*, 2011, **32**, 1769–1771.
- (55) D. Sheppard, R. Terrell and G. Henkelman, *The Journal of chemical physics*, 2008, **128**, 134106.
- (56) G. Henkelman and H. Jonsson, *The Journal of chemical physics*, 2000, **113**, 9978–9985.
- (57) B. J. Berne, G. Ciccotti and D. F. Coker, *Classical and quantum dynamics in condensed phase simulations*, World Scientific, 1998.
- (58) J. L. Banks, H. S. Beard, Y. Cao, A. E. Cho, W. Damm, R. Farid, A. K. Felts, T. A. Halgren, D. T. Mainz, J. R. Maple et al., *Journal of computational chemistry*, 2005, **26**, 1752–1780.
- (59) J. Haile, *Molecular dynamics simulation*, Wiley, New York, 1992, vol. 18.

- (60) W. G. Hoover, *Physical review A*, 1985, **31**, 1695.
- (61) H. J. Berendsen, J. v. Postma, W. F. van Gunsteren, A. DiNola and J. Haak, *The Journal of chemical physics*, 1984, **81**, 3684–3690.
- (62) S. Melchionna, G. Ciccotti and B. Lee Holian, *Molecular Physics*, 1993, **78**, 533–544.

3 | Development and Application of TAP Numerical Analysis

The next three chapters will be dedicated to the work that I have undertaken throughout the PhD course. The large majority of the work was performed developing analysis tools (*i.e.* CCI-TAP and CCI-TPR) and here I will explain explicitly the development and testing procedures, and how they have been applied to real catalytic systems. I will also outline the work performed under the NOVACAM project, both experimental and theoretical.

3.1 Development of Analysis Software

This first section is dedicated to the development of the analysis software, which applies the theory used in Section 2.2 (CCI-TPR) and Section 2.3 (CCI-TAP) and implements it into a graphical user interface outlined in Section 2.1. The development of a graphical user interface was seen as the best way to present the analysis tools in an easily usable fashion. The development of this user interface - and the underlying MATLAB code is outlined in this section.

3.1.1 Importing User Data

Before any kind of analysis can be performed, the user must first load in their data. While this may sound elementary, in practice it can be quite difficult. As different machines will have different file formats, and the different file formats will have varying verbosity's when it comes to outputting the data, trying to generate an *all-purpose* input script is an almost impossible task. Instead it was decided to define a *lowest available standard* data type which any file type can be easily generated by a user as a method to load in any data type, alongside having pre-programmed standards for the common file-types that I have used.

When loading in data - the first step grab the location of the file on the computer. This is easily done in MATLAB by using the *uigetfile* function which opens up a file explorer window and allows the user to locate and select the file. Once the file has been located it needs to be opened and read using the *fopen* and *textscan* functions respectively.

```
1 % Opens file explorer window – try/catch is used to save the filepath.
2 [fileName , filePath] = uigetfile({'*.*'}, 'Please select TAP pulse');
3
4 % Open the file and set endFile to 0
5 fid = fopen([filePath fileName]);
6 endfile = 0;
7
8 % Saves the path to handles structure to keep for later
9 handles.path = filePath;
10
11 % Start timer – For Debugging
12 tic
13
14 % Reads through each line of the data
15 fprintf('Reading in File\n');
16 while endfile ~= 1
17     % Grabs each line and saves it as a string
18     textdata = textscan(fid, '%s');
19     data = textdata{1,1};
20     % When end of file is reached sets endFile to 1
21     endfile = feof(fid);
22 end
23 fprintf('Finished reading file with %d lines\n', length(data))
```

Listing 3.1: Snippet of script used to read in TAP data

The listing 3.1 shows how the data is initially loaded. The *textscan* function reads in each line in the file as string and saves it as a cell array, and as such is universal for each format. Once the file has been completely read the script attempts to work out which file-type is being read. This is done by looking for specific markers in each of the text files, for example in the .Tst file type (used by Idaho National Labs TAP-2 machine) the second line of the file will always be a line of asterisks (*) as shown in listing 3.2.

```
Ethylene-2s-23C-1PtSiO2-1.Tst
*****
10 Number of Pulses
1000 Number of Data Points
0 Pulse Delay
2 Collection Time
-0.001468
-0.002438
0.000472
...
```

Listing 3.2: Snippet of .Tst filetype used in INL TAP data

Using these specific markers allows the script to identify which file type is being read. Once the script knows which file type is being read, it can then search for the various parameters that will be saved in the file (*e.g.* Pulse Number, collection time, heating rate). Once the specific parameters are saved the bulk of the data is then stored by combining the cell array *data* (from listing 3.1) in to one large string using the *strjoin* function, and then reading the large string for floating point numbers using *sscanf*. The reasoning behind this seemingly convoluted method is that the functions *strjoin* and *sscanf* are highly optimised and run much faster than a standard loop.

```
1 ...
2 elseif inldata == 0
3     %Data is from INL
4     npoints = str2double(data{7,1});
5     peakno = (length(data)-17)/npoints;
6     handles.datatype = 2;
7     delay = str2double(data{12,1});
8     tpulse = str2double(data{15,1});
9     str = strjoin(data(18:length(data)));
10    d = sscanf(str, '%f ');
11    data = d;
```

```
12 else
13     fid = fopen([filePath fileName]);
14     endfile = 0;
15     fprintf('Data is from Cardiff... loading into arrays\n');
16     %Reads LAS data type
17     d = sscanf(textdata, '%f ');
18     data = d;
19 end
```

Listing 3.3: Snippet of script used to read in formatted TAP data

If it cannot find any of these marks it defaults to the *lowest available standard* file type, which is a text file which contains nothing but the raw output of the machine (e.g. Intensity). The loading of TPR data follows exactly the same format, albeit with small changes regarding variables.

3.1.2 Processing the Users Data

Once the raw data has been imported into MATLAB, it needs to be processed. For TAP data this involves taking the raw data, splitting it apart into each individual peak, and generating the time array. For TPR data, this involves removing the experimental parameters by converting the raw data to a $d\alpha/dT$ thermogram, and generating the temperature array.

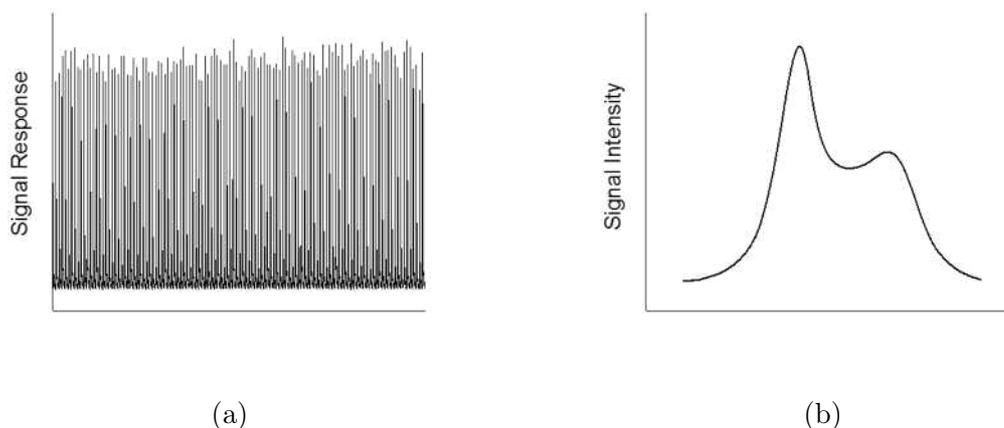


Figure 3.1: Example plots of raw data sets a) Raw TAP experimental data b) Raw TPR experimental data

The raw TAP data as seen in figure 3.1a is a simple 1d array of the experimental

response. The first step is to generate the time array. This is done by reading (or asking the user to manually input) the total number of peaks, and the pulse response time for each peak. The time array can then be interpolated simply using the *linspace* function. The next step is to split the data up into individual peaks, and save into a 2D array, and then generate a time array for each peak.

```

1 %% Generates the peaks
2 % Sets time axis
3 time = linspace(0, tpulse*peakno, length(data));
4 number = length(data)/peakno;
5 dloc = find(time > delay);
6 add = [];
7
8 % Splits the data into peaks if we have more than 1 peak
9 if peakno > 1
10     for j = 1:peakno
11         i = j*number;
12         pt = data(i-(number-1):i, 1);
13         peak(1:number, i/number) = pt;
14     end
15
16 % Required for data from Cardiff TAP-2: Removes final peak.
17 if handles.datatype == 0;
18     %Removes the last peak from peak variable
19     peak(:, peakno) = [];
20     %Removes last peak from the data and time
21     data((((peakno-1)*number)+1):peakno*number) = [];
22     time((((peakno-1)*number)+1):peakno*number) = [];
23 end
24
25 averagepeak = mean(peak, 2);
26 timesingle = linspace(0, tpulse, length(averagepeak));
27 % If we have only one peak take the whole data as the peak.
28 else
29     peak = data;
30     averagepeak = data;
31     timesingle = time;
32 end

```

Listing 3.4: Script used to split up TAP response data.

Once the data has been split up and stored in its correct locations it is stored in a structure which is then saved to the handles workspace as outlined in Section 2.1. For the TPR data the data does not need to be split up (figure 3.1b), but a temperature array instead of time is required. The temperature array is either read directly from the file or generated by knowing the heating rate and the total run

time. Once the temperature array has been generated for TPR data, the da/dT curve is created using a simple numerical integration script. Again the data is stored in a structure which is then saved to the handles workspace.

```
1 %% Creates the da/dT profile for the data
2 % Creates the alpha profile from the raw data
3 alph = cumtrapz(Temp, Signal)/trapz(Temp, Signal);
4
5 % Differentiates the alpha profile creating dalpha/dt
6 dadT = diff(alph)./diff(Temp);
7 dadT(end+1) = 0;
```

Listing 3.5: Script used for numerical integration of TPR response data.

3.1.3 Developing the Front-End GUI

The data itself after processing is stored in the handles of the GUI, which acts a repository where the required variables and arrays can be taken and placed back at will. Importing and storing the data is useful on its own, but attempting to present it and manipulate it in a user friendly format is where a large amount of the difficulty in developing software arises. The GUIDE GUI creation tool in MATLAB has been employed extensively in the generation of this software.

In order to get the user to communicate with the hidden scripts, various control objects are included on the interface (*e.g.* push-buttons). These control objects are functions which are activated on a predefined event (*e.g.* clicking on the button), and allow a simple method of manipulating and performing complex functions by a user without a predefined knowledge of the scripts and mathematics.

The data also needs to be presented to the user, this is performed by using an *axes* object, which acts as a graph allowing any relevant data to be shown for that specific script, an example of some of these plotting and data manipulation commands are highlighted in the appendix listing 7.7.

As the actual generation of a user-interface is not the subject of this thesis, the scripts used for manipulating, moving, and plotting data have no need to be

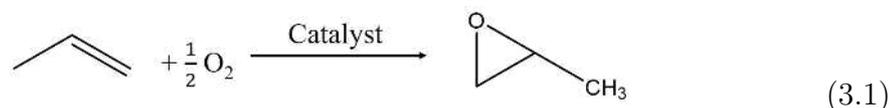
discussed in any great detail. It is the mathematics and programming outlined in later sections which are of most interest relating to this thesis.

3.2 Application of TAP Data Analysis

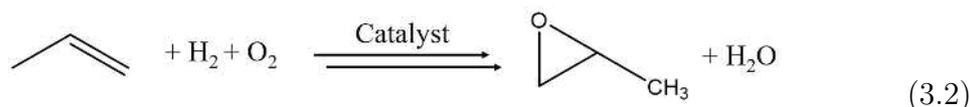
Using the methodology outlined in the previous chapter, this section will discuss how these current methods (and some newly developed ones) have been implemented and subsequently used in the analysis of real TAP data. The analysis software currently has multiple functions available, ranging from simple data manipulation and application of current knowledge, to the development of new frequency analysis tools for baseline removal, noise correction, and the gathering of kinetic information.

3.2.1 Analysis of Experimental Response

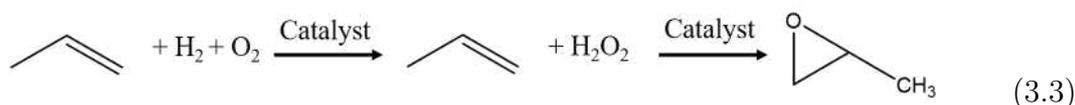
Due to the high time resolution and transient nature of the TAP experiment, often the raw intensity response, without any complex kinetic analysis, is highly insightful into reaction mechanisms and kinetics. The first example of this is in an experiment taken as part of a larger project in which propylene was being oxidised to its epoxide form (propylene oxide). It was known that this reaction can occur through simple oxidation using O_2 gas:



Although the reaction would proceed, it did so with a low selectivity and conversion rate. Later it would be shown[1] that it is possible to greatly increase the selectivity and conversion of propylene to propylene oxide by including a stream of H_2 into the reaction mixture:



The actual mechanism for this increased conversion and selectivity could not be discerned from the data in the previous paper. This reaction was one of the first projects that usage of the TAP reactor was found to be applicable. It was theorised that the H_2 was generating H_2O_2 on the surface of the catalyst, and then it was the H_2O_2 that was performing the oxidation of propylene to propylene oxide.



The problem with attempting to prove that the H_2O_2 intermediate exists is that H_2O_2 will decompose rapidly over almost all catalytic surfaces even at room temperature, and as such it is very difficult to detect. The TAP reactor was a method to get around these limitations, by pulsing the reaction mixture over the AuPd–TiO₂ catalyst, the high time resolution and transient nature of the TAP experiment was hoped to provide the correct environment for the detection of H_2O_2 .

As can be seen from figure 3.2 it was possible to detect the production of H_2O_2 using the TAP reactor. From measuring mass 34 (H_2O_2) while pulsing a mixture of 2% H_2 in air (recorded at mass 2), and comparing it to a blank signal recorded at mass 100. It was thought that any increase in intensity between the blank signal and the signal at mass 34 would indicate a production of H_2O_2 . Being able to detect the H_2O_2 not only confirmed that it was forming on the surface of the catalyst, possibly being the reason for speeding up the oxidation reaction, it also laid the groundwork for attempting to produce H_2O_2 in the gas phase. This work was published as part of a larger collaboration under the title: *Gas Phase stabiliser-free production of hydrogen peroxide using gold-palladium catalysts* in *Chemical Science*.^[2]

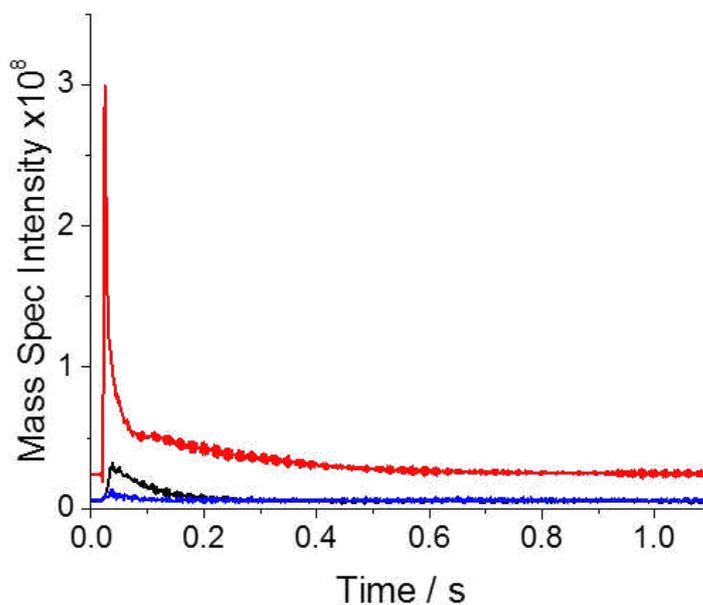


Figure 3.2: Pulsing of 2% H_2 in air over AuPd-TiO_2 at 60°C . Red line – mass spec signal at M/Z 2 (H_2), black line – mass spec signal at M/Z 34 (H_2O_2), blue line – "baseline" mass spec signal taken at M/Z 99.

3.2.2 Baseline Prediction and Tail Expansion

When performing a TAP experiment, there will always be a baseline response which arises from the mass spectrometer. One of the first steps required before analysing TAP data is to try and remove the baseline. If the experimental response returns to the baseline over the collection time of the experiment then this process is elementary. Taking an average of the last portion of the tail of the response curve, and removing that from the peak can shift it down. As each data set will have differing response times and collection frequencies, the number of data points required to get an accurate estimate will vary. Too many points and sometimes it is possible include some of the tail from the TAP response, too few and the noise will overpower the baseline and the reading will not be accurate. To counteract this non-standard amount of points, the user is able to manually select how many points they wish to include in the calculation using the *axes* object on the GUI. The process shown in figure 3.3 is performed using the *ginput* command in MATLAB:

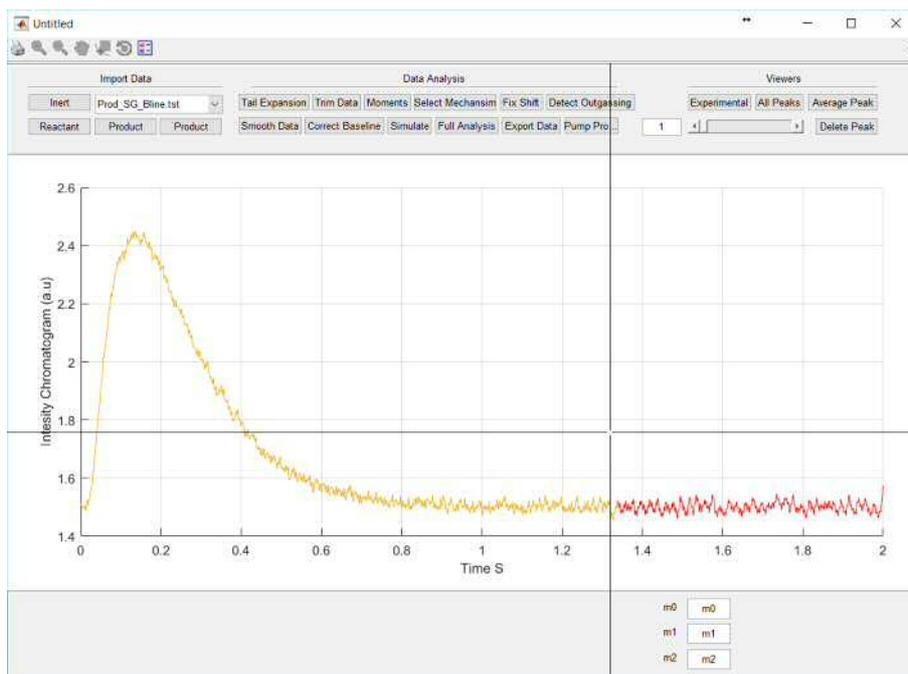


Figure 3.3: Selection of baseline cutoff using CCI-TAP

```

1 ...
2 while holder <= 10
3     pause(0.5)
4     start = ginput(1);
5     try
6         delete(bplot)
7     end
8
9     if isempty(start)
10        break;
11    else
12        bline = start(2);
13        bline = linspace(bline, bline, length(time));
14        bplot = plot(time, bline, 'k', 'LineWidth', 3);
15    end
16 end
17
18 points = find(time > max(time) - 0.2);
19 pend = peak(:, 1);
20 pend = pend(min(points):(max(points)));
21 pend = mean(pend);
22 diff = peak - bline;
23
24 for i = 1:peakno
25     if handles.datafilter == 0;
26         p = smooth(peak(:, 1));
27     end
28     pend = peak(:, i);
29     pend = pend(min(points):(max(points)));
30     pend = mean(pend);

```

```

31     bline = pend-diff;
32     peak(:, i) = peak(:, i) - bline(1);
33 end
34
35 k = findall(gca, 'type', 'line');
36 delete(k);
37
38 plot(time, peak);
39
40 sStructData.peak = peak;
41 averagepeak = mean(peak, 2);
42 sStructData.averagepeak = averagepeak;

```

Listing 3.6: Baseline correction script used in CCI-TAP

This process becomes more difficult when the TAP response does not return to the baseline during the collection time. In order to correct for this the tail of the curve needs to be expanded until it reaches the baseline and then the process can be repeated. For *simple* experiments (*e.g.* adsorption/diffusion), it has been shown that the tail of the curve can be replicated by a simple exponential decay.[3] This method was used on pre-corrected data, but by including a shifting parameter in the equation it is possible to calculate the baseline.

$$F(t) = a \exp(bt) + c \quad (3.4)$$

Using a simple optimisation algorithm it is possible to calculate the variables a , b , and c , the latter of which is considered to be the baseline. For more complex reactions, this process becomes more difficult. Instead of fitting a simple single exponential, the tail of the response can be thought of as a sum of two exponential curves, and as such the function becomes

$$F(t) = a \exp(bt) + c \exp(dt) + f \quad (3.5)$$

with f being the baseline.

Alongside calculating the baseline, expanding the tail is very important for calculating the moments of the response curves, and in normalising the experimental

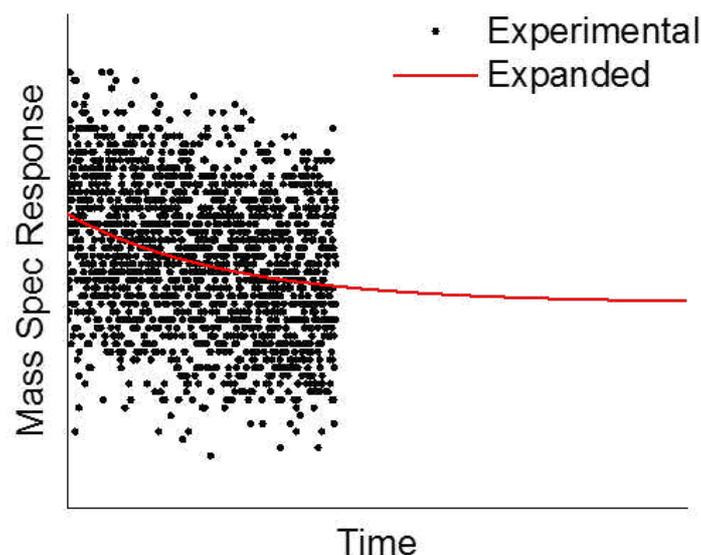


Figure 3.4: Tail expansion using double exponential function

data. For the moment equations to be correct the peaks all have to be baseline correct and return to the baseline during the collection time. Using equations 3.4 and the baseline can be expanded (figure 3.4) and the new expanded data can be used to calculate the moments.

3.2.3 Correcting for Noise

One of the largest barriers to accurate TAP data analysis is reducing the noise in the peaks. The noise itself can be broken up into two sections, *physical* noise which we characterise as electric interference and vibrational interference, and *experimental* noise which relates to the noise in the mass spectrometer. The first type of noise is considered to be regular and cyclic, and as such is relatively easy to remove, the latter is random and non-cyclic and as such is very difficult to remove.

There are many standard methods for removing noise from data (also known as data smoothing), with the most popular being a Savitzky-Golay[4] method, in which a series of polynomials are used to fit sub-sets of adjacent data points, smoothing

out the data. While this method is sufficient for data where the noise is minimal, and the collection time is short, for data in which longer collection times are required this method is not sufficient.

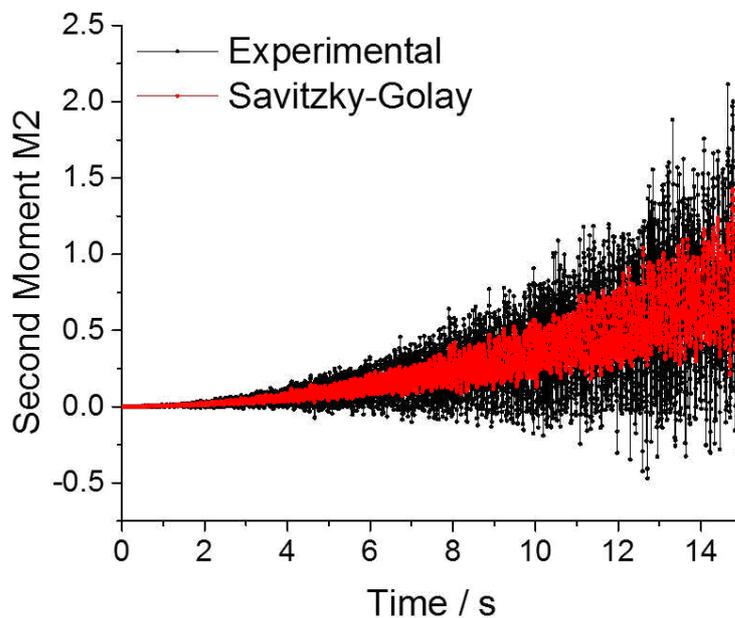


Figure 3.5: Comparison of second moment for experimental response and Savitzky-Golay smoothed response

Figure 3.5 shows a comparison between the second moment ($F_{exit} \times t^2$) for the experimental response and the Savitzky-Golay smoothed data (data span of 15 with a 2nd order polynomial). As it can be seen, the Savitzky-Golay method does decrease the amount of noise in the data, but as the second moment is highly influenced by the tail end of the response (as t increases) any small amount of noise is greatly amplified when carrying the moment analysis of the data. In order to remove the small noise fluctuations the Savitzky-Golay smoothing can be repeated multiple times, but then the shape of the initial response can be altered giving false results when calculating the kinetics.

In order to get accurate results for the second moment (and hence the kinetics), all of the noise from the TAP response curve needs to be removed. A new method, coined as the *Laplace Fourier Filtering* (LFF) method, based on the TAP response

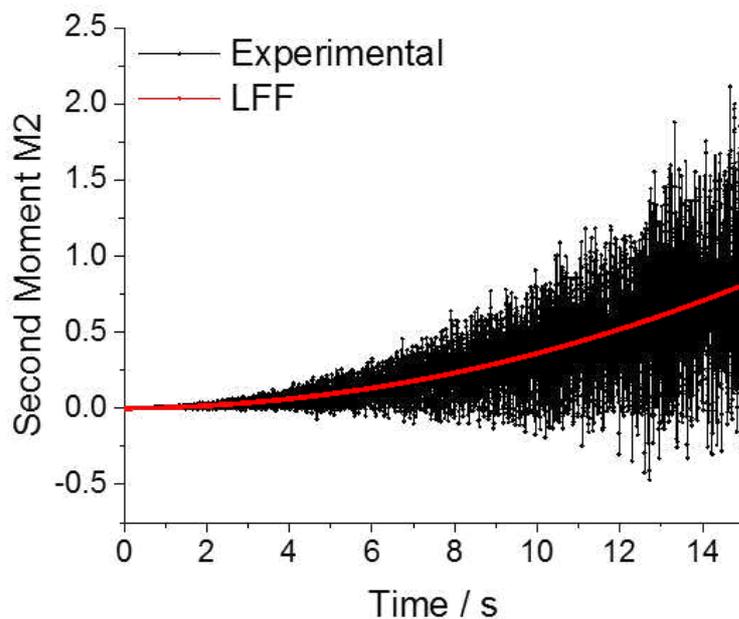


Figure 3.6: Comparison of second moment for experimental response and LFF method

equations outlined in the previous chapters, has been developed as part of this thesis, and using this method it is possible to remove almost all noise present in the data. Figure 3.6 shows the effect of this method on the second moment, and when comparing figure 3.5 and 3.6 it is obvious that this new method is far superior in terms of noise reduction.

3.2.4 Laplace Fourier Filtering

The LFF method works by using the Laplace equations that have been solved for exit flow (see section 2.3.6) in a thin zone reactor to simulate the exit flow. The equations are shown below for clarity.

$$\bar{F}_{exit} = \frac{1}{\cosh(\sqrt{2s\tau_{in}}) + \left(\frac{r^r(C_{CS}, s)\tau_{cat}}{\sqrt{2s\tau_{in}}} \right) \sinh(\sqrt{2s\tau_{in}})}$$

for the exit flow of the reactant, and:

$$\bar{F}_{exit} = \frac{\left(\frac{\sqrt{2}r^p(C_{CS}, s)\tau_{cat}}{\sqrt{s\tau_{in}n}} \right)}{\left(\left(\cosh(\sqrt{2s\tau_{in}}) + \left(\frac{r^r(C_{CS}, s)\tau_{cat}}{\sqrt{2s\tau_{in}}} \right) \sinh(\sqrt{2s\tau_{in}}) \right) \right)} \times \cosh \left(\frac{\sqrt{2s\tau_{in}D_{in}^r}}{D_{in}^p} \right)$$

for the exit flow of the product.

The main idea of the LFF method is that all of the analysis is performed in the frequency domain rather than the time domain, differing from standard regression fittings. This is because of the relationship between the Laplace and Fourier domains. The equation for a Fourier transform is as follows:

$$F(\omega) = \int_{-\infty}^{\infty} F(t)e^{-j\omega t} dt \quad (3.6)$$

and the Laplace transform:

$$F(s) = \int_{-\infty}^{\infty} F(t)e^{-st} dt \quad (3.7)$$

with:

$$s = \sigma + i\omega \quad (3.8)$$

The variable s can be broken up into two components, a *real* component which is represented by σ , and an imaginary component $i\omega$. When comparing equations and it can be seen that when the real component of the Laplace variable s is set to zero, the Laplace and Fourier equations are related by the imaginary number i .

This means that if we transfer the experimental exit flow into the Fourier domain via a Fast Fourier Transform (FFT), it is possible to simulate the corresponding frequency domain curve using the Laplace exit flow equations.

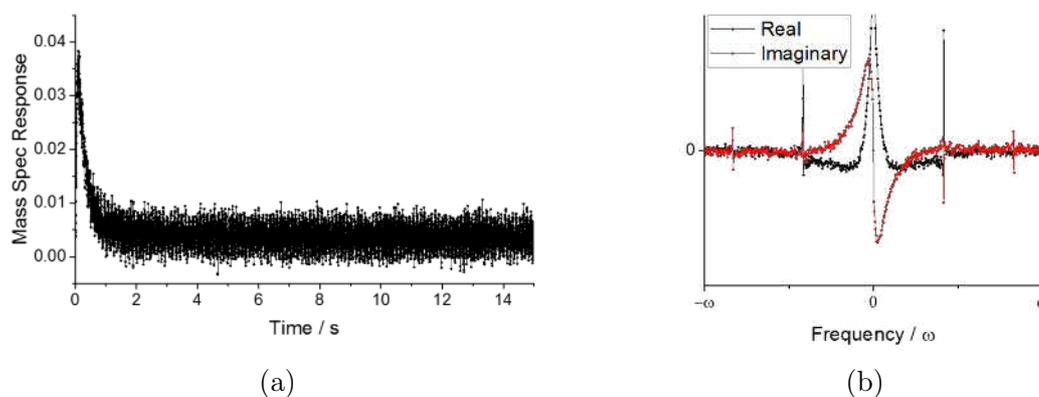


Figure 3.7: Comparison of time and frequency domains for an example TAP exit flow curve a) An example of exit flow intensity v time for a TAP experiment b) FFT of example exit flow curve, with real and imaginary components plotted as intensity v frequency

When performing a FFT on the experimental data, a complex array is formed, consisting of a *real* and *imaginary* part (figure 3.7b). This array can be plotted as a function of frequency ω . The spikes in the frequency domain relate to regular signals that appear in the experimental response, and these tend to be the *physical* components of the noise mentioned earlier. It is elementary to remove these peaks in the frequency domain by locating the maxima and *skipping* over them by taking a linear fit between the neighbouring points. This is performed by taking the second order difference in one of the components of the frequency domain (as spikes occur simultaneously in both imaginary and real it is faster to use just a single component) and then by simply iterating through all of the frequencies it is possible to find where the maxima occur. Using this method the peaks relating to the *physical* noise can be removed.

Figure 3.8 shows that even though the *physical* noise has been removed from the frequency spectrum, there is still noise present, and this is considered to be the difficult to remove *experimental* noise. As this noise is non-regular it cannot be

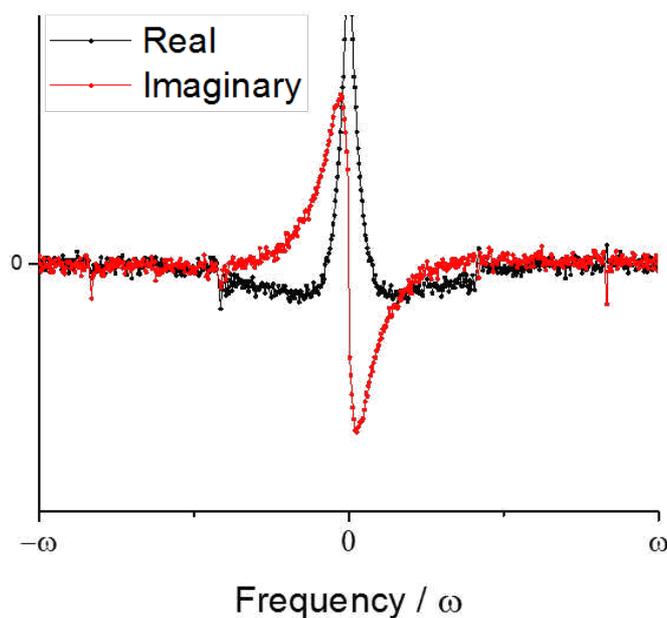


Figure 3.8: Intensity v Frequency for TAP experimental response with *physical* noise removed

removed via the previous method, and a more complex method is required. It can be stated that the shape of the frequency domain response is what defines the shape of the time domain response, and that a TAP experimental response can be simulated using the Laplace exit flow equations. As such it can be considered that using the Laplace exit flow equations it is possible to recreate the frequency response for the experimental data, but with all of the noise removed.

The first step in performing the curve fitting is to normalise the frequency response of the experimental and simulated curves, standardising the data. As the data is normalised, the explicit intensities of the peaks are removed, and as such only the shape of the frequency responses are being compared. As it is only the shape of the response, not the intensity, that is being measured, any kinetic information that could have been gathered from the explicit values of the variables in the Laplace function is considered moot. Once the curves have been standardised a global multistart optimisation method, alongside the *fmincon* function in MATLAB are used to minimise the residuals between the real and imaginary components of

the experimental and simulated frequency responses.

The particular variables of interest in the reactant Laplace equation are the kinetic model function $r^r(C_{CS}, s)$ and the residence times in the inert and catalytic zones. From the solution to the equations at $s = 0$ it was stated that via a Taylor Expansion a second order polynomial can be used to describe the kinetic model function[5]:

$$r^r(C_{CS}, s) = r^0 + r^1 s + r^2 s^2 \quad (3.9)$$

While this holds true at $s = 0$ and as such the basic kinetic coefficients and kinetics derived from these parameters are still true, when attempting to simulate a curve numerically this function no longer holds as s needs to be sampled from $-\infty$ to $+\infty$. Figure 3.9 shows a comparison between the explicit and Taylor Expansion

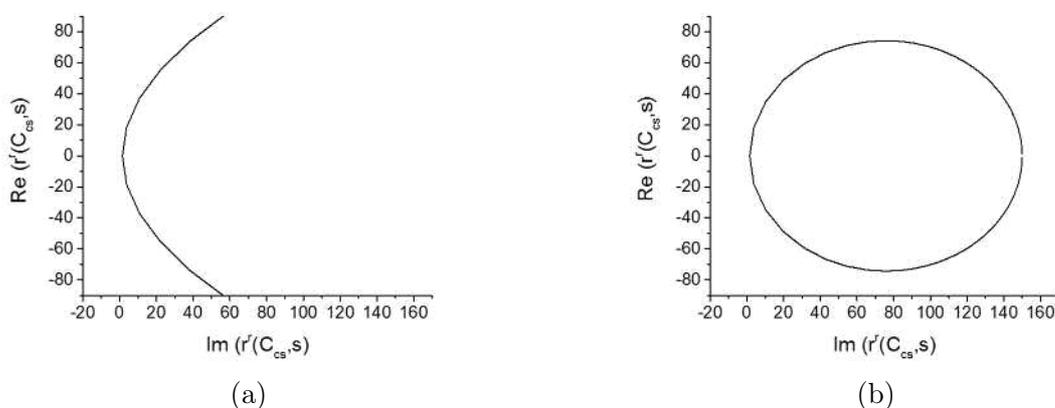


Figure 3.9: Second order polynomial v explicit rate model calculated numerically in frequency domain a) Three coefficient model from Taylor Expansion b) Explicit kinetic model for adsorption/desorption/reaction

models when simulated in the frequency domain. Instead of using the three coefficient model, the LFF method uses a more standardised model, which should fit the majority of kinetic models and has shown to be able to replicate the majority of the explicit kinetic models with high accuracy.

$$r^r(C_{CS}, s) = \frac{a + bs}{c + ds} \quad (3.10)$$

Using the global optimisation method it is then possible to recreate the full fre-

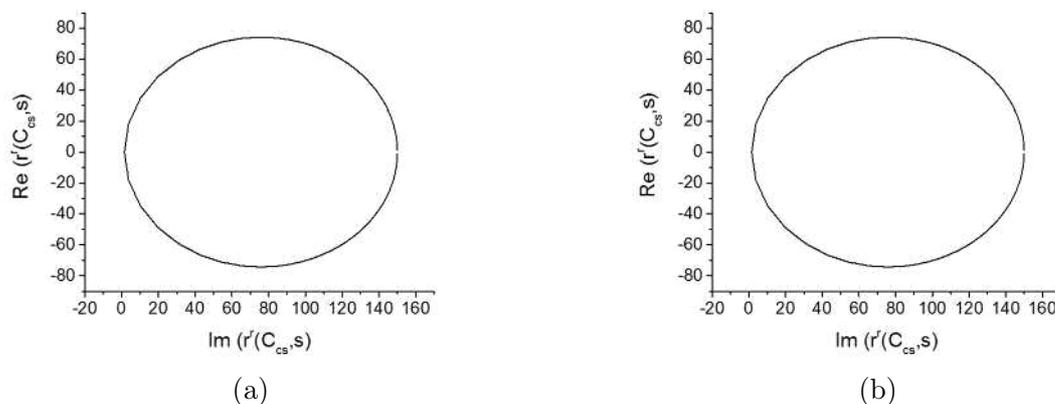


Figure 3.10: LFF four coefficient v explicit rate model calculated numerically in frequency domain a) Generalised kinetic model used in LFF method b) Explicit kinetic model for adsorption/desorption/reaction

quency curve with the noise removed. As the only interest is in replicating the shape of the curve, the parameters that are in equation 3.10 are considered have no direct kinetic meaning, although one can attempt to correlate them to a given kinetic model. The curves are compared using a simple residual sum of squares method, with a scaling factor which favours the lower frequencies. This scaling is important as the lower frequencies have a much larger effect on the overall shape of the time domain response, and hence should be considered more *important* by the global optimisation algorithm. Once the curve has been simulated in the frequency domain, it is simple to return it into the time domain using an Inverse Fast Fourier Transform (IFFT). Figure 3.12 shows that the TAP response curve has now been recreated in the time domain, but with all of the noise removed. In order to confirm that the correct curve has been created the residuals between the simulated and experimental response can be studied. Figure 3.13 clearly shows that the residuals are centred on zero, and as such it can be stated that the curve that is being generated using the LFF modified Laplace exit flow function is fully recreating the *true* experimental curve that is being hidden by the noise.

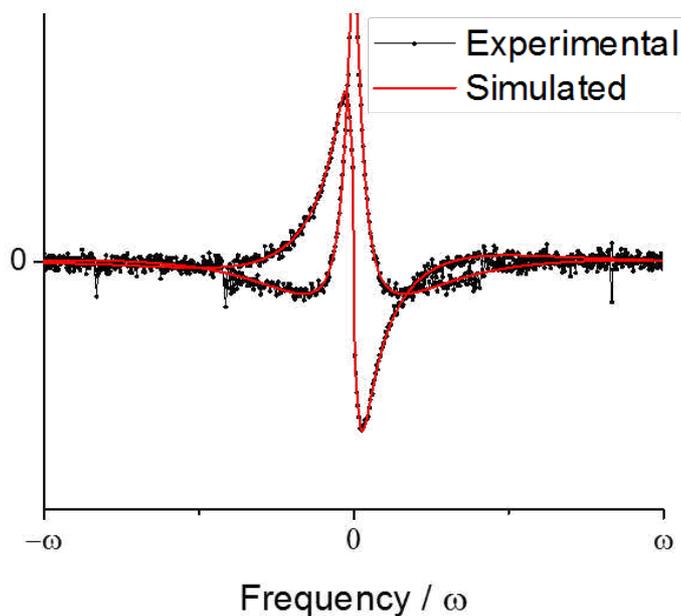


Figure 3.11: Intensity v Frequency for TAP experimental and simulated responses

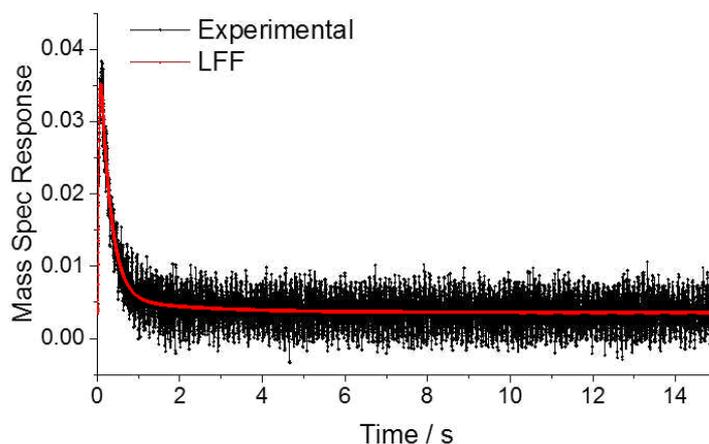


Figure 3.12: Intensity v time for TAP experimental and baseline-shifted simulated responses

Another benefit of using the LFF model over standard noise reduction techniques is that the simulated Laplace function is simulated with a zeroth baseline. In figure 3.12 the baseline has been added back into the frequency domain curve by *borrowing* the zeroth frequency point from the experimental frequency domain curve. This means that by using the new LFF method it possible to fully recreate an experimental response curve with all the noise removed, and baseline corrected, allowing a

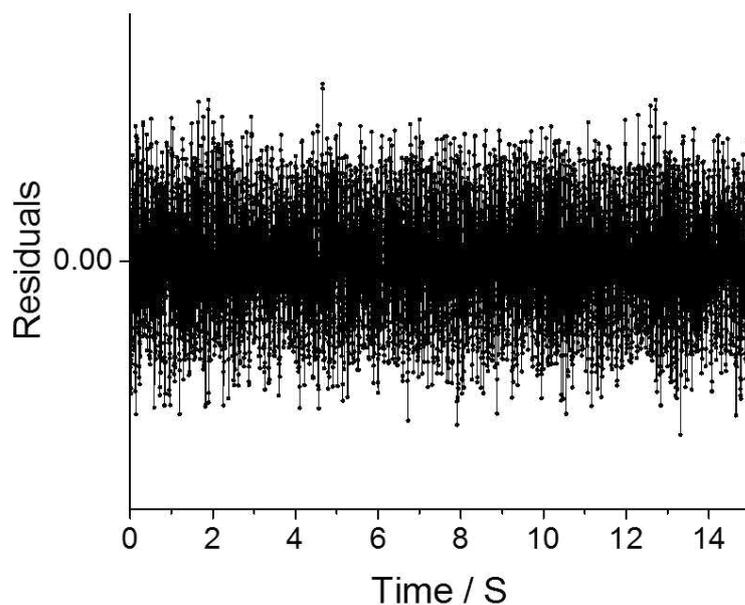


Figure 3.13: Residuals v time for the TAP experimental and simulated response moment-based kinetic analysis to be performed with a level of accuracy not possible using previous methods. A sample script for performing the LFF method is included in the appendix 7.8

3.2.5 Confirmation of the LFF Regression Technique

In order to confirm the validity of the LFF regression technique, a set of inert, product, and reactant curves will be simulated, and then they will be treated with noise which by superimposing a blank mass spectra of their respective simulated M/Z values. In order to simulate the curves a reversible adsorption / reaction model was chosen, therefore the two Laplace reactivity ($r(C_{cs}, s)$) terms in the equations describing exit flow in a thin zone TAP reactor are:

$$r^r(C_{cs}, s) = \frac{k_a(k_d + s)}{k_d + k_r + s} \quad (3.11)$$

$$r^p(C_{cs}, s) = \frac{k_a k_d}{k_d + k_r + s} \quad (3.12)$$

where k_a , k_d , and k_r are the rate constants for adsorption, desorption, and reaction respectively. It should also be noted that equations 3.11 and 3.12 both can be recreated by the generalised LFF function seen in equation 3.10, with the former containing all of the constants, and the latter having the parameters b set to zero. The curves were simulated using values of 100, 25, and 10 for the k_a , k_d , and k_r respectively, with residence times of $\tau_{in} = 0.3698\text{s}$ and $\tau_{cat} = 0.043\text{s}$, and for the physical parameters for the reactor bed a void fraction value of $\eta_{in} = 0.8$, and the lengths of the reactor and catalyst zone were $L_r = 4.3\text{cm}$ and $L_{cat} = 0.4\text{cm}$, finally the masses for the gases were taken to be 40, 44, and 28 for the inert, reactant, and product respectively. The resulting simulated curves can be seen in figure 3.14.

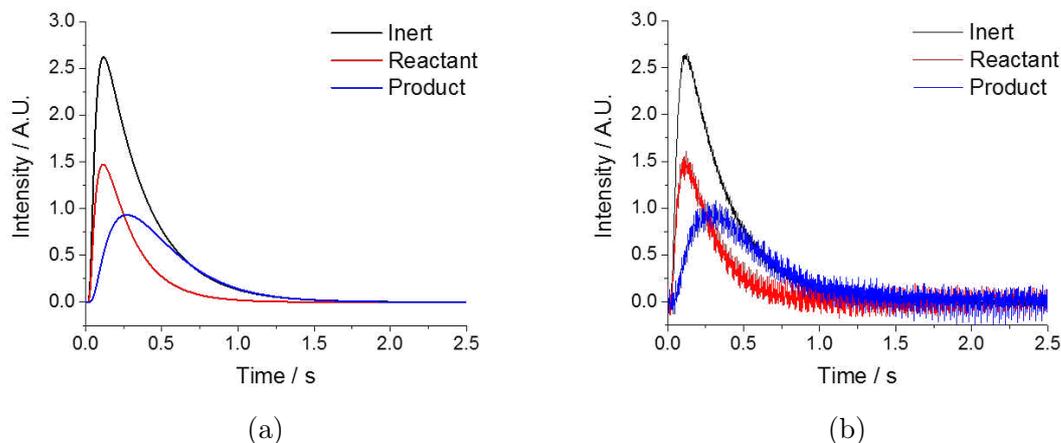


Figure 3.14: Inert, reactant and product curves simulated using TZTR exit flow equations, and a reversible adsorption/reaction mechanism a) Original simulated curves b) Original curves with added noise

In order to test the validity of the LFF technique, the basic kinetic coefficients were calculated for the simulated, noisy, LFF corrected, and a Savitzky-Golay (15 data point span, and a 2nd degree polynomial) corrected curves. Those kinetic coefficients would be used to calculate the various kinetic constants via the method shown in section 2.3.8. The curves generated from the LFF algorithm can be seen in figure 3.15, and figure 3.16 for the SG corrected curves, and finally the basic kinetic

coefficients and rate constants can be found in table 3.1

Table 3.1: by means of and kinetic parameters calculated from moments of simulated TAP responses

	r_0^r s^{-1}	r_1^r -	r_2^r s	r_0^p s^{-1}	r_1^p -	r_2^p s	k_a s^{-1}	k_d s^{-1}	k_r s^{-1}
Original	28.6	2.04	-0.058	28.6	-0.82	-0.351	100.0	25.0	10.0
Original + Noise	28.6	1.09	1.075	28.6	-1.42	0.604	30.2	0.1	1.1
SG Corrected	28.5	0.42	1.042	28.6	2.86	0.502	28.0	0.0	0.4
LFF Corrected	28.6	2.05	-0.060	28.3	-0.83	-0.353	98.9	24.3	9.9

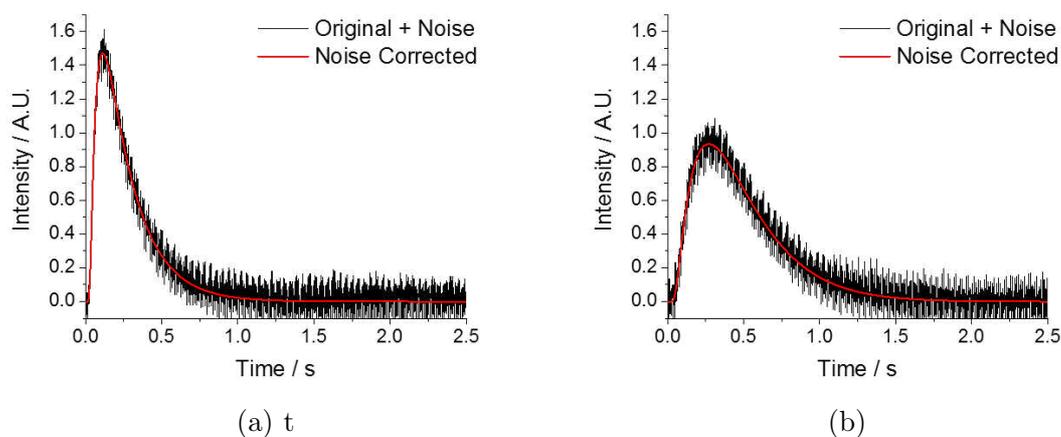


Figure 3.15: Simulated curves + noise compared with the curve calculated from the LFF regression technique a) Simulated reactant b) Simulated product

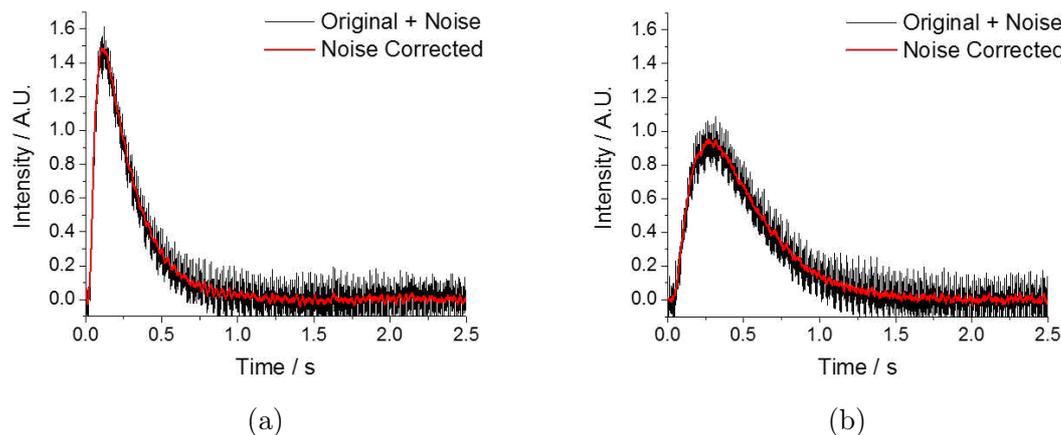


Figure 3.16: Simulated curves + noise compared with the curve calculated from the SG smoothing a) Simulated reactant b) Simulated product

As can be seen from figures 3.15 and 3.16b the SG filtering and LFF algorithm both seem to recreate the shape of the curves by simple examination by sight, even

with the experimental noise added to the peaks. The by means of in table 3.1 show that the zeroth moment is largely unaffected by the noise present in the data, but when the higher degree moments are taken into consideration the noise has a large effect, which is then compounded when calculating the kinetic by means of giving the erroneous results seen. This error due to noise is even present in the SG filtered curves, even though the noise has been lowered dramatically. This arises because the SG filtering can only smooth a curve, based on the current data points, and therefore any underlying structure that has arisen from the background noise becomes very difficult to remove completely. It can then be stated that when analysing experimental response data, if the zeroth moment is the only parameter being studied, to save on computational demand the LFF algorithm is not required, but if any further kinetic study wants to be performed then the LFF algorithm will be mandatory as even with high quality filtering techniques, the higher order moments and kinetic by means of will be largely effected by the noise.

3.2.6 Moment Analysis and Calculation of Kinetics

The mathematics for the moment analysis for calculating kinetics from TAP experimental response curves has been outlined the section 2.3, and in this section the method for calculating these moments is outlined. For the sake of clarity the moment equations are displayed here: Definition of a Moment:

$$M_n = \int_0^{\infty} t^n F(t) dt$$

For the reactants:

$$M_0^r = \frac{1}{1 + \tau_{cat} r_0^r}$$

$$\frac{M_1^r}{M_0^{R2}} = \tau_{cat} r_1^r + \tau_{in} \left(1 + \frac{\tau_{cat} r_0^r}{3} \right)$$

$$\frac{M_2^r}{M_0^{r^2}} - 2 \frac{M_1^{r^2}}{M_0^{r^3}} = -2\tau_{cat}r_2^r + \frac{\tau_{in}}{3} \left(\frac{\tau_{in}\tau_{cat}r_0^r}{5} + 2\tau_{cat}r_1^r + \tau_{in} \right)$$

For the Products:

$$\begin{aligned} M_0^p &= M_0^r \tau_{cat} r_0^p \\ \frac{M_1^p}{M_0^p} &= \frac{r_1^p}{r_0^p} + \frac{\tau_{in}}{12} \left(8M_0^r + 3 + 9 \frac{D_{in}^r}{D_{in}^p} \right) + \tau_{cat} M_0^r r_0^r \\ \frac{M_2^p}{M_0^p} &= 2 \frac{r_2^p}{r_0^p} + \frac{19D_{in}^{r^2} \tau_{in}^2}{16D_{in}^p{}^2} \dots \\ &+ \frac{D_{in}^r \tau_{in}^2}{8D_{in}^p r_0^p} (r_0^p) ((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat}) - 12r_1^p \dots \\ &- \frac{r_1^p}{6r_0^p} ((3 + 8M_0^r) \tau_{in} \dots \\ &+ 12M_0^r r_1^r \tau_{cat}) + \tau_{in}^2 \left(\frac{5}{48} + \frac{1}{45} M_0^r (23 + 40M_0^r) \right) \dots \\ &+ \frac{1}{6} M_0^r (3 + 16M_0^r) r_1^r \tau_{in} \tau_{cat} + 2M_0^r \tau_{cat} (M_0^r r_1^{r^2} \tau_{cat} - r_2^r) \end{aligned}$$

Where τ_{cat} , τ_{in} , are residence times in the catalyst and inert zones, and r_n are the basic by means of for the reactants r^r and products r^p . Once the data has been noise and baseline corrected, the next step is to calculate the moments. The moments are always normalised to the intensity of the inlet pulse, which means that for a Knudsen non-interacting curve the zeroth moment (M_0) should always be equal to one. This means that by knowing the composition of the reacting gas mixtures the zeroth moment for the reactant can be calculated using the explicit experimental moments.

$$M_{0r} = R_{exp} \times R_{gas} \times R_{ms} \quad (3.13)$$

Where M_{0r} is the inlet pulse normalised moment for the reactant and:

$$R_{exp} = \frac{M_{Er}}{M_{Ei}} \quad R_{gas} = \frac{P_i}{P_r} \quad R_{ms} = \frac{T_i}{T_r} \quad (3.14)$$

Where R_{exp} is the ratio of the experimental responses for the reactant M_{Er} and the inert M_{Ei} , R_{gas} is ratio of the gas that is the inert P_i and the amount that is the reactant P_r , and R_{ms} is the ratio of the collection by means of for the inert and reactants (fragmentation pattern, ionisation cross section, transmission coefficient, detection coefficient) the latter of which is highly variable for each gas detected, an example of calculating this can be found in the literature[6]. For example if an experimental response of 4 is calculated for the inert, and an experimental response of 0.5 is calculated for the reactant, and a 2:1 ratio of inert to reactant is pulsed into the reactor with a collection coefficient of 3:1, then the inlet pulse normalised moment for the reactant can be calculated as 0.75:

$$M_{0r} = \frac{0.5}{4} \times \frac{2}{1} \times \frac{3}{1} = 0.75 \quad (3.15)$$

Using this method the normalised moments are calculated from the experimental response without any quantification of the TAP experimental response required, which is notoriously difficult to perform. Once the experimental curves have been area normalised to the inlet pulse, the first and second moments are simple to calculate using the standard moment equation.

Once these moments have been calculated, the next step is to attempt to calculate the basic kinetic by means of, for the reactant this is simple and only requires two parameters, τ_{cat} and τ_{in} which are calculated using:

$$\tau_{in} = \frac{\varepsilon_{in} L_r^2}{2D_{in}} \quad \tau_{cat} = \frac{L_r L_{cat}}{2D_{in}} \quad (3.16)$$

Both equations shown in 3.16 are a function of the length of the catalyst zone, L_{cat} ,

the length of the reactor L_r , the fractional voidage of the reactor ε_{in} , and finally the Knudsen diffusivity D_{in} . The three parameters L_r , L_{cat} , and ε_{in} are simple physical constants and are easy to measure for a given reactor setup, the diffusivity's are more difficult to calculate and therefore the following method attempts to perform the calculations without them.

The residence time in the inert zone of the reactor for the inert peak can be calculated from the ratio of the first moment to the second moment[7]:

$$\tau_{in} = \frac{M_{1i}}{M_{0i}} \quad (3.17)$$

The residence time in the inert zone for the reactant peak can then be calculated using:

$$\tau_{in,r} = \tau_{in,i} \frac{1}{\sqrt{\frac{M_{wi}}{M_{wr}}}} \quad (3.18)$$

Where M_{wi} is the molecular weight of the inert gas, and M_{wr} is the molecular weight of the reactant gas. This relationship stems from the fact that the diffusivity is purely a function of the molecular weight:

$$D = \frac{\varepsilon}{t_r} \frac{d_i}{3} \sqrt{\frac{8RT}{\pi M_w}} \quad d_i = \frac{2\varepsilon}{3(1-\varepsilon)} d_p \quad (3.19)$$

Where the variables ε , t_r , d_i , R , and T are constant for a given pulse. Once the residence time in the inert zone for the reactant has been calculated, the residence times in the catalyst zone can be calculated by simple rearrangement of equations 3.16:

$$\tau_{cat} = \frac{L_{cat}\tau_{in}}{\varepsilon_b L_r} \quad (3.20)$$

Using this method it is possible to get all of the required parameters for calculating the basic kinetic by means of for the reactant using only simple to measure physical parameters. The calculations for the coefficients are outlined in the 2007 paper by Shekhtmen *et. al.*[8], and are outlined in section 2.3.8. The equations are repeated below for clarity.

$$r_0^r = \frac{1 - M_0^r}{\tau_{cat} M_0^r}$$

$$r_1^r = \frac{M_1^r}{\tau_{cat} M_0^{r2}} - \tau_{in} \left(\frac{1}{\tau_{cat}} + \frac{r_0^r}{3} \right)$$

$$-r_2^r = \frac{M_2^r}{2\tau_{cat} M_0^{r2}} - \frac{M_1^{r2}}{\tau_{cat} M_0^{r3}} + \frac{\tau_{in}}{6} \left(\frac{\tau_{in} r_0^r}{5} + 2r_1^r + \frac{\tau_{in}}{\tau_{cat}} \right)$$

For the products:

$$r_0^p = \frac{M_0^p}{\tau_{cat} M_0^r}$$

$$-r_1^p = r_0^r \left[\frac{M_1^p}{M_0^p} - \frac{\tau_{in}}{12} \left(8M_0^r + 3 + 9 \frac{D_{in}^r}{D_{in}^p} \right) - \tau_{cat} M_0^r r_1^r \right]$$

$$r_2^p = \frac{r_0^p}{2} \left[\frac{M_2^p}{M_0^p} - \frac{19D_{in}^r{}^2 \tau_{in}{}^2}{16D_{in}^p{}^2} \dots \right.$$

$$\left. - \frac{\tau_{in} D_{in}^r}{8D_{in}^p} \times \left((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat} - 12 \frac{r_1^p}{r_0^p} \right) \dots \right.$$

$$\left. - \frac{1}{6} M_0^r (3 + 16M_0^r) r_1^r \tau_{in} \tau_{cat} - 2M_0^r \tau_{cat} (M_0^r r_1^{r2} \tau_{cat} - r_2^r) \dots \right.$$

$$\left. + \frac{r_1^p}{6r_0^p} ((3 + 8M_0^r) \tau_{in} + 12M_0^r r_1^r \tau_{cat}) \dots \right.$$

$$\left. - \tau_{in}{}^2 \left(\frac{5}{48} + \frac{1}{45} M_0^r (23 + 40M_0^r) \right) \right]$$

Once the kinetic coefficients for the reactants have been calculated, they can be used subsequently to calculate the kinetic coefficients for the product. The kinetic coefficients for the products also include the ratio of the diffusivity of the gases, but as equation 3.19 shows, the diffusivity is purely a function of the molecular weight of the gas. The ratio is simple to calculate using the following equation:

$$\frac{D_{in}^r}{D_{in}^p} = \frac{\sqrt{\frac{1}{M_{wi}}}}{\sqrt{\frac{1}{M_{wr}}}} = D_{in}^{ratio} \quad (3.21)$$

Using these methods it is possible to calculate the basic kinetic coefficients for the reaction, and hence understand the kinetics, with only measuring the length of the reactor bed L_r , the length of the catalyst zone L_{cat} and finally the fractional voidage of the reactor bed ε_b , all of which are simple to measure or estimate.

3.2.7 Kinetic Analysis of Experimental Data

Using the methods outlined in this thesis it is possible to get highly detailed kinetic information for a specific reaction with minimal user input. Using only three parameters (L_r , L_{cat} , and ε_b) a large quantity of kinetic information can be gathered for the reaction. Due to the transient nature of the TAP experiment, each pulse is considered a snapshot of the catalyst at that specific state, and by performing a mutli-pulse experiment, it is possible to get snapshots of the catalyst at varying temperatures, catalyst states, and surface concentrations just to name a few, and now for the first time it is possible to gather in-depth kinetic information from these snapshots with only minimal user input.

The LFF algorithm, which has been shown to be vital for understanding in-depth kinetic information on a process, has been applied to a system where a cat-

alyst (4Pt–Mo₂C) was heated from room temperature to 400°C, a 1:1 mixture of CO/Argon was pulsed, and M/Z(mass divided by charge) values of 40, 28, and 44 were recorded via a quadrupole mass spectrometer for a total collection time of 10s per pulse. The CO (M/Z 28) is expected to form CO₂ (M/Z 44) via the Boudouard Reaction, during which the CO reversibly dissociates into C and O on the surface and the dissociated O then reacts with the incoming CO to form CO₂.



Throughout the experiment as the temperature is increased linearly, this means that it is simple to assign a temperature to a pulse number. The M_0 (the total area under the response curve) is plotted as a function of pulse number in figure 3.17, there are a total of 1152 pulses (383 at each mass) recorded throughout the experiment. As the current version of the LFF algorithm is very computationally demanding a selection of 14 peaks recorded at various temperatures were taken to be filtered in order to be able to plot the trends with temperature while reducing the computational demand. These 14 peaks should be able to recreate the full kinetics of the pulse sequence, as assuming that the pulses are performed in a state defining regime (where each pulse does not change the state of the catalyst significantly) each individual pulse should have similar kinetic characteristics the nearby surrounding pulses. The r values were calculated for the reactants and the products from the moments of the response curves using the equations outlined in the previous sections. For the physical parameters of the reactor bed, the length of the reactor was measured to be 4.3 cm, the length of the catalyst zone was measured to be 0.3 cm, and finally the void fraction of the reactor bed was measured to be 0.52.

From studying the zeroth moments shown in 3.17b it is clear that after approximately 130°C with increasing temperature the conversion of CO increases relatively linearly (from decreasing M_0), but when studying the M_0 for CO₂ there is no longer a linear relationship with temperature. This could be an indication of some change

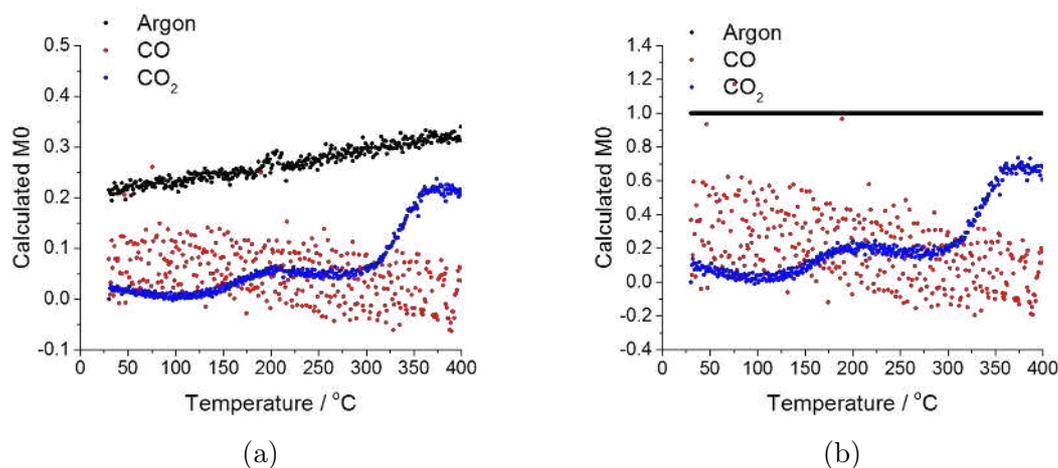


Figure 3.17: Zeroth moments taken from experimental data for 4 Pt–Mo₂C a) Calculated M_0 from experimental data b) Calculated M_0 normalised to inlet (inert) pulse size.

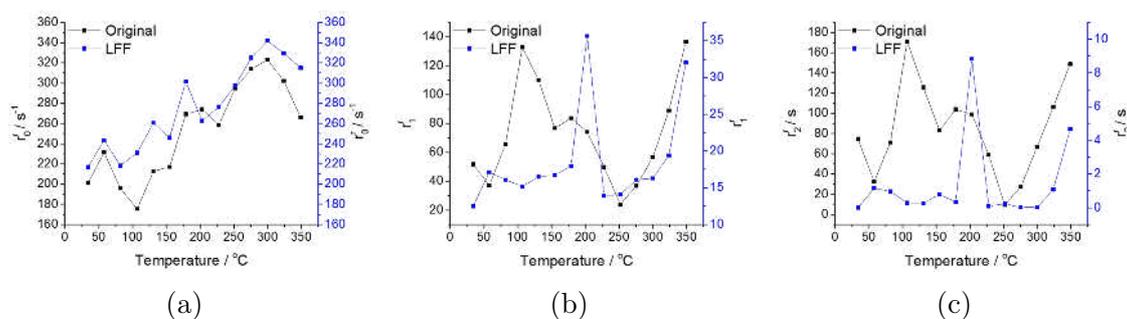


Figure 3.18: Calculated basic kinetic coefficients for the reactants on the original and the LFF corrected data. a) Calculated r_0 coefficients b) Calculated r_1 coefficients c) Calculated r_2 coefficients

in either the surface (*e.g.* increasing surface area and henceforth activity) or the reaction mechanism, although it is important to note that these are more likely not mutually exclusive. To understand if the reaction mechanism changes one can study the basic kinetic coefficients.

Similar to the testing performed in the previous section, the calculation of the zeroth basic kinetic coefficient (r_0^r) was largely unaffected by the LFF algorithm (figure 3.18a), with the general trend being increasing adsorption with increasing temperature. Where the LFF algorithm starts to show its effectiveness is understanding coefficients which rely on the higher order moments (*i.e.* r_1^r and r_2^r which rely on M_1 and M_2 respectively). Not only do the magnitudes of the coefficients

vary drastically when comparing the original data to the LFF filtered data, but the trends with temperature also have large variance. The increase in magnitude can be attributed to the multiplicative effect on the noise with increasing time and moment (figure 3.19).

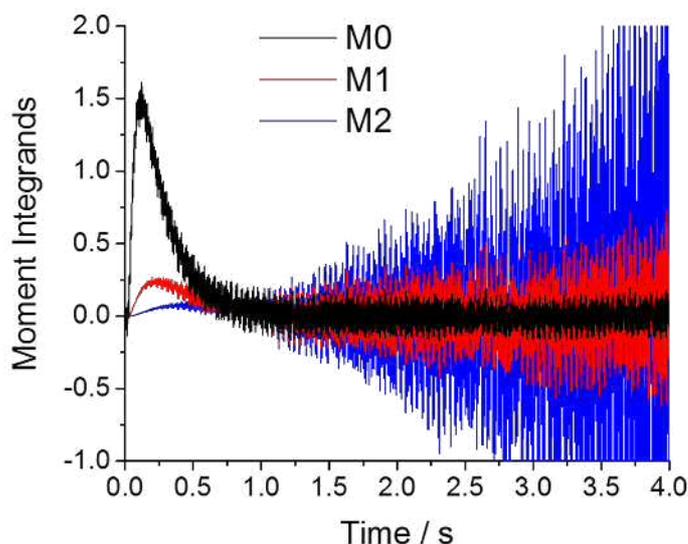


Figure 3.19: Moment integrands v time for the first three moments for the reactant pulse.

The most interesting coefficient for understanding a surface based catalytic reaction is the second coefficient r_2^r as this relates to the apparent time delay caused by processes occurring on the surface (or internal) and can provide more insight than looking purely at a conversion coefficient (r_0^r). It is clear from the second coefficient (figure 3.18c) that at 200°C there is some large shift in the reaction mechanism that is occurring on the surface which causes a much larger apparent time delay on the surface of the material. This large shift in surface delay of the CO is followed by a *plateau* in conversion to CO₂ when comparing the general trend with temperature. This could point towards some phase transition on the active site of the material at this specific temperature which causes the recombination of C and O to form CO or the formation of CO₂ to be much slower.

In a similar fashion to the coefficients shown in figure 3.18 the coefficients calculated for the products (figure 3.20) are greatly affected by the LFF algorithm. The

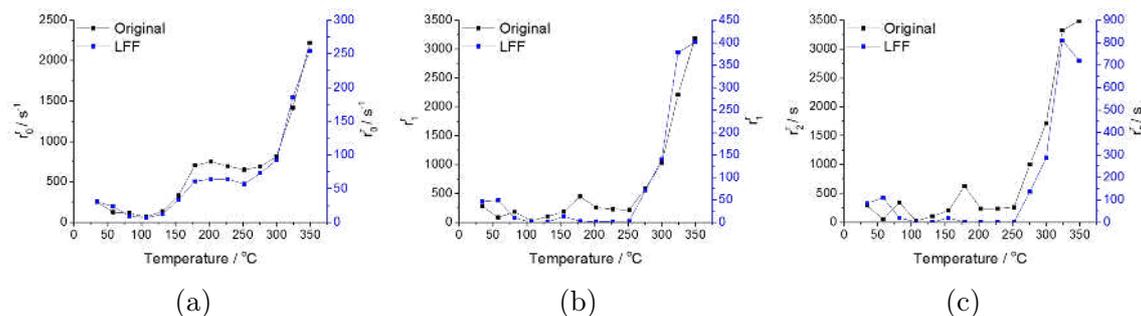


Figure 3.20: Calculated basic kinetic coefficients for the products on the original and the LFF corrected data. a) Calculated r_0^p coefficients b) Calculated r_1^p coefficients c) Calculated r_2^p coefficients

trend in the r_0^p coefficient is similar in the original and in the filtered data, but if one looks at the scales on figure 3.20a the absolute values of the coefficients differ massively, and this scaling factor is a trend throughout all of the product basic kinetic coefficients. The calculated r_0^p values, which relate to the rate of production of the CO_2 , also shows a plateau at approximately 200° mirroring the experimental M_0 values. In the higher order coefficients (r_1^p and r_2^p) it can be seen that the apparent time delay relating to processes occurring on the surface of the material falls to zero, this means that the reformation of CO_2 is happening very quickly on the surface. This would indicate that the change in the material that causes the plateau in CO_2 causes the CO splitting to slow down limiting the increase in rate of formation that one would expect with increasing temperature. With more time to investigate the calculated data, and by comparison with similar materials or reactions one could massively increase the quality of the analysis. The idea being put forward in this section is that the LFF algorithm is vital for calculating the higher order moments (and kinetic coefficients) in long time response data.

3.3 Summary

In this chapter the basis for designing software for numerical analysis has been outlined, and the idea is that by automating some of the complex mathematics in

a robust fashion, any human errors can be removed. Alongside the development of software this chapter also discusses the Temporal Analysis of Products reactor, and complementary software (CCITAP) designed to process and analyse the raw experimental data. The TAP reactor has been shown to be an extremely powerful tool for catalysis, the transient nature of the pulses allow the probing of catalyst states, and the small and rapid pulse allows intermediates that would be lost in other analysis methods to be discovered. It was shown in section 3.2.1 that the perceived intermediate for the epoxidation of propylene (H_2O_2) which was not possible to observe in standard flow reactor systems was detected using the TAP reactor. This ability to delve deeply in to various catalytic mechanisms is bolstered massively by the deep numerical analysis. By automating this into the CCITAP code, it becomes readily available for any user to perform with little knowledge on the actual functions themselves. Finally the last sections are devoted to the Laplace Fourier Filtering (LFF) algorithm developed throughout this thesis. By building upon the current literature of the well defined diffusion of the gas through the reactor and applying a new generalised kinetic model alongside a new technique of filtering data in the Fourier domain the quality of experimental data is increased dramatically, and the more in-depth analysis involving the higher moments now becomes a possibility for almost any data set. The overall aim of this project has been to make TAP more desirable to the wider community by removing some of the large barriers to entry that exist in the form of the complex numerical analysis and problems relating to a lack of proper filtering. Although there are still some limitations to the current methodology (the large computational demand for one) it is hoped that the work outlined in this chapter has provided a strong base for others to build upon.

Bibliography

- (1) T. Hayashi, K. Tanaka and M. Haruta, *Journal of Catalysis*, 1998, **178**, 566–575.
- (2) A. Akram, S. J. Freakley, C. Reece, M. Piccinini, G. Shaw, J. K. Edwards, F. Desmedt, P. Miquel, E. Seuna, D. J. Willock et al., *Chemical Science*, 2016.
- (3) K. Morgan, A. Goguet, C. Hardacre, E. V. Kondratenko, C. McManus and S. O. Shekhtman, *Catalysis Science & Technology*, 2014, **4**, 3665–3671.
- (4) J. Chen, P. Jönsson, M. Tamura, Z. Gu, B. Matsushita and L. Eklundh, *Remote sensing of Environment*, 2004, **91**, 332–344.
- (5) S. O. Shekhtman, *Interrogative kinetics: A new methodology for catalyst characterization*, 2003.
- (6) L.-C. Wang, M. L. Personick, S. Karakalos, R. Fushimi, C. M. Friend and R. J. Madix, *Journal of Catalysis*, 2016, **344**, 778–783.
- (7) J. T. Gleaves, G. S. Yablonskii, P. Phanawadee and Y. Schuurman, *Applied Catalysis A: General*, 1997, **160**, 55–88.
- (8) S. Shekhtman, N. Maguire, A. Goguet, R. Burch and C. Hardacre, *Catalysis Today*, 2007, **121**, 255–260.

4 | Development and Application of TPR Numerical Analysis

4.1 Application of TPR Data Analysis

Similar to the previous section, the methodology for analysing TPR experimental curves has been implemented into the CCI-TPR program. The software can be used to baseline correct, deconvolve, and analyse experimental thermograms to a level of accuracy not available in current methods.

4.2 Baseline Correction

In TPR experiments, attempting to remove the baseline is one of the most important tasks, as the shape of the baseline has a drastic effect on the shape of the reduction peaks. The major problem in baseline correction is that unlike TAP there is no definite *flat* baseline due to the nature of the experimental method. As such an exact mathematical method is not available for baseline removal, and therefore it is completely subjective to the user what is considered to be the baseline. To counteract this, multiple methods of removing the baseline are included in the CCI-TPR software, ranging from simple linear fitting to more complex polynomial methods.

In order to make the removal of the baseline more user-friendly, it is completely

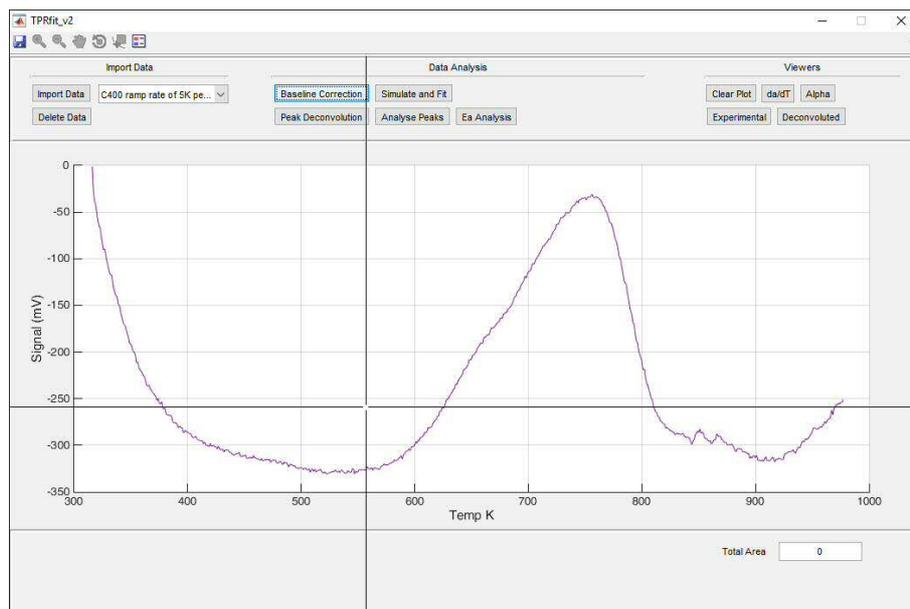


Figure 4.1: Baseline correction in CCI-TPR software

interactive (figure 4.1 in the CCI-TPR interface. The user will select start and end points and which method they would like to use, and then the baseline correction is performed.

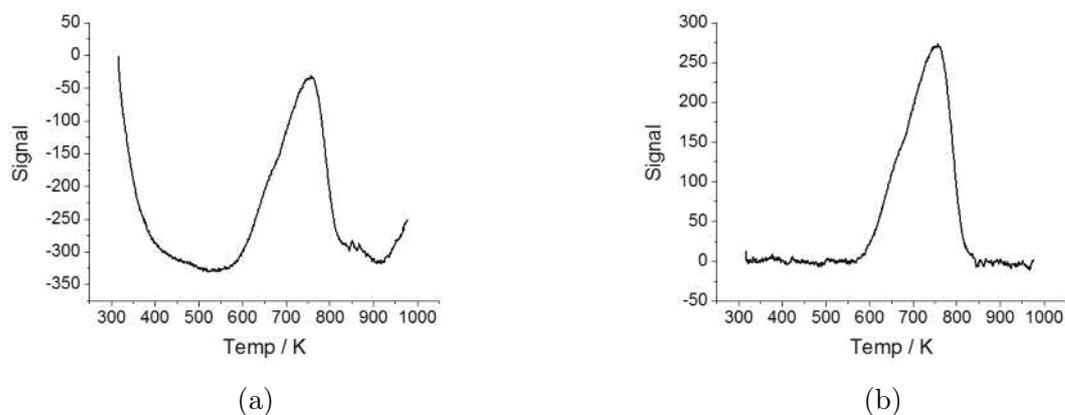


Figure 4.2: Baseline correction of TPR data a) Raw TPR signal b) Baseline corrected TPR signal

Once the baseline has been removed using the desired method (figure 4.2) it is possible to create the correct da/dT which is required for the data analysis.

4.3 Deconvolution of TPR thermograms

The majority of materials studied by TPR tend not to reduce via a single process, and as such the experimental thermogram can be considered to be combination of several different reduction processes. Deconvolution of a thermogram is a highly desired process, as it allows the breakdown of each individual reduction step in the material, which can be used to gain insight into the kinetics of the reduction processes of the material, which can give invaluable insight into the properties of the materials properties.

When attempting to simulate a TPR thermogram using the methods outlined in section 2.2, the standard kinetic equation can be used:

$$\frac{d\alpha}{dt} = \frac{d\alpha}{dT} \beta = A f(\alpha) \exp\left(\frac{-E_a}{RT}\right) \quad (4.1)$$

If there are multiple processes, equation 4.1 can be re-written as:

$$\frac{d\alpha}{dt} = \frac{d\alpha}{dT} \beta = \sum_{i=1}^N \left(A_i f(\alpha_i) \exp\left(\frac{-E_{a,i}}{RT_i}\right) \right) P_i \quad (4.2)$$

Where i is the i^{th} index of N total number of reduction processes, and P_i is the ratio of the i^{th} process to the the total process, *i.e.*

$$P_i = \frac{\int_{T_0}^T \frac{d\alpha_i}{dt}}{\int_{T_0}^T \frac{d\alpha_N}{dt}} \quad (4.3)$$

Using equation 4.2 it is possible to recreate a complex multi-process TPR reduction thermogram.

If the reduction profile can be recreated using equation 4.2, at first sight, it could

be theorised that a certain set of parameters for A and E_α , and a given kinetic model $f(\alpha)$, will give a specific shape for a reduction profile and therefore using standard optimisation and regression techniques the underlying kinetic constants for a reduction process can be calculated using equation 4.2.

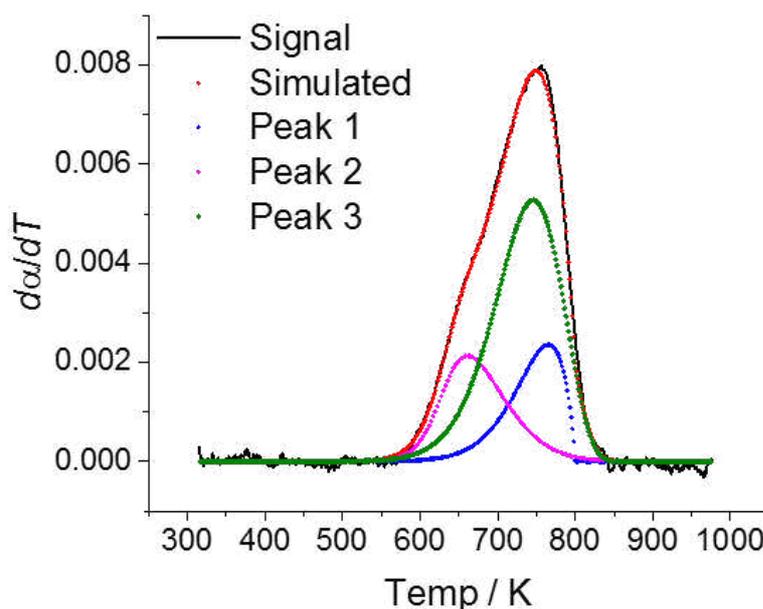


Figure 4.3: Simulated TPR thermogram using a three process kinetic model

Figure 4.3 shows a TPR thermogram that has been recreated using the kinetic equation 4.2, with an assumption of three reduction processes. There are multiple issues with this method, which will be outlined in the following sections, and in order to get an accurate deconvolution and estimation of kinetic parameters a new method has been developed.

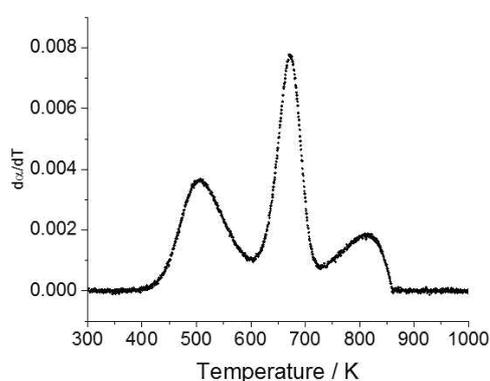
In order to accurately deconvolve the thermogram, the total number of peaks needs to be known. The new method outlined in this thesis states that instead of studying the shape of the thermogram, one should be studying the activation energy profile instead. The activation energy profile is calculated from a thermogram which has been repeated at least three heating rates, using the Kissinger, Flynn-Wall-Ozawa, Friedman, and *nonlinear-procedure* methods outlined in section 2.2. Assuming that applying these methods is accurate, it is elementary to calculate the

activation energy profile using a MATLAB script (see 7.9). The activation energy profile is then interpolated using a shape-preserving piecewise cubic interpolation method, which allows accurate estimation of the energy profile at specific degrees of reduction (it has been found that each 200th degree of reduction provides a sufficient resolution for analysis).

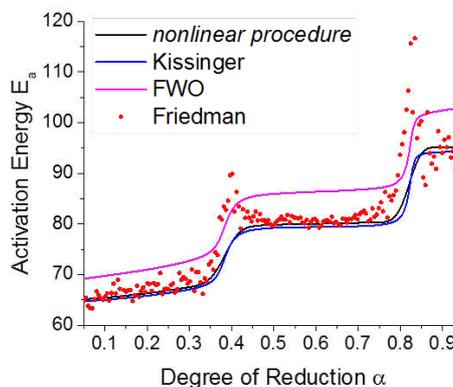
In order to test the method, a three-peak thermogram was simulated using equation 4.2 and the kinetic parameters shown in table 4.1. The thermogram was simulated at three heating rates of 5, 10 and 15 K min⁻¹, and random noise was included in order to replicate experimental data. The activation energy was then calculated at each 200th degree of reduction.

Table 4.1: Coefficients used to simulate TPR thermogram using kinetic equation

Peak	A	E_a	m	n	γ	p
1	20000	65.0	0.29	1.79	0.080	0.40
2	2000	95.0	0.00	0.54	0.010	0.20
3	8000	80.0	0.60	1.00	0.000	0.40



(a)



(b)

Figure 4.4: Testing of deconvolution procedure a) Simulated TPR thermogram at 5 K min⁻¹ with added random noise b) Calculated activation energy profile

In order to perform the deconvolution of the thermogram, the activation energy profile can be assessed. For well separated peaks with a low value of γ , $E_a(\alpha)$ will essentially be constant as a function of degree of reduction, with the shape of the profile shifting as two processes begin to overlap. As the peaks become more convoluted the shifts expected between the different TPR processes become more gradual

and it can be difficult to understand where one process begins and another ends, and estimating the number of processes occurring can be difficult. What was proposed is that instead of studying the $E_a(\alpha)$ profile, the $dE_a/d\alpha$ profile is calculated instead. As the rate of change of activation energy for a single process is considered to be constant, if there is any significant peak (or minima) in the $dE_a/d\alpha$ profile it can be assigned to a change from one reduction process to another. When calculating the $dE_a/d\alpha$ profile it is recommended that you use a combination of the four methods (Friedman, Kissinger, FWO, and the nonlinear procedure) is used when calculating the activation energy profile in order to minimise the effect of the errors in each individual methods.

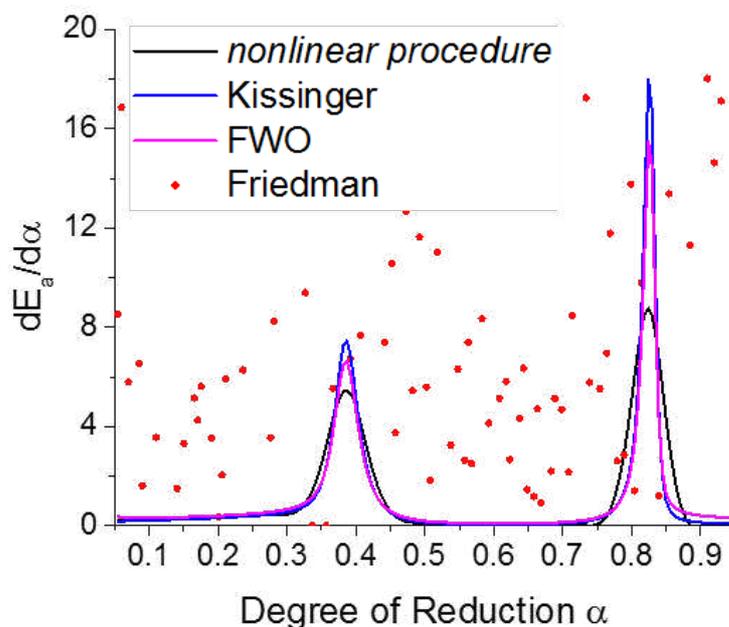


Figure 4.5: $dE_a/d\alpha$ profile calculated for the simulated TPR thermogram

From the four methods used to calculate the $dE_a/d\alpha$ profile, it can be seen that the Friedman method provides little to no insight into the number of processes occurring. This is because the Friedman method is based on the shape of the peak, and therefore it is highly sensitive to noise, causing the lack of structure in the $dE_a/d\alpha$ profile. Using the other three methods it can be seen that there are two peaks in the $dE_a/d\alpha$ profile, and therefore it can be assumed that there are at least three reduction processes occurring in the thermogram. The plot in figure 4.5 is

only shown for degrees of reduction $0.05 < \alpha < 0.95$, as at low and high values of α the signal to noise ratio is much lower and hence gives anomalous results. This method allows deconvolution of the number of reduction processes, as long as there are distinctly separate processes occurring with different activation energies. If two processes which are highly convoluted and have very similar activation energies, they cannot be separated by this method, or any other current method, but at these extreme cases the idea of *separate* reduction processes becomes more a philosophical question than a chemical one and hence are outside the scope of this method.

4.4 Calculation of Kinetic Parameters

Once an estimation of the number of reduction processes has been performed using the method outlined in the previous section, it could be theorised that the kinetic parameters can be calculated via regression and optimisation as shown in figure 4.1, but this method is still flawed even if the number of reduction processes are known. Instead in the new method, it was proposed that the regression and optimisation should be performed on the activation energy profile (4.4b).

The temperatures of reduction, and the activation energy for the reduction process at that temperature can be assumed to act as a *fingerprint* for that material, and as such only a specific combination of processes will create that activation energy profile. Using this principle, it is possible to attempt to simulate an activation energy profile using equation 4.2. Recalling section 2.2 calculating the activation energy uses the integral form of the kinetic equation, unfortunately as there is no direct solution for the integral form of the kinetic equation, it is easier to simulate the $d\alpha/dT$ profiles, combine then, adding rudimentary noise via random number generator with a range of ± 0.0002 , and then finally using a cumulative integral, recreate the alpha profiles at three different heating rates. The kinetic model used in the algorithm is the generalised Sestak Bereggren model, as it has been shown to

be able to fit to the majority of TPR thermograms. An example script is shown in listing 4.1:

```

1 ...
2 for i = 1:length(beta)
3     for j = 1:c1 % Loop over number of processes
4         %Unpacks variables
5         A = variables(j,1)*10;
6         Ea = variables(j,2);
7         m = variables(j,3);
8         n = variables(j,4);
9         g = variables(j,5);
10        prop = variables(j,6);
11        A = exp(A);
12        Ea = Ea*10000;
13
14        %Simulates Curves using Euler's method
15        [dadTs,~] = TPR_function(A,Ea,m,n,g,Temp(i,:),beta(i));
16        psim(i,:) = psim(i,)+dadTs.*prop';
17    end
18    % Create final alpha profile using cumulative integral
19    asim(i,:) = cumtrapz(Temp(i,:),psim(i,))/trapz(Temp(i,:),psim(i,));
20 end
21 ...

```

Listing 4.1: Loop used to simulate alpha profile for multi-process TPR curve

Once the alpha profile has been generated, an activation energy profile can be calculated using the one of the four standard methods. For the regression algorithm only the Friedman profile is used, as it is based on the shape of the $d\alpha/dT$ profile and hence is considered to be more accurate than the relationship between temperature and degree of conversion that the other methods are based on.

Figure 4.6 shows that even with the added random noise the algorithm is able to predict the correct activation energy profile, and hence the correct shape for the $d\alpha/dT$ curve. When comparing the actual to the fitted parameters from table 4.2, it can be seen that the algorithm is able to recreate the initial parameters with a high level of accuracy. Due to the fine nature of the value of γ , the regression algorithm is able to get a close estimate due to the robustness of the local and global optimisation algorithms, as at the low values of γ there will be many local minima surrounding the global minima.

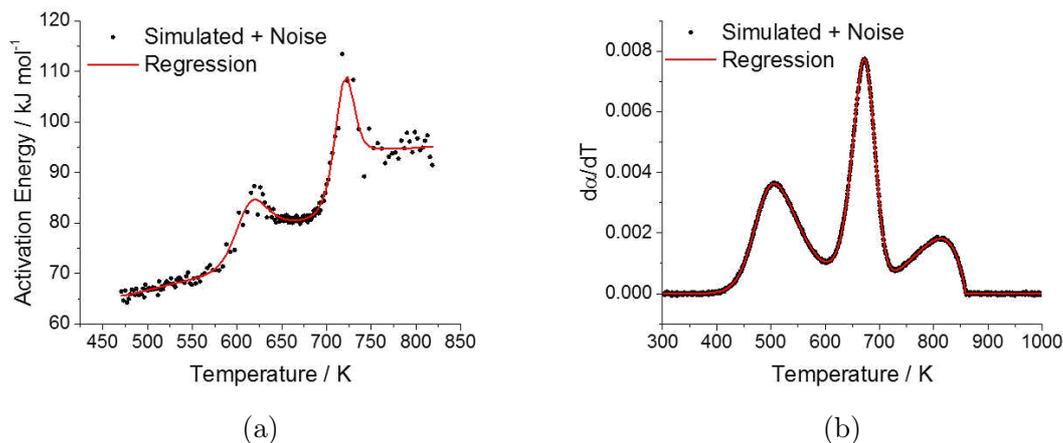


Figure 4.6: Comparison of simulated + noise three peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles

Table 4.2: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
Actual: 1	20000	65.0	0.29	1.79	0.080	0.40
2	2000	95.0	0.00	0.54	0.010	0.20
3	8000	80.0	0.60	1.00	0.000	0.40
Fitted: 1	21016	65.2	0.30	1.87	0.079	0.41
2	1836	94.0	0.03	0.52	0.018	0.20
3	8356	79.8	0.62	0.98	0.009	0.39

In order to test this process further, the algorithm was repeated using a two-peak and a single peak curve using the same parameters from table 4.1. For the two

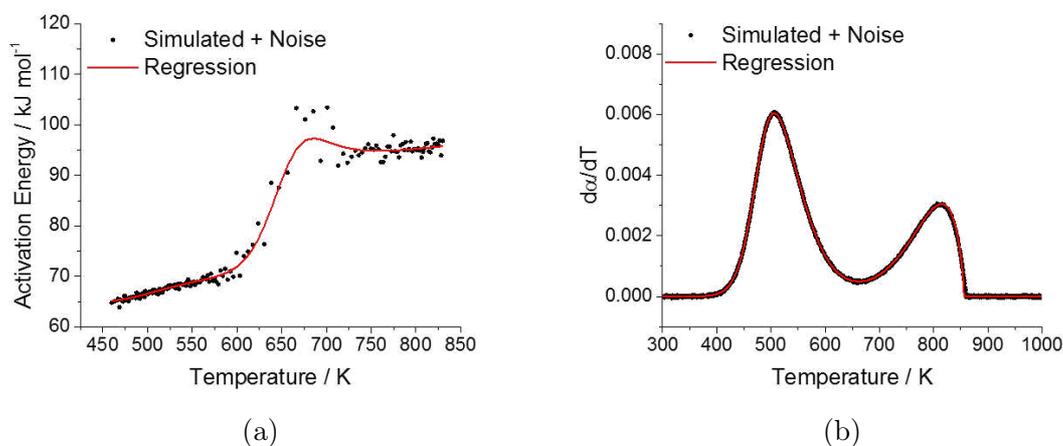


Figure 4.7: Comparison of simulated + noise two peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles

peak curve, by studying figure 4.7 and comparing the parameters in table 4.3 the

Table 4.3: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
Actual: 1	20000	65.0	0.29	1.79	0.080	0.67
2	2000	95.0	0.00	0.54	0.010	0.33
Fitted: 1	19935	64.9	0.30	1.76	0.084	0.67
2	1833	94.1	0.02	0.46	0.023	0.33

algorithm is able to recreate the two peak curve with a high level of accuracy. Again the γ parameter seems to be difficult to fix, again this is most likely due to the noise causing multiple local minima around the global minima. The parameters P , A and E_a are replicated with very high levels of accuracy, which are the more important parameters, the reasoning why will be outlined in the next section.

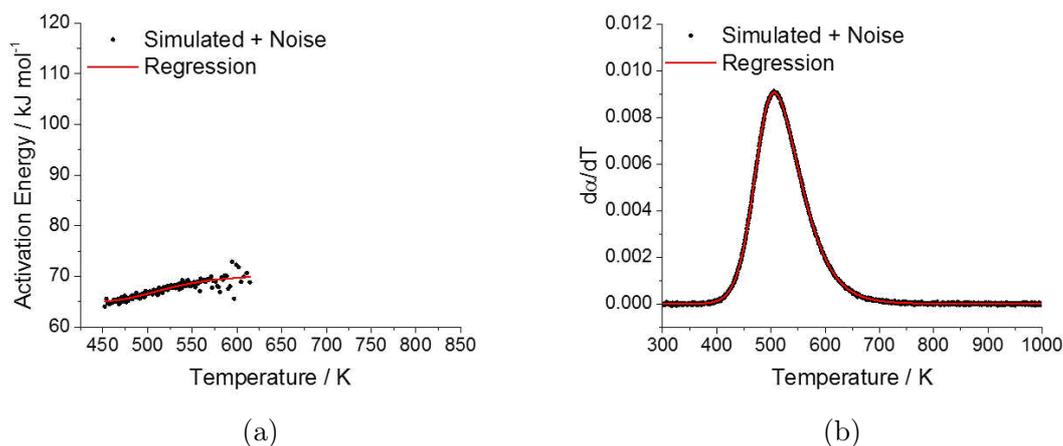


Figure 4.8: Comparison of simulated + noise one peak profiles and the ones calculated using the regression algorithm a) Activation energy profiles b) $d\alpha/dT$ profiles

Table 4.4: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
Actual: 1	20000	65.0	0.29	1.79	0.080	0.67
Fitted: 1	1993	64.9	0.30	1.76	0.084	0.67

Using the simulated data, the new method developed as part of this thesis has been shown to replicate TPR thermograms with high levels of accuracy, allowing an accurate deconvolution of the thermogram, while also providing accurate estimates for the kinetic parameters.

4.5 Further Analysis

Using the new method outlined in the previous sections, it is possible to get accurate estimations of the kinetic parameters for the reduction of a material, but the kinetic analysis does not have to stop there. As the curves have been fully deconvolved into their respective processes, the individual processes can then be run through Malek's procedure in order to try and assign a *real* kinetic model to the reduction process, rather than applying the Sestak Bereggren generalised kinetic model.

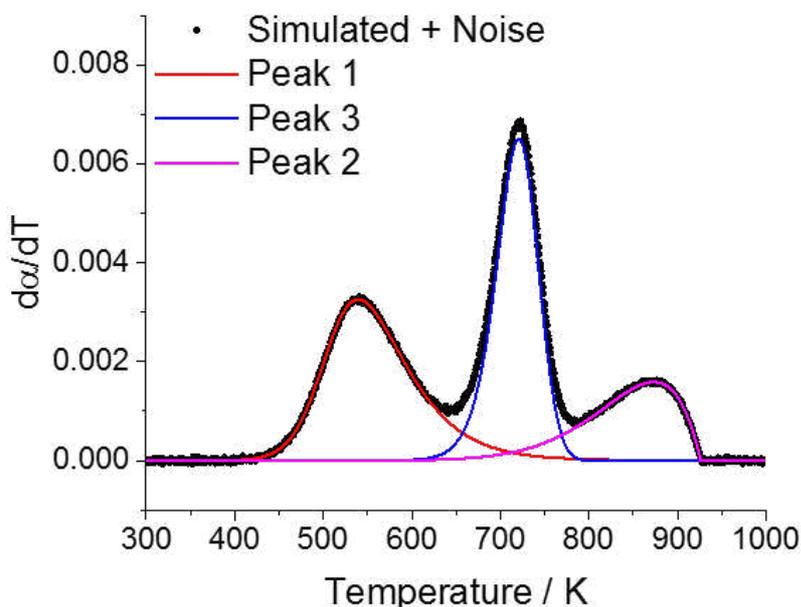


Figure 4.9: $dE_a/d\alpha$ profile with deconvolved processes overlayed

Using the equations outlined in Malek's procedure [1], which are shown below for clarity, it is possible to generate the master plots for the three curves that have been deconvolved from the thermogram.

$$y(\alpha) = \left[\left(\frac{d\alpha}{dT} \right) \beta \right] \exp \left(\frac{E_a}{RT} \right) \quad (4.4)$$

$$z(\alpha) = \pi(x) \left[\left(\frac{d\alpha}{dT} \right) \beta \right] \frac{T}{\beta} \quad (4.5)$$

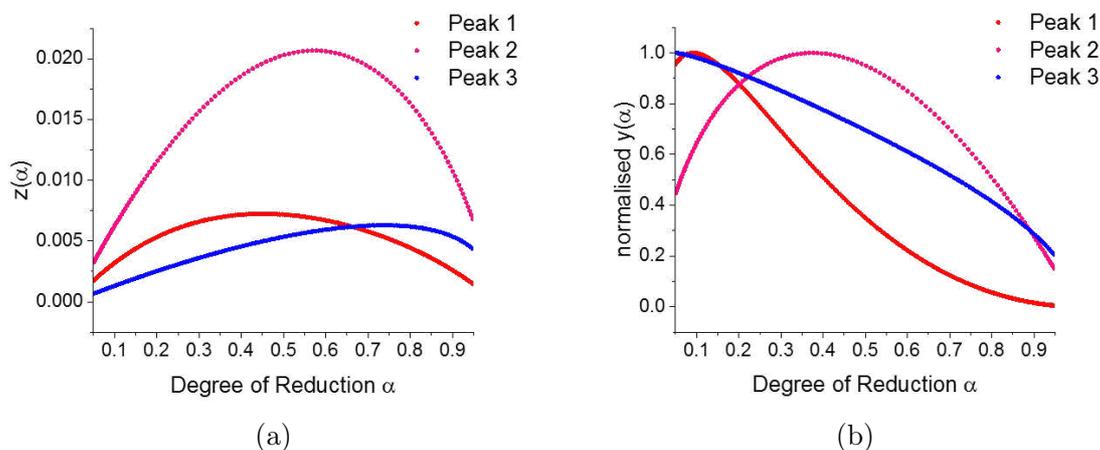


Figure 4.10: Master plots calculated using Malek's procedure a) Master $z(\alpha)$ plot for deconvolved curves b) Master $y(\alpha)$ plot for deconvolved curves

Malek outlined a flow chart in his 1992 paper which can be used to assign a kinetic model to a specific reduction curve using the master plots $z(\alpha)$ and $y(\alpha)$, which is shown in section 2.2. By following this flowchart a kinetic model can be assigned to each of the individual peaks. Once the model has been assigned, the kinetic constants for that model can then be calculated by analysing the shape of the $z(\alpha)$, and the $y(\alpha)$ master plots.

Table 4.5: Parameters and kinetic model calculated via Malek's procedure

Peak	$z(\alpha_m)$	$y(\alpha_m)$	$y(\alpha)$ shape	Kinetic Model
1	0.449	0.091	Concave	SB(m,n)
2	0.736	0.000	Concave	RO($n < 1$)
3	0.568	0.376	Concave	SB(m,n)

Not only is it possible to get the kinetic model, it is also possible to calculate the kinetic parameters. For the RO($n < 1$) model the kinetic parameter can be calculated using Gorbachev's procedure[2], shown in section 2.2, which is shown below for clarity

$$\alpha_p = 1 - \left[1 + \frac{1+n}{n} x_p \pi(x_p) \right]^{1/(n-1)} \quad (4.6)$$

Using this method for the second peak, which was estimated to be the RO($n < 1$) model from Malek's procedure, the kinetic exponent n was calculated to be 0.54, giving the kinetic function.

$$f(\alpha) = (1 - \alpha)^{0.54}$$

which is the same as the kinetic model used to simulate the curve as shown in table 4.2.

$$f(\alpha) = \alpha^0 (1 - \alpha)^{0.54}$$

For the other two curves, no specific kinetic model could be calculated and hence it defaults to the generalised SB(m, n) model. The kinetic constants can still be calculated through Malek's procedure through performing a linear fit of:

$$x_{SB} = \ln \left[\left(\frac{d\alpha}{dt} \right) \exp \left(\frac{E_a}{RT} \right) \right] \quad y_{SB} = \ln [\alpha^p (1 - \alpha)] \quad (4.7)$$

where

$$p = \frac{y(\alpha_m)}{1 - y(\alpha_m)} \quad (4.8)$$

from the linear fit the values for m and n can be calculated via:

$$n = \frac{dx_{SB}}{dy_{SB}} \quad m = pn \quad (4.9)$$

Taking the values of E_a calculated from the regression algorithm, the values for the parameters used in the SB equation were as follows:

$$\text{Peak1 : } f(\alpha) = \alpha^{0.22} (1 - \alpha)^{2.22}$$

$$\text{Peak2 : } f(\alpha) = \alpha^{0.66}(1 - \alpha)^{0.97}$$

As can be seen from the above values, the kinetic parameters for the second peak line up accurately with the original values, whereas the values for peak 1 do not line up. This is due to peak 1 having an E_a scaling coefficient γ which is greater than zero. This means that over the course of the reduction process the activation energy will change, and therefore the model 4.7 needs to be adjusted:

$$x_{SB} = \ln \left[\left(\frac{d\alpha}{dt} \right) \exp \left(\frac{E_a(1 + \gamma\alpha)}{RT} \right) \right] \quad y_{SB} = \ln [\alpha^p(1 - \alpha)] \quad (4.10)$$

With the inclusion of the E_a scaling coefficient calculated from the regression algorithm the values for the coefficients m and n were calculated as follows:

$$\text{Peak1 : } f(\alpha) = \alpha^{0.30}(1 - \alpha)^{1.87}$$

When compared to the values from 4.2 it is clear that including the scaling coefficient is important in getting accurate estimations of the kinetic parameters from Malek's procedure.

By comparing the values from Malek's procedure, and the new method, they match up nicely with the initial parameters used to simulate the curves. Although the values do not overlap perfectly, they still provide a decent estimate of the reduction function. As Malek's procedure requires the fully deconvolved processes, and it does not seem to be as accurate at recreating the initial kinetic parameters, it is easy to dismiss as an unnecessary extra step. However, the benefit of using the procedure is that it can be used to assign a *real* kinetic model to a deconvoluted function.

4.6 Application to Real Data

As the new method has been shown to be able to replicate the initial kinetic parameters used to simulate a complex thermogram, it can now be used to understand the reduction processes occurring in real catalytic systems.

The catalytic materials used are a set CeO₂ catalysts, which have been calcined at three temperatures, (400, 500, and 600°C). CeO₂ was chosen as the material of interest as it is a commonly used in the three way catalytic converter as an oxygen storage medium. TPR can be used to assess the ease at which oxygen can be removed from this material. By comparing these three materials information on the effect of preparation method on the functionality can be obtained. The CeO₂ catalysts have been reduced at three heating rates (5, 10, and 15 K min⁻¹) which allows the full kinetic analysis to be performed.

When studying figure 4.11 from the $dE_a/d\alpha$ profile it could be argued that there are one or two peaks in the $dE_a/d\alpha$ profile, indicating two or three processes. As noise is a large problem in experimental data, the analysis was performed using both two and three processes. Increasing the number of processes will always increase the quality of the fit. Therefore having a close estimate of the number of processes occurring, performing the analysis, and seeing which gives the best fit with the lowest number of peaks is the best practice.

Table 4.6: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
1	9	49.5	0.46	0.45	0.141	0.69
2	114687	115.8	0.01	0.53	0.000	0.31

The first step was to use the least amount of processes as calculated from figure 4.11, estimated to be 2. This is based on the assumption that any peak in the $dE_a/d\alpha$ profile (*e.g.* at $\alpha = 0.25$) indicates a change in reduction process (as the activation energies for two processes are expected to be different). Figure 4.12

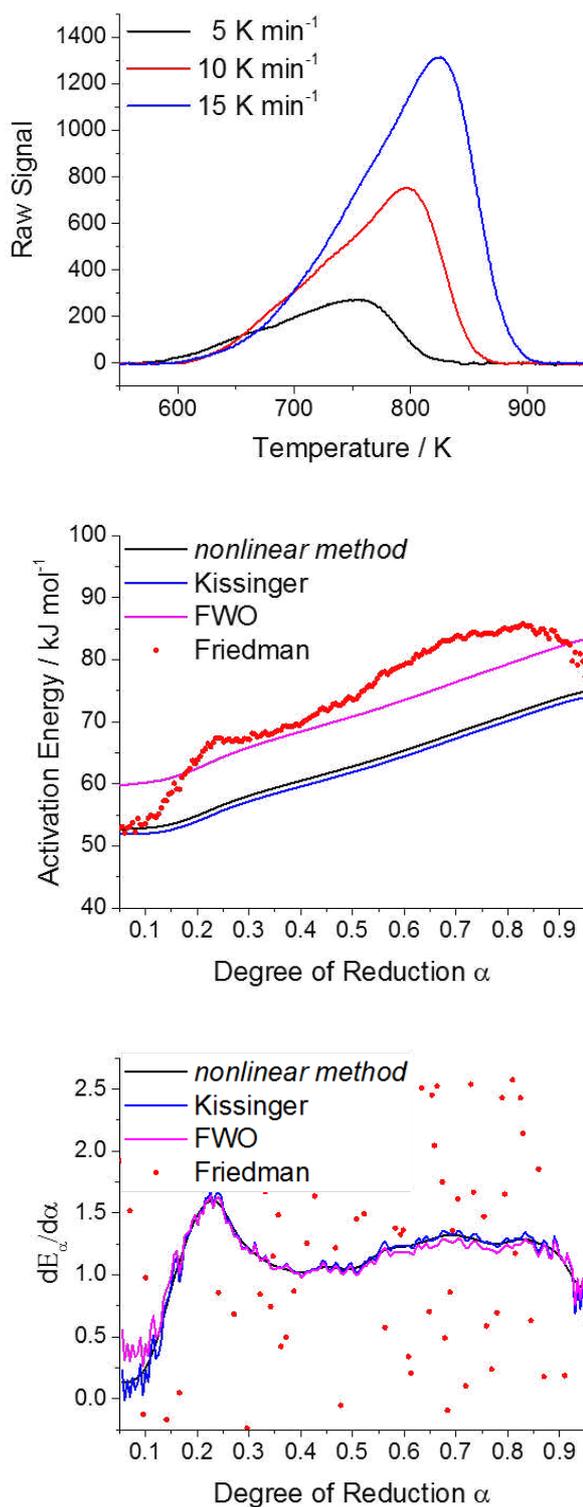


Figure 4.11: CeO₂ catalyst calcined at 400°C baseline corrected data, activation energy, and $dE_a/d\alpha$ profiles respectively.

shows the calculated profiles, and table 4.6 shows the calculated parameters. When fitting the experimental data it was found that including the Kissinger method

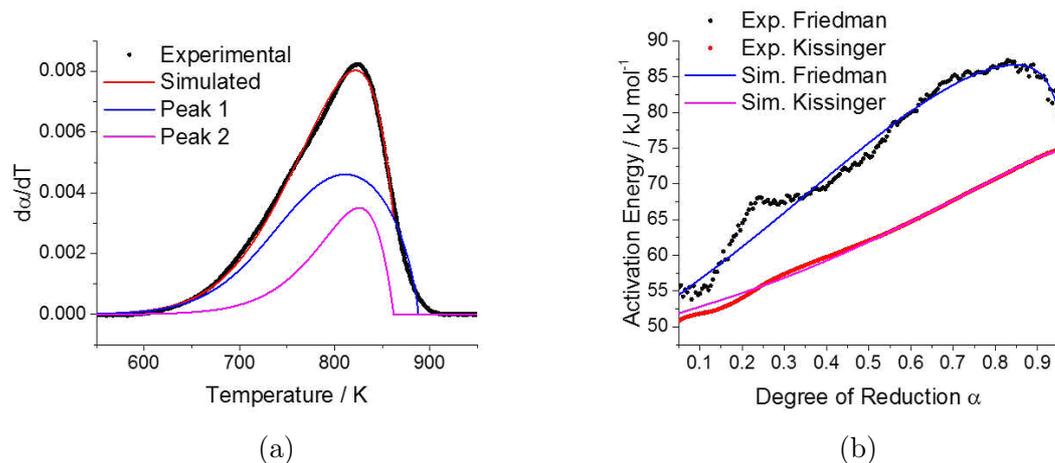


Figure 4.12: Experimental (Exp.) and simulated two process (Sim.) $d\alpha/dT$ and activation energy profiles overlaid for CeO_2 calcined at 400°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

for analysing the curves in the regression algorithm was useful for comparing the curves. As the Friedman method is correlated to the shape of the curve, and the Kissinger is correlated to the temperatures of the reduction it makes sense to use a combination of the two as the target for the regression algorithm. From figure 4.12 it can be seen that the simulated activation energy profile misses a feature at approximately $\alpha = 0.25$, therefore it is assumed that two processes do not replicate the real reduction profile accurately.

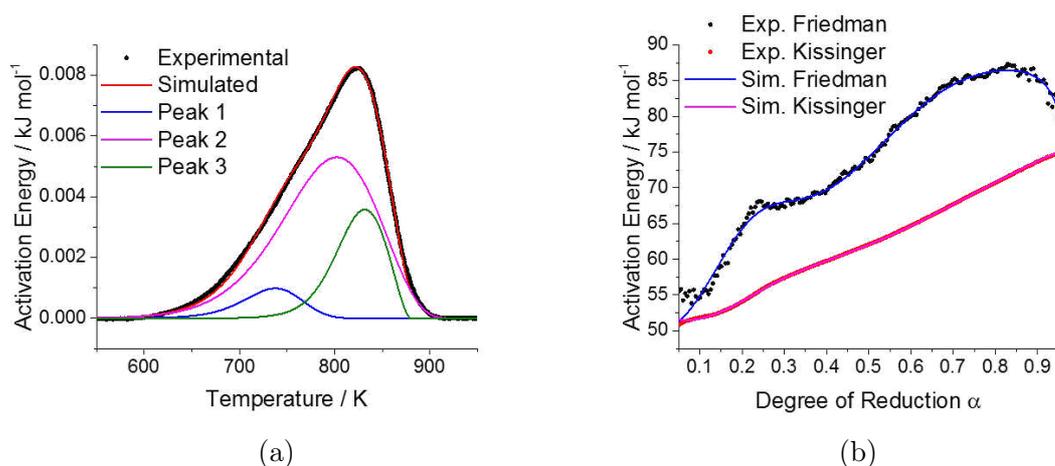


Figure 4.13: Experimental (Exp.) and simulated three process (Sim.) $d\alpha/dT$ and activation energy profiles overlaid for CeO_2 calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

Table 4.7: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
1	14	43.8	0.75	0.87	0.000	0.06
2	24	53.6	0.39	0.77	0.029	0.67
3	4918	90.8	0.53	0.73	0.012	0.27

Figure 4.13, and in particular figure 4.13b, clearly shows a much more accurate fit with three processes when compared with 4.12b, particularly when comparing the residuals between the peaks ($res = |sim - exp|$) which are calculated to be 96 and 203 respectively. As there could be one or two peaks in the $dE_a/d\alpha$ profile, it can be assumed that the three-process model provides the best estimate for the deconvolved thermogram. Analysing the peaks, they all seem to reduce via similar kinetic models, with the m and n parameters having no distinct pattern with changing activation energy E_a and Arrhenius pre-exponential A . The one pattern that can be observed, is that with an increasing activation energy the activation energy scaling parameters γ also increases. While individually the kinetic parameters can be used to estimate some of the properties of the catalytic material, where the analysis really shines is when it is used to compare similar materials.

The next catalyst to be analysed was the CeO_2 calcined at $500^\circ C$. The $dE_a/d\alpha$ does seem to show a small peak at 0.2α , and therefore the first test should be performed using a two process regression model, similar to the previous catalyst.

Table 4.8: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
1	6	45.2	0.55	0.47	0.233	0.48
2	573	80.4	0.26	0.52	0.015	0.52

Similar to the previous catalyst, the two process regression method does not accurately recreate the activation energy profile as seen in figure 4.15b, particularly in the region of 0.1α to 0.5α . The parameters in table 4.8 seem to be consistent with the ones calculated for the previous catalyst, table 4.6 indicating that similar processes are attempting to be modelled. Based on this assumption, a three process

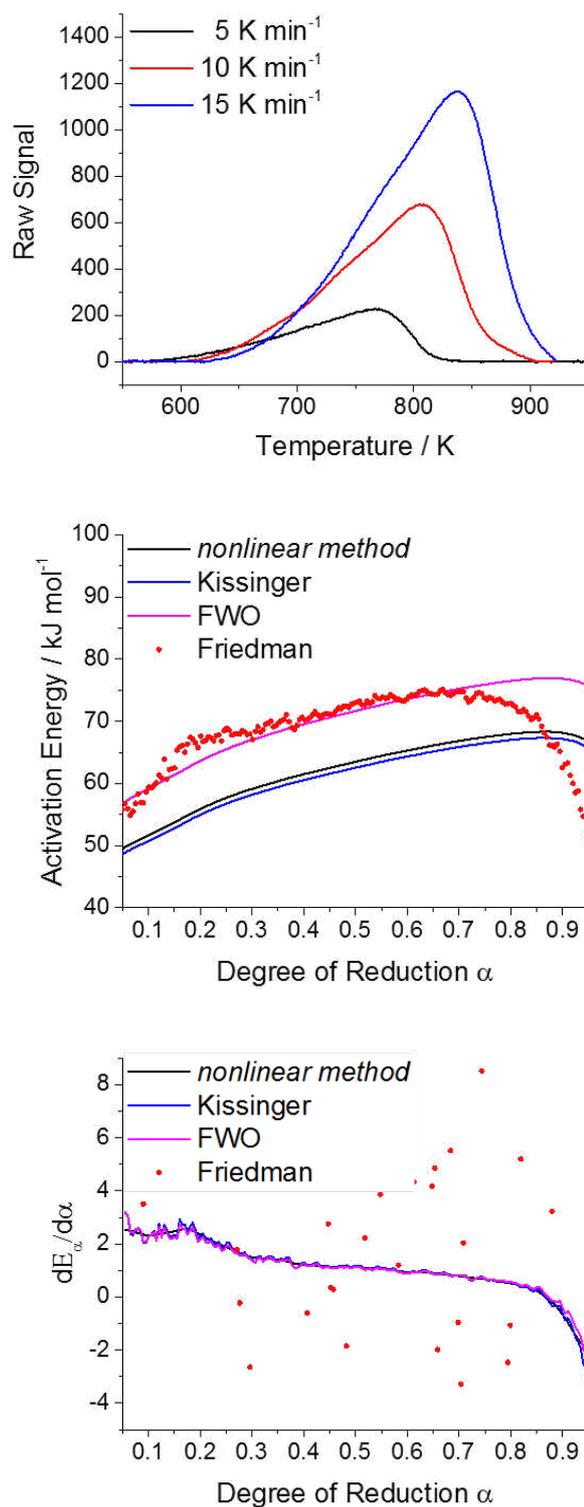


Figure 4.14: a) CeO₂ catalyst calcined at 500°C baseline corrected data b) activation energy c) $d\alpha/dT$ profiles respectively.

regression model will also be used, as even though the $dE_{\alpha}/d\alpha$ profile appears to be flat, there could be a small change hidden by the noise. As mentioned previously

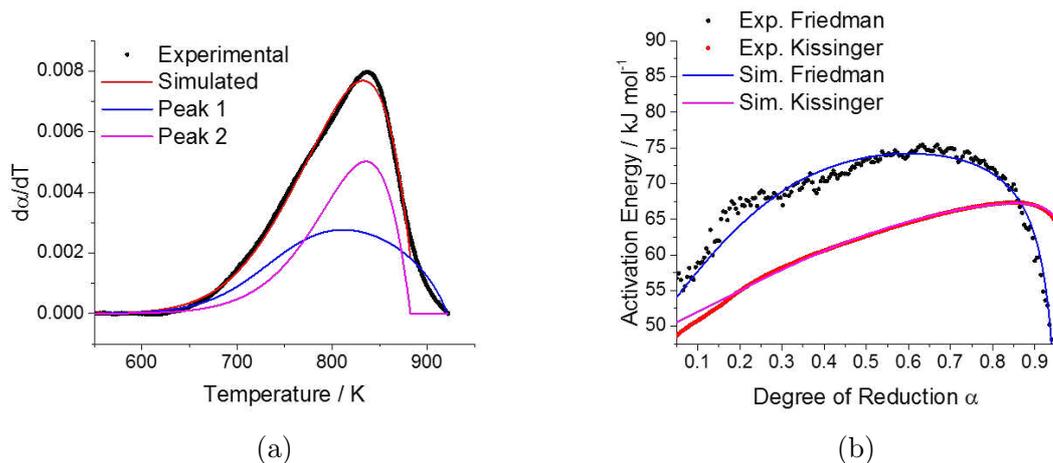


Figure 4.15: Experimental (Exp.) and simulated two process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO_2 calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

this method of assigning the number of reduction processes is subjective, and the $dE_a/d\alpha$ should only be used a guideline, with the rule of thumb being using the least number of processes required to get an accurate fit.

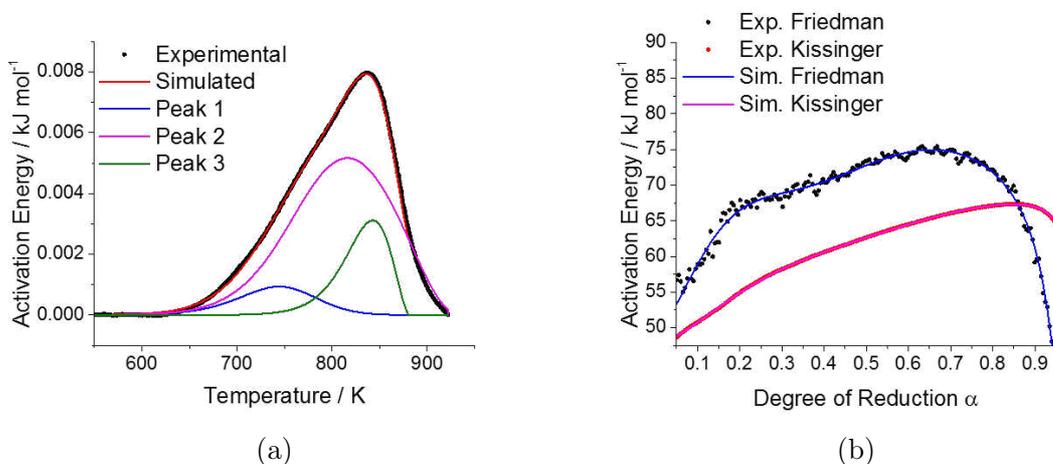


Figure 4.16: Experimental (Exp.) and simulated three process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO_2 calcined at 500°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

Table 4.9: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
1	8	42.5	0.66	1.01	0.026	0.12
2	53	60.3	0.51	0.76	0.089	0.69
3	605	76.5	0.67	0.67	0.009	0.19

Figure 4.16b clearly shows a much more accurate fit to the activation energy profile than figure 4.15b, and the parameters calculated in table 4.9 show a strong correlation to the ones calculated in table 4.7. At first glance this indicates that when the material is calcined at 400°C and 500°C there is little change in the actual structure of the catalyst.

The final catalyst to be analysed is the CeO₂ which has been calcined at 600°C. It is important to note that the raw response for the catalyst calcined at 600°C was much lower than the catalysts calcined at 400 and 500°C. Assuming that all other experimental parameters have been kept consistent (*e.g.* equipment used, amount of catalyst, amount of gas) this would indicate that less material has been reduced overall as the total amount of H₂ adsorbed is less. This alone would indicate that the thermal stability of the material has been increased by the high temperature calcination, subsequently indicating a possible phase change. This is further supported by the emergence of a shoulder peak (figure 4.17) at high temperatures for the three thermograms. As the $dE_a/d\alpha$ profile for the 600°C calcined catalyst is extremely noisy, it is difficult to get an initial estimate to the number of processes occurring during the reduction of the material. By applying what we have seen from the previous materials, where a three-process reduction method fits the main peak, assuming that the main peak seen in figure 4.17 is the same as the one seen in figure 4.11 and 4.14, and that the shoulder peak the emergence of a new phase, it can be assumed that a four-process model will fit the resulting data.

It was found that the complex nature of the activation energy profile caused large problems for the optimisation algorithm, and it was difficult to extract any kinetic information from the thermogram. This most likely arose from the low signal to noise ratio, and the fact that the baseline was highly complex in the original thermogram.

Attempting to model the entire thermogram was found to be unfeasible, and therefore no kinetic information could be obtained. Applying the principle that the second peak that appears in the reduction profile is the formation of a new phase,

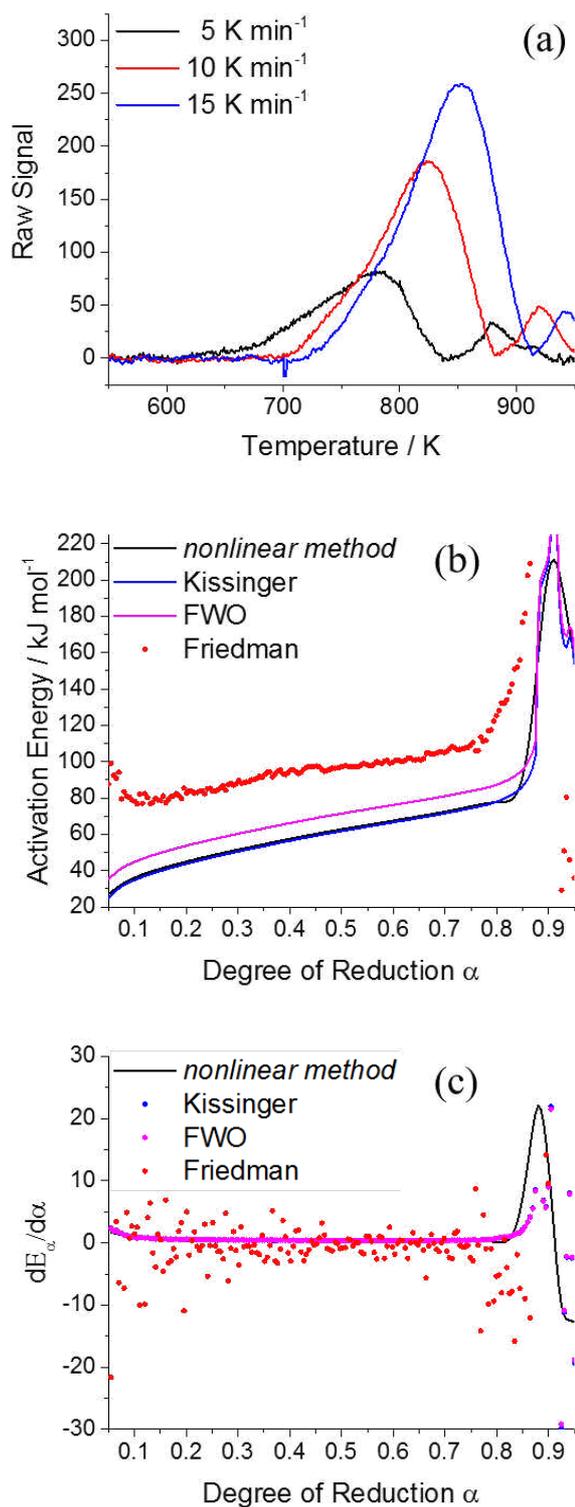


Figure 4.17: a) CeO_2 catalyst calcined at 600°C baseline corrected data b) activation energy c) $d\alpha/dT$ profiles respectively

and that the main peak is the reduction of the same material seen in the material when it is calcined at 400 and 500°C , the thermogram was split so that the only

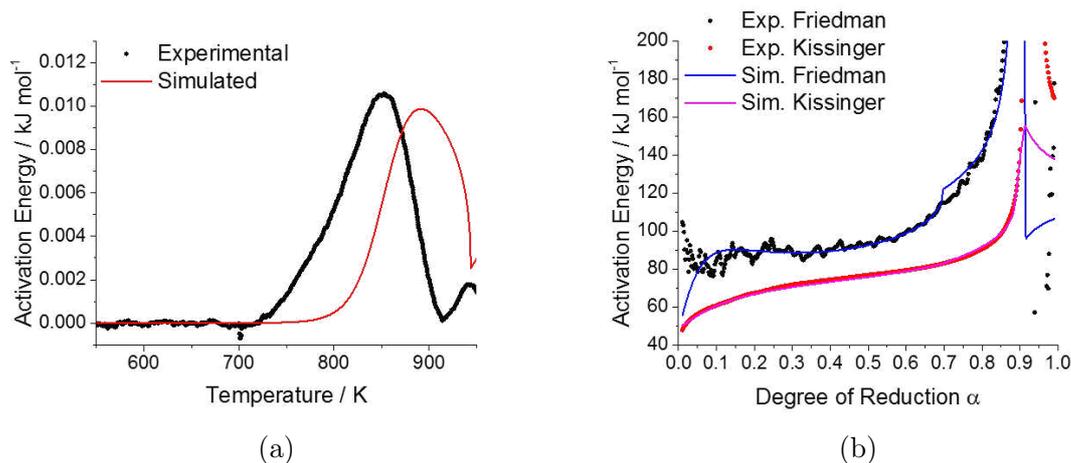


Figure 4.18: Experimental (Exp.) and simulated four process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO_2 calcined at 600°C a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

peak being modelled was the secondary one.

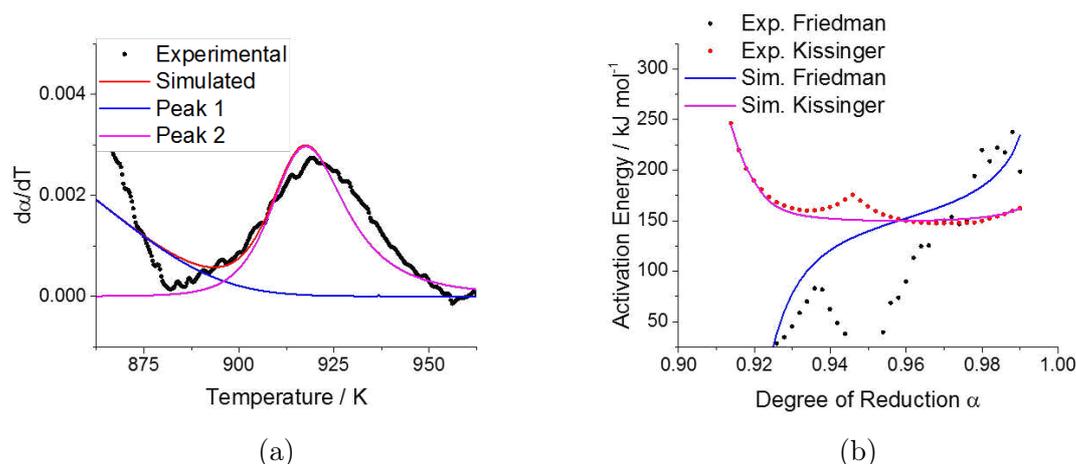


Figure 4.19: Experimental (Exp.) and simulated two process (Sim.) $dE_a/d\alpha$ and activation energy profiles overlaid for CeO_2 calcined at 600°C zoomed on second peak a) $d\alpha/dT$ profiles with deconvoluted peaks b) Activation energy profiles for Kissinger and Friedman methods.

Table 4.10: Coefficients calculated from regression algorithm

Peak	A	E_a	m	n	γ	p
1	5319901	144.2	0.95	1.69	0.000	0.08

Using this method some kinetic parameters were able to be calculated for the secondary process. It would seem that the second phase has a much higher activation energy for reduction, which is to be expected, and the Arrhenius pre-exponential A

is much higher when compared to the other processes from the thermograms for the material reduced at 400 and 500°C. The material reduced at 600°C is presented as the extreme case when it comes to deconvolution of TPR thermograms, when the baseline is subjective, and the signal to noise ratio is very low.

4.7 Understanding the Reduction Process

In the previous section, the thermograms for CeO₂ have been analysed using the new method outlined in this thesis. In this section the kinetic parameters and deconvolved curves are used to analyse the process for the reduction of CeO₂. The kinetic parameters calculated for the two materials are shown below for clarity.

Table 4.11: Coefficients calculated from regression algorithm for CeO₂ calcined at 400 and 500°C

Peak	A	E_a	m	n	γ	p
400°C: 1	14	43.8	0.75	0.87	0.000	0.06
2	24	53.6	0.39	0.77	0.029	0.67
3	4918	90.8	0.53	0.73	0.012	0.27
500°C: 1	8	42.5	0.66	1.01	0.026	0.12
2	53	60.3	0.51	0.76	0.089	0.69
3	605	76.5	0.67	0.67	0.009	0.19

The first step was to run the deconvolved peaks through Malek's procedure, in order to see if a kinetic model can be applied to each reduction process. Table 4.12 shows the computed kinetic models for the various processes.

When comparing the reduction processes on each of the materials, the first process (Peak 1) does not seem to follow any explicit reduction model and hence defaults to the SB(m,n) model. It is promising that the kinetic function predicted from Malek's procedure in table 4.12 replicates the kinetic function calculated through the regression method in table 4.11, further supporting the validity of the regression method. Although the SB(m,n) model is not derived from any explicit reduction

Table 4.12: Kinetic models calculate from Malek's procedure for CeO₂ calcined at 400 and 500°C

Peak	$z(\alpha_m)$	$y(\alpha_m)$	model	function
400°C: 1	0.590	0.463	SB(m,n)	$\alpha^{0.750}(1-\alpha)^{0.869}$
2	0.634	0.340	JMA($M > 1$)	$1.709(1-\alpha)[- \ln(1-\alpha)]^{1-1/1.709}$
3	0.635	0.420	JMA($M > 1$)	$2.199(1-\alpha)[- \ln(1-\alpha)]^{1-1/2.199}$
500°C: 1	0.551	0.394	SB(m,n)	$\alpha^{0.662}(1-\alpha)^{1.017}$
2	0.586	0.402	SB(m,n)	$\alpha^{0.508}(1-\alpha)^{0.755}$
3	0.642	0.499	JMA($M > 1$)	$3.233(1-\alpha)[- \ln(1-\alpha)]^{1-1/3.233}$

method, some attempt has been made at trying to understand the kinetic parameters m and n . [3] The parameter m has been indicated to relate to the influence of the reduced area of the material, with the n parameter indicating the influence of the oxide on the reduction process. The explicit values of the kinetic parameters are arbitrary, but when used in comparison for similar kinetic processes, more information can be divulged from them.

For process 1 (peak 1) the value for m decreases and the value of n increases for the CeO₂ calcined at 500°C when compared to the material calcined at 400°C. This indicates both a decrease in the influence of the reduced material and an increase in the influence of the metal oxide on the reduction process. The very similar activation energy would indicate that chemically the same reduction process is occurring, and that this difference in influence is purely physical. This is backed up by the change in the Arrhenius pre-exponential A which is lower in the 500°C catalyst, which also correlates to a larger value for γ which indicates that as more of the material is reduced the reduction process becomes more difficult. Perhaps one could assume that the reduction for the first process is around a specific type of ceria grain, and in the catalyst calcined at 500°C the grains are larger, as would be expected by higher temperature calcination. This would create a lower surface area (hence the lowered A value) and if the reduction occurs on the grain boundary, as the larger grain is reduced the distance between the grain boundary and the oxide increases which would increase the energy required for the reduction, which would not be seen

in the smaller grains.

For the second process, which make up 67% and 69% of the materials respectively for the catalyst calcined at 400°C follows a Johnson-Mehl-Avrami type reduction process, and at 500°C does not follow any specific mechanism. This would indicate that the bulk of the material has changed when the calcination temperature is increased. This is further backed by the increase in the the kinetic coefficients A , E_a and the kinetic parameter m . As the influence of the metal is increased in the catalyst calcined at 500°C, one can assume a polymorphic change where more Ce metal is present on the surface has occurred.

The final processes, both follow a Johnson-Mehl-Avrami reduction model, with the catalyst calcined at 500°C having an increased value of M . The values of M have been shown to correlate to specific reduction processes[4], with a value of $M \approx 2$ indicating a grain edge nucleation method, and a value of $3 > M > 4$ indicating a decreasing nucleation rate. This is backed up by the decrease in the kinetic parameter A when $M = 3.233$, and the different activation energies indicating different reduction processes. As the P parameters seem to be correlated between process 1 and 3, perhaps process 3 is the reduction of the material on the grain boundary of the material reduced in process 1. As the size of the particles reduced in process 1 seem to have increased in the material calcined at 500°C this would be reflected in less area for the material reduced in process 3.

Here the surface structure and possible changes in the material have been discussed in great detail, of course it would be impossible to confirm these polymorphic changes in the material without some other form of characterisation (XRD, XAFS *etc.*), but the idea here is to show how much information it is possible to gain from what is usually considered to be a basic experiment with little to no kinetic information. Using the new methodology outlined in this thesis, the quality and quantity of information available from TPR thermograms has been greatly increased, and hopefully will prompt a more in-depth approach when it comes to kinetic analysis.

4.8 Summary

Similar to the previous chapter, there has been an initial focus on the development of software, with the aim of making the analysis of Temperature Programmed Reactions more accessible. The current literature has been very poor in applying the described techniques either due to a lack of understanding or simply being unaware that they exist. It is hoped that by taking the very well established theory behind the non-isothermal analysis and creating a new technique for analysis, these methods will become more popular. It has been shown using simulated data that this new method can provide in-depth information about the reduction process of a material, with one of the major perks being an ability to deconvolute a reduction processes with a high level of accuracy not possible with standard shape regression techniques. The new methodology has been applied to various CeO_2 materials which were calcined at varying temperatures, and it was found that for the samples calcined at 400 and 500°C it was possible to deconvolve three different reduction processes, and therefore three different phases of the material. The differing kinetic parameters calculated from the regression model found that when increasing the calcination temperature from 400°C to 500°C there was some phase change in the material, but this could not be proved without further characterisation techniques. The limitation of the methodology was demonstrated when studying the CeO_2 sample calcined at 600°C, as if the reduction spectrum is not clean, and if the reduction process does not return to the baseline the assumptions made during the numerical integration (that the whole process is completed) are no longer true. A future study could perhaps be in attempt to counteract this problem by developing a technique of expanding TPR thermograms theoretically, as is seen in the TAP literature.

Bibliography

- (1) J. Malek, *Thermochimica acta*, 1992, **200**, 257–269.
- (2) V. Gorbachev, *Journal of Thermal Analysis and Calorimetry*, 1983, **27**, 151–154.
- (3) G. Munteanu, L. Ilieva, R. Nedyalkova and D. Andreeva, *Applied Catalysis A: General*, 2004, **277**, 31–40.
- (4) J. W. Christian, *The Theory of Transformations in Metals and Alloys Ediz 2. an Advanced Textbook in Physical Metallurgy Parte 1. Equilibrium and General Kinetic Theory*, Pergamon Press, 1975.

To increase the quality of the catalyst, the most important step is understanding how the reactant interacts with the catalyst surface. As in-situ methods require highly specialised equipment and training, DFT and Molecular Dynamics was used to attempt to understand both the reaction mechanism and the catalytic system.

5.2 Understanding the Catalyst Material

The first step in attempting to perform a reaction is to design a catalyst. Once Cu–ZrO₂ was chosen as the desired material for the catalyst, the catalyst was synthesised by multiple methods in order to gauge which would be the most active, and then by using standard tools such as ICP, XRD, XPS and TEM the structure of the catalyst was analysed. There were two methods to synthesise the catalyst, the methanothermal method (Me) and the oxalate gel precipitation method (Og), the actual steps required in these methods are outside the scope of this thesis and are well described in the paper by Satoshi *et. al.*[1] It was found that the Me method would purely substitute Cu into the tetragonal ZrO₂ lattice, whereas the Og method would substitute into the tetragonal lattice and deposit CuO onto the surface of the catalytic material.

For each of the catalysts being studied, the lattice parameters a and c were calculated via powder x-ray diffraction (PXRD). It was theorised that there was a maximum Cu content that could be incorporated into the tetragonal ZrO₂ lattice, as with increasing Cu content there was little to no shift seen in the lattice parameter c as the copper content was increased past 21% (table 5.1). In order to confirm these findings, DFT was used to simulate each of the Cu–ZrO₂ catalyst materials with various amounts of copper doping to see if the shift in lattice parameter could be replicated.

The calculations were preformed using the CASTEP[2–10] simulation package, with the Perdew-Burke Ernzerhof (PBE) functional and auto generated pseudopo-

Table 5.1: Physio-chemical properties of Cu–ZrO₂ catalysts

Catalyst	Cu / (Cu + Zr) / %		Lattice Parameter / Å	
	Prepared	Actual	<i>a</i>	<i>b</i>
ZrO ₂	0	0	3.60	5.15
Cu–ZrO ₂ (Me)	20	7.6	3.59	5.08
Cu–ZrO ₂ (Og)	10	10.7	3.60	5.13
Cu–ZrO ₂ (Og)	20	21.0	3.59	5.08
Cu–ZrO ₂ (Og)	30	31.3	3.59	5.08
Cu–ZrO ₂ (Og)	40	41.6	3.60	5.07
Cu–ZrO ₂ (Og)	50	51.8	3.60	5.08

tentials. A plane wave basis set was used with a three dimensional periodic boundary, with a cutoff value of 630 eV, with finite basis set corrections included for all calculations. The G06 long range dispersion force correction by Grimme was included, along with a Hubbard U value of 7 eV for the d orbitals on the Cu metal. The different levels of doping of Cu in the ZrO₂ lattice were simulated using a 3×3×3 supercell (Zr₅₂O₁₀₄) in which a number of Zr⁴⁺ ions were substituted with Cu²⁺ ions so that the percentage of Cu to Zr was as close as possible to the experimental values. In order to make sure the unit cell was charge neutral an appropriate number of O²⁻ ions were also removed. The Cu atoms were positioned so that the second nearest neighbours for each Cu atom were made up purely of Zr atoms. The atom positions and the unit cell geometries were both included in the optimisation procedure in order to calculate lattice parameters for each level of Cu doping. The lattice parameters reported were taken as an average of all possible charge compensating defects available for each Cu dopant site. To compare the stability of each complex the mixing energy was calculated using:

$$E_{mixing} = E_{\text{Cu}_x\text{Zr}_{n-x}\text{O}_{2n-x}} - ((n-x)E_{\text{ZrO}_2} + xE_{\text{CuO}}) \quad (5.1)$$

$E_{\text{Cu}_x\text{Zr}_{n-x}\text{O}_{2n-x}}$ is taken as the calculated energy for the optimised doped unit cell, E_{ZrO_2} is the energy of the optimised reference ZrO₂ unit cell, and E_{CuO} is the energy of the optimised reference CuO unit cell. Cu doping percentages are measured using

the same formula as table 5.1.

The first step was to create the reference ZrO_2 and CuO unit cells. In order to keep the values for the energies consistent, $3 \times 3 \times 3$ supercells were used for both reference structures. The optimisation of these structures was performed in a single calculation using the CASTEP package.

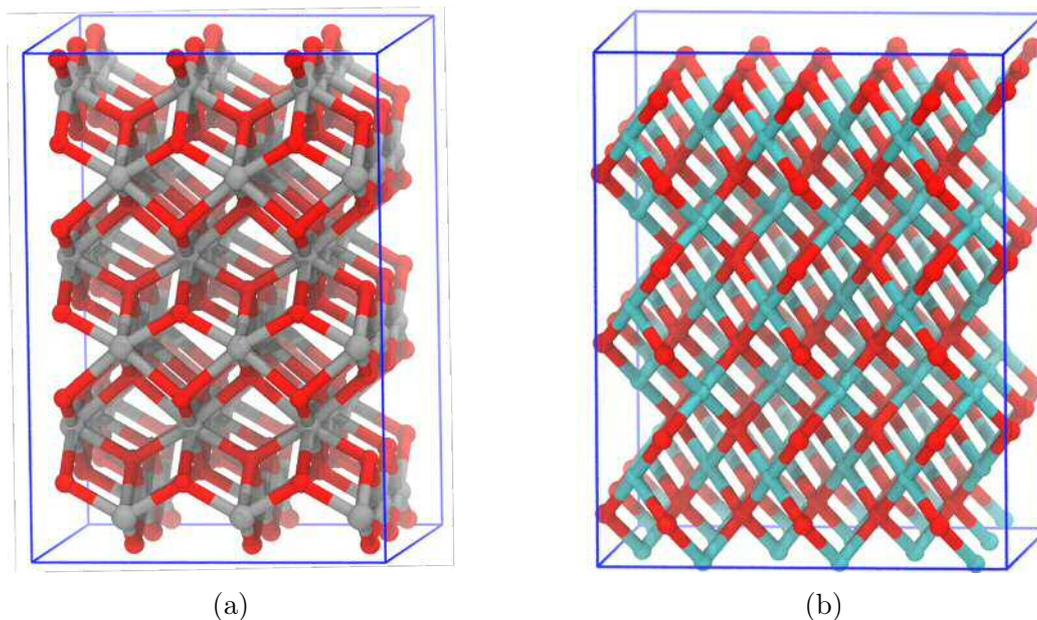


Figure 5.1: Standard unit cells used in energy calculations a) $\text{Zr}_{54}\text{O}_{108}$ unit cell, Zr - Grey and O - Red b) $\text{Cu}_{108}\text{O}_{108}$ unit cell, Cu - Cyan and O - Red

Table 5.2: Simulated and (experimental) unit cell parameters for the ZrO_2 and CuO unit cells calculated from the $3 \times 3 \times 3$ supercell

System	Unit Cell Parameters / \AA		
	a	b	c
$\text{Zr}_{54}\text{O}_{108}$	3.59 (3.58)	3.59 (3.58)	5.27 (5.16)
$\text{Cu}_{108}\text{O}_{108}$	—	—	5.18 (5.13)

For comparison with the experimental (taken from ICSD[11] 9993 and 653723) values for the lattice parameters, the calculated lattice parameters were calculated by dividing the $3 \times 3 \times 3$ supercell lattice parameters by 3 to give the lattice parameters for the basic unit cells Zr_2O_4 and Cu_2O_2 and are shown in table 5.2. For the CuO the a and b parameters are removed as a orthorhombic non-standard cubic unit cell was used in the calculation in order to increase the speed and accuracy of the

optimisation method and therefore no comparison can be made. CASTEP over predicts the lattice parameter c for both the $Zr_{54}O_{108}$ and the $Cu_{108}O_{108}$ systems by approximately 2%, but for consistency sake the c values relative to the starting structure are taken instead.

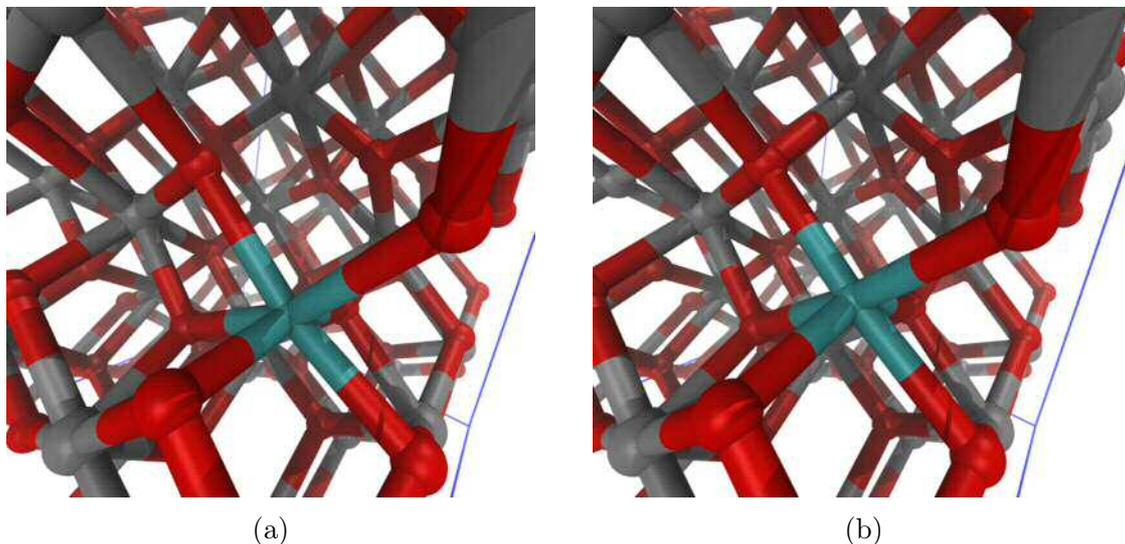


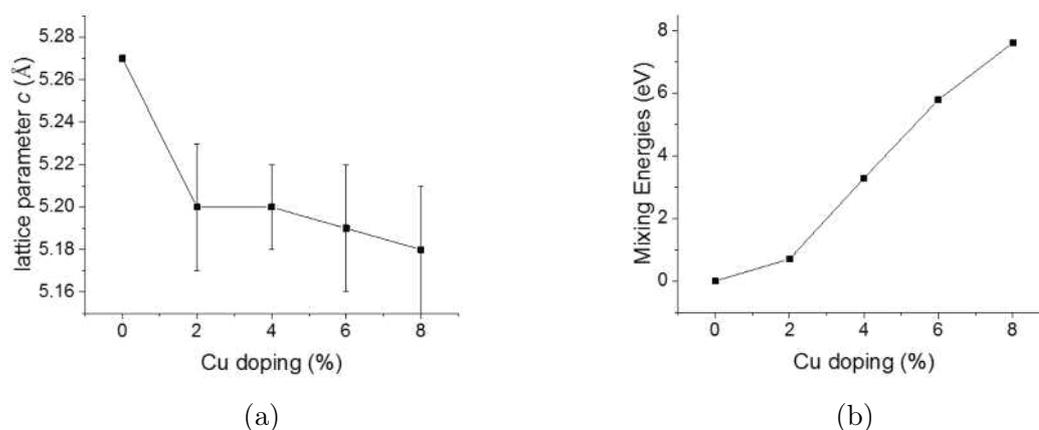
Figure 5.2: Optimisation of Cu doped ZrO_2 a) Initial $CuZr_{53}O_{107}$ unit cell, Zr - Grey, O - Red and Cu - Cyan b) Optimised $CuZr_{53}O_{107}$ unit cell, Zr - Grey, O - Red and Cu - Cyan

To recreate the doping, one of the Zr^{4+} ions was replaced with a Cu^{2+} ion, and then a O^{2-} ion was removed in order to keep the unit cell charge neutral. The defected system (figure 5.2a) is allowed to optimise, allowing both the atomic positions and the unit cell parameters to change, and the final structure 5.2b is then analysed. This process was repeated for multiple Cu^{2+} ion loadings, with the lowest energy structure from each of the subsequent loadings carried forward. The lattice parameters and the mixing energy (E_{mixing}) has been recorded as a function of doping % (rounded) and are shown in figure 5.3 and table 5.3.

The lattice shift that can be seen in table 5.1 was replicated by the DFT results. The contraction of the lattice parameter c can be seen on incorporation of the Cu^{2+} ions into the bulk ZrO_2 unit cell structure, and the insensitivity of the lattice parameter a has also been reproduced. In the optimised structures, it was seen as a common factor the Cu^{2+} ion adopting the square planar geometry (figure 5.2),

Table 5.3: Simulated with (error) physiochemical properties for standard ZrO_2 and CuO unit cells normalised

System	Doping %	Mixing Energy eV	Unit Cell Parameters / Å		
			<i>a</i>	<i>b</i>	<i>c</i>
$\text{Zr}_{54}\text{O}_{108}$	0	0.00	3.59	3.59	5.27
$\text{CuZr}_{53}\text{O}_{107}$	2	0.71	3.59 (0.02)	3.59 (0.02)	5.20 (0.03)
$\text{Cu}_2\text{Zr}_{52}\text{O}_{106}$	4	3.29	3.58 (0.02)	3.61 (0.02)	5.20 (0.03)
$\text{Cu}_3\text{Zr}_{51}\text{O}_{105}$	6	5.79	3.57 (0.02)	3.61 (0.02)	5.19 (0.03)
$\text{Cu}_4\text{Zr}_{50}\text{O}_{104}$	8	7.61	3.58 (0.02)	3.61 (0.02)	5.18 (0.03)

Figure 5.3: Simulated physiochemical properties for standard ZrO_2 and CuO unit cells normalised a) Lattice parameter *c* as a function of Cu doping % b) Mixing energy as a function of Cu doping %

migrating just outside of the Zr^{4+} ion site. As the number of Cu^{2+} ions is increased the mixing energy also increases, even after normalisation to Cu content, and it was observed that any doping in excess of 20% caused drastic shifts in the shape of the unit cell, which suggests that a high Cu content in the ZrO_2 lattice can not be supported. This could be in part due to the fact that at high levels of doping it is difficult to get an arrangement of Cu^{2+} ions such as that the second nearest neighbours are purely Zr^{4+} ions, and therefore the shift from the octahedral Zr site to the square planar Cu site cannot be facilitated without drastic rearrangement of the system. This assumption would mean that as the amount of Cu is increased in the catalytic material, after a certain point (21% loading from table 5.1) the amount of bulk Cu is constant, and therefore any increase in Cu content can be considered

to purely result in an increase in surface Cu.

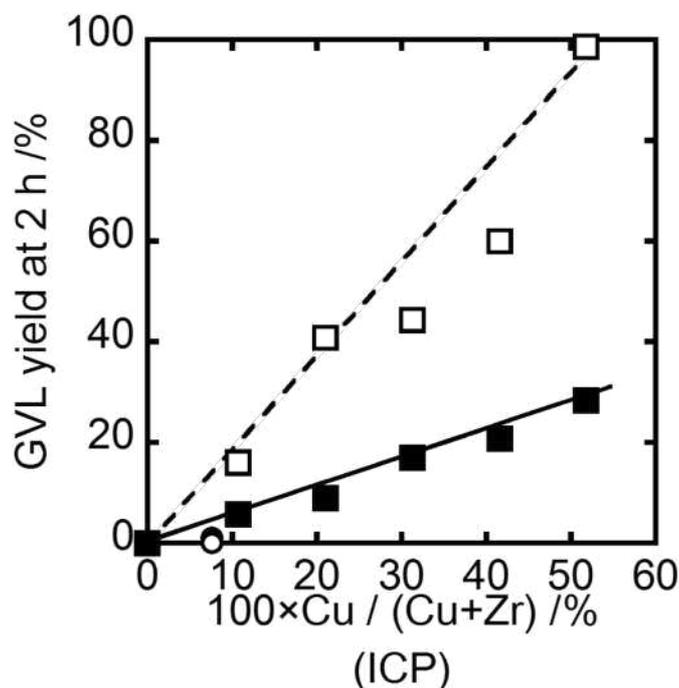


Figure 5.4: gVL yield as a function of Cu / (Cu + Zr) determined by ICP. Open square - Og preparation method, Closed square - Me preparation method taken from[1]

Figure 5.4 shows that the catalyst prepared by the Og method were more active for the hydrogenation of LA to gVL, and that increasing copper content seemingly increased the activity of the catalyst. Indicating that the surface copper is more important than the lattice copper for the actual reaction. The Og method was then refined by studying the effect of changing Cu / (Cu + Zr) ratio, ageing time, and calcination temperature[12] until an optimum catalyst had been designed with (Cu / (Cu + Zr) of 50, an ageing time of 6 hours, and calcination at 400°C). One of the key points from the work was that although the absolute value for the conversion of LA to gVL was highest with these parameters, when the conversion was normalised to Cu surface area (figure 5.5[12]) the rate was seemingly constant.

This idea that the rate was a function of Cu surface area further indicated that the surface copper was the most important component of the catalyst in the reduction of LA to gVL. Some attempts were made at modelling the surface of the material but the Cu–ZrO₂ system was found to be too complex and the number of

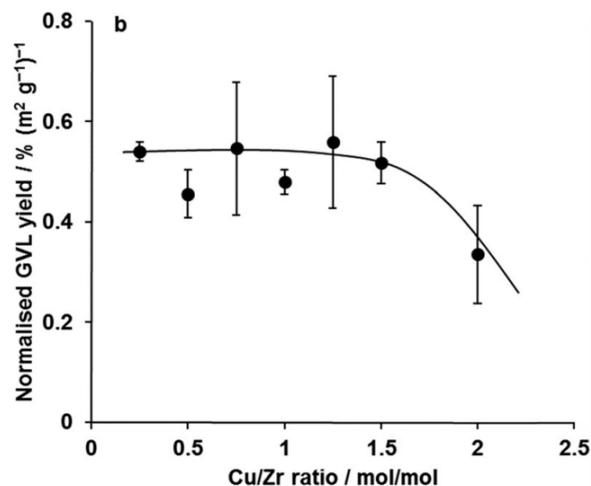


Figure 5.5: gVL yield normalised by BET surface area taken from [12]

possible systems was far too large for any useful analysis to be performed. Instead it was opted to take Cu nanoclusters instead, as it was assumed they are well studied in the literature, and their highly symmetrical nature makes them computationally friendly.

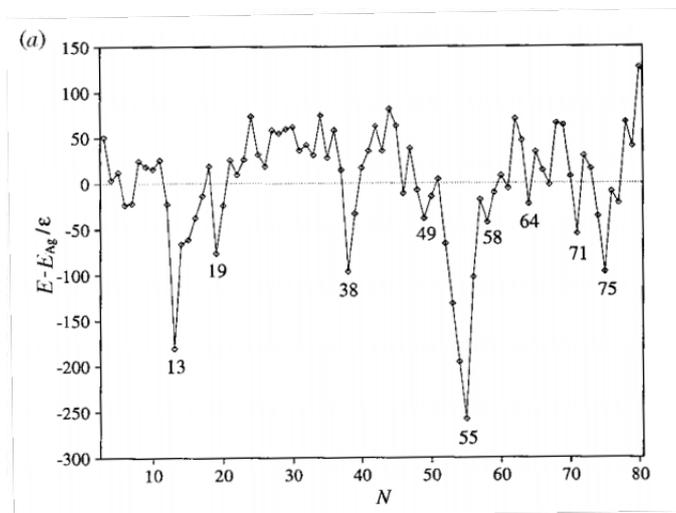


Figure 5.6: Energies of the various nanoclusters normalised to cluster size taken from [13]

There are a massive range of sizes and shapes when it comes to nanoclusters. The shapes of the clusters were taken from the Sutton-Chen [13] global minima transition metal clusters. The sizes of the Sutton-Chen clusters vary from 9 to 80 atoms, but as shown in figure 5.6 [13] specific clusters are more stable than others, the sizes in particular being 13, 55, 38, and 75 atoms. The most popular clusters used are the 13

atom and the 38 atom due to their high symmetry and various facet sites available, but for this work the 75 atom cluster was used.

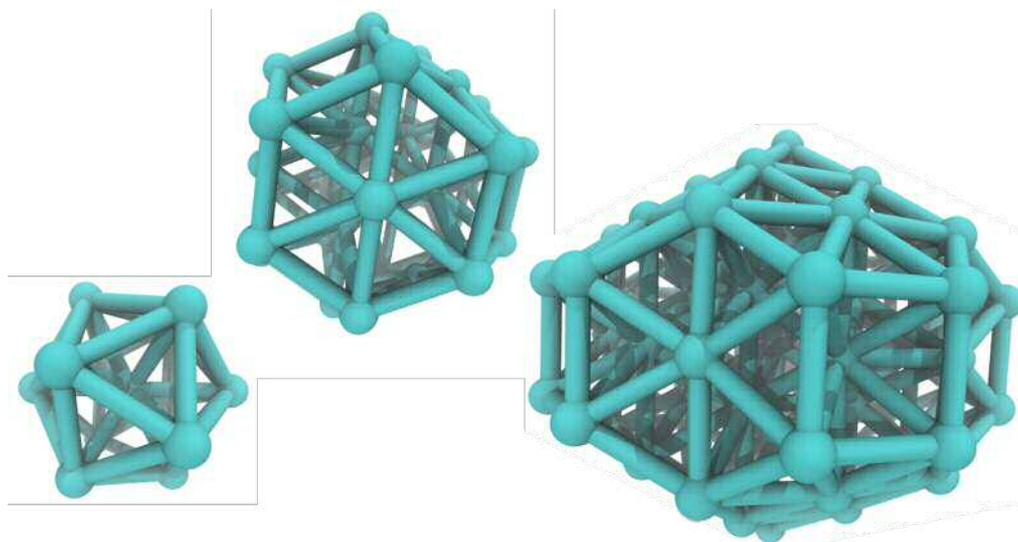


Figure 5.7: 13, 38, and 75 atom nanocluster shapes

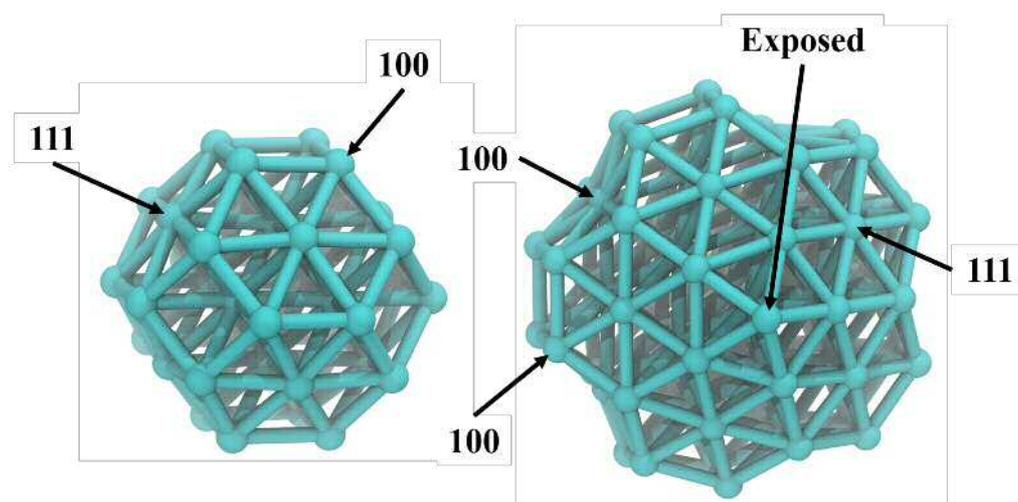


Figure 5.8: Various sites available on 38 and 75 atom nanoclusters

It was decided that the 75 atom nanocluster (figure 5.7) would be used as it had the largest range of sites on a single cluster. Figure 5.8 shows the comparison between the 38 atom and the 75 atom clusters, as it can be seen the 38 atom cluster only contains the (111) and (100) facets, whereas the 75 atom cluster has (111), (100), (110), and an *exposed* atom site. As the (111) and (100) facets from the 38 atom cluster are replicated in the 75 atom cluster, along with two extras it was deemed that the 75 atom cluster would be the desired material for modelling the surface copper.

5.3 The Reaction Mechanism

Once the catalytic material had been modelled the next step was to attempt to understand the reaction mechanism. Some previous work[14] had shown that when acetone (a simple ketone with enol isomer) was adsorbed onto a metal surface, it preferred to be in the enol form, and therefore this was taken as the first step in the reaction mechanism. Moving on from the enol, there are two more steps, a cyclisation via dehydration, and a hydrogenation which are required to form the gVL.

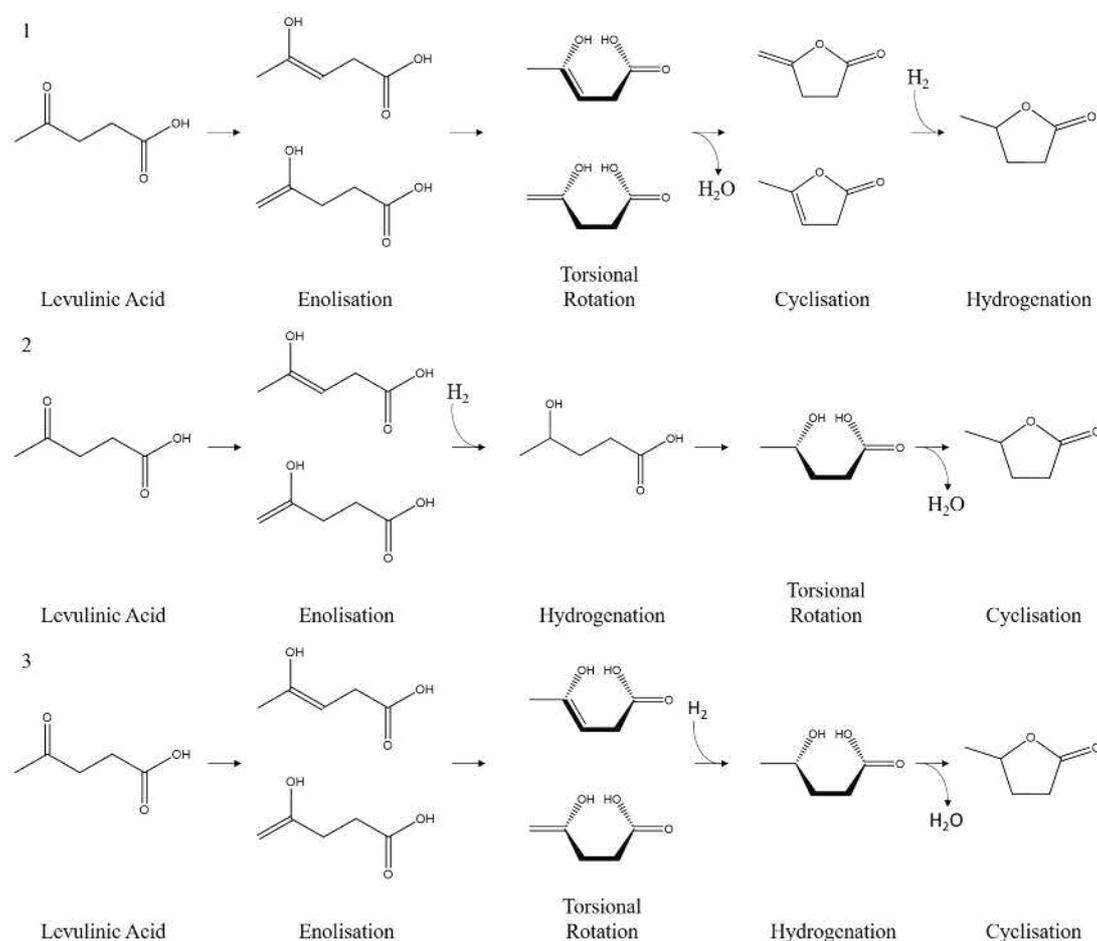


Figure 5.9: Possible mechanisms for conversion of LA to gVL (1-3)

Figure 5.9 shows the three possible mechanisms that were derived for this process by varying the order of hydrogenation, torsional rotation, and cyclisation. The first step in understanding the mechanism was to perform a simple *gas* phase analysis

of the molecules. This involves placing them in a large periodic box (to remove self interactions) and performing a geometry optimisation using DFT. The optimisation was performed using the VASP package, using a plane wave cutoff of 600 eV, the Perdew-Burke Ernzerhof (PBE) functional, in a 15 Å/ unit cell. In order to make sure the energies of the molecules were comparable the stoichiometry was kept constant (*e.g.* LA + H₂ and gVL + H₂O). Using this a simple reaction profile was developed.

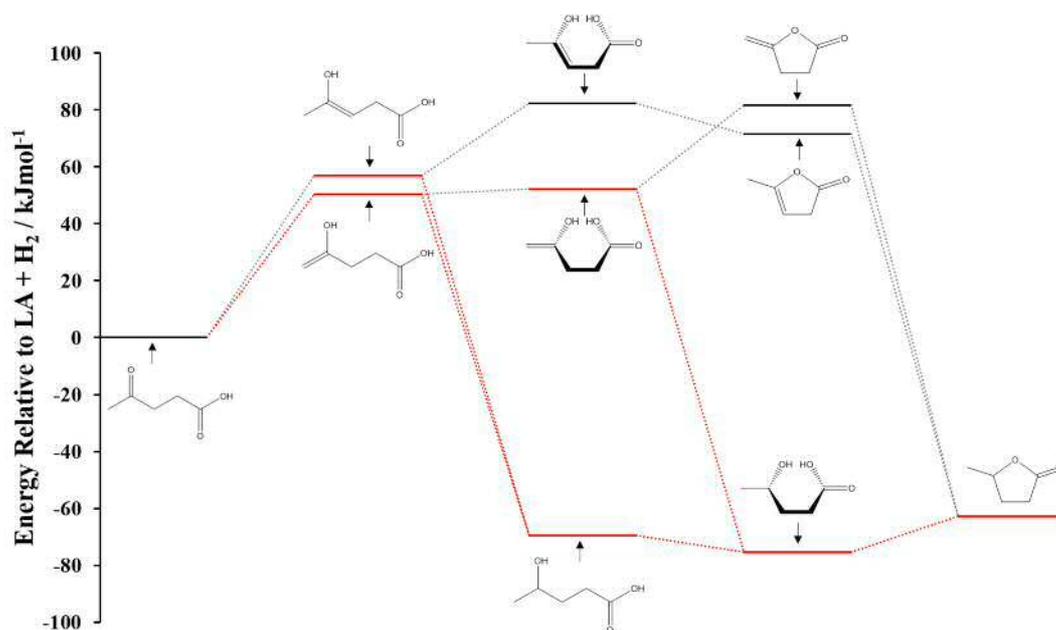


Figure 5.10: Gas phase reaction mechanism for conversion of LA to gVL with most likely mechanism shown in red.

Figure 5.10 shows the energies for the various molecules involved in the reaction mechanism, initial tests showed that the enol form of LA was less stable in the gas phase, but as the previous work showed that the enol only forms on the surface of the metal this was expected. The order of the hydrogenation and cyclisation step could be derived from the gas phase calculations. The hydrogenated forms of the enols were found to be much more stable than their cyclised counterparts, therefore it can be assumed that the reaction mechanism follows a hydrogenation and then cyclisation mechanism.

While this method can be used to provide insight into the reaction mechanism,

it is still massively flawed. The simulations are all static images, ran at 0K in the gas phase, and the system in question is in the liquid phase, at 200°C, at a pressure of 35 bar and also include the catalyst material. Therefore a more accurate system needs to be developed. The first step was to remove the factors of temperature and pressure by performing molecular dynamics, the molecular dynamics also allowed inclusion of the solvent (H₂O). In order to accurately recreate the system a specific flow chart was followed for each molecule:

- Take the optimised geometry from the gas phase
- Surround the molecule with 59 water molecules and place in to a 15Å/ unit cell so that no molecules overlap
- Perform a classical dynamics simulation
- Allow the system to fully relax using an NVE simulation
- Recreate the experimental conditions using an NPT simulation
- Perform ab-initio dynamics on the relaxed system using NVT
- Analyse the properties calculated from the ab-initio dynamics run

Creating the initial image by surrounding the molecule with water was performed using an in-group script *soak* which fills the unit cell with H₂O molecules while making sure no atoms occupy the same space. For this experiment 59 water molecules were used to simulate the solvent, as it was found to be the best trade off between accuracy and computational time for the Ab-Initio dynamics. The next step is to perform the molecular dynamics run, when running the system for the first time a NVE dynamics run was performed (fixed number of atoms, volume and energy) this allows the system to relax, as performing a NPT (fixed number of atoms, pressure and temperature) with an unphysical system can cause the simulation to fail. The classical dynamics simulations were performed using the DL_POLY package, using

OPLS2005 force fields for the atoms, and the TIP4P mode for the water, for a total of 2×10^6 0.5 fs timesteps, giving a total simulation time of 10 ns, as the simulation was run twice with (NPT and NVE) a total simulation time of 20 ns was used. The large total simulation time arises from the fact that liquids tend to have much longer relaxation times than gases or solids. After the classical simulations were finished the physical parameters of the systems were recorded. **[tab:mech1]**

Table 5.4: Physical parameters of classical dynamical simulations: EN - Enol, HYD - Hydrogenated Intermediate, EXP - Experimental

System	Volume \AA^3	Temperature $^{\circ}\text{C}$	Pressure bar
LA	1824.4	199	32.5
EN1	1793.8	199	33.9
EN2	1816.5	199	34.0
HYD	1830.5	198	33.8
GVL	1846.1	199	34.2
EXP	—	200	35.0

Table 5.5 shows that the temperature of the physical system is replicated very accurately. There is a slight underestimation in the pressure of the unit cell, this is most likely due to the small system size and unfortunately is unavoidable for this particular simulation, but it is well within experimental error. With the systems at their correct experimental conditions the next step is to transfer from classical to Ab-Initio dynamics. The biggest barrier is the conversion between the file types, as DL_POLY has it's own specific output file, and VASP (which is used for the Ab-Initio Dynamics) has it's own specific input file type a converter is required. The script used is another in-group script *analyse_hist* which can perform the conversion between the HISTORY file from DL_POLY and the POSCAR file required for the VASP job. The Ab-Initio dynamics is run using a NVT simulation rather than NPT, as pressure is a function of temperature and volume, as long as the two are kept constant the pressure from the NPT classical dynamics run should be conserved. The NVT dynamics is performed at 275°C , this increase in temperature is because the PBE functional is used, and it is known to over-predict the binding between

water molecules[15], and the tritium isotope of hydrogen is used allowing for longer timesteps, finally Grimme's D3 dispersion correction[16] is included to account for long range van der Waals forces as they have been shown to increase the accuracy of the simulation.[17] The simulations are for approximately 10000 steps giving a total simulation time of 20 ps. The 20 ps simulation time was chosen as the best trade off between computation and simulation time. The calculations are run using a plane wave basis set with a cutoff of 600 eV.

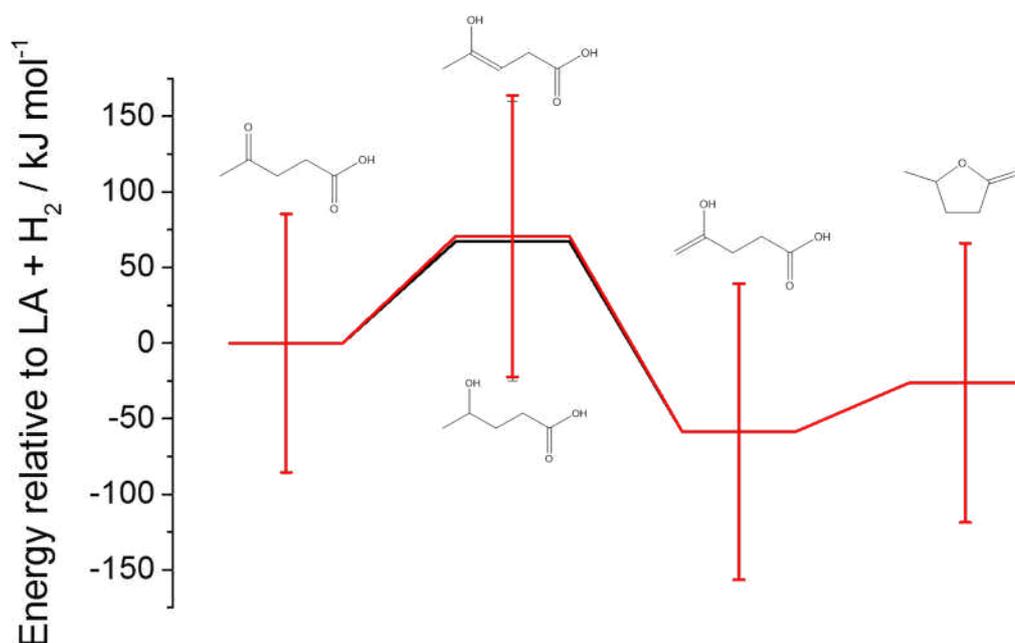


Figure 5.11: Liquid phase reaction mechanism for conversion of LA to gVL with errors bars showing the standard deviation of the energies over the course of the simulation.

The median energies, with the standard deviation overlaid are shown in figure 5.11, the reaction mechanism follows a similar pattern as the gas phase mechanism, with the enol form being less stable than the initial LA, and the hydrogenated intermediate being more stable. Another point to notice is the large standard deviation in energies, again this arises from the small system size and was to be expected. As the range of possible values for the energies is so large, the interaction between the molecules and the water, rather than the explicit values of the energies are more important.

In order to record the interaction between the water the radial pair distribution function, $g(r)$, was taken between the oxygen atoms on the molecules and the hydrogen atoms in the water (see figure 5.12). It can be considered that the water stabilises the oxygen atoms on the molecules via hydrogen bonding, and as such the degree of interaction between the hydrogen atoms on the water, and the oxygen on the molecules is a good indicator to the relative stability of the molecules.

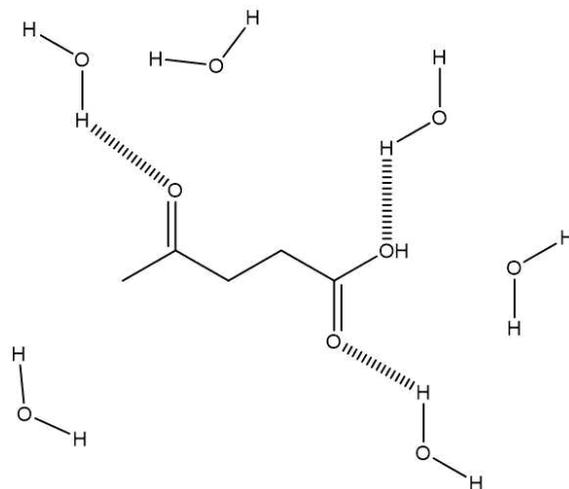


Figure 5.12: Schematic of interaction between the molecules and water

The radial distribution function is defined as the probability of finding a specific particle at a defined distance r from a given particle normalised to the if the atoms were evenly spaced.

$$g(r) = 4\pi r^2 \rho dr \quad (5.2)$$

where r is the distance from the reference particle, and dr is the width of the sampling space, and ρ is the number density of the particles

$$\rho = \frac{N}{V} \quad (5.3)$$

with N being the number of particles, and V being the volume of the sampling space. The radial distribution function is a standard tool for displaying averaged interactions between particles in a system, and they are calculated using the Visual

Molecular Dynamics[18] visualisation software. The two systems of interest are the hydrogenated intermediate and the enol, as they provide information on the reaction pathway and kinetics.

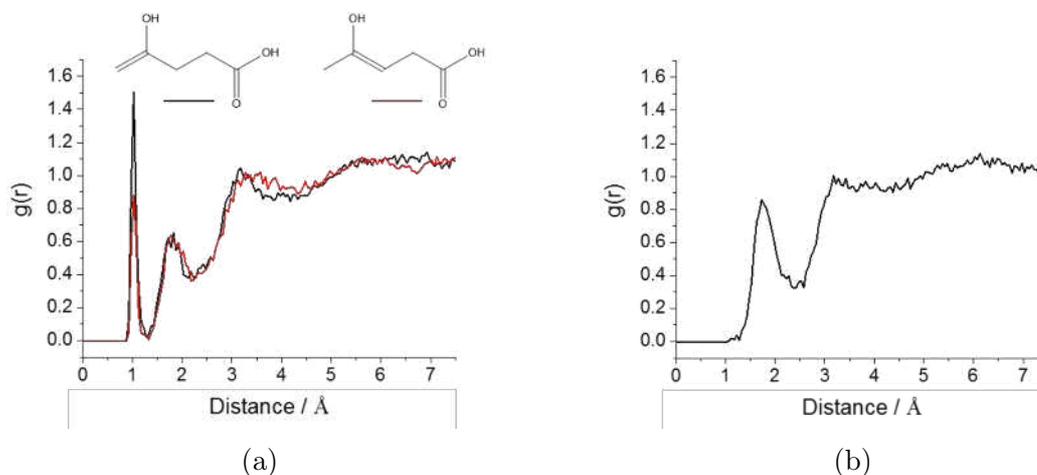


Figure 5.13: Radial distribution functions for the interaction between the oxygen atoms on the molecule, and the hydrogen atoms on the water. a) Both enol systems superimposed on top of each other b) Hydrogenated intermediate

It was interesting to see in figure 5.13 that the water interacts more strongly with the enol forms rather than with the hydrogenated intermediates, which would seem to be counter intuitive to the energies reported in figure 5.11 in which the hydrogenated intermediate was reported to be more stable. When comparing the two enol forms to each other, the enol with the external double bond (shown in black on figure 5.13) interacts more strongly with the water than the internal (shown in red). Studying the interaction between the oxygen on the molecules themselves will give information relating to the degree of *intramolecular* interactions on the molecule.

When studying figure 5.14a, the internal enol is considered to be much more rigid than the external enol shown by the sharp peaks in the radial distribution function, indicating that either there are strong intramolecular bonds holding the molecule together, or what is considered most likely is that the internal double bond is causing the carbon chain backbone of the molecule to stay rigid. When comparing the external enol to the hydrogenated intermediate, there is a much broader range

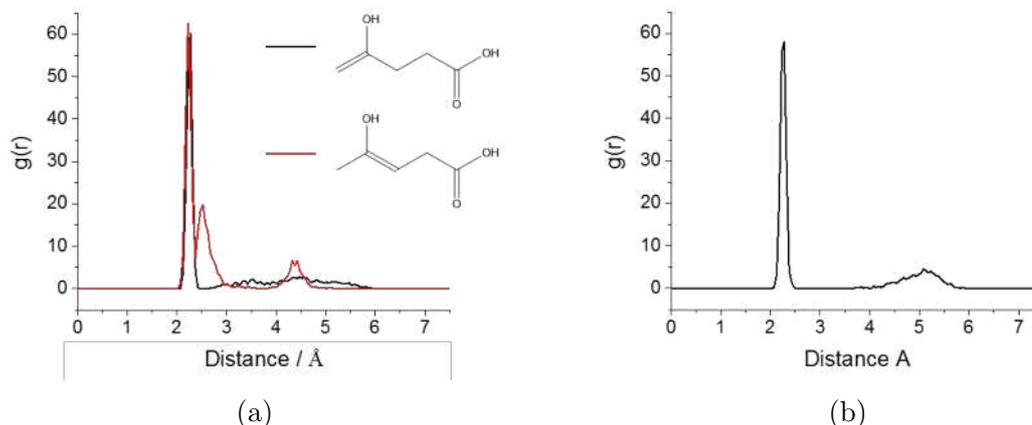


Figure 5.14: Radial distribution functions for the interaction between the oxygen atoms on the molecule a) Both enol systems superimposed on top of each other b) Hydrogenated intermediate.

over which the O-O interactions take place, indicating that the molecule is more flexible when in water. From studying these interactions it can be inferred that the enol is more stable in water than the hydrogenated intermediate, due to its increased flexibility and interaction with the water, the difference in energy arises from the fact that the enol is an unfavourable form of the ketone, and it can be assumed to be stabilised by the binding to the metal surface. From the data so far it can be assumed that the hydrogenated intermediate is the transition state for the reaction, but the rate determining step is not known to be the hydrogenation of the enol or the cyclisation to form the gVL.

5.4 Inclusion of Copper

The analogue for the reactive site on the catalyst was decided to be the Cu_{75} nanocluster mentioned previously. In order to understand the effect that the copper would have on the reaction mechanism the binding energies for each of the molecules was calculated on each of the various sites on the Cu nanocluster.

$$E_{\text{binding}} = E_{\text{molecule+surface}} - (E_{\text{molecule}} + E_{\text{surface}}) \quad (5.4)$$

Equation 5.4 is used to calculate the binding energy, where it is the difference between energies of the molecule and the surface (in this case the Cu₇₅ nanocluster) summed, and the energy of the bound molecule to the surface, with a negative binding energy indicating a favourable interaction.

Table 5.5: Binding energies for the various molecules bound to the Cu₇₅ nanocluster, with the 1, 2 and 3 binding sites shown in appendices 7.2:7.5.

System	Facet Binding Energy / eV			
	(100)	(111)	(110)	<i>exposed</i>
LA - 1	-0.67	-0.47	-0.19	-0.71
LA - 2	-0.45	-0.39	-0.40	-0.77
LA - 3	-0.29	-0.35	-0.35	-0.48
E1	-0.88	-0.75	-0.36	-0.83
HYD - 1	-0.77	-0.40	-0.53	-0.61
HYD - 2	-0.39	-	-0.31	-0.33
HYD - 3	-0.67	-0.56	-0.43	-0.76
gVL - 1	-0.51	-0.36	-0.38	-0.73
gVL - 2	-0.69	-0.54	-0.52	-0.19
H ₂ - 1	-0.22	-0.11	0.07	-0.25
H ₂ - 2	-0.99	-0.95	-0.83	-0.99

A full sampling of each of the binding modes for the LA, enol, hydrogenated intermediate, gVL, and H₂, on each of the binding sites, (100), (111), (110), and *exposed* was performed. The binding energies for each of the sites and binding modes were calculated, and from the binding energy calculations the most likely pathways were found. The first step was to understand where the H₂ was most likely to dissociate on the catalyst.

For all sites except the (110) the binding energy of the H₂ molecule is negative, and once the H₂ has been dissociated it is stable on the surface of the cluster as it's binding energy is much lower than molecular H₂. Although the (110) site has a positive binding energy, it is only 0.07 eV (approx. 7 kJ mol⁻¹) and at the temperature of 200°C and 35 bar pressure the hydrogen is most certainly going to bind to the surface and dissociate. The high hydrogen pressure required in the reaction can be explained by looking at the binding energies for molecular H₂ when compared

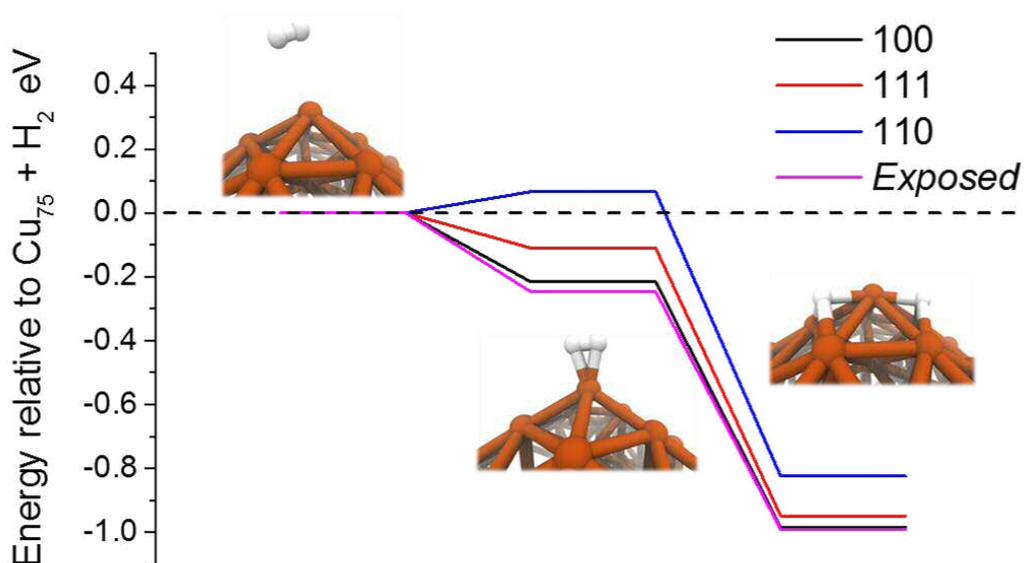


Figure 5.15: Mechanism for adsorption and dissociation of a H_2 molecule on the (100), (111), (110), and *exposed* sites on the Cu_{75} nanocluster, *exposed* site is pictured.

to the reactants and products. On every site the reactants and products bind more strongly than the molecular hydrogen, which means that the molecular hydrogen would not be able to displace one of these products to facilitate the dissociation into atomic hydrogen required for the reaction to continue, but with a high hydrogen pressure the surface coverage of hydrogen can be increased greatly, facilitating the initial binding of hydrogen to the Cu cluster.

The first mechanism to be analysed is the hydrogenation over the (100) site as shown in figure 5.16. Step 1 is taken as the dissociation of the hydrogen, the same as shown in figure 5.15. Afterwards it was calculated that the LA would most likely bind to the Cu surface through the ketone group as shown in step 2a and table 5.5. The LA can also bind in its enol form (step 2b), but from figure 5.16 it can be seen that the binding of the enol is less favourable than the LA. This is to be expected, as seen in the gas phase mechanism from the previous section the enol is much less favourable in the gas phase, and it also interacts strongly with H_2O meaning it will most likely be stabilised, therefore it would be premature to discredit the enol pathway. Afterwards the enol / LA is hydrogenated forming the hydrogenated

intermediate, due to the geometry of the LA and enol, the hydrogenated intermediate would most likely be bound to the surface through the hydroxyl group, and then through a surface re-arrangement would bind through the acid group (step 4), which is found to be more stable. The hydrogenated intermediate would then go under cyclisation through the loss of water forming the gVL as seen in step 5. Finally the gVL would go through a final surface rearrangement, binding to the surface more favourably as seen in step 6.

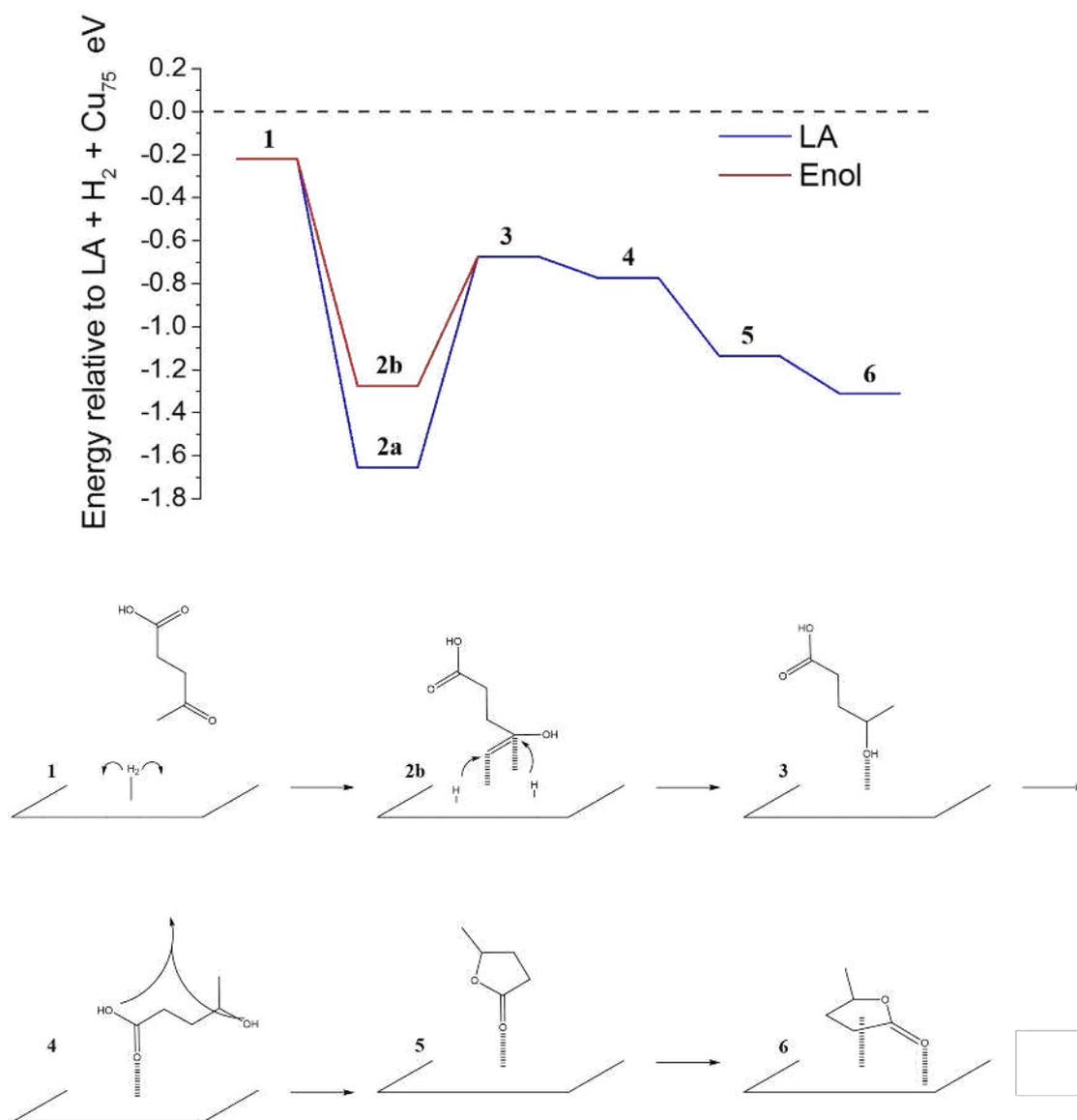


Figure 5.16: Mechanism for hydrogenation of LA to gVL over (100) site on Cu_{75}

The second mechanism is over the (111) site, and can be seen in figure 5.17, it can be seen that this mechanism involves only 4 steps compared to the 6 seen on

the (100) site, this is because there are less surface re-arrangements required. The overall energy of the mechanism is higher, as all of the reactants and products bind less strongly to the (111) site, indicating this is a less likely pathway than the (110) site. Similar to the mechanism seen in figure 5.16 the LA can bind through both through its ketone group and the enol form, seeing a similar pattern to the previous mechanism. The hydrogenated intermediate was found to bind more strongly through the hydroxyl group than the acid group (step 3), opposite to the previous mechanism, this means that no major surface rearrangement would be required and therefore the hydrogenated intermediate can directly cyclise, forming the gVL as shown in step 4.

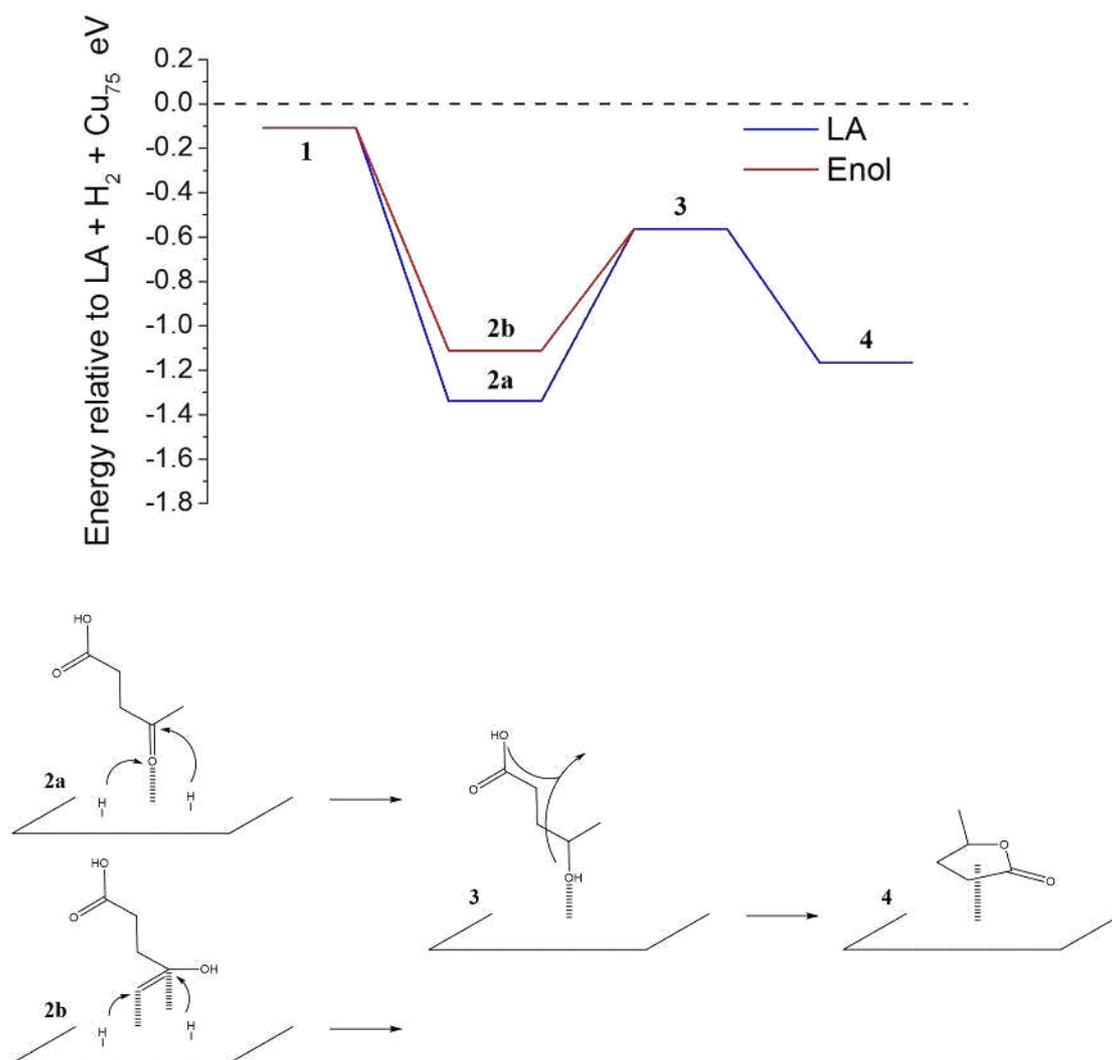


Figure 5.17: Mechanism for hydrogenation of LA to gVL over (111) site on Cu_{75}

The next site to be studied the (110) site, from the binding energy calculations

it was shown that this site is unfavourable for hydrogen dissociation when compared to the other sites, but due to the harsh conditions involved in the experiment it would most likely be accessible. When considering the (110) site it is also important to keep note of the geometry of the site, figure 5.18 shows that due to the repulsion between the Cu atoms it can be difficult for a molecule to get close enough to the surface to explicitly bind. Therefore when looking at the interaction between the molecules and the (110) surface it was found that there was no explicit bonding present, only long range ($> 2.8 \text{ \AA}$) interactions were found.

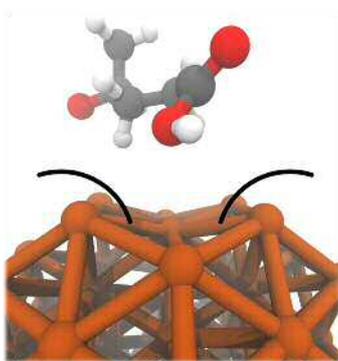


Figure 5.18: Geometry of (110) site on Cu_{75}

As the long range van-der-Waals interactions are always stabilising, the interaction with the surface is still favourable when compared to the individual gas phase molecules, as such a mechanism for the hydrogenation over the (110) surface still has a net negative energy when considering the binding energies and therefore can be considered possible. While the hydrogenation is still favourable over the (110) surface, when compared to the other three site (figures 5.16, 5.17, and 5.20) this mechanism is the least favourable and goes via a different route (figure 5.19).

The final site studied was the *exposed* atom site, it is expected that this site would be the most reactive as the Cu atom is on an edge and acts as an analogue for single atom sites that are sometimes seen in catalytic materials. The first observation is that the binding energy for the LA is the highest of all the sites, and that the strongest binding was not through the ketone group as seen on the other sites, but through the acid group instead. The second observation is that the gVL is very

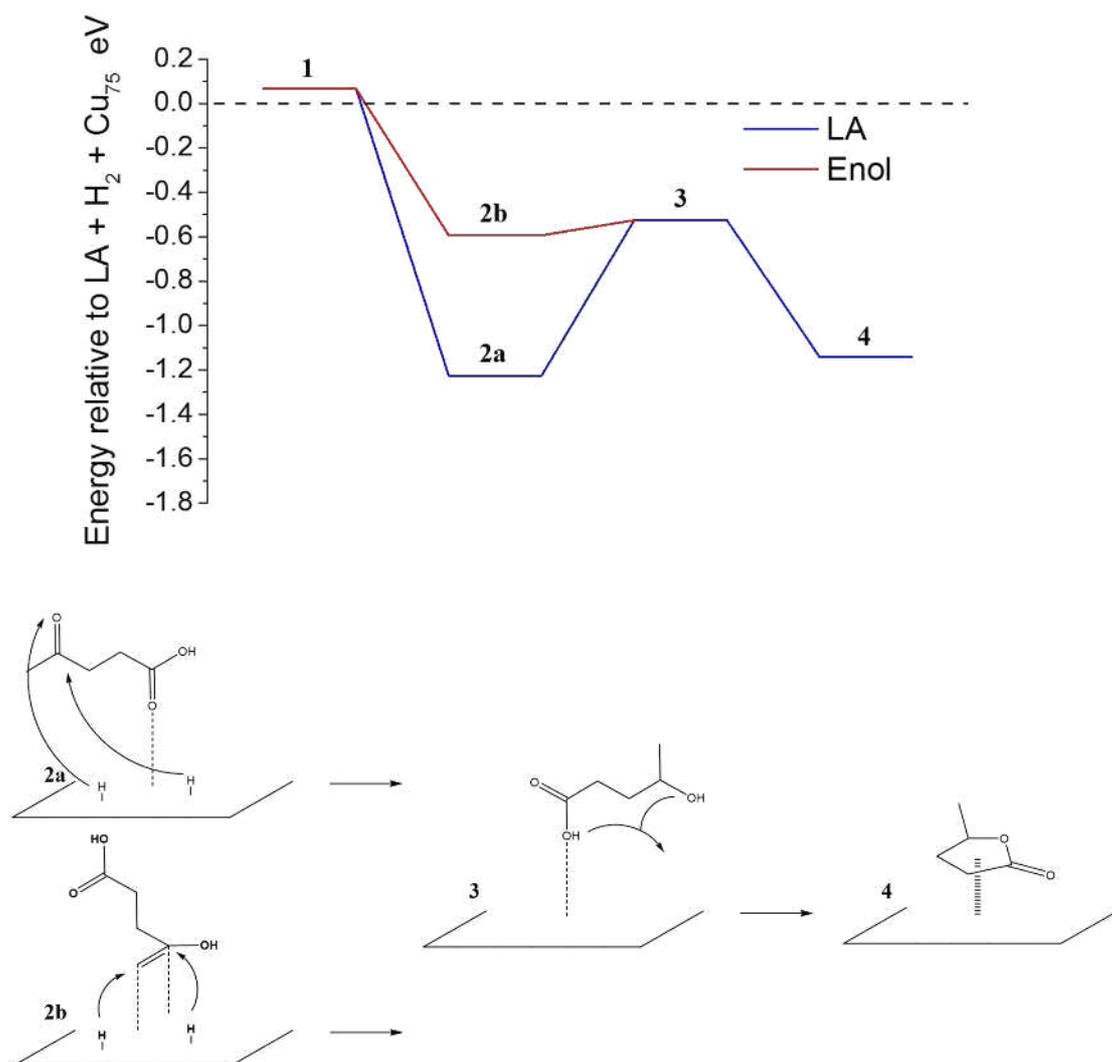


Figure 5.19: Mechanism for hydrogenation of LA to gVL over (110) site on Cu_{75}

strongly bound the surface through the carbonyl group, rather than a combination of the carbonyl and the 5 membered ring, this most likely arises from the geometry of the site rather than any electronic effect.

The gas phase binding energy calculations have been useful to show that there are many possible mechanisms available when converting LA to gVL using a Cu nanoparticle. In order to understand which of these pathways is the most likely to occur, the kinetics of the mechanism need to be better understood. Unfortunately calculating the kinetics of these reactions is not elementary as there are multiple steps with multiple variables involved.

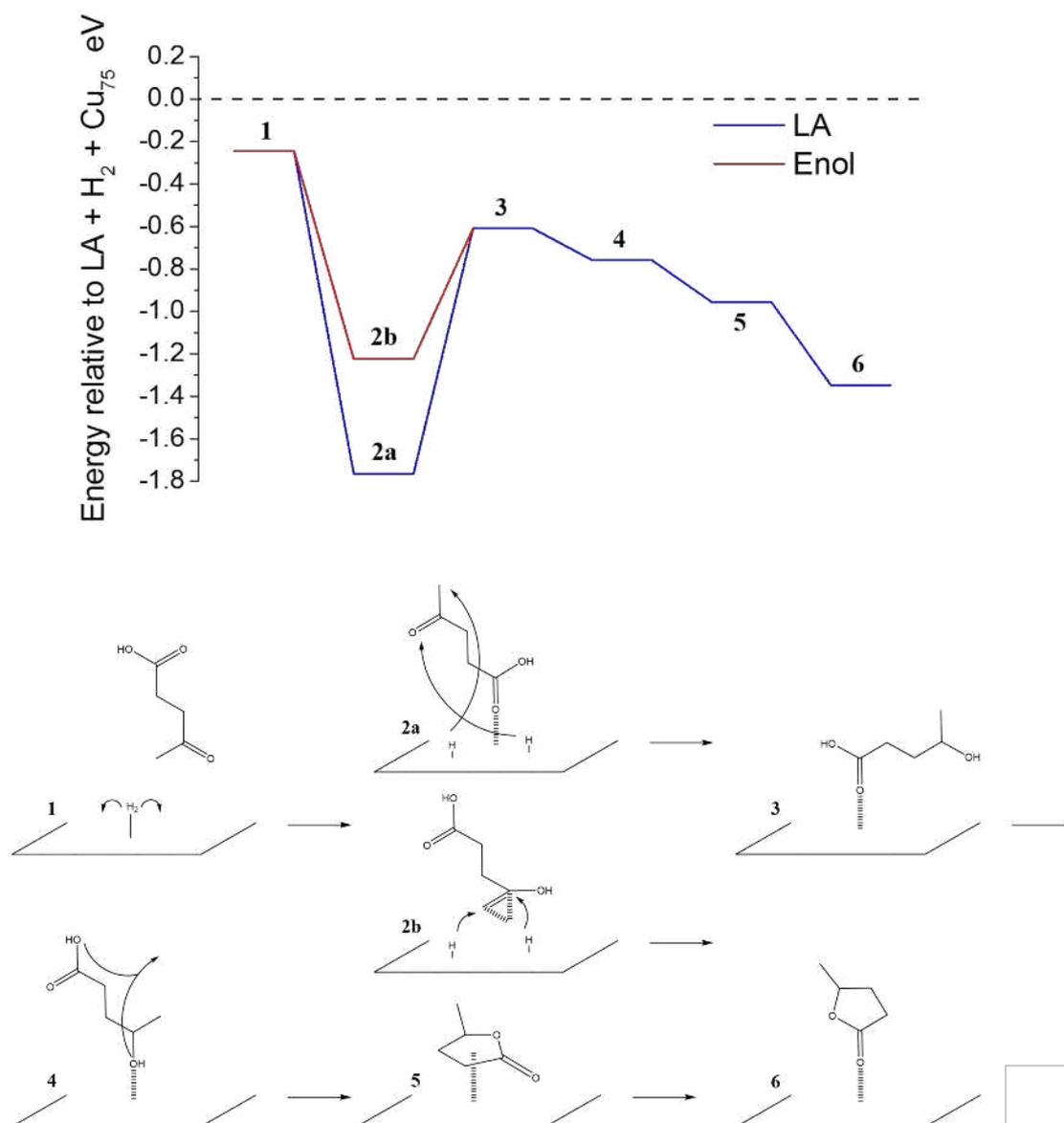


Figure 5.20: Mechanism for hydrogenation of LA to gVL over *exposed* site on Cu_{75}

5.5 Calculation of Energy Barriers

When it comes to understanding the kinetics of a reaction, the activation energy barrier is one of the most important pieces of information to obtain from a potential energy surface. Calculating activation energy barriers is one of the more difficult tasks when it comes to computational simulations, and there are numerous methods available to the user (see section 2.4.9). For the specific reaction at hand, there are three main barriers of interest:

- The hydrogen dissociation on the Cu nanoparticle
- The hydrogenation of the LA (or enol)
- the cyclisation of the hydrogenated intermediate to form gVL

The first barrier calculated was the ring closing of the hydrogenated intermediate to form gVL. The calculations were performed using the same parameters as the previous gas phase calculations (600 eV planewave cutoff, PBE functional, DFT-D3 correction). The NEB calculations were performed using the VASP Transition State Tools (VTST) package which is a third-party version of the VASP planewave DFT package.[19–22] It uses a climbing image method combined with the standard nudged elastic band method to calculate the transition state between a start and end point structure. In order to make sure the correct pathway is calculated the NEB calculation is run multiple times until it converges upon a specific pathway.

In order to get the transition pathway between two points (the hydrogenated intermediate, and the gVL) an initial pathway first needs to be set, this is done by using the in-house developed script *inter_vasp* which takes the start and end image, and interpolates between the two. This initial pathway is relaxed using the NEB algorithm to find the lowest energy pathway for the reaction.

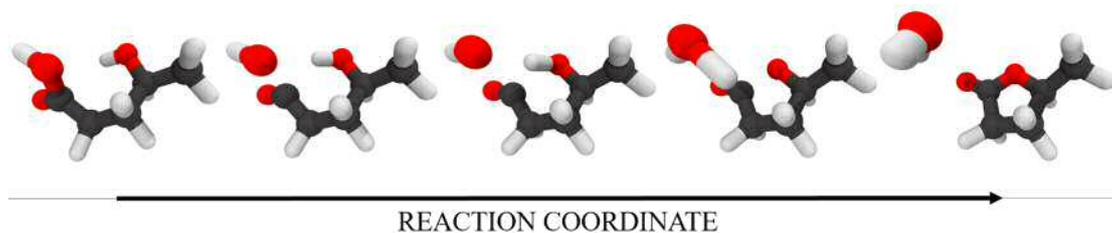


Figure 5.21: Example of interpolated images between start and end point (hydrogenated intermediate to gVL) as generated by *inter_vasp*

Each of the steps in the NEB calculation are referred to as *images* and each image will have a calculated energy, and it is using this energy that the reaction pathway can be shown. In order to study a reaction coordinate (or coordinates) is required.

This is some way of monitoring how far the reaction has progressed. For the ring closing mechanism the reaction coordinate can be defined as the distance between O_1 and C_5 as O_1 is the oxygen atom which forms the ring in gVL from Baldwin's Rules, or the distance between C_5 and O_2 as O_2 is the oxygen atom which leaves and forms water (see figure 5.22).

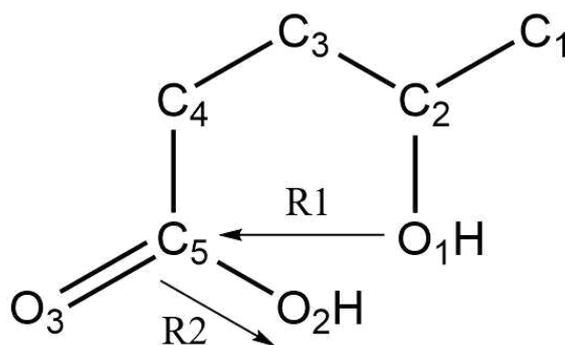


Figure 5.22: Hydrogenated intermediate with labelled atoms.

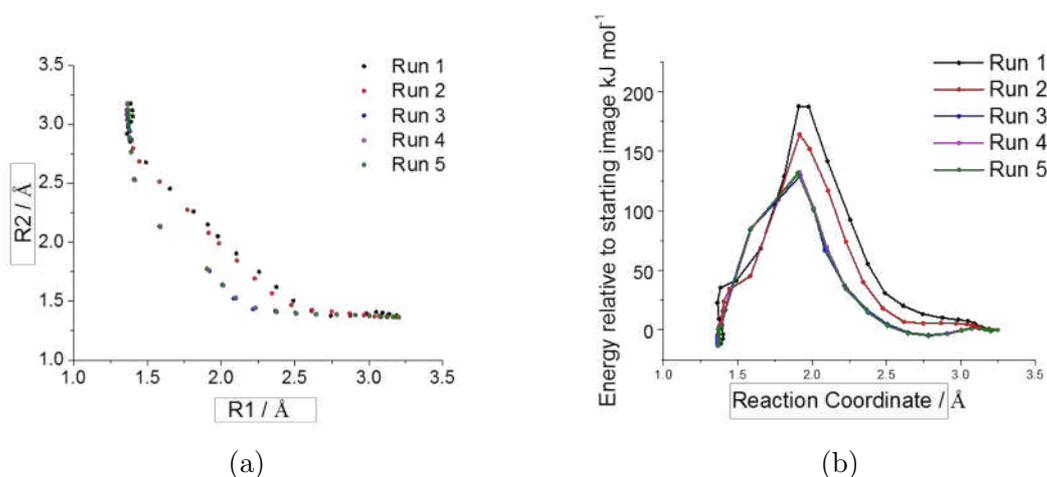


Figure 5.23: Results from NEB calculations for ring closing of hydrogenated intermediate to form gVL a) Reaction coordinates $R1$ and $R2$ (5.22) as a function of NEB run b) Energy as a function of reaction coordinate $R1$.

When studying the reaction coordinate $R1$ and $R2$ as a function of NEB run it can be seen that it clearly converges on a specific pathway after 3 three runs (figure 5.23a), and that the energy as a function of NEB run converges in a similar fashion with a calculated barrier of 130 kJ mol^{-1} . Alongside the ring closing mechanism, some NEB runs were performed to calculate the barrier for hydrogen dissociation and hydrogen migration on a number of Cu surfaces as part of some initial work.

The aim was to understand if the dissociation of hydrogen over the copper was the rate limiting step of the reaction. As NEB calculations are extremely expensive computationally the dissociation on the Cu_{75} cluster could not be performed and instead it was opted to use a Cu surface slab instead. Using a smaller system size, and removing the van-der-Waals DFT-D3 corrections (which are expected to be minimal for a H atom) greatly decreased the computational requirement. While the calculations are not directly comparable with the previous NEB calculation, they can be used to give an initial understanding on whether or not it is useful to pursue the calculations using the full scale system.

With the (110) surface being the most unlikely pathway for the dissociation of hydrogen on the cluster, it was decided that if the barrier for the dissociation was far lower than the barrier for the cyclisation of the hydrogenated intermediate, it most likely not going to be the rate limiting step. The (110) surface was cut from an optimised Cu bulk unit cell (figure 5.24) and a 15 Å vacuum gap was added to stop interactions between neighbouring surface slabs, the reaction coordinate (R1) for the NEB was taken as the distance between the two H atoms.

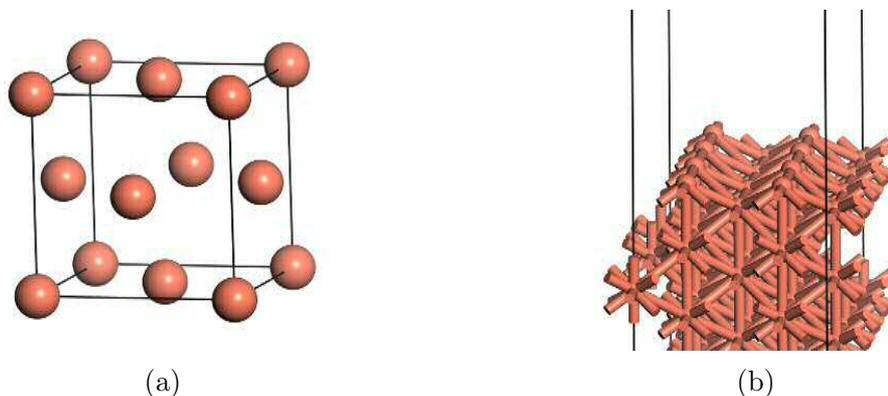


Figure 5.24: Unit cells for both the bulk and surface used in the NEB calculations
a) Cu 1x1x1 bulk unit cell b) 6 Layer (110) surface cut from bulk unit cell

Figure 5.25 shows that over the Cu (110) surface the barrier for hydrogen dissociation was calculated to be 48 kJ mol^{-1} which is much lower than the calculated barrier for the ring closing of hydrogenated intermediate (125 kJ mol^{-1}). Although this was performed over the (110) surface rather than the facet, it is a strong indi-

cator that the dissociation of hydrogen is not the rate limiting factor in the reaction over the Cu nanocluster.

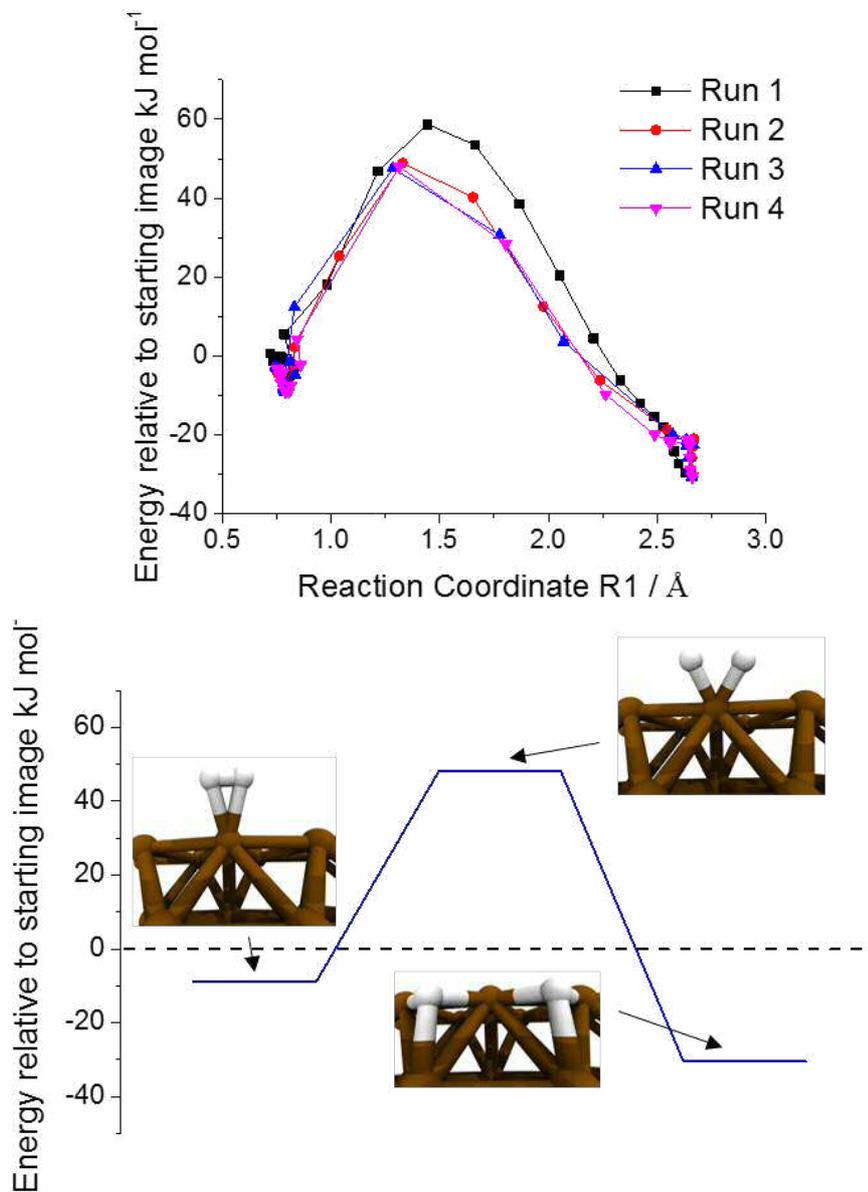


Figure 5.25: Mechanism for dissociation of hydrogen over Cu (110) surface slab

The final step is the hydrogenation of the LA / enol to form the hydrogenated intermediate. Unfortunately this mechanism would require multiple NEB calculations, as it involves multiple complex reaction coordinates, meaning it is not feasible using the standard NEB method. In order to attempt to model this type of reaction a different method of sampling transition states needs to be attempted.

5.6 Dynamic Transition State Sampling

Moving on from NEB calculations, the chosen method for sampling the transition state was umbrella sampling (see section 2.4.9). By locking the reaction into specific ranges along the reaction coordinate and then performing a sampling using MD it is hoped that the full reaction coordinate can be sampled quickly and effectively. Another advantage is that as the energy surface is sampled perpendicular to the reaction co-ordinate, an estimate of the free energy barrier is obtained, whereas in NEB there are no entropic contributions and only the energy barrier is obtained. In order to test this technique first it was decided to use a very simple reaction pathway, the transfer of hydrogen across a 1,3-pentadiene molecule (figure 5.26).

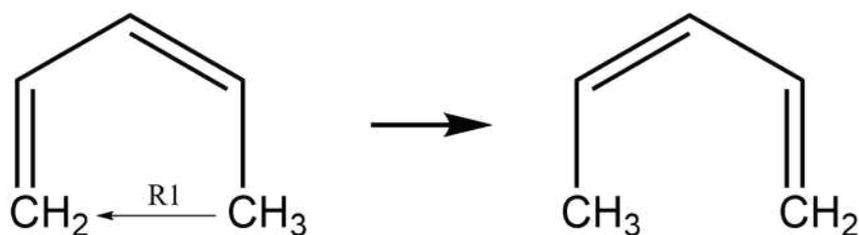


Figure 5.26: Transfer of hydrogen in 1,3-pentadiene complex with reaction coordinate labelled

The first step is to perform a *slow-growth* molecular dynamics run, during which the reaction coordinate R1 (5.26) is decreased from 3.05 to 1.10 Å. This provides an initial reaction coordinate from which to perform the calculation, similar to how `inter_vasp` interpolates between the start and end images when setting up an NEB calculation. The slow growth simulation can then be split up into separate images which act as the initial trajectory for the transition path sampling. In order to make sure the full reaction coordinate is covered, the potential barriers which keep the image locked into a specific reaction coordinate overlap (see figure 5.27).

Once the initial calculations have been set up the molecular dynamics simulations performed on each image. While the molecular dynamics simulations are considered to be less accurate at calculating energies than the standard DFT calculations that

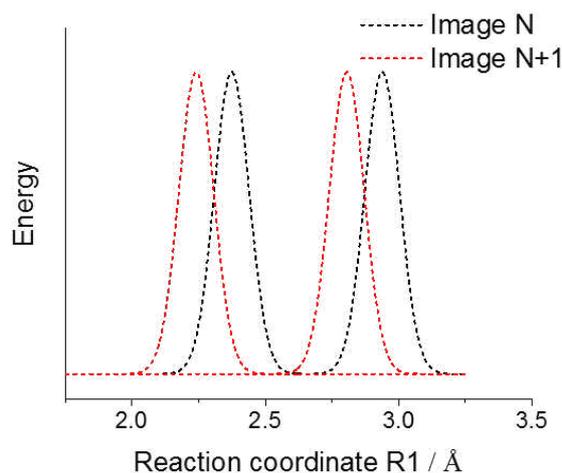


Figure 5.27: Example of potential wells for neighbouring images emphasised for clarity

are used in the NEB method, the speed at which they can sample the reaction pathway is a significant advantage.

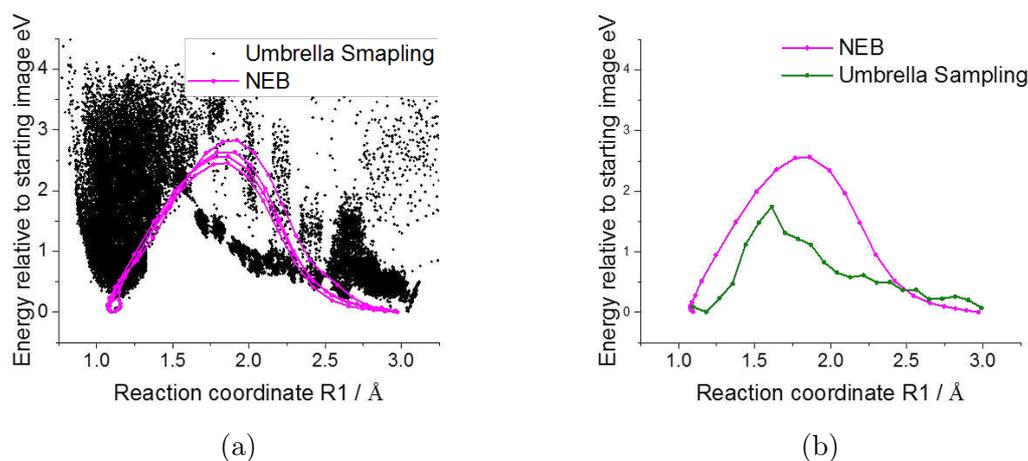


Figure 5.28: Results from Umbrella sampling and NEB methods for determining reaction pathways a) Energies from calculations b) Minimum pathways

Figure 5.28 shows the results from the Umbrella sampling overlaid on top of the results captured from the NEB calculations. As it can be clearly seen from figure 5.28a there is a drastic increase in the quantity of points in the umbrella sampling when compared to the four NEB runs, meaning much more of the reaction coordinate has been sampled. When the minimum pathway from the NEB and umbrella sampling are overlaid on top of each other 5.28b it becomes much more obvious the benefits of using umbrella sampling over the standard NEB technique,

as a lower energy pathway was found. When comparing the computational cost of the two methods, the HPC computer ARCHER uses a unit of kAUs (which are proportional to CPU hours) in order to measure the cost of running a simulation. The NEB calculations were performed using 24 nodes (1 for each image) and required 12 hours to run, and were repeated four times, giving a total cost of **414.7 kAUs**, whereas the umbrella sampling method required 24 nodes, and only 30 minutes of wall time in order to calculate the energy barrier, ending up with a total cost of **4.3 kAUs**.

The sample system of 1,3-pentadiene clearly demonstrated the benefits of using umbrella sampling over the standard NEB technique, and it was then used to model the ring closing step in the hydrogenation of LA to gVL. It was decided that the reaction coordinate that would be used in the initial slow growth simulation would be the R1 coordinate labelled in figure 5.22. As the umbrella sampling uses molecular dynamics to sample each image, it was decided that to increase the accuracy of the model, the system in which the explicit solvent is included (see section 3.4.2) was used, which meant that the effect of the solvent on the ring closing mechanism could also be included.

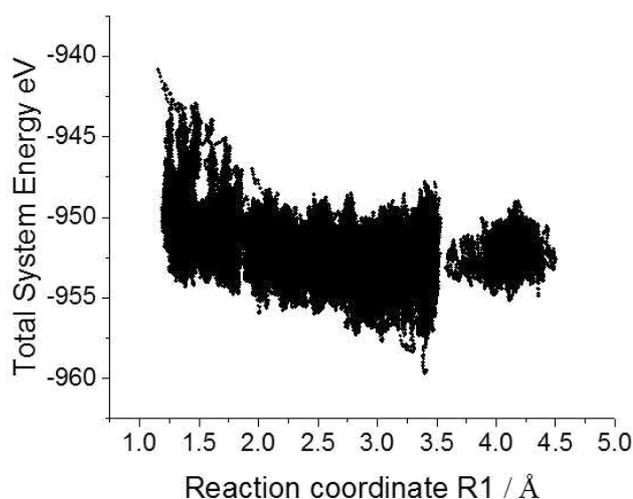


Figure 5.29: Umbrella sampling for ring closing of hydrogenated intermediate

Figure 5.29 shows the results from the umbrella sampling of the hydrogenated

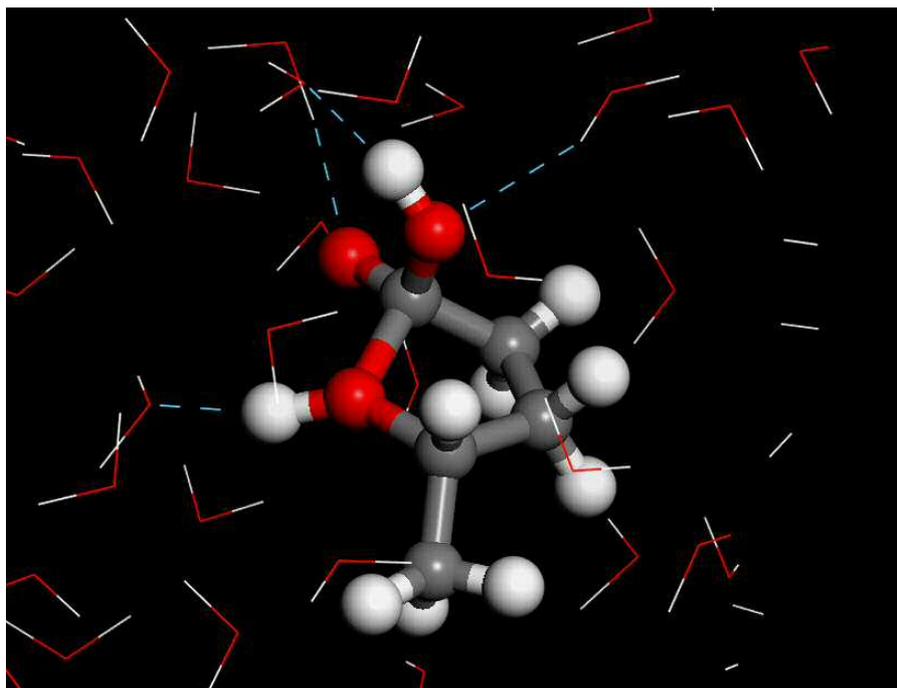


Figure 5.30: Lowest energy configuration of the reaction intermediate, with hydrogen bonding displayed in blue

intermediate. For the sampling a total of 315000 points were taken along the whole reaction coordinate R1. It was interesting to find that even with the coordinate R1 being at its minimum value (equivalent distance as seen in gVL), the OH group had not formed OH_2 and dissociated. This is a combination of the fact that the MD calculations are not performed on long enough timescales to capture dissociation, and that the complex is highly hydrogen bonded to the solvent, greatly stabilising the intermediate as seen in figure 5.30.

While in the gas phase NEB calculation, the ring closing was intramolecular, it was clearly in the umbrella sampling that this was certainly not the most likely pathway, as the molecule interacts much more strongly with the solvent than itself. The predicted mechanism most likely involved the water acting as a catalyst as seen in 5.31. This observation was important as it mimicked the experimental results, during which it was found that performing the reaction under dry methanol greatly decreased the conversion rate and selectivity of the reaction.

Overall it can be considered that NEB calculations are very useful when consid-

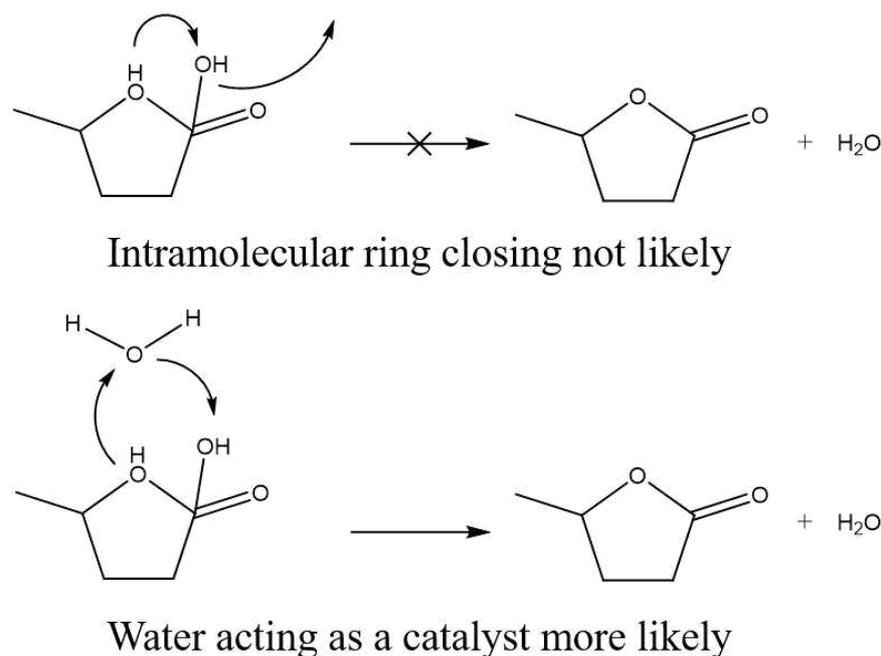


Figure 5.31: Comparison of the intramolecular and intermolecular ring closing mechanisms and the effect of the H_2O solvent.

ering barriers for small and simple reaction mechanisms, but when more the reaction mechanism is more complex, or the initial estimate for the reaction barrier is far away from the real barrier the NEB calculation becomes limited by its drastic computational cost. Umbrella sampling is one method of counteracting these issues of complexity and computational cost, but as it can be seen with the ring closing of the LA there are still some hurdles to overcome.

5.7 Summary

This final results chapter has been focused on the computational calculations performed as part of this thesis. The project was based around a Cu-ZrO_2 catalyst which had been shown to be active for the hydrogenation of Levulinic Acid. Although the catalyst was active, its role in the reaction and the structure of the material was not well understood, it was seen from PXRD experiments that after increasing the amount of copper past 20% in the sample there was no change in the

lattice constant of the material, in order to understand why there was no change seen DFT calculations were performed using CASTEP. It was found from these calculations that as the amount of copper in the bulk lattice was increased, the shift in the lattice constant was seen but the material becomes more unstable and the doping energy increased with increasing copper content. Therefore it was assumed that past 20% copper content in the material the maximum amount of copper that could be incorporated into the bulk had reached a maximum, and any other copper would form on the surface.

The next set of calculations was focused on understanding the reaction mechanism. This involved an approach during which the complexity of the model being used to simulate the reaction was increased stepwise so that it ranged from simple gas phase 0K calculations to dynamic calculations including the solvent for the reactant/product molecules. It was found for the reactant calculations that the formation of the GVL from LA was first initiated by the formation of the enol, and then it would be hydrogenated to the acid, and then finally there would be a quick cyclisation to form the GVL. To incorporate and understand the role of Cu on the surface a Cu_{75} nanoparticle was used to model the active site. Multiple combinations of sites and binding modes were tested on the Cu_{75} nanoparticle, and at first glance it seemed that the majority of the sites were able to bind the LA, GVL, and the intermediates strongly.

In order to really understand reaction kinetics, the barriers for various processes need to be calculated. This is not an arbitrary process and the current most popular method (Nudged Elastic Band) has been applied to the ring closing step of the hydrogenated intermediate, but it was found to be very computationally demanding. In order to counteract this a different technique (umbrella sampling) was attempted. It was found that the umbrella sampling decreased the computational cost by a factor of two, and provided more in-depth information on the reaction mechanism as the calculations are performed using molecular dynamics. The technique was initially

applied to the ring closing of the hydrogenated intermediate studied in the solvent, but unfortunately due to time constraints only the initial portion of this work could be completed. It is hoped that this work lays the foundation for future calculations to include dynamics into barrier calculations through umbrella sampling, rather than sticking with the standard static image techniques. This work also provides a lot of room for expansion on the understanding of the mechanism for the hydrogenation of LA to GVL. From the DFT it is found that the barrier to H_2 dissociation is small and therefore it can be suggested that the ring closing is the most difficult step in the reaction mechanism, but this does not comply with calculated experimental data with the reaction is zeroth order with concentration of LA and first order in relation to H_2 pressure. This indicates that either the supply of hydrogen into the solvent is very important, or that when the solvent is included the barrier to H_2 dissociation is largely effected.

Bibliography

- (1) S. Ishikawa, D. Jones, S. Iqbal, C. Reece and et al, *Green Chemistry*, Submitted 2016.
- (2) P. Hohenberg and W. Kohn, *Phys. Rev.*, 1964, **136**, B864–B871.
- (3) W. Kohn and L. J. Sham, *Phys. Rev.*, 1965, **140**, A1133–A1138.
- (4) G. P. Francis and M. C. Payne, *J. Phys.-Condes. Matter*, 1990, **2**, 4395–4404.
- (5) H. J. Monkhorst and J. D. Pack, *Phys. Rev. B*, 1976, **13**, 5188–5192.
- (6) M. C. Payne, M. P. Teter, D. C. Allan, T. Arias and J. D. Joannopoulos, *Rev. Mod. Phys.*, 1992, **64**, 1045–1097.
- (7) S. J. Clark, M. D. Segall, C. J. Pickard, P. J. Hasnip, M. J. Probert, K. Refson and M. Payne, *Z. Kristall.*, 2005, **220**, 567–570.
- (8) B. G. Pfrommer, M. Cote, S. G. Louie and M. L. Cohen, *J. Comput. Phys.*, 1997, **131**, 233–240.
- (9) S. Grimme, *J. Comput. Chem.*, 2006, **27**, 1787.
- (10) E. McNellis, J. Meyer and K. Reuter, *Phys. Rev. B*, 2009, **80**, 205414.
- (11) G. Bergerhoff, R. Hundt, R. Sievers and I. Brown, *Journal of Chemical Information and Computer Sciences*, 1983, **23**, 66–69.
- (12) D. R. Jones, S. Iqbal, S. Ishikawa, C. Reece, L. M. Thomas, P. J. Miedziak, D. J. Morgan, J. K. Edwards, J. K. Bartley, D. J. Willock et al., *Catalysis Science & Technology*, 2016.

- (13) J. P. Doye and D. J. Wales, *New journal of chemistry*, 1998, **22**, 733–744.
- (14) E. L. Jeffery, R. K. Mann, G. J. Hutchings, S. H. Taylor and D. J. Willock, *Catalysis today*, 2005, **105**, 85–92.
- (15) E. Schwegler, J. C. Grossman, F. Gygi and G. Galli, *The Journal of chemical physics*, 2004, **121**, 5400–5409.
- (16) S. Grimme, J. Antony, S. Ehrlich and H. Krieg, *The Journal of chemical physics*, 2010, **132**, 154104.
- (17) I.-C. Lin, A. P. Seitsonen, M. D. Coutinho-Neto, I. Tavernelli and U. Rothlisberger, *The Journal of Physical Chemistry B*, 2009, **113**, 1127–1131.
- (18) W. Humphrey, A. Dalke and K. Schulten, *Journal of Molecular Graphics*, 1996, **14**, 33–38.
- (19) D. Sheppard and G. Henkelman, *Journal of computational chemistry*, 2011, **32**, 1769–1771.
- (20) D. Sheppard, R. Terrell and G. Henkelman, *The Journal of chemical physics*, 2008, **128**, 134106.
- (21) G. Henkelman and H. Jonsson, *The Journal of chemical physics*, 2000, **113**, 9978–9985.
- (22) G. Henkelman, B. P. Uberuaga and H. Jonsson, *The Journal of chemical physics*, 2000, **113**, 9901–9904.

6 | General Conclusions

The work encompassing this thesis has gone in to detail about some of the various techniques available for understanding kinetics in heterogeneous catalysis: Temperature Programmed Reduction (TPR), Temporal Analysis of Products (TAP), and molecular simulations using classical dynamics and Density Functional Theory (DFT). The new methodologies developed as part of this work for TAP and TPR take the current experimental responses and greatly increase the quantity and quality of calculable kinetic information, through either deconvolution of kinetic parameters (TPR), or via noise filtering (TAP). For the molecular simulations the current methodologies were not expanded upon but were applied in a way that the model system used was of a higher complexity (dynamic solvent simulations), therefore providing a more accurate representation of the kinetics. TAP, TPR, and molecular simulations can all be focused on materials used in gas phase heterogeneous catalysis. TPR provides information on in the kinetics of how the catalytic material is reduced (and how it behaves under reducing conditions), TAP is much more strongly focused on the kinetics of the surface reaction occurring, and finally molecular simulations can be applied to both.

The TPR analysis provides a breakdown on how a material is reduced, this has a direct application in materials which are used as oxidisers such as the CeO_2 example outlined in section 4.6, but can also be used to understand how a material will act under catalytic conditions. The deconvolution technique developed as part of this thesis (see section 4.4) means that it is now possible to accurately measure

kinetic parameters (activation energies, Arrhenius pre-exponentials) and, in some cases, even predict a specific reduction model (e.g. 2D diffusional) for each process. From the application of the deconvolution method to the CeO₂ material it was found that there were three different reduction processes in the thermogram, and they were corresponded to three different phases of the material. From the study of the kinetic parameters it was hypothesised that there was a phase change in the material when the calcination temperature was changed from 400°C to 500°C. TPR is a common technique in catalysis, which alone has its strengths and drawbacks. The main strength being that the deconvolution procedure will be applicable to a large proportion of the research community, but the main drawback being that as the technique is standardised the general communities knowledge of non-isothermal kinetic analysis is limited. This main drawback is why the development of analysis software has been one of the major focuses in this work. For both TPR and TAP an analysis tool, in the form of a graphical user interface has been developed. This means that anyone with minimal prior training can perform a kinetic analysis on their experimental data, therefore having this software freely available greatly increases the impact of the analysis techniques used in both TPR and TAP.

The TAP reactor, which is focused on the kinetics of the surface reactions, is much more directly applicable to catalysis than TPR. By attempting to bridge the gap between ultra high vacuum (UHV) single crystal experiments, and in-situ studies performed in flow/batch reactors, the TAP combines the accuracy of UHV studies with the more a closer analogue of the *real* system. The issue with the TAP reactor is that while the mathematics of the transport through the reactor, and therefore how the apparent kinetics are influenced by transport effects, are well established there is a hard limit on the level of kinetic information that can be accurately gathered from current techniques. In section 2.3.8 the idea of a basic kinetic coefficient is established, where these three coefficients provide different kinetic information about a surface reaction (i.e. r_0 provides an apparent rate constant, r_1 relates gas substances with preceding intermediates, and r_2 is an apparent time delay for a

process on a material) and are inherently model free, but if they are combined together they can be applied to a specific reaction model to calculate rate constants. Unfortunately due to the characteristic nature of mass spectrometers there is always noise present in TAP response curves, meaning it is not possible to accurately calculate the r_1 and r_2 coefficients. A new methodology, *Laplace Fourier Filtering* (LFF), for filtering data has been developed as part of this thesis 3.2.4 which removes the effect of noise in the response curves, and for the first time the r_1 and r_2 coefficients and subsequently (with an applied reaction model) the rate constants for each elementary step in the reaction can be accurately calculate from experimental responses (section 3.2.5). Only the model free coefficients were calculated as part of this work (3.2.7) for a (4 Pt–Mo₂C) system in which CO is oxidised to CO₂ via the Boudard reaction, as its complex nature meant a model could not be easily applied, but the model free coefficients provide more information on the nature of the reaction than the standard unfiltered data alone. On its own this level of analysis is highly sought after in kinetics but where the TAP really excels is that fact that this is combined with the state defining nature of the TAP pulse (see section 1.2). As the pulse is state defining, it can be considered a snapshot of the catalyst at some specific state, but upon repeated pulsing the state of the catalyst can be changed. This means that for every state of the catalyst, a full kinetic analysis can be performed (calculation of kinetic coefficients / rate constants), and therefore it is now possible to understand how the kinetics of a reaction evolve with a catalytic system, providing unprecedented levels of kinetic information. TAP is not without its shortcomings as it is performed under UHV conditions and is a transient technique, it cannot be said to fully replicate the reaction conditions (high pressure, steady state) often seen in batch/flow type reactors, but with the aid of the LFF algorithm it is hoped that the large amount of transient kinetics now available can be used to understand these batch/flow type systems.

For an understanding of a full catalytic system molecular simulations provide an avenue for understanding the catalyst material alongside the kinetics of the surface

reactions. This is often used when other techniques would fall short (TAP and TPR), but they do not come without their own difficulties. For understanding a catalytic material simple DFT calculations of a bulk solid are very commonplace in the literature, and have been applied here to calculate why there is an upper limit of how much copper can be incorporated into a ZrO_2 lattice (section 5.2), but the more interesting aspect of the molecular simulations is when they're applied to a specific reaction. For this particular work the conversion of Levulinic Acid (LA) to gamma-Valerolactone (GVL) over a Cu-ZrO_2 catalyst was taken as the system of interest, in which the active site of the reaction was considered to be Cu nanoparticles. The idea was to gradually increase the complexity of the model system in order to save computation time and increase the understanding of the system allowing implausible reaction mechanisms to be ignored. The first step was to start with a simple analysis during which the energetics of the reaction were calculated on their own in the gas phase (section 5.3) and then subsequently in the liquid phase using classical and ab-initio dynamics of solvated molecules. The next step was to revert to the gas phase (to save computation time) and include the effect of the Cu nanoparticle by studying how the molecules would bind to the surface of the cluster (section 5.4). On their own these kind of studies give information on whether or not a reaction is thermodynamically possible, information which is of use regarding catalysis, but in order to understand the real kinetics of a reaction the barriers to various processes need to be calculated. In molecular simulations calculating barriers is not an elementary concept (see section 2.4.9), it involves simulating the potential energy surface in order to search for a transition state, and then from the energy of the transition state it is possible to calculate an activation energy barrier. For a reaction with a well defined reaction coordinate this, while not trivial, is possible using standard techniques for example Nudged Elastic Band (NEB). This was performed for the ring closing of the hydrogenated intermediate (considered to be a rate limiting step in the conversion of LA to GVL from DFT, see section 5.5). This sampling of the potential energy surface for the transition state is

very expensive computationally, even for a simple gas phase ring closing consisting on the order of a dozen atoms. Therefore it greatly limits its application to systems such as the conversion of LA to GVL which are performed in the liquid phase. One idea was to perform dynamic transition state sampling 5.6 which combines dynamic simulations with transition state search by performing a dynamics constrained at each point along the reaction coordinate. This greatly increases the sampling of the potential energy surface. This technique was performed on the LA to GVL system but unfortunately due to lack of time and computational resources only initial testing could be performed. The preliminary calculations showed that the solvent interacts very strongly with the hydrogenated intermediate, by studying how the solvent interacts with the hydrogenated intermediate throughout the ring closing mechanism it was hypothesised that the ring closing is *intermolecular* rather than *intramolecular*. This could explain experimental data that showed when the reaction is performed in a dry solvent, there was a large reduction in conversion of the LA. This interaction would have not been possible to model using the standard NEB technique due to the large number of degrees of freedom present when the solvent is included, therefore for these systems the dynamics transition state sampling presented in this thesis can be considered important.

Overall these three methods may seem to be separate entities in the field of catalysis each with their own specific goals, but this work has attempted to link the three together under the general denomination of kinetics. Understanding the kinetics of a system is paramount when it comes to designing a better catalyst. Using the methodologies outlined in this thesis the potential for calculable kinetic information has been demonstrated, which upon their application it is hoped the field of catalysis can be advanced towards its final goal of developing an highly active and selective catalyst.

6.1 Future Work

While this work has gone into depth about various techniques and methods for performing kinetic analysis, there are three main areas that could benefit from some expansion. The first of which is the LFF algorithm itself. While it has been shown that the algorithm can filter curves to a high degree, allowing for the higher order moments to be calculated, it was found that it was very computationally expensive. Therefore the main focus should be on refining the variable space used during the regression of the Fourier domain curves, which would cause a significant increase in speed allowing for the filtering of large data sets, instead of simply taking a selection of curves. The second area of expansion would be in the TPR deconvolution procedure. While the deconvolution procedure has been shown to be accurate, the software which has been developed to aid the application of this procedure is still not in a state where it could be publicly released. While this is a factor unrelated to the mathematics or chemistry involved in the procedure, it is still vital overall as the model will be of no use if it cannot be applied with ease. The final area would be the dynamic transition state sampling of LA to GVL. Only initial data could be gathered in the time frame of this thesis but the technique showed promising results.

There is a hypothetical experiment which could combine the three methods outlined in this thesis to provide in-depth kinetic analysis of a material. An example would be studying a material (*e.g.* CeO_2) which can perform a simple oxidation reaction (*e.g.* CO oxidising to form CO_2). Assuming that the material does not reduce uniformly in a single process, a deconvolution of a TPR thermogram can give information on the various reduction pathways on the material, and some information on the kinetics (activation energies) for each of these processes. Then using a TAP reactor it would be possible to linearly heat the sample (CeO_2) while repeatedly pulsing a reducing gas (CO). Using the LFF algorithm the curves could be cleaned and accurate measurements of the r_0 , r_1 , and r_2 taken, and with application

of some specific model (*e.g.* reversible adsorption / reaction) rate constants can be calculated. If one was to plot these rate constants as a function of temperature, it would be hoped that they would coincide with various reduction processes found in the deconvolution procedure, demonstrating a link between TPR and TAP. Finally DFT can be used to model a number of potential surfaces of the material and the energetics of reduction can be roughly calculated by simply removing oxygen atoms from the surface, these energetics can then be related to the various TPR reduction processes by studying the order in which the surfaces are more likely to reduce. Then using the dynamic transition state sampling one could calculate barriers for the various reaction steps that are happening in the TAP reactor. As the TAP reactor is performed under UHV conditions, the difficulty that arrives with simulating a solvent is completely removed meaning it should be simpler than the case discussed in this thesis. It is then possible to see if the simulated reaction has similar kinetics to the ones calculated from the TAP experiment. By combining these three techniques, the information available becomes far greater than the sum of its individual components. By relating the kinetics of the surface reduction calculated from TPR to kinetics of the reaction calculated from TAP the understanding of the reaction has been increased greatly, but then using theory to recreate the kinetics it is possible to visualise and understand reactions on a molecular level. Separately these three techniques can be considered powerful tools, but when they are combined the level of kinetic information is far greater than the sum of their individual parts, and it is this in-depth information that will allow the creation of better catalytic materials.

7 | Appendix

Process	Mechanism	Apparent kinetic coefficient from r_0	apparent "intermediate-gas" coefficient from $ r_1 $	Apparent time delay from $ r_2/r_1 $
Irreversible adsorption + irreversible reaction	$A \xrightarrow{k_a} AZ$ $AZ \xrightarrow{k_r} B$	k_a	0; (A) $\frac{k_a}{k_r}$ (B)	$\frac{1}{k_r}$ (AZ)
Reversible adsorption + irreversible reaction	$A \xrightleftharpoons[k_d]{k_a} AZ$ $AZ \xrightarrow{k_r} B$	$\frac{k_a k_r}{k_d + k_r}$	$\frac{k_a k_d}{(k_d + k_r)^2}$ (A) $\frac{k_a k_r}{(k_d + k_r)^2}$ (B)	$\frac{1}{k_d + k_r}$ (AZ)
Reversible adsorption + irreversible reaction with a buffer-step	$A \xrightleftharpoons[k_d]{k_a} AZ$ $AZ \xrightleftharpoons[k_2^-]{k_2^+} CZ$ $AZ \xrightarrow{k_r} B$	$\frac{k_a k_r}{k_d + k_r}$	$\frac{k_a k_d}{(k_d + k_r)^2} \left(\frac{k_2^+}{k_2^-} + 1 \right)$ (A) $\frac{k_a k_r}{(k_d + k_r)^2} \left(\frac{k_2^+}{k_2^-} + 1 \right)$ (B)	$\frac{1}{k_d + k_r} \frac{(k_2^+ + k_2^-)^2 + (k_2^-)^2}{k_2^- (k_2^+ + k_2^-)} + \frac{k_2^+}{k_2^- (k_2^+ + k_2^-)}$ (AZ)
Reversible adsorption + two irreversible consecutive reactions	$A \xrightleftharpoons[k_d]{k_a} AZ$ $AZ \xrightarrow{k_{rB}} CZ + B$ $CZ \xrightarrow{k_{rC}} C$	$\frac{k_a k_{rB}}{k_d + k_{rB}}$	$\frac{k_a k_d}{(k_d + k_{rB})^2}$ (A) $\frac{k_a k_{rB}}{(k_d + k_{rB})^2}$ (B)	$\frac{1}{k_d + k_{rB}}$ (AZ) $\frac{1}{k_d + k_{rB}} + \frac{1}{k_{rC}} - \frac{1}{k_d + k_{rB} + k_{rC}}$ (CZ)

			$\frac{k_a k_{rB}}{(k_d + k_{rB})^2} \frac{k_d + k_{rB} + k_{rC}}{k_{rC}}$ (C)	
Reversible adsorption + two parallel irreversible reactions	$A \xrightleftharpoons[k_d]{k_a} AZ$ $AZ \xrightarrow{k_{rB}} B$ $AZ \xrightarrow{k_{rC}} C$	$\frac{k_a (k_{rB} + k_{rC})}{k_d + k_{rB} + k_{rC}}$ (A \rightarrow B, C) $\frac{k_a k_{rB}}{k_d + k_{rB} + k_{rC}}$ (A \rightarrow B) $\frac{k_a k_{rC}}{k_d + k_{rB} + k_{rC}}$ (A \rightarrow C)	$\frac{k_a k_d}{(k_d + k_{rB} + k_{rC})^2}$ (A) $\frac{k_a k_{rB}}{(k_d + k_{rB} + k_{rC})^2}$ (B) $\frac{k_a k_{rC}}{(k_d + k_{rB} + k_{rC})^2}$ (C)	$\frac{1}{k_d + k_{rB} + k_{rC}}$ (AZ)
Reversible adsorption	$A \xrightleftharpoons[k_d]{k_a} AZ$	0	$\frac{k_a}{k_d}$ (A)	$\frac{1}{k_d}$ (AZ)
Pseudo-equilibrium model of microporous diffusion	Equilibrium adsorption into the pore entry + Fickian diffusion through the pores	0	$(1 - \varepsilon_{cat}) \delta K_{p,ent}$; δ is the porosity, $K_{p,ent}$ is an equilibrium coefficient of adsorption into the pore entry	$\frac{\tau_p}{3}$, τ_p is the residence time in pores: $\tau_p = \frac{L_p^2}{D_p}$
Adsorption model of microporous diffusion	$A \xrightleftharpoons[k_d]{k_a} AZ$ $AZ \xrightleftharpoons[k_2^-]{k_2^+} Z + A_{pore}$ + Fickian diffusion	0	$(1 - \varepsilon_{cat}) \frac{k_a}{k_d} \left(1 + \frac{k_2^+}{k_2^-} \delta \right)$	$\frac{\tau_p + \delta}{3} \frac{k_2^-}{k_2^+} + \frac{1}{k_d} \left(1 + \frac{k_2^+}{k_2^-} \delta \right)$ $1 + \frac{k_2^-}{\delta k_2^+}$

Model of gas mesoporous diffusion	Fickian diffusion through the pores in the porous gas-like phase + adsorption on the pore surface	0	$(1 - \varepsilon_{\text{cat}}) \delta(1 + K_{\text{eq}})$ K_{eq} is an equilibrium coefficient of adsorption	$\tau_{\text{p,app}} = \frac{1}{3} \frac{K_{\text{eq}}}{k_d(1 + K_{\text{eq}})}$ $\tau_{\text{p,app}}$ is an apparent residence time in the pore: $\tau_{\text{p,app}} = \frac{L_p^2(1 + K_{\text{eq}})}{D_p}$
Model of gas and surface mesoporous diffusion. Diffusion limitation.	Fickian diffusion through the pores in the porous gas-like and surface phases + adsorption-desorption on the pore walls	0	$(1 - \varepsilon_{\text{cat}}) \delta(1 + K_{\text{eq}})$ K_{eq} is an equilibrium coefficient of adsorption	$\tau_{\text{p,app}} = \frac{1}{3}$ $\tau_{\text{p,app}}$ is an apparent residence time in the pore: $\tau_{\text{p,app}} = \frac{L_p^2(1 + K_{\text{eq}})}{D_p + K_{\text{eq}}D_s}$

Figure 7.1: Three coefficient solutions for various reaction mechanisms

```

SYSTEM NEB_run cluster
ISTART = 1      # 0: New job, 1: Continuation job
PREC = Accurate
GGA = PE       # Use the PBE functional 91 is PW91
ENCUT = 500    # Set planewave cutoff by hand
ICHARG = 2     # Initiate charge as superposition of atomic charges

#D3 corrections
IVDW = 11     # Turns on D3
VDW_S6 = 1.0  # S6 parameter
VDW_SR = 1.217 # SR parameter
VDW_S8 = 0.722 # S8 parameter
VDW_RADIUS = 15.0 # Cutoff radius for pair interactions
VDW_CNRRADIUS = 10.0 # Cutoff radius for coordination number

#Relaxation parameters
NSW = 10
NELM = 50
ISPIN = 2
IBRION = 3     # Turn off VASP optimisers
#EDIFF = 1E-4  # Electronic relaxation threshold, 1E-4 is default
#EDIFFG = -1.5E-2 # Electronic relaxation threshold, 1E-3 is default
ISIF = 2      # Calculate atom forces and relax but not shape/size cell

# Settings for speed
ALGO = Fast    # RMM-DIIS algorithm for electrons
LREAL = .FALSE. # evaluate projection operators in real space
NSIM = 4      # blocked algorithm update, four bands at a time
ISYM = 0     # Turns off symmetry

# Recommended option for preventing force noise
ADDGRID = .TRUE.

# NEB Calculation
IMAGES=10
ICHAIN=0
SPRINT=-5.0 # Spring constant used for nudging images
LCLIMB=.FALSE.

```

```
# Optimiser QM
IOPT=3 # Set to QM optimiser
MAXMOVE=0.2
TIMESTEP=0.1
```

Listing 7.1: INCAR file for performing NEB calculations with QM optimiser

```
SYSTEM NEB_run cluster
ISTART = 1 # 0: New job, 1: Continuation job
PREC = Accurate
GGA = PE # Use the PBE functional 91 is PW91
ENCUT = 500 # Set planewave cutoff by hand
ICHARG = 2 # Initiate charge as superposition of atomic charges

#D3 corrections
IVDW = 11 # Turns on D3
VDW_S6 = 1.0 # S6 parameter
VDW_SR = 1.217 # SR parameter
VDW_S8 = 0.722 # S8 parameter
VDW_RADIUS = 15.0 # Cutoff radius for pair interactions
VDW_CNRRADIUS = 10.0 # Cutoff radius for coordination number

#Relaxation parameters
NSW = 10
NELM = 50
ISPIN = 2
IBRION = 3 # Turn off VASP optimisers
#EDIFF = 1E-4 # Electronic relaxation threshold, 1E-4 is default
#EDIFFG = -1.5E-2 # Electronic relaxation threshold, 1E-3 is default
ISIF = 2 # Calculate atom forces and relax but not shape/size cell

# Settings for speed
ALGO = Fast # RMM-DIIS algorithm for electrons
LREAL = .FALSE. # evaluate projection operators in real space
NSIM = 4 # blocked algorithm update, four bands at a time
ISYM = 0 # Turns off symmetry

# Recommended option for preventing force noise
ADDGRID = .TRUE.

# NEB Calculation
IMAGES=10
ICHAIN=0
SPRINT=-5.0 # Spring constant used for nudging images
LCLIMB=.FALSE.

# Optimiser LBFGS
IOPT=1 # Set to LBFGS optimiser
MAXMOVE=0.2
ILBFGSMEM=20 # Number of steps saved when building hessian matrix
LGLOBAL=.TRUE. # Optimise globally using all images
LAUTOSCALE=.TRUE.
INVCURV=0.01
```

```
LLINEOPT=.FALSE.
FDSTEP=5E-3
```

Listing 7.2: INCAR file for performing NEB calculations with LBFGS optimiser

```
SYSTEM NEB_run cluster
ISTART = 1      # 0: New job , 1: Continuation job
PREC = Accurate
GGA = PE       # Use the PBE functional 91 is PW91
ENCUT = 500    # Set planewave cutoff by hand
ICHARG = 2     # Initiate charge as superposition of atomic charges

#D3 corrections
IVDW = 11     # Turns on D3
VDW_S6 = 1.0  # S6 parameter
VDW_SR = 1.217 # SR parameter
VDW_S8 = 0.722 # S8 parameter
VDW_RADIUS = 15.0 # Cutoff radius for pair interactions
VDW_CNRRADIUS = 10.0 # Cutoff radius for coordination number

#Relaxation parameters
NSW = 10
NELM = 50
ISPIN = 2
IBRION = 3     # Turn off VASP optimisers
#EDIFF = 1E-4  # Electronic relaxation threshold, 1E-4 is default
#EDIFFG = -1.5E-2 # Electronic relaxation threshold, 1E-3 is default
ISIF = 2      # Calculate atom forces and relax but not shape/size cell

# Settings for speed
ALGO = Fast    # RMM-DIIS algorithm for electrons
LREAL = .FALSE. # evaluate projection operators in real space
NSIM = 4       # blocked algorithm update, four bands at a time
ISYM = 0       # Turns off symmetry

# Recommended option for preventing force noise
ADDGRID = .TRUE.

# NEB Calculation
IMAGES=10
ICHAIN=0
SPRINT=-5.0 # Spring constant used for nudging images
LCLIMB=.FALSE.

# Optimiser CG
IOPT=2 # Set to CG optimiser
MAXMOVE=0.2
FDSTEP=5E-3
```

Listing 7.3: INCAR file for performing NEB calculations with CG optimiser

```
CONFIG file generated for Q4ethanol
      2      3      252  -197582.203976
12.400000000000  0.000000000000  0.000000000000
```

```

-0.000000001000      12.400000000000      0.000000000000
-0.000000001000      -0.000000001000      12.400000000000
CT      1
  2.528420076      -4.750736068      -1.441954980
  8.43930608358      0.788814465446      -8.36339874372
 -7315.64742770      -1815.60864539      1652.30806835
CM      2
  3.278372728      -5.162288474      -2.718315305
  7.92922422675      -2.14136936080      -2.32329216856
 -13906.4392845      -20987.2016358      47292.9254372
CM      3
  3.259559424      -4.715412736      -3.895723670
  5.34017216013      2.45500803565      -6.30006189642
 -4189.06476814      12775.9095533      -61636.1739850
CT      4
  3.948091106      -5.309191490      -5.208454062
  5.42577932121      -0.277216164941      -2.53990775610
 15572.4835459      14454.5127871      30744.3856327
C      5
  3.638344620      -4.849083667      5.793965232
 -1.54566544712      2.00628722019      -3.48276175468
 -16259.3269200      -1270.39038245      3152.60510704
OHE      6
  3.874810292      6.139264371      -2.335160415
 -1.77580554207      -2.02680902096      -0.228318152102
 19227.7031051      -15257.8044121      -13462.9269453
O      7
  4.429692887      -4.888960463      4.877817271
 -9.48210152001      1.49673507122      2.13146411304
 2296.04265841      3945.23039799      -8309.07736312
...

```

Listing 7.4: Snippet from CONFIG file for NPT DL_POLY run

```

Generated by DL_FIELD v3.40
Units kcal/mol
Molecular types 2
Molecule name not_define
nummols 1
atoms 16
CT      12.01150      -0.18000      1      0
CM      12.01150      0.04880      1      0
...
bonds 15
harm    1      2      634.00      1.51000
harm    1      9      680.00      1.09000
...
angles 23
harm    2      1      9      70.00000      109.50000
harm    2      1      10     70.00000      109.50000
harm    2      1      16     70.00000      109.50000
harm    9      1      10     66.00000      107.80000
...

```

```

dihedral 29
cos3      9      1      2      3      0.00000      0.00000      0.00000      0.50000      0.50000
cos3      9      1      2      6      0.00000      0.00000      0.00000      0.50000      0.50000
...
finish
Molecule name TIP4
nummols 59
atoms 4
OT4      15.99940      0.00000      1      0
HT4      1.00797      0.52000      1      0
HT4      1.00797      0.52000      1      0
Q4       0.00000      -1.04000      1      0
rigid 1
4 1 2 3 4
finish
vdw 78
CT      CT      lj      0.0660      3.5000
CM      CT      lj      0.0708      3.5249
...

```

Listing 7.5: Snippet from FIELD file for NPT DL_POLY run

```

SPC model water in cubic cell

temperature      473
pressure         0.03454
ensemble npt berendsen 0.5 1.5

steps           20500000
equilibration   50000
multiple        1
scale           20
print           2000
stack           1000
stats           1000

timestep        5.0000E-04
primary         6.2000E+00
cutoff          4.0000E+00
delr            1.2000E+00
rvdw            4.0000E+00

trajectory nstraj= 1      istraj= 5000 keytrj=0

reaction field
eps constant    7.0000E+01
cap             1.0000E+04
shake tolerance 1.0000E-05
quaternion tolerance 1.0000E-05

print rdf

job time        9.0000E+05
close time      2.0000E+01

```

```
finish
```

Listing 7.6: CONTROL file for NPT DL_POLY run

```

1 function AllPeak_Callback(hObject, eventdata, handles)
2 %Asks which data you want to see 3D or 2D
3 prompt = {'Please enter 1 for 2D or 2 for 3D viewer:'};
4 dlg_title = 'Input initial parameters for simulation';
5 num_lines = 1;
6 def = {'1'};
7 answer = inputdlg(prompt,dlg_title,num_lines,def);
8 pans = str2num(answer{1,1});
9
10 k = findall(gca, 'type', 'line');
11 delete(k);
12
13 sStructData = handles.TAPdata;
14 averagepeak = mean(sStructData.peak,2);
15 timesingle = linspace(0,sStructData.tpulse,length(averagepeak));
16
17 if pans == 1
18 plot(timesingle,sStructData.peak)
19 xlabel('Time S')
20 ylabel('Intesity Chromatogram (a.u)')
21 grid on
22 hold on
23
24 elseif pans == 2;
25 peak = sStructData.peak;
26 peak = peak';
27 [m,n] = size(peak)
28 xMat = repmat(timesingle',1,m);
29 hold off
30 Y = 1:m;
31 yMat = repmat(Y,numel(timesingle),1);
32 size(xMat)
33 size(yMat)
34 plot3(xMat,yMat,peak)
35 ylabel('Number of Pulses')
36 xlabel('Intesity Chromatogram (a.u)')
37 xlabel('Time S')
38 view(26, 42);
39
40 legend('off');
41 end
42
43 % Update handles structure
44 guidata(hObject, handles);

```

Listing 7.7: Example script used in plotting TAP experimental response data using a control object

```
1 %% Latest version of LFF method
```

```

2 % p = TAP experimental response
3 % tnew = TAP time response
4 % type = Which type of curve to fit (product or reactant)
5
6 % Lets do some curve fitting
7 tlen = length(p); % Length of signal
8 tmax = max(tnew);
9 Fs = tlen/max(tnew); % Sampling Frequency
10 T = 1/Fs; % Sampling period
11
12 Y = fft(p)/(tlen/tmax);
13 Y = fftshift(Y);
14 Y(1+end) = Y(1);
15 Y(1) = [];
16 Yscale = real(Y(end/2-1));
17 Y = Y/Yscale;
18
19 Yr = real(Y);
20 Yi = imag(Y);
21
22 k = linspace(-tlen/2+1,tlen/2,tlen);
23 k(k==0) = 1E-10;
24 s = (2*k*pi/(tmax))*1i;
25
26 % Sets up which part of the curve to fit to
27 loc = find(k > -500 & k < -0);
28 loc(loc==tlen/2) = [];
29
30 % Removes outliers
31 kloc = abs(diff(diff(real(Y(loc)))));
32 i = 1;
33 while i < 15
34     ktmp = find(kloc==max(kloc));
35     if ktmp > 450
36         kloc(kloc==max((kloc))) = 0;
37     else
38         i = i+1;
39         loc(find(loc==loc((kloc==max((kloc)))))+1) = [];
40         kloc(kloc==max((kloc))) = [];
41     end
42 end
43
44 % Scaling parameter to make begining of curve more important
45 scaling = linspace(5,10,length(loc));
46 scaling(end-15:end) = linspace(scaling(end-15),max(scaling)*3,16);
47
48
49 if type == 1
50     tin = rand();
51     tcat = rand();
52     a = rand()-0.5;
53     b = rand()-0.5;
54     c = rand()-0.5;

```

```

55     d = rand() - 0.5;
56
57     Ff = zeros(1, length(p)/2);
58
59
60     vars = [tin, tcat, a, b, c, d];
61     LB = [0, 1, -10, -10, -10, -10];
62     UB = [2, 1, 10, 10, 10, 10];
63 else
64     tin = rand();
65     tcat = rand();
66     a1 = rand() - 0.5;
67     b1 = rand() - 0.5;
68     c1 = rand() - 0.5;
69     d1 = rand() - 0.5;
70     a2 = rand() - 0.5;
71     b2 = rand() - 0.5;
72     c2 = rand() - 0.5;
73     d2 = rand() - 0.5;
74     D = rand();
75     Ff = zeros(1, length(p)/2);
76
77
78     vars = [tin, tcat, a1, b1, c1, d1, a2, b2, c2, d2, D];
79     LB = [0, 1, -10, -10, -10, -10, -10, -10, -10, -10, 0];
80     UB = [2, 1, 10, 10, 10, 10, 10, 10, 10, 10, 10];
81 end
82
83
84 problem = createOptimProblem('fmincon', ...
85     'objective', @(vars) DataCleanSum(vars, Y, s, Ff, loc, scaling, type), ...
86     'x0', vars, 'lb', LB, 'ub', UB, 'Aeq', [], 'Aineq', [], 'beq', [], 'bineq', [], ...
87     'options', ...
88     optimoptions(@fmincon, 'MaxIter', 5000));
89
90 gs = GlobalSearch('Display', 'iter', 'NumTrialPoints', 1000);
91 [variables, ~] = run(gs, problem);
92
93 vars = variables
94
95 if type == 1
96     tin = vars(1);
97     tcat = vars(2);
98     a = vars(3)*100;
99     b = vars(4)*100;
100    c = vars(5)*100;
101    d = vars(6)*100;
102 else
103     tin = vars(1);
104     tcat = vars(2);
105     a1 = vars(3)*100;
106     b1 = vars(4)*100;
107     c1 = vars(5)*100;

```

```

108     d1 = vars(6)*100;
109     a2 = vars(7)*100;
110     b2 = vars(8)*100;
111     c2 = vars(9)*100;
112     d2 = vars(10)*100;
113     D = vars(11);
114 end
115
116 for i = 1:length(s)
117     if type == 2
118         Ff(i) = (cosh(sqrt(s(i)*tin*(0.5)*D))*...
119             sinh(sqrt(s(i)*tin/2))*...
120             (sqrt(2)*(...
121             ((a2+b2*s(i))./(c2+d2*s(i)))...
122             )*tcat/sqrt(s(i)*tin)))/...
123             ((cosh(sqrt(2*s(i)*tin))+...
124             ((...
125             ((a1+b1*s(i))./(c1+d1*s(i)))...
126             )*tcat/sqrt(2*s(i)*tin))*...
127             sinh(sqrt(2*s(i)*tin))*cosh(sqrt(2*s(i)*tin*D)));
128     else
129         Ff(i) = 1/(cosh(sqrt(s(i)*2*tin))+...
130             ((...
131             ((a+b*s(i))./(c+d*s(i)))...
132             )*tcat)/sqrt(2*tin*s(i)))...
133             *sinh(sqrt(2*s(i)*tin)));
134     end
135 end
136
137 Fscaling = real(Ff(end/2-1));
138
139 fk = @(s) 1/cosh(sqrt(s*2*tin));
140
141 nY = Ff/Fscaling;
142
143 S1 = abs((real(Y(loc))-real(nY(loc)))).*scaling;
144 S2 = abs((imag(Y(loc))-imag(nY(loc)))).*scaling;
145
146 S = sum(S1)+sum(S2);
147
148 nY(end/2) = Y(end/2);
149 Fnew = abs(iffn(nY)*tlen/tmax*Yscale);
150
151 plot(tnew,p)
152 hold on
153 plot(tnew,Fnew,'—','LineWidth',2)
154 hold off
155
156 return
157 plot(timesingle,Fint)

```

Listing 7.8: Script used in the LFF method for TAP data analysis

```

1 [rl,c1] = size(Alpha);

```

```

2 Anew = linspace(0,1,200);
3 for i = 1:rl
4     %Grabs current variable
5     Tcurrent = Temp(i,:);
6     Acurrent = Alpha(i,:);
7     Scurrent = Signal(i,:);
8
9     try
10    % Interpolates the data points to the new ones Anew
11    T(i,:) = pchip(Acurrent, Tcurrent, Anew);
12    A(i,:) = Anew;
13    S(i,:) = pchip(Tcurrent, Scurrent, T(i,:));
14    catch
15        % If interpolation fails we remove non-unique values then try again
16        [AcurrentU, IA, ~] = unique(Acurrent);
17        TcurrentU = Tcurrent(IA);
18        ScurrentU = Scurrent(IA);
19
20        T(i,:) = pchip(AcurrentU, TcurrentU, Anew);
21        A(i,:) = Anew;
22        S(i,:) = pchip(TcurrentU, ScurrentU, T(i,:));
23    end
24 end
25
26 % Performs fitting for each subsection of A
27 % Initial starting guess
28 Ea = 50;
29
30
31 % Performs fitting for each trapezial integral
32 for i = 1:199
33     T1 = T(:, i);
34     [var_min, Sv] = fminsearch(@(Ea) EaSum(T1, Ea, Beta), Ea);
35     EaC(i) = var_min;
36     TC(:, i) = T1;
37     SC(i) = Sv;
38 end
39
40 loc = find(A(1,:) > 0.05 & A(1,:) < 0.95);
41 AA = A(1, loc);
42 TT = T(1, loc);
43 SS = S(1, loc);
44 Ea = EaC;
45 EaC = Ea(loc);
46
47
48 % Sets up variables for Ea calculation
49 for i = 1:length(B);
50     % Setup of Kissinger calc
51     K1(i,:) = log(B(i)./(T(i, :).^2));
52     K2(i,:) = -1./(R.*T(i, :));
53
54     % Setup of FWO calc

```

```

55     FW1(i,:) = log(B(i));
56     FW2(i,:) = -1.052./(R.*T(i,:));
57
58     % Setup of Friedman Calc
59     F1(i,:) = log(S(i,:).*B(i));
60     F2(i,:) = -1./(R.*T(i,:));
61 end
62
63 % Calculates gradient of points to get activation energy
64 for i = 1:length(Anew);
65     x = K2(:,i);
66     y = K1(:,i);
67     Val = diff(y)'/diff(x)';
68     GK(i) = Val(1);
69
70     x = FW2(:,i);
71     y = FW1(:);
72     Val = diff(y)'/diff(x)';
73     GFW(i) = Val(1);
74
75     x = F2(:,i);
76     y = F1(:,i);
77     Val = diff(y)'/diff(x)';
78     GF(i) = Val(1)/1000;
79 end

```

Listing 7.9: Example script used in the calculation of activation energy profile for TPR thermograms

```

SYSTEM VASP MD RUN NPT
# Electronic stuff
PREC = Normal # Only use Low after testing
ENCUT = 600
GGA = PE
ALGO = Fast # Can use Very Fast if taking too long
LREAL = Auto
EDIFF = 1E-5 # Default is too large (causes drift)
ISMEAR = 0 # Gaussian smearing
SIGMA = 0.05

# D3 Parameters
IVDW = 11 # Turns on D3
VDW_S6 = 1.0 # S6 parameter
VDW_SR = 1.217 # SR parameter
VDW_S8 = 0.722 # S8 parameter
VDW_RADIUS = 15.0 # Cutoff radius for pair interactions
VDW_CNRRADIUS = 10.0 # Cutoff radius for coordination number

# MD Parameters
IBIRON = 0 # Run MD
ISIF = 2 # Relax cell shape
LANGEVIN_GAMMA = 20 20 20 20 20 # How often to add friction to atoms (per ps: each atom)
MDALGO = 3 # Choose thermostat

```

```

POTIM = 0.5      # Timestep (fs)
NBLOCK = 50     # Updates XDATCAR every 50 steps
NSW = 30000     # Number of Steps
TEBEG = 550     # Start Temp
TEEND = 550     # End Temperature
ISYM = 0        # Remove symmetry
MAXMIX = 40     # Reuse mixer from one MD step to the next
NELMIN = 4      # Perform at least 4 electronic steps (can increase if system is comp

# Printing parameters
LCHARG = .FALSE. # Stops writing CHGCAR file
LWAVE = .FALSE. # Stops writing WAVECAR file
LBLUEOUT = .TRUE.

# Speed settings: should be ~ sqrt of cores
NCORE = 16      # Number of cores thing

```

Listing 7.10: INCAR file from Ab initio MD run

```

SYSTEM NEB_run
ISTART = 1      # 0: New job, 1: Continuation job

#Planewaves and precision settings
PREC = Accurate
GGA = PE        # Use the PBE functional 91 is PW91
ENCUT = 600    # Set planewave cutoff by hand
ICHARG = 1      # Initiate charge as superposition of atomic charges

#Relaxation parameters
NSW = 10
NELM = 50
ISPIN = 2
IBRION = 2     # 1: Quasi NR, 2: Conjugate gradient
#EDIFF = 1E-4  # Electronic relaxation threshold, 1E-4 is default
#EDIFFG = -1.5E-2 # Electronic relaxation threshold, 1E-3 is default
ISIF = 2       # Calculate atom forces and relax but not shape/size cell

# Settings for speed
ALGO = Fast    # RMM-DIIS algorithm for electrons
LREAL = .FALSE. # evaluate projection operators in real space
NSIM = 4       # blocked algorithm update, four bands at a time
ISYM = 0       # Turns off symmetry

# Recommended option for preventing force noise
ADDGRID = .TRUE.

#NEB Calculation
IMAGES=24
ICHAIN=0
LCLIMB=.TRUE.
ISMEAR=0
SIGMA=0.02

```

Listing 7.11: INCAR file from ring closing NEB calculation

```

SYSTEM = Cu110_diss
ISTART= 1 !uncomment to restart from previous after first run

#plane waves and precision settings
ENCUT = 500 !cutoff energy plane wave
PREC = ACCURATE
GGA = PE #PBE
ICHARG = 1 #start with atomic densities , change to 1 to restart from previous CHGCAR

#efficient relaxation from unreasonable starting guess
IBRION = 2 #use 1:quasi newton 2:cg 3:damped md
NELMIN = 4
NELM = 200
EDIFF = 1E-4
EDIFFG = -1.5E-3
#POTIM = 0.01
NSW = 20 #number of ionic steps

# Settings for speed
ALGO = Fast # RMM-DIIS algorithm for electrons
LREAL = .FALSE. # evaluate projection operators in real space
NSIM = 4 # blocked algorithm update, four bands at a time

#prevent force noise
ADDGRID=.TRUE.

#other settings
ISIF = 2 #relax th atoms not the cell
ISPIN = 2 #spin non-polarised
IMAGES=24 #number of images you have
SPRING=-5 #-5 sets to run NEB, 0=Climbing NEB
NPAR = 8
ISMEAR = 0
SIGMA = 0.02
ISYM = 0

```

Listing 7.12: INCAR file from hydrogen dissociation NEB calculation

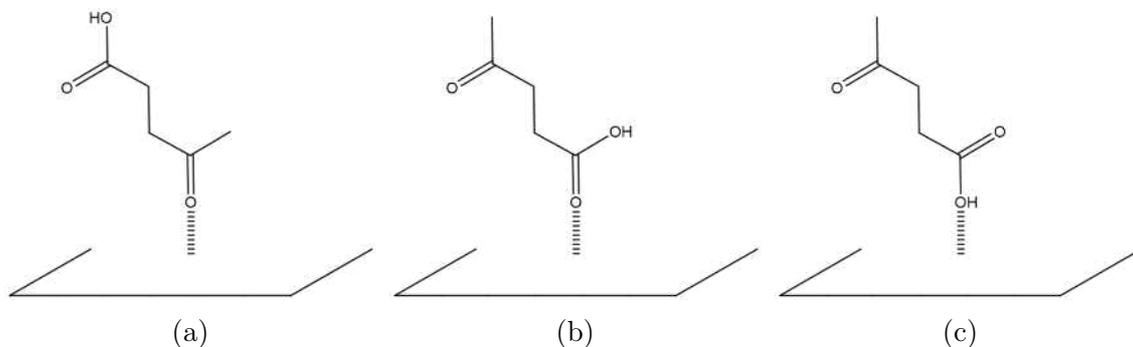


Figure 7.2: Binding modes for the Levulinic Acid a) 1 ketone group b) 2 carbonyl in carboxyl group c) 3 alcohol in carboxyl group.

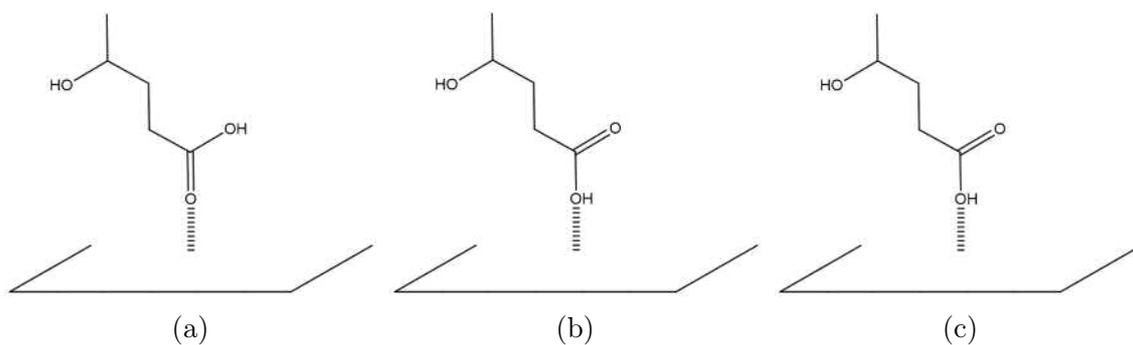


Figure 7.3: Binding modes for the hydrogenated intermediate a) 1 carbonyl in carboxyl group b) 2 alcohol in carboxyl group c) 3 alcohol group.



Figure 7.4: Binding modes for the GVL a) 1 carbonyl group b) 2 through ring.



Figure 7.5: Binding modes for the H₂ a) 1 molecular b) 2 dissociated.