

Creation and Evaluation of a Combustion and NO_x Prediction Model for a GDI Engine



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:

Date:

Acknowledgements

The author thanks project supervisor Malcolm McDonald for his support, guidance and contagious enthusiasm for all things car related, throughout the year. Gratitude is also extended to Jordan Ballantyne for sharing his MATLAB knowledge when the author had questions.

Abstract

Creation of a combustion and NO_x prediction model is outlined from a combination of measured engine data and calculations from first principles. Required engine parameters are first computed. Two dissociation models are created from a known iterative method using different degrees of accuracy and compared. The extended Zeldovich mechanism is then employed to predict NO_x using results gathered so far. Calculated results are compared with measured engine data and the model is evaluated.

Contents

Acronyms.....	v
Introduction	1
1 Engine Model.....	2
1.1 Polytropic Index	2
1.1.1 Mass Fraction Burned.....	7
1.1.2 Cylinder Temperature	12
1.1.3 Heat Released	19
2 Oxides of Nitrogen	21
2.1 NO _x Formation	25
2.1.1 Thermal NO	25
2.2 Dissociation Model.....	27
2.2.1 Reaction Rates	27
2.2.2 Constructing Useful Equations.....	28
2.2.3 Calculating Partial Pressure Equilibrium Constants (K_p).....	30
2.2.4 Goodger's Dissociation Model	32
2.2.5 Adaptation of the Goodger Method	33
2.2.6 Results.....	38
3 Thermal NO _x Model	44
3.1 O ₂ and N ₂ Concentrations	44
3.2 Extended Zeldovich Mechanism	47
3.3 Literature Review	50
3.4 Results.....	56
3.4.1 Analysis	58
4 Extended Goodger Dissociation Method	64
4.1 Methodology	64
4.2 Results.....	68
5 References	Error! Bookmark not defined.
6 Appendix A – Ford Fox Data	Error! Bookmark not defined.

6.1	Lambda.....	Error! Bookmark not defined.
6.2	Air Fuel Ratio.....	Error! Bookmark not defined.
6.3	Fuel Mass (kg/h).....	Error! Bookmark not defined.
6.4	Exhaust Gas Recirculation.....	Error! Bookmark not defined.
7	Appendix B – Stoichiometry	79
8	Appendix C – Calculating Gamma	81
9	Appendix E – MATLAB Script for Dissociation Model.....	95
10	Appendix F - Dissociation Model: Tabulated Results	133

Acronyms

AFR – Air Fuel Ratio

BSFC – Brake Specific Fuel Consumption

DI – Direct Injection

ECU – Engine Control Unit

EVO – Exhaust Valve Opening

FER – Fuel Equivalence Ratio

HCCI – Homogenous Charge Compression Ignition

ICE – Internal Combustion Engine

IVC – Inlet Valve Closing

SOC – Start of Combustion

MFB – Mass Fraction Burned

Introduction

Combustion engines face ever-tightening emissions regulations and the threat of eventual extinction if renewable technologies can achieve levels of performance, infrastructure and cost comparable with gasoline and diesel vehicles while lowering pollutants. Though the shift to full fleet hybridization is well underway for many car manufacturers, recent advances in combustion technology seem set to prolong the life of the ICE regardless of the political landscape.

Mazda's spark-controlled compression ignition, Nissan's variable compression ratio engine (Kojima, et al., 2018) and MAHLE's turbulent jet ignition technology (Attard & Parsons, 2010) all offer unprecedented efficiency and emission control for ICEs, at the expense of increased complexity. Increased complexity inevitably means increased development and manufacturing costs. The importance of a combustion model that can accurately predict engine behaviour – thus reducing these costs by facilitating calibration – is self-evident.

Outlined in this report is the creation, deployment and analysis of a 2-zone NO_x prediction model based on provided engine data and calculations from first principles.

Table 0-1 - Ford Fox 1.0 litre specifications

Bore	0.0719	m
Stroke	0.082	m
Crank Radius	0.041	m
Bore Area	0.00406	m^2
Cyl Disp.	0.000333	m^3
Cyl Disp.	332.9366	cc
Total Displacement	998.8097	cc
Power	125	PS
Torque	170	Nm

Data is taken from a Ford Fox 1.0 litre turbocharged GDI engine (Table 0-1) and used to model the engine's parameters. Creation of a six-species dissociation model is combined with calculated parameters to generate data for use in NO_x prediction. A NO_x prediction model is then created using the extended Zeldovich mechanism. Results are plotted against known engine data, analysed and the methods of implementing the extended Zeldovich mechanism are discussed at length. Finally, the dissociation model is extended to include an additional species and the NO_x prediction model is implemented once again to draw a comparison with previous results.

1 Engine Model

A series of calculations must be performed on the pressure and volume data to ascertain additional thermodynamic data.

1.1 Polytropic Index

Calculation of the polytropic index is necessary to distinguish rise in cylinder pressure due to piston compression p_{cp} versus rise in cylinder pressure due to combustion p_{cb} (Rassweiler & Withrow, 1938). Once pressure rise due to combustion is determined, the SOC can be found and the MFB calculated.

Compression and expansion of the cylinder is said to be approximately polytropic (Lancaster, et al., 1975), in that it obeys the relationship (section 8.1):

$$pV^n = \text{Constant}$$

Equation 1-1

Where $p = \text{pressure}$, $V = \text{volume}$, $n = \text{polytropic index}$.

Mathematically, the polytropic index can be calculated for every degree. From Equation 1-1, and Boyle's law:

$$p_1 V_1^n = p_2 V_2^n$$

Equation 1-2

Then n can be calculated at every degree by calculating the natural log of pressure and volume:

$$n = \frac{\ln p_2 - \ln p_1}{\ln V_1 - \ln V_2}$$

Equation 1-3

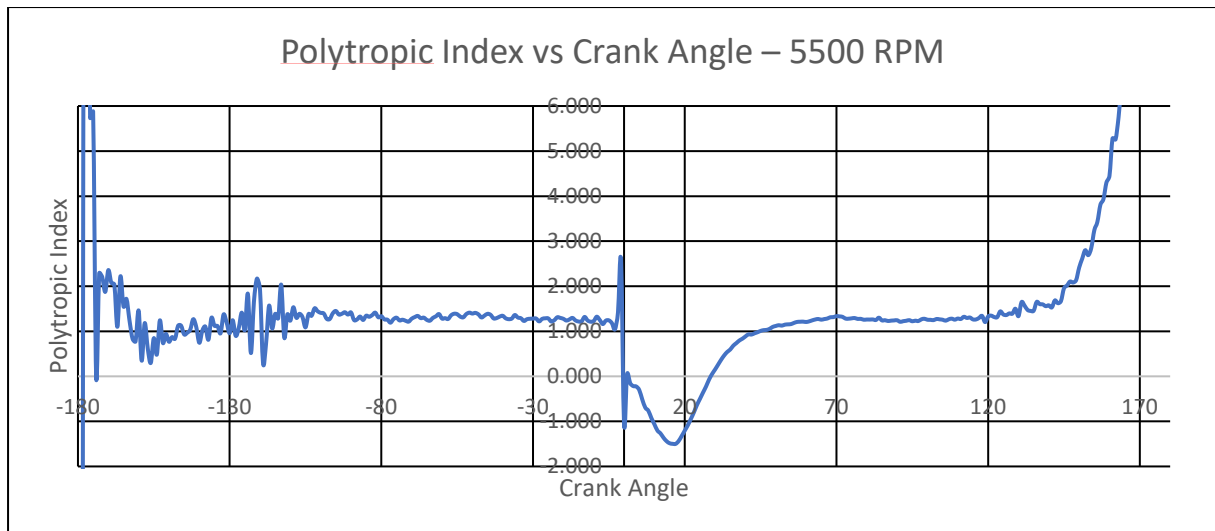


Figure 1-1 - Calculated polytropic index (n) variability over the cycle

The challenge is to determine a single value for the polytropic index over the compression and expansion strokes from calculated values that vary every degree (Figure 1-1). A mean calculation can be made over a number of degrees – However, where this calculation is made and the range of crank angle degrees it's made over can alter the result significantly. Brunt & Platts (1999) state the compression and expansion indexes should be calculated immediately before and after combustion but fail to specify more precise terms.

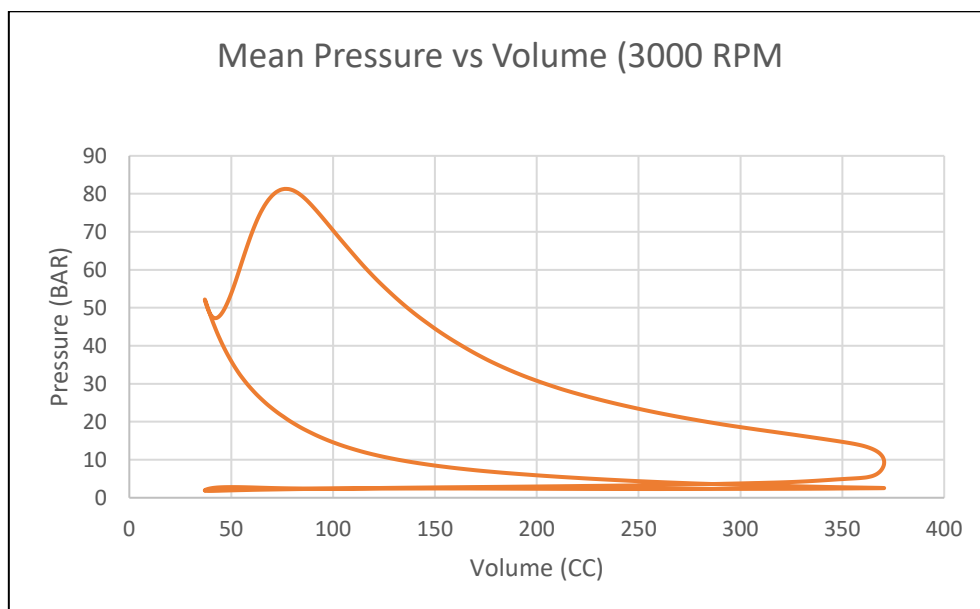


Figure 1-2 - PV diagram for Ford Fox engine. Note that start of combustion begins ATDC, causing difficulty for mean calculations

The method outlined in this work attempts to derive SOC from an expression using n which also makes this problematic. That said, SOC can be approximated from a PV diagram (Figure 1-1) or log PV diagram (Figure 1-2) to iterate the difficulty of selecting an average.

Table 1-1 - Calculated polytropic index averages over 5, 10, 20, 25 degrees before combustion

° Before Combustion	n	Average		
25	1.282194	1.20049		
24	1.277105			
23	1.265723			
22	1.269111			
21	1.258402			
20	1.275131	1.182985		
19	1.241069			
18	1.263211			
17	1.265449			
16	1.227782			
15	1.265667			
14	1.240575			
13	1.262592			
12	1.21581			
11	1.230884			
10	1.17722	1.117153		
9	1.140828			
8	1.092649			
7	1.079647			
6	0.936031			
5	0.112485	1.149032		
4	1.309224			
3	1.545096			
2	1.416784			
1	1.361572			

Table 1-2 demonstrates the illusive variability of n and the problem with defining a mean value.

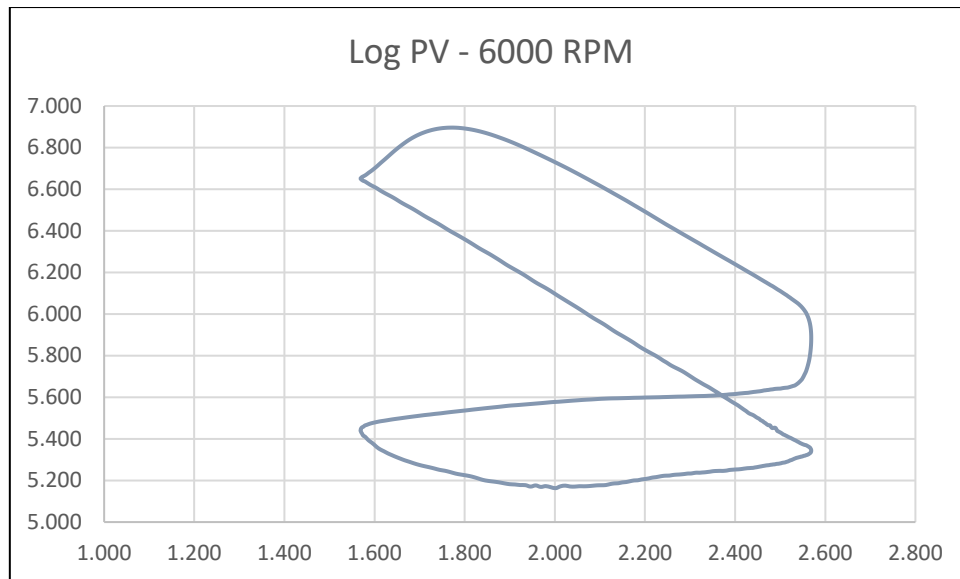


Figure 1-3 – Log PV Diagram for 6000 RPM

A more favourable method is outlined by Grau et al (2002). Logarithmic values for pressure and volume are calculated and a log PV diagram plotted. The compression and expansion strokes are visually inspected to determine suitable upper and lower values to be used for a linear best fit that are devoid of interference from:

- SOC/EOC
- Intake/exhaust strokes
- Data errors

For example, from Figure 1-3 compression values of 6.5 and 5.7 were selected for upper and lower values of the compression stroke, considered to be free from interreference. A best fit was performed on these values using the least squares method (Figure 1-2) to extract an average value – 1.31 – that finds agreement with observations from Heywood (1988), where $n = 1.3 \pm 0.05$. The method is repeated for the expansion stroke so that each index can be used on its respective stroke for determining SOC and EOC.

At this point, a mean can be taken for both cycles from the two or each can be employed on their respective sides of TDC. The latter method was used in this model.

Polytropic sensitivity is discussed in the next section.

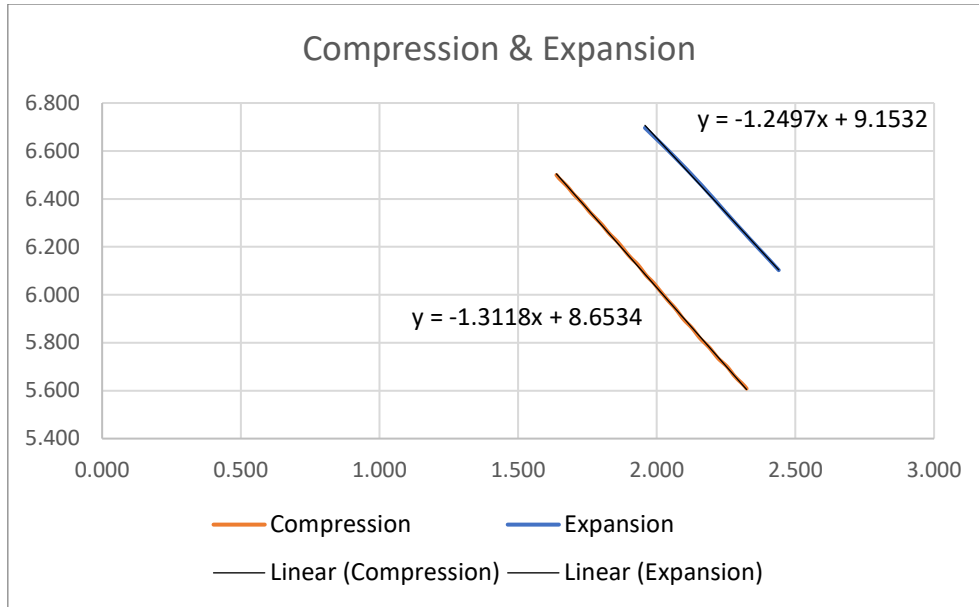


Figure 1-4 - Linear best fit of selected ranges. Expansion stroke included for comparison.

1.1.1 Mass Fraction Burned

Rassweiler and Withrow (1938) put forward that pressure change during combustion is equal to both the change in pressure due to compression and change in pressure due to combustion:

$$\Delta p = \Delta p_{cp} + \Delta p_{cb}$$

Equation 1-4

It was further suggested that change in pressure due to compression could be attributed to the polytropic relationship. For a polytropic process of two states the equation becomes:

$$p_1 V_1^n = p_2 V_2^n = \text{Constant}$$

Equation 1-5

This is re-arranged to find p_2 , which represents the instantaneous pressure change due to polytropic compression, now re-defined as p_{2n} to distinguish the polytropic state from the actual data state:

$$p_{2n} = p_1 \cdot \left(\frac{V_1}{V_2}\right)^n$$

Equation 1-6

Subtracting calculated p_{2n} from the recorded data p_2 makes it possible to observe when the recorded dataset deviates from the polytropic relationship. If experiencing only polytropic

compression, it is expected that the recorded data would be approximately equal to the calculated p_{2_n} at every increment. When experiencing additional pressure rise due to combustion, the recorded data will reflect this as higher values than the calculated p_{2_n} as it deviates from the polytropic relationship:

$$\Delta p_{cp} = \Delta p_{2_n}$$

Equation 1-7

$$\Delta p_{cb} = \Delta p - \Delta p_{2_n}$$

Equation 1-8

When $\Delta p_{cb} > 0$, combustion has begun. Determining the exact degree of SOC is problematic – combustion is not instantaneous and begins relatively slowly. A pre-determined value of pressure increase must be decided upon to indicate SOC. In this model, SOC is deemed to be the degree before $\Delta p_{cb} > 0.0$. This is based on the data trends before that point during polytropic compression and subsequent pressure increases that increase dramatically.

Identifying EOC is performed the same way, using decreasing values.

Once both SOC and EOC are identified, combustion pressures can be summed from SOC to EOC at each degree to get total combustion pressure: $p_{cb_{total}}$. Mass fraction burned for each degree can now be calculated:

$$m_{b_\theta} = \frac{\sum \Delta p_{cb_\theta}}{\sum \Delta p_{cb}}$$

Equation 1-9

Where $\sum \Delta p_{cb_\theta}$ is the cumulative total of combustion pressure at a CA degree. The result is a proportional calculation of mass fraction burned over the combustion period that ends when

$$\frac{\sum \Delta p_{cb_\theta}}{\sum \Delta p_{cb}} = 1.$$

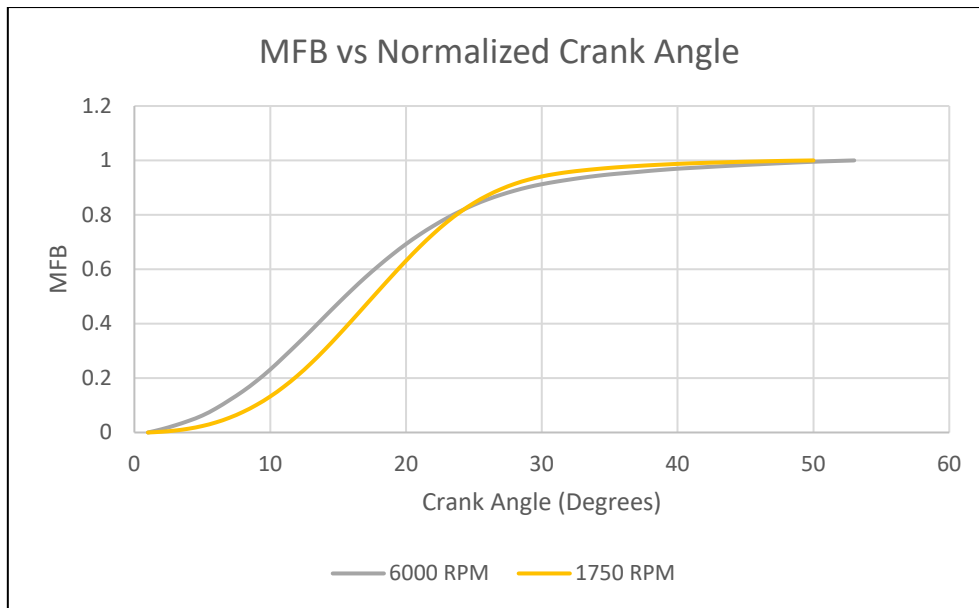


Figure 1-5- MFB at different RPM plotted against normalized crank angle for comparison of burn pattern. (Note: 6000 RPM should read 6500 RPM)

Observations about each combustion event can be made from the MFB. The air/fuel mix of 6500 RPM burns faster than 1750, because of increased piston speed, advanced ignition (compared to the others) and the rich FER, indicated by the Ford Fox data. The piston is responsible for 'feeding' mixture to the flame – an effect that increases with RPM, allowing combustion to remain almost constant in terms of crank angle degrees (Heywood, 1988). Advancing the ignition allows the mixture to burn in a smaller volume, giving the flame front less distance to travel to consume the mixture. The 10% rich FER at 6500 RPM ($\phi = 1.11$) gives what Stone and Ball (2004) succinctly refer to as 'best oxygen utilization' – Increasing the chance that all oxygen will contribute to oxidising fuel with the penalty of increased emissions and worse BSFC.

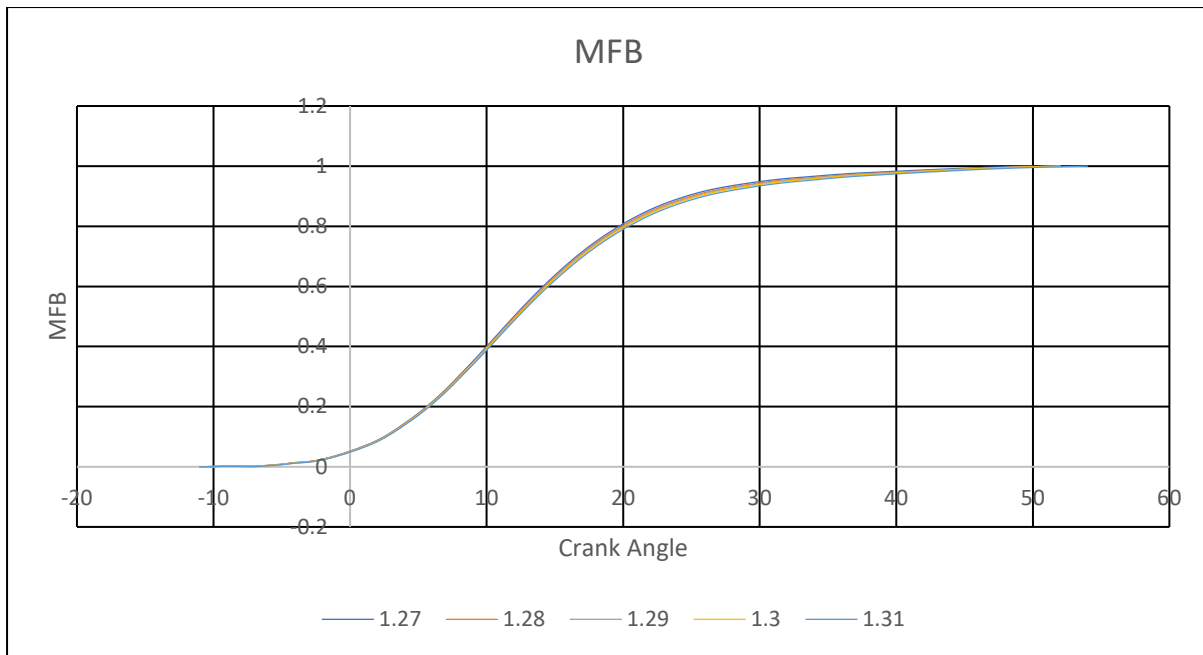


Figure 1-6 - Polytropic index influence on MFB

Figure 1-5 illustrates the relatively low sensitivity of MFB to the calculated polytropic index over a small range that has seen various use in the literature and that concerns over the index prevision are unwarranted. Brunt and Rai (1998) outline the incorrect use of a mean polytropic index and its effect on heat release after EOC but also point out the redundancy. As n can be calculated for every degree (Equation 1-3), there is no need to use a mean value.

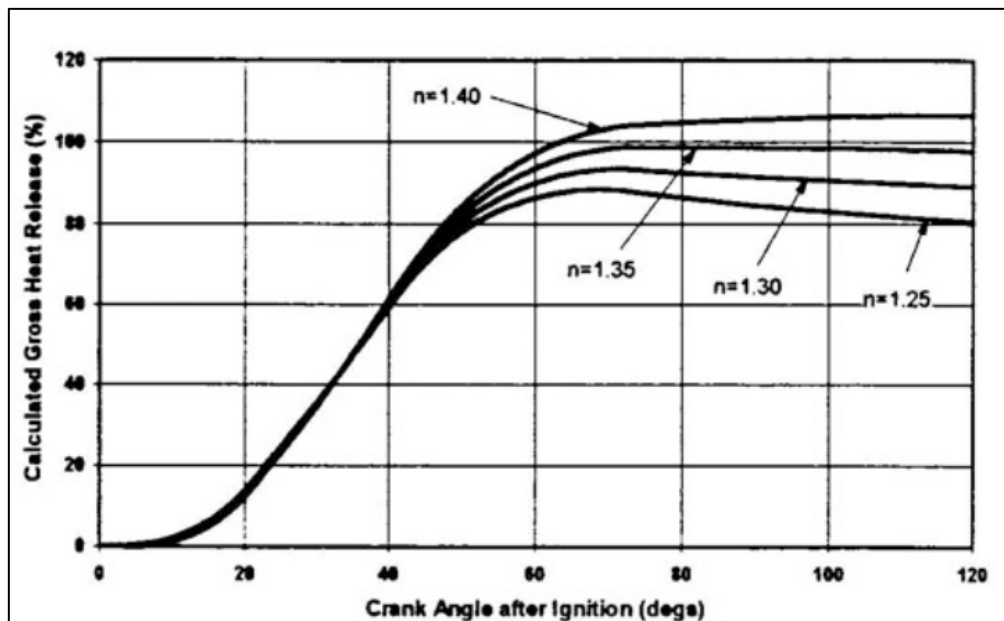


Figure 1-7 - Effect of mean polytropic index on heat release (Brunt and Rai, 1998)

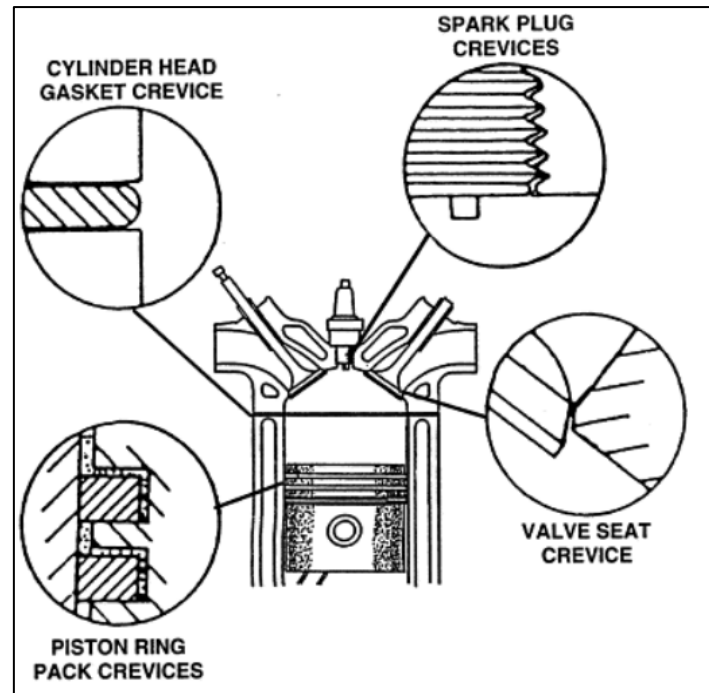


Figure 1-8 - Combustion chamber crevices according to Alkidas (1999)

Several assumptions are made to make this calculation possible, most obviously that all the fuel undergoes complete combustion. The model does not account for geometry-related crevices that can trap unburned hydrocarbons and prevent them undergoing combustion, thus reducing the air/fuel mixture (Alkidas, 1999).

Heywood (1988) states that pressure rise over the combustion period is proportional to the amount of fuel chemical energy released rather than amount of mass burned. It could be argued that fuel chemical energy released would in-turn be proportional to amount of mass burned for a truly homogenous mixture, but this raises several questions as to the method's validity for:

1. Direct injection, where the fuel may not have time or turbulence enough to mix totally with the air.
2. High RPM, where mixing time is reduced further by engine speed.

Though the polytropic index is approximated to a constant value every effort has been made to ensure accurate representation. That said, the MFB seems insensitive to errors that renders this a non-issue.

1.1.2 Cylinder Temperature

Calculation of the MFB allows the active air/fuel mixture inside the cylinder to be imagined as two distinct zones with different properties at any given point during combustion: A burned zone and an unburned zone. The MFB is used to determine proportions of each zone's volume, temperature and specific gas constant at each degree.

The ideal gas law can be re-arranged to provide an equation for temperature:

$$T = \frac{pV}{mR}$$

Equation 1-10

Where T = temperature, p = pressure, V = volume, m = mass, R = specific gas constant.

Pressure and volume data is provided by the dataset. Mass can be calculated using additional Ford Fox data and specific gas constant can be worked out via the combustion chemistry.

1.1.2.1 Cylinder Mass

Cylinder mass consists of an air/fuel mixture that is pre-determined by requirements outlined by the engine's ECU. Cylinder mass is considered constant upon IVC to EVO by assuming negligible blowby effects. AFR and fuel flow data for the Ford Fox 1L engine is available. Fuel mass is given in kg/h so must first be converted:

$$\text{Fuel Mass (per Cycle per Cylinder)} = \frac{m_{fuel_{kg/h}}}{60 \cdot RPM \cdot 1.5}$$

Equation 1-11

Air mass is calculated:

$$\text{Air Mass} = AFR \cdot \text{Fuel Mass (per Cycle per Cylinder)}$$

Equation 1-12

The two are added together for total charge mass per cycle per cylinder.

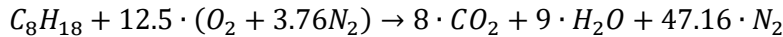
1.1.2.2 Specific Gas Constant

Specific gas constant is calculated by considering the molecular make-up of the reactants and products of combustion and calculating an atomic balance. Lambda (λ) values from the Ford Fox data are converted to FER (ϕ):

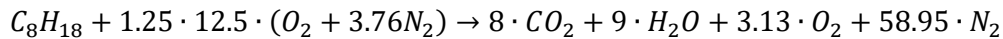
$$\phi = \frac{1}{\lambda}$$

Equation 1-13

Heywood (1988) presents two equations for the atomic balance of isooctane: stoichiometric and 25% lean respectively:

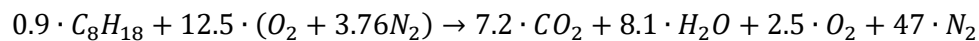


Equation 1-14



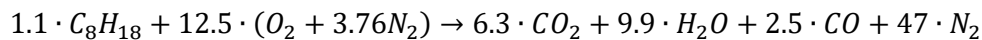
Equation 1-15

Equation 1-15 is manipulated to include excess O_2 as a function of FER (ϕ) rather than an excess of air to simplify calculation:



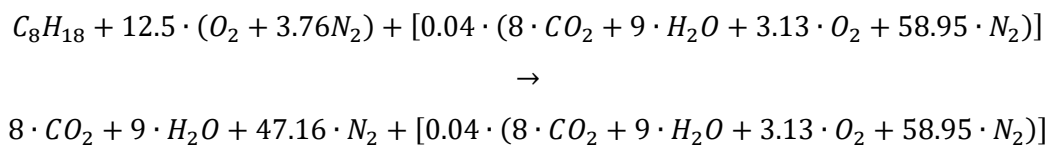
Equation 1-16

Heywood (1988) asserts that the products from a rich combustion (CO_2, H_2O, CO, H_2, N_2) cannot be calculated from an atomic balance alone, requiring in-depth modelling of the dissociation process. Omission of H_2 from the combustion products allows a simplified rich balance to be calculated (McDonald, 2017), although lacks precision (10% rich):



Equation 1-17

Combining FER data with these atomic balance equations – selecting the correct occasion for the correct combustion state – allows the products to be approximated. EGR data is also available for the Ford Fox and can be included in the balance by assuming no change of state. For example, if EGR percentage was 4%:



Equation 1-18

Gas constant for a specific gas can be calculated (Rogers & Mayhew, 1992):

$$R = \frac{\tilde{R}}{\tilde{m}}$$

Equation 1-19

Where R = specific gas constant (J/kg.K), $\tilde{R}$ = universal gas constant (8314 J/kg.K), $\tilde{m}$ = molar mass (kg/kmol). Total molecular mass is calculated by multiplying atomic masses by

the number of reactants or products present. E.g. To work out R for stoichiometric combustion products including 4% EGR:

Table 1-2 - Molecular Mass of Stoichiometric Products

Molecule	Molecular Mass
CO ₂	44
H ₂ O	18
N ₂	28

$$\text{Product Molecular Mass} = (8 \cdot 44) + (9 \cdot 18) + (47 \cdot 28) = 1830 \frac{\text{kg}}{\text{kmol}}$$

Equation 1-20

$$\text{EGR Molecular Mass} = 0.04 \cdot 1830 = 73.2 \frac{\text{kg}}{\text{kmol}}$$

Equation 1-21

$$\text{Total Molecular Mass} = 1830 + 73.2 = 1903.2 \frac{\text{kg}}{\text{kmol}}$$

Equation 1-22

Total molecular mass is divided by the number of product molecules to get an average molecular mass per molecule:

$$\text{Total Number of Molecules} = 8 + 9 + 47 + [0.04 \cdot (8 + 9 + 47)] = 66.56$$

Equation 1-23

$$\frac{\text{Molecular Mass}}{1 \text{ Molecule}} = \frac{1903.2}{66.56} = 28.59 \frac{\text{kg}}{\text{kmol}}$$

Equation 1-24

This can now be used in Equation 1-19 to get the specific gas constant for the products:

$$R = \frac{8314}{28.59} = 290.8 \frac{\text{J}}{\text{kg} \cdot \text{K}}$$

Equation 1-25

The method remains the same for reactants and products at any FER, giving R_{UB} and R_B ,

1.1.2.3 Average Cylinder Temperature

Average cylinder temperature between IVC and EVO can now be calculated using the rearranged ideal gas law (Equation 1-10). Outside of these events, mass exchange takes place negating the cylinder mass calculation made previously, rendering the temperature calculation inaccurate. Burned and unburned zones for the two-zone model will have distinct

temperatures as mentioned – average cylinder temperature allows these temperatures to be calculated by determining the last known temperature before SOC.

Other steps must first be taken to work out burned and unburned zone temperatures.

1.1.2.4 Two-Zone Temperatures

Once gamma (γ) is calculated (Appendix C – Calculating Gamma), unburned zone temperature (derived in section 8.2) can be determined by using the last known temperature before SOC as T_{UB1} :

$$T_{UB2} = T_{UB1} \cdot \left(\frac{p_2}{p_1} \right)^{\frac{\gamma-1}{\gamma}}$$

Equation 1-26

Unburned volume can now be calculated using the ideal gas law:

$$V_{UB} = \frac{m_{UB} \cdot R_{UB} \cdot T_{UB}}{p}$$

Equation 1-27

Where:

$$m_{UB} = (1 - x) \cdot m$$

Equation 1-28

x = mass fraction burned, m = total cylinder mass.

Burned volume is made known by subtracting V_{UB} from the recorded volume data at each degree.

Finally, burned temperature is calculated using burned properties in the ideal gas law:

$$T_B = \frac{p \cdot V_B}{x \cdot m \cdot R_B}$$

Equation 1-29

Where:

$$m_B = x \cdot m$$

Equation 1-30

1.1.2.5 Results and Analysis

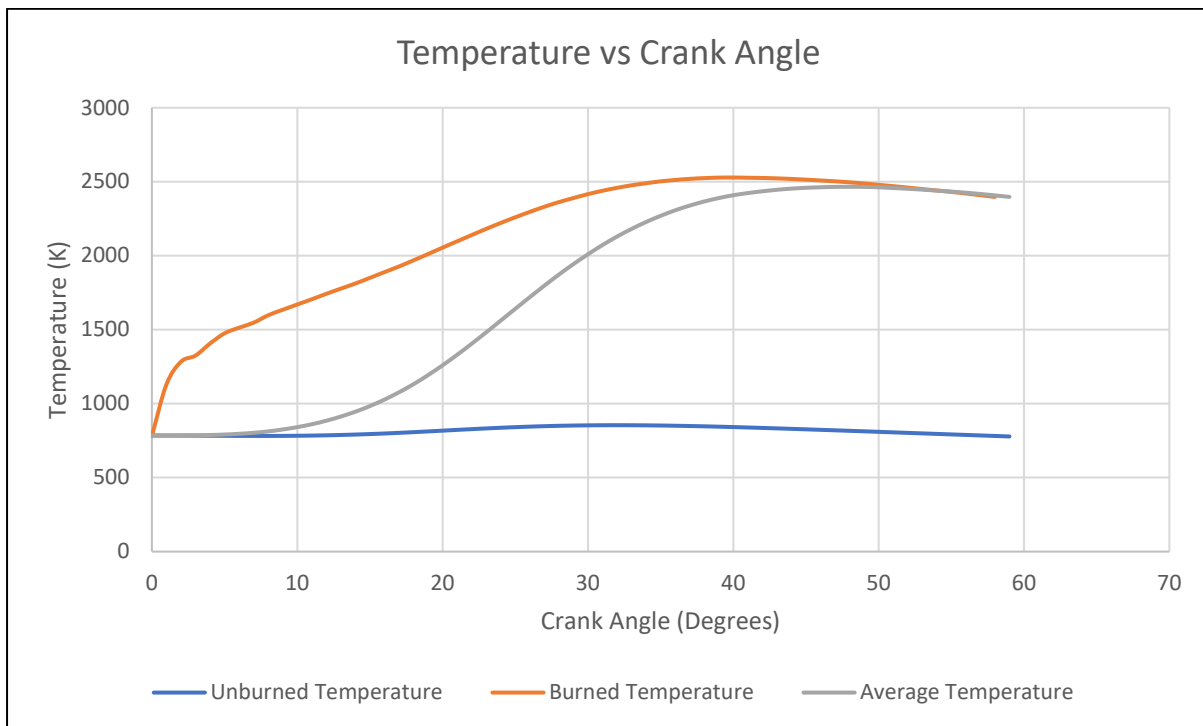


Figure 1-9 - Temperature during combustion (5000 RPM)

Half of the results (3500-6500 RPM) find good agreement with the literature (Figure 1-9 and Figure 1-10). Results in the lower portion of the RPM band begin with excessively high temperatures (Figure 1-11) because of the way large V_B values combine with small MFB values in Equation 1-29. Unfortunately, these overpredictions are a result of ignition delay at these lower RPM values to avoid knock by burning in a larger volume and are unavoidable.

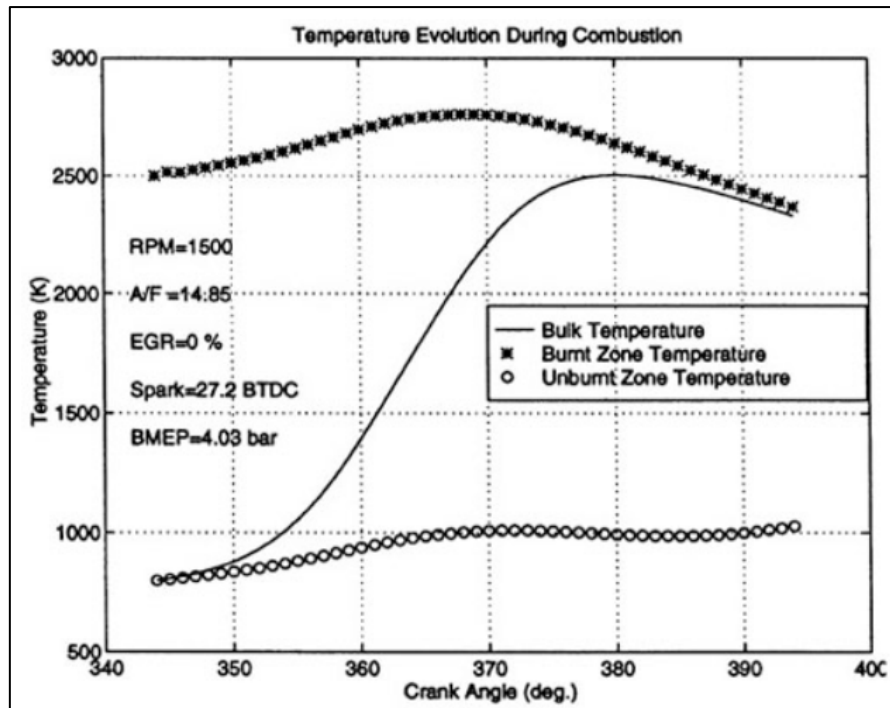


Figure 1-10 - Temperature evolution during combustion (Guezennec and Hamama, 1999)

These excessively high temperatures must be disregarded with the penalty of working with fewer values over the combustion period. Despite this adaptation, low RPM burned zone temperatures still suffer from a lack of normalization compared to high RPM results and even 'acceptable' temperatures are high for the retarded spark, lean conditions. This is confirmed by observing measured exhaust gas temperatures (Table 1-3) for the troubled conditions.

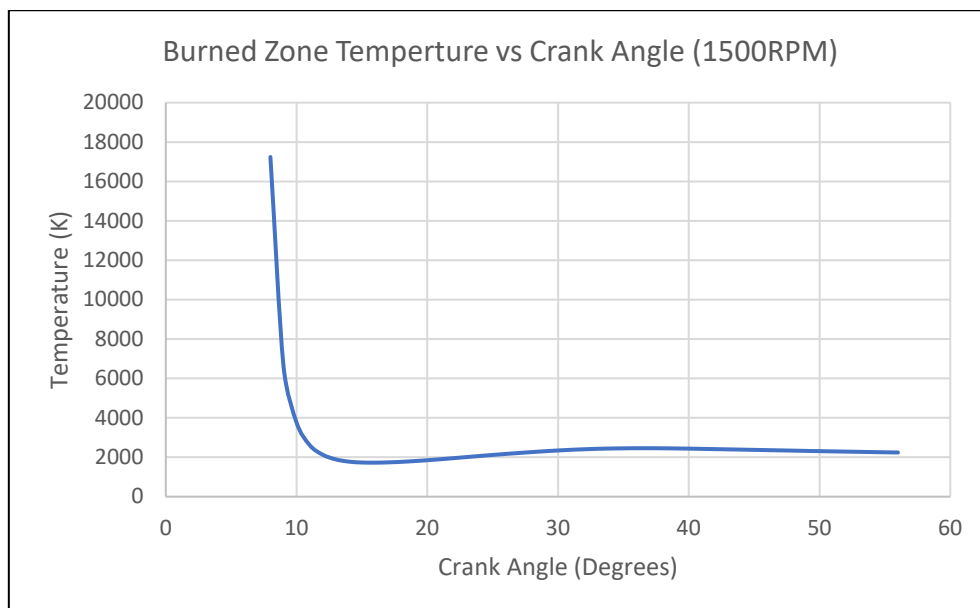


Figure 1-11 - Example of miscalculated burned zone temperatures

Given the thermal sensitivity of the extended Zeldovich mechanism and lean FERs, it is forecast that the discussed RPM sites will exhibit excessively high NO prediction. Research will be undertaken to formulate a solution.

Table 1-3 - Exhaust temperatures from Ford Fox data, cylinder 1

RPM	Exhaust Temperatures (°C)
1000	485.4
1500	624.4
1750	711.9
2000	798.4
2500	870.4
3000	881.9
3500	900.3
4000	919.9
4500	910.3
5000	930.5
5500	926
6000	906.3
6500	861.2

1.1.3 Heat Released

Heat released (ΔQ_{NET}) from the engine during combustion is calculated using an expression constructed in terms of γ (Appendix B – Calculating Gamma), n and work done ($p dV$). A derivation of the expression used can be found in section 8.4.

$$dW = p \cdot dV$$

Equation 1-31

$$dQ_{NET} = \frac{\gamma - n}{\gamma - 1} \cdot p \cdot dV$$

Equation 1-32

γ is derived from the unburned reactants before SOC and from the burned products after EOC. Valve timings are required to determine heat release between IVO-SOC and EOC-EVO. During combustion, MFB (x) is used to calculate γ at each degree:

$$\gamma_{CB} = (x \cdot \gamma_B) + [(1 - x) \cdot \gamma_{UB}]$$

Equation 1-33)

dQ_{NET} is then summed throughout the combustion process to get total heat lost during combustion.

dQ_R describes the heat released from the fuel and is determined by first calculating fuel burned per degree (m_{FB}) using the MFB (x):

$$m_{FB} = dx \cdot m_{fuel}$$

Equation 1-34

At every degree, ΔQ_R can now be calculated by multiplying the fuel burned per degree by the calorific value (Q_{LHV}) of the fuel (Appendix B – Stoichiometry):

$$dQ_R = m_{FB} \cdot Q_{LHV}$$

Equation 1-35

ΔQ_R can now be summed across combustion for total heat released by the fuel. Losses can be calculated:

$$dQ_L = dQ_R - dQ_{NET}$$

Equation 1-36

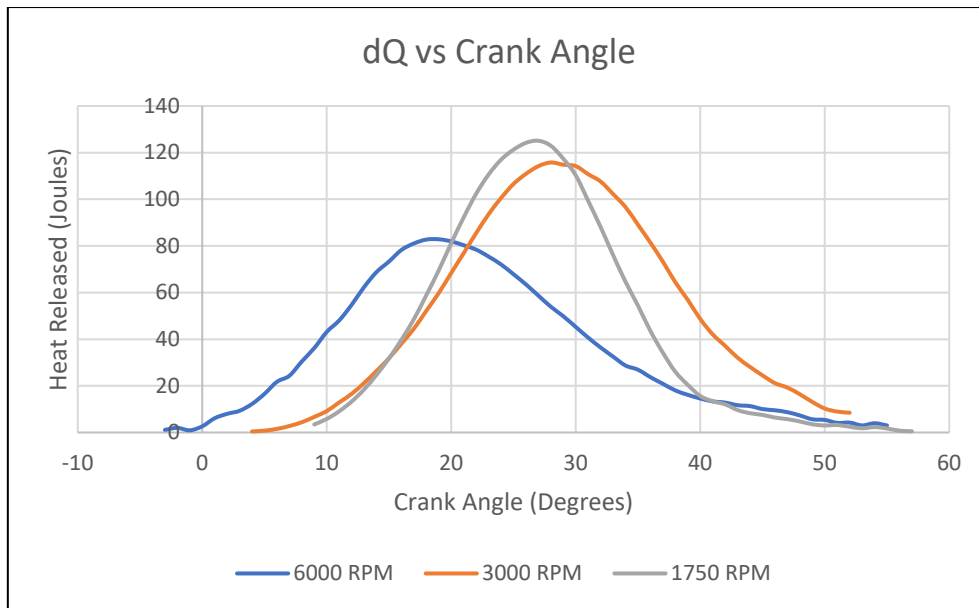


Figure 1-12 - Results of heat release during combustion

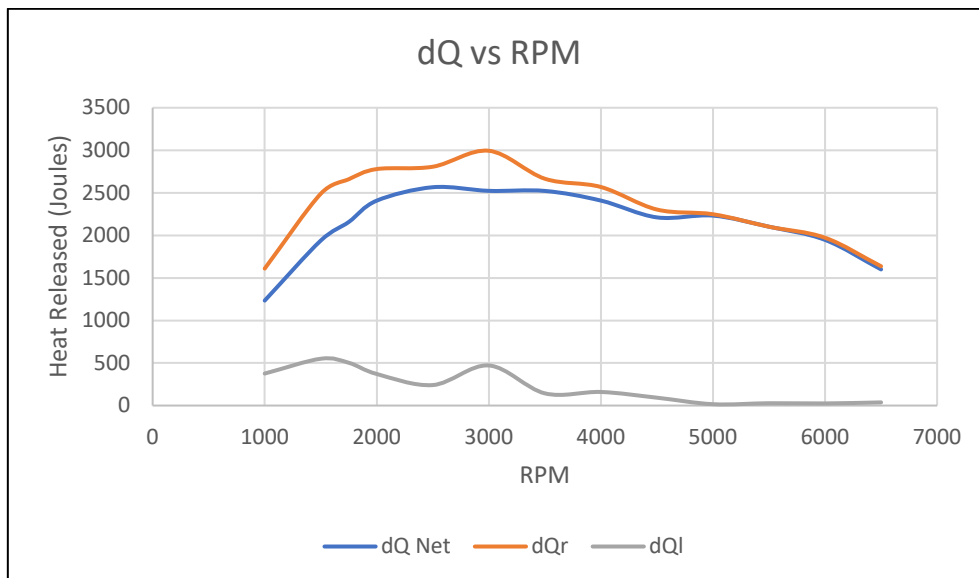


Figure 1-13 - Heat releases and losses across RPM

Without including IVO and EVO data, results do not paint a full picture particularly in terms of losses. That said, observable losses (Figure 1-13) are as expected: Larger losses at low RPM with larger times for heat transfer, progressing to very minor losses at high RPM with no time for heat transfer.

Brunt et al (1998) examines sensitivity of heat release to gamma by comparing simulation results of constant values. Attention is drawn to the inversely proportional relationship to $(\gamma - 1)$, whereby heat release is exaggerated using a low gamma value and vice-versa.

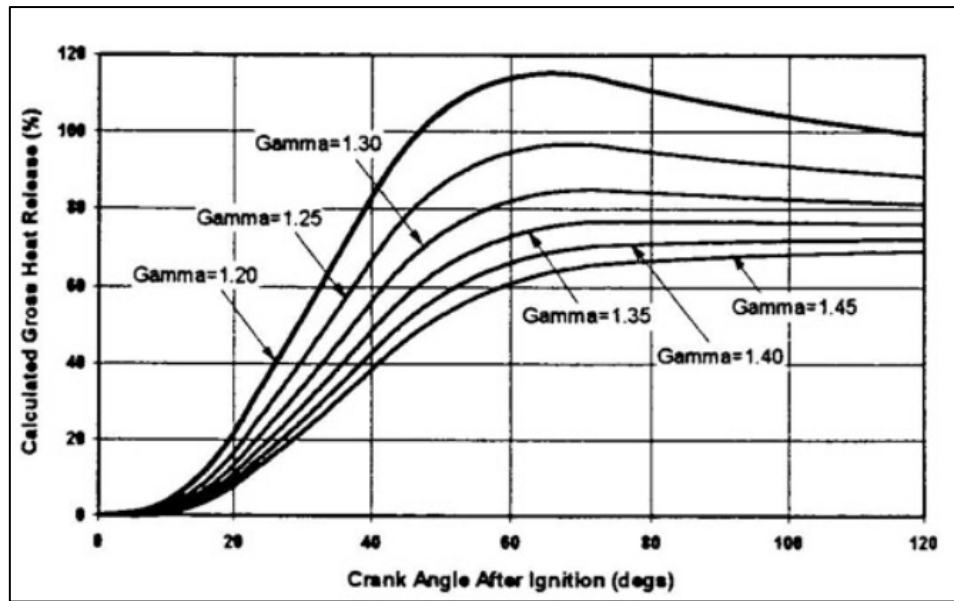


Figure 1-14 - Sensitivity study of gamma. (Brunt et al, 1998)

The method of Brunt et al to achieving a variable gamma is a similar use of best fit but created directly as a function of temperature and FER.

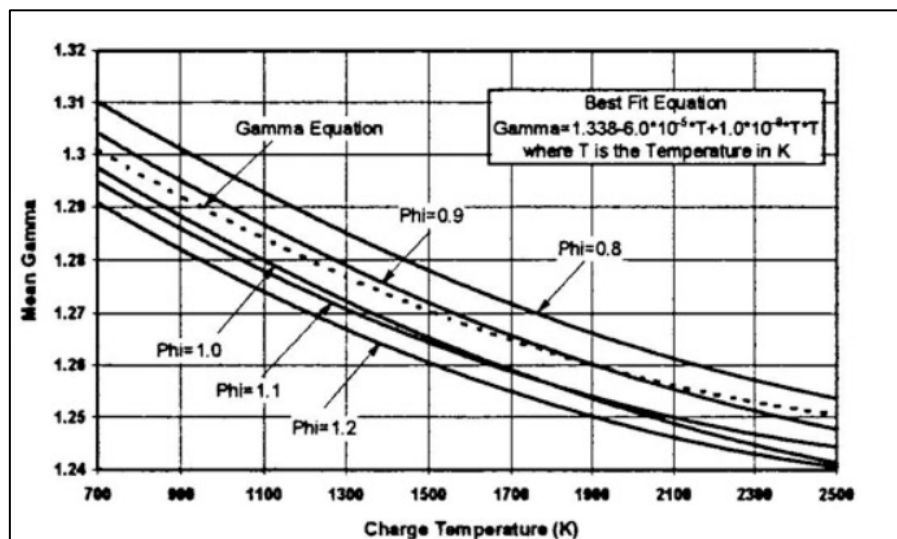


Figure 1-15 - Best fit gamma. Note the proximity to Phi=0.9 (Brunt et al, 1998)

The equation was implemented in the Fox Excel model for comparison.

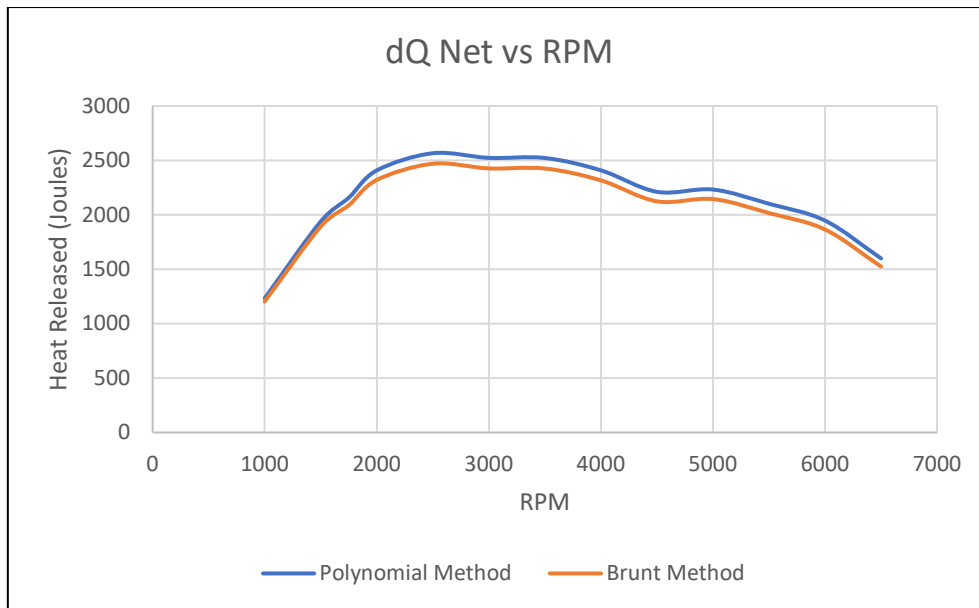


Figure 1-16 - Comparison of gamma best fit

Brunt's method slightly under predicts heat release at all but very lean conditions. This agreement at $\phi \approx 0.9$ was foreseeable from the fit line's proximity to the same FER in Figure 1-15. The accuracy of the method is surprising, giving the variance pictured in the best line fit.

Being a product of the specific heat of chemical species, gamma is affected by dissociation (Heywood, 1988). This is not accounted for in the simple heat release calculation, in which γ_B is calculated via 4 out of 5 species, depending on FER. Heywood (1988) shows the drastic effect dissociation can have on the ratio of specific heats – Figure 1-17.

Brunt et al (1998) have shown that the kind of dissociation-related decrease in γ (pictured) at combustion temperatures calculated for the Ford Fox would result in an increase of heat release. Inclusion of the dissociation model (section 2.2) for the calculation of gamma is decidedly complicated since the mole fractions are calculated through both pressure and temperature data. Proving this relationship would be an interesting piece of further work.

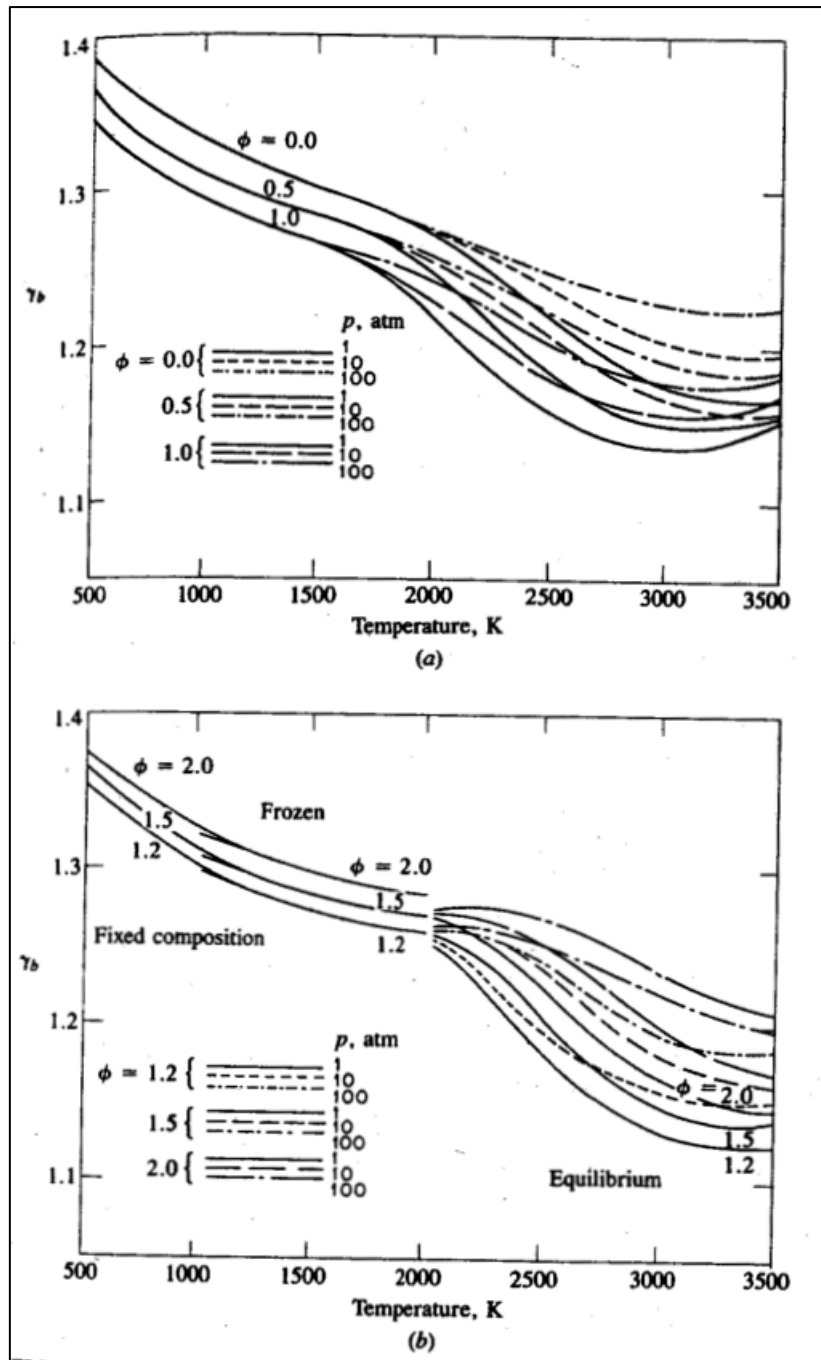


Figure 1-17 - Effect of dissociation on gamma

2 Oxides of Nitrogen

Nitrogen oxide (NO) and nitrogen dioxide (NO₂) – collectively known as NO_x – are products that can form when nitrogen (N₂) and oxygen (O₂) dissociate into free radicals allowing formation of NO_x. Concerns over NO_x production is widely researched and well justified.

Modern reports are reluctant to identify a definite link between NO_x and asthma development, due to presence of other toxic emissions, particularly sulfur oxides (EPA, 2016). Von Mutius (1998) recognises similar problems in various cases but offers a compelling piece of evidence in the form of Germany. When reunified, study of Western German residents (where NO_x was prevalent due to higher levels of automotive use) revealed higher instances of asthma and associated ailments when compared to Eastern German counterparts, who were subject to high levels of sulfur dioxide and particulate matter.

Koenig (1999) supplies further evidence by citing an increase in asthma and respiratory-related emergency room visits during peak hourly NO₂ concentration in Northern California, Athens and Barcelona. Boningari and Smirniotis (2016) cite the molecules' water solubility as the reason it is easily able to penetrate the lungs, causing bronchitis, emphysema and decreased lung function. While NO_x causation on asthma is up for debate, little argument can be found against NO_x association with respiratory problems,

On a more global scale, N₂O in particular is a greenhouse gas that the EPA calculates that has a warming effect on the atmosphere that is three-hundred times that of CO₂ per molecule and does not decompose for 114 years (EPA, n.d.). That an overwhelming large proportion of NO_x emissions are NO (98-99% for SI engines) makes little difference given that it is almost entirely converted to NO₂ in the atmosphere (Weisser, 2001).

With all this in mind, it is of little surprise that attention has shifted to reducing NO_x production in the automobile industry. Emission standards are set to tighten even more and the ability to accurately predict engine pollutants has never been more important.

A precise NO_x model can aid engine design and calibration during development and reduce testing costs. Latest aims in this area are centred around real-time prediction of NO_x as part of on-board diagnostics (OBD) (e.g., (Kihás & Et al, 2016)). Uses of such technology vary from better management of SCR, to scrutiny of the physical NO_x sensor or outright replacement of the sensor altogether (Camporeale, et al., 2017).

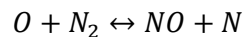
2.1 NO_x Formation

The literature shows general agreement in three mechanisms of NO_x formation for spark ignition engines. NO_x prediction in this work is modelled entirely on thermal NO_x formation, but other mechanisms are reviewed and discussed in section 3.3 to better analyse and validate the model.

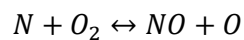
2.1.1 Thermal NO

Thermal NO formation takes place in both the flame front and burned zone during combustion. High pressure experienced by the air/fuel mixture causes a thin flame front ($\approx 0.1\text{mm}$), limiting time spent in this zone for chemical species (Heywood, 1988). Chemical species produced early in the cycle experience increased pressure and temperature in the burned zone as combustion continues. Higher temperatures invite dissociation of the O₂ and N₂, leading to NO formation. Thermal NO is largely a product of the conditions experienced in the burned zone.

Zeldovich (1946) described two reactions responsible for NO production in the burned zone:

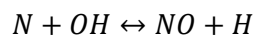


Reaction 1



Reaction 2

Lavoie et al (1970) added a third reaction – because of its significant contribution to overall NO – to what would become known as the extended Zeldovich mechanism:



Reaction 3

The extended Zeldovich mechanism is widely used for ICE NO_x prediction due to its dominant presence in adiabatic combustion where excess oxygen is present (Borman & Ragland, 1998). Introduction of DI engines and lean combustion technologies such as stratified charge has meant that excess oxygen is no longer just a product of lean FER – It is possible for lean combustion to occur locally within the cylinder, both deliberately and not (Zhao, et al., 1999).

Merker et al (2006) note that speed coefficients for the Zeldovich reactions vary from study to study:

Reaction i	$k_{i,r}$ [cm ³ /mol s]	Author
1	$1.8 \cdot 10^4 \exp\left[-\frac{38,400}{T}\right]$	Baulch et al. (1991)
	$0.544 \cdot 10^{14} T^{0.1} \exp\left[-\frac{38,020}{T}\right]$	GRI-MECH 3.0 (2000)
	$0.76 \cdot 10^{14} \exp\left[-\frac{38,000}{T}\right]$	Heywood (1988)
2	$6.4 \cdot 10^9 T \exp\left[-\frac{3,150}{T}\right]$	Baulch et al. (1969)
	$9.0 \cdot 10^9 T \exp\left[-\frac{3,280}{T}\right]$	GRI-MECH 3.0 (2000)
	$1.48 \cdot 10^8 T^{1.5} \exp\left[-\frac{2,860}{T}\right]$	Pattas (1973)
3	$3.0 \cdot 10^{13}$	Baulch et al. (1991)
	$3.36 \cdot 10^{13} \exp\left[-\frac{195}{T}\right]$	GRI-MECH 3.0 (2000)
	$4.1 \cdot 10^{13}$	Heywood (1988)

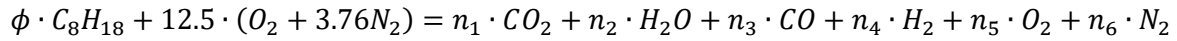
Figure 2-1 - Forward reaction rate coefficients (Merker et al, 2006). Note: Typographic error for Baulch, reaction 1: Should read: $1.8 \cdot 10^{14} \cdot \exp\left(-\frac{38370}{T}\right)$ according to source (Baulch et al, 1991)

Warnatz et al (2006) draw attention to the high activation energy required to break the triple bonded N₂ molecule in the first reaction (Figure 2-1), dissociating atmospheric nitrogen into atomic nitrogen. All thermal NO production is therefore reliant on the ability for this rate-limiting reaction to occur before combustion is complete. Given that combustion reactions are said to take place much faster than the decomposition of N₂ (Stone & Ball, 2004), the reaction's – and thermal NO production – dependence on temperature becomes self-evident, as does its name.

Thermal NO, therefore, is produced when cylinder temperatures are high enough to increase the reaction rate enough for Reaction 1 to take place before combustion ends.

2.2 Dissociation Model

To accurately predict thermal NO it is necessary to model the dissociation of combustion products that occurs at sufficiently high temperatures. This is done via an iterative method outlined by Goodger (2001), for which the atomic balance using isooctane is:



Equation 2-1

Carbon, hydrogen, oxygen and nitrogen can be balanced:

$$Carbon = 8 = n_1 + n_3$$

Equation 2-2

$$Hydrogen = 18 = 2 \cdot n_2 + 2 \cdot n_4$$

$$9 = n_2 + n_4$$

Equation 2-3

$$Oxygen = 25 = 2 \cdot n_1 + n_2 + n_3 + 2 \cdot n_5$$

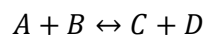
Equation 2-4

$$Nitrogen = 47 = n_6$$

Equation 2-5

Six equations are required to solve six unknowns – two more equations are added by using calculated partial pressure equilibrium constants for CO₂ and H₂O. To grasp this, it is first prudent to review reaction rates.

2.2.1 Reaction Rates



Reaction 4

Reaction rate refers to the production or decomposition rate for a chemical species for a given reaction and is seen to be proportional to the instantaneous concentrations of forming species, raised to a power (Warnatz, et al., 2006). E.g., for Reaction 4:

$$\frac{d[C]}{dt} = k_f \cdot [A]^a \cdot [B]^b$$

Equation 2-6

$$\frac{d[A]}{dt} = -k_r \cdot [C]^c \cdot [D]^d$$

Equation 2-7

Where: square brackets denote an instantaneous molar concentration of each species, k_f , k_r = experimentally determined forward and reverse reactions rate and a, b, c, d = experimentally reaction orders (reaction orders = stoichiometric coefficients for elementary reactions, i.e., reactions that form directly with no intermediate steps involving other species (Warnatz, et al., 2006)).

If k_f and k_r are equal at dynamic equilibrium, where the ratio of reactants and products remains constant despite reactions continuing to take place, this results in a 'dynamic' steady state:

$$\therefore \frac{[C]^c \cdot [D]^d}{[A]^a \cdot [B]^b} = \frac{k_f}{k_r} = K_c$$

Equation 2-8

Where K_c is the concentration equilibrium constant for Reaction 4, typically used for liquids. Dealing with gases, equilibrium constants are defined in terms of pressure (and can be calculated from thermodynamic data, seen later), giving the partial pressure equilibrium constant:

$$K_p = \frac{p_C^c \cdot p_D^d}{p_A^a \cdot p_B^b}$$

Equation 2-9

Where p_A , p_B , p_C and p_D are partial pressures of each species, given by the type reaction:

$$\frac{p_A}{p} = \frac{n_A}{n_t}$$

Equation 2-10

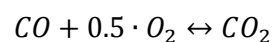
$$p_A = p \cdot \frac{n_A}{n_t}$$

Equation 2-11

Where p = total pressure, n_A is the number of moles for species A and n_t = total number of product moles.

2.2.2 Constructing Useful Equations

In the case of:



Reaction 5

It can be seen that:

$$K_{p(\text{co}_2)} = \frac{p_{\text{CO}_2}}{p_{\text{CO}} \cdot (p_{\text{O}_2})^{0.5}}$$

Equation 2-12

Following the form of Equation 2-11 and referring to Equation 2-1:

$$p_{\text{CO}_2} = p \cdot \frac{n_{\text{CO}_2}}{n_t} = p \cdot \frac{n_1}{n_t}$$

Equation 2-13

$$p_{\text{CO}} = p \cdot \frac{n_{\text{CO}}}{n_t} = p \cdot \frac{n_3}{n_t}$$

Equation 2-14

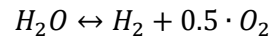
$$p_{\text{O}_2} = p \cdot \frac{n_{\text{O}_2}}{n_t} = p \cdot \frac{n_5}{n_t}$$

Equation 2-15

Equation 2-13, Equation 2-14, Equation 2-15 can be substituted into Equation 2-12 to get:

$$K_{p(\text{co}_2)} = \frac{p \cdot \frac{n_1}{n_t}}{p \cdot \frac{n_3}{n_t} \cdot \left[p \cdot \frac{n_5}{n_t} \right]^{0.5}} = \frac{n_1}{n_3 \cdot \left[p \cdot \frac{n_5}{n_t} \right]^{0.5}}$$

Equation 2-16



Reaction 6

The method can be repeated for $K_{p(\text{H}_2\text{O})}$ (Reaction 6) to get:

$$K_{p(\text{H}_2\text{O})} = \frac{n_2}{n_4 \cdot \left[p \cdot \frac{n_5}{n_t} \right]^{0.5}}$$

Equation 2-17

For use in Goodger's method, each equation is re-arranged to get an expression in terms of unknowns:

$$\frac{n_1}{n_3} = K_{p(\text{co}_2)} \cdot \left[p \cdot \frac{n_5}{n_t} \right]^{0.5}$$

Equation 2-18

$$\frac{n_2}{n_4} = K_{p(\text{H}_2\text{O})} \cdot \left[p \cdot \frac{n_5}{n_t} \right]^{0.5}$$

Equation 2-19

Equations for carbon and hydrogen must be represented in terms of $\frac{n_1}{n_3}$ and $\frac{n_2}{n_4}$ to give expressions that can be calculated by known values. This is done by dividing Equation 2-2 and Equation 2-3 by n_3 and n_4 respectively:

$$\frac{8}{n_3} = \frac{n_1}{n_3} + \frac{n_3}{n_3}$$

$$8 = n_3 \cdot \left(\frac{n_1}{n_3} + 1 \right)$$

Equation 2-20

$$\frac{9}{n_4} = \frac{n_2}{n_4} + \frac{n_4}{n_4}$$

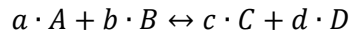
$$9 = n_4 \cdot \left(\frac{n_2}{n_4} + 1 \right)$$

Equation 2-21

Values for $K_{p(\text{CO}_2)}$ and $K_{p(\text{H}_2\text{O})}$ must be sought before the method can be outlined and can be calculated from Gibbs free energy values, in keeping with the approach from first principles.

2.2.3 Calculating Partial Pressure Equilibrium Constants (K_p)

Glassman and Yetter (2011) outline the method for calculation of partial pressure equilibrium constant (K_p) by considering the change in free energy for an example equilibrium reaction:



Reaction 7

Where A, B, C, D are substances of some combustion product and a, b, c, d are stoichiometric coefficients given by the molecular balance. Change in Gibbs free energy is calculated by:

$$-\Delta G^\circ = a \cdot G_A^\circ + b \cdot G_B^\circ - c \cdot G_C^\circ - d \cdot G_D^\circ$$

Equation 2-22

ΔG° is the standard state free energy change and G_X° denotes Gibbs free energy of formation for species X , determined experimentally (Rogers & Mayhew, 1992) and published in temperature dependent thermodynamic tables.

The partial pressure equilibrium constant is then described by the Arrhenius form:

$$K_p = \exp \left(\frac{-\Delta G^\circ}{\tilde{R}T} \right)$$

Equation 2-23

Partial pressure equilibrium constants can be calculated at suitable temperatures, relative to the boundaries of study. In this case, eleven temperature sites were required, ranging from 2000K to 3000K at 100K increments. A MATLAB script (Appendix E – MATLAB Script for Dissociation Model) was created to work out K_p at each temperature for Reaction 5 and Reaction 6, using values from Table 2-1. Table 2-2 shows the standard state free energy changes ($-\Delta G^\circ$) and partial pressure equilibrium constants (K_p) for both reactions.

Table 2-1 - Gibbs Energy of Formation (Glassman & Yetter, 2011)

Temperature (Kelvin)	$\Delta_f G^\circ_{CO}$ (KJ/mol)	$\Delta_f G^\circ_{CO_2}$ (KJ/mol)	$\Delta_f G^\circ_{O_2}$ (KJ/mol)	$\Delta_f G^\circ_{H_2O}$ (KJ/mol)	$\Delta_f G^\circ_{H_2}$ (KJ/mol)	$\Delta_f G^\circ_O$ (KJ/mol)
2000	-286.034	-396.333	0	-135.528	0	121.552
2100	-294.372	-396.304	0	-129.721	0	114.86
2200	-302.672	-396.262	0	-123.905	0	108.159
2300	-310.936	-396.209	0	-118.082	0	101.45
2400	-319.164	-396.142	0	-112.252	0	94.734
2500	-327.356	-396.062	0	-106.416	0	88.012
2600	-335.514	-395.969	0	-100.575	0	81.284
2700	-343.638	-395.862	0	-94.729	0	74.551
2800	-351.729	-395.742	0	-88.878	0	67.814
2900	-359.789	-395.609	0	-83.023	0	61.072
3000	-367.816	-395.461	0	-77.163	0	54.327

Table 2-2 - Calculated Partial Pressure Equilibrium Constant Results & Goodger's Published Values

T (Kelvin)	CO_2		H_2O		Goodger	
	$-\Delta G^\circ$ (KJ/mol)	K_p	$-\Delta G^\circ$ (KJ/mol)	K_p	$K_{p(CO_2)}$	$K_{p(H_2O)}$
2000	110.299	760.0083	135.528	3465.421	765.6	3467
2100	101.932	343.1715	129.721	1685.581	345.94	1686.6
2200	93.59	166.7974	123.905	874.9534	168.27	874.98
2300	85.273	86.4332	118.082	480.6437	87.097	480.84
2400	76.978	47.36342	112.252	277.453	47.753	277.43
2500	68.706	27.26372	106.416	167.3096	27.543	167.49
2600	60.455	16.39078	100.575	104.8696	-	-
2700	52.224	10.24166	94.729	68.03131	10.351	68.077
2800	44.013	6.623724	88.878	45.50921	-	-
2900	35.82	4.417852	83.023	31.2939	-	-
3000	27.645	3.029421	77.163	22.05849	3.0549	22.029

Calculated values for K_p show very good agreement with values published by Goodger (2001), for use with the same method. Small discrepancies between the values can easily be attributed to different sources of the experimental data from which Gibbs free energy is recorded or – more likely – rounding errors. That some temperature sites are missing from Goodger's data illustrates the necessity in creating an equilibrium constant calculated as not to be constrained by provided temperatures.

Increments of 100K allow more accurate interpolation between known sites. From first principles, the equilibrium constant calculator can attain values at any temperature for which the Gibbs energy of formation is available.

With six equations for six unknowns and K_p values for all required temperature sites, Goodger's method can now be approached.

2.2.4 Goodger's Dissociation Model

Goodger (2001) describes the dissociation model method in the following steps:

1. Values for temperature and pressure are selected.
2. A value for $\frac{n_5}{n_t}$ is assumed (guessed).
3. Selected pressure value is used to calculate:

$$\left(p \cdot \frac{n_5}{n_t}\right)^{0.5}$$

Equation 2-24

4. Values for $\frac{n_1}{n_3}$ and $\frac{n_2}{n_4}$ are calculated using Equation 2-18 and Equation 2-19, along with calculated values for $K_{p(\text{CO}_2)}$ and $K_{p(\text{H}_2\text{O})}$ at the selected temperature.
5. n_3 can now be worked out by re-arranging Equation 2-20:

$$n_3 = \frac{8}{\left(\frac{n_1}{n_3} + 1\right)}$$

Equation 2-25

6. n_1 is attained simply:

$$n_1 = 8 - n_3$$

Equation 2-26

7. n_4 is calculated by re-arranging equation Equation 2-21:

$$n_4 = \frac{9}{\left(\frac{n_2}{n_4} + 1\right)}$$

Equation 2-27

8. n_2 is approached via:

$$n_2 = 9 - n_4$$

Equation 2-28

9. n_5 can be calculated manipulating the oxygen atomic balance - Equation 2-4:

$$n_5 = \frac{25 - 2 \cdot n_1 - n_2 - n_3}{2}$$

Equation 2-29

10. n_6 can be calculated using the nitrogen atomic balance - Equation 2-5.

11. n_t is calculated by adding all calculated products.

12. $n_t = \frac{\text{Calculated } n_5}{\text{Assumed } \frac{n_5}{n_t}}$ is calculated and compared with the sum of products worked out

$$(\sum n_j = n_1 + n_2 + n_3 + n_4 + n_5 + n_6).$$

13. Steps 2-9 are repeated until $n_t = \sum n_j$. At this point, the iterative method is complete and correct values have been found for all products.

2.2.5 Adaptation of the Goodger Method

The Microsoft Excel software package lends itself well to this kind of iterative process. Initial attempts at the model were made using K_p values published by Goodger. Steps 1-8 of the Goodger method are followed but the final steps are adapted to better suit the spreadsheet program:

9. A new $\frac{n_5}{n_t}$ value is created using calculated values from steps 6 and 8.

10. 'Calculated' $\frac{n_5}{n_t}$ is subtracted from 'assumed' $\frac{n_5}{n_t}$, termed 'error'. For the correct values, error will approximate zero.

11. The original 'guess' value is added to the guess and steps 2-10 are repeated until step 10 yields a negative result. At this point, the closest positive value to zero will be correct and the iterative method is finished.

Modification of Goodger's steps is necessary for the spreadsheet method. Since an absolute 'zero' is unlikely to be found, questions arise as to the definition of zero. These modifications allow location of the lowest positive 'error' value – I.e., closest to zero (Figure 2-2).

	n1	5.909062104					Closest +VE to zero	0.00000
	n2	8.928079211					Closest -VE to zero	-0.00000
	n3	2.490937896						
	n4	0.521920789						
	n5	0.881429342						
	n6	47						
	nt	65.73142934						
10	11	12	13	14				
n4	n2	n5	n6	nt (Calculated)	Calculated n5/nt Error	Error	Error	Error
							+VE Values	-VE Values
6.440375899	3.0096241	6.53957504	47	71.39	0.091604062	0.091594	0.091594062	
5.689789086	3.7602109	6.067292099	47	70.92	0.085554481	0.085534	0.085534481	
0.523213605	8.9267864	0.884371389	47	65.73	0.013453713	0.000124	0.000123713	
0.523028317	8.9269717	0.88394991	47	65.73	0.013447388	0.000107	0.000107388	
0.52284323	8.9271568	0.883528827	47	65.73	0.013441068	9.11E-05	9.10679E-05	
0.522658343	8.9273417	0.883108141	47	65.73	0.013434754	7.48E-05	7.4754E-05	
0.522473655	8.9275263	0.88268785	47	65.73	0.013428446	5.84E-05	5.8446E-05	
0.522289168	8.9277108	0.882267953	47	65.73	0.013422144	4.21E-05	4.21438E-05	
0.522104879	8.9278951	0.881848451	47	65.73	0.013415847	2.58E-05	2.58474E-05	
0.521920789	8.9280792	0.881429342	47	65.73	0.013409557	9.56E-06	9.55691E-06	
0.521736897	8.9282631	0.881010626	47	65.73	0.013403272	-6.7E-06	-6.72781E-06	
0.521553204	8.9284468	0.880592303	47	65.73	0.013396993	-2.3E-05	-2.30067E-05	
0.521369708	8.9286303	0.88017437	47	65.73	0.01339072	-3.9E-05	-3.92799E-05	
0.521186409	8.9288136	0.879756829	47	65.73	0.013384453	-5.6E-05	-5.55472E-05	
0.521003307	8.9289967	0.879339678	47	65.73	0.013378191	-7.2E-05	-7.18087E-05	
0.520820402	8.9291796	0.878922916	47	65.73	0.013371935	-8.8E-05	-8.80645E-05	

Figure 2-2 - Definition of zero: Showing the crossover from positive to negative

The initial 'guess' itself is problematic at low temperatures at the rich end of FERs used in the Ford Fox data – where atmospheric oxygen (n_5) levels are extremely low. Values used for lean calculations – initial guess ≈ 0.0001 – result in no positive 'error' values (step 10) at lower temperatures, for rich combustion. This can be seen in comparing the stoichiometric results of Figure 2-4 to that of $\phi = 1.05$ in Figure 2-5, where both use the same initial guess.

While low oxygen presence during rich combustion is in line with expectation, dissociation takes place and NO_x is still produced (Karvountzis-Kontakiotis & Ntziachristos, 2012), albeit at low quantities. Attempts at using smaller initial guesses to calculate missing results proved unwieldy, creating disproportionate demands on computing power. Coupled with the desire to work out results for 1 bar increments to improve accuracy of final NO_x prediction, another method was sought.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1																							
2		Carbon	8.00 a			Temperature	3000				n1	4.957526038				Closest +VE to zero		0.0000008988					
3		Hydrogen	18.00 b			K _{CO2}	3.0549				n2	8.294114729				Closest -VE to Zero		-0.0000123871					
4		X	12.5 c			K _{H2O}	22.029				n3	3.042473962											
5		N2	47			K _{N2}	7.211				n4	0.705885271											
6											n5	1.874179616											
7											n6	47											
8											nt	65.87417962											
9																							
10	1	2	3	4	5	6	7	8	9	10	11	12	13	14									
11	Error	Guess n5/nt	Pressure	(n5/nt)*P	((n5/nt)*P)^0.5	n1/n3	n3	n1	n2/n4	n4	n2	n5	n6	nt (Calculated)	Calculated n5/nt	Error	Error	Error					
12																	+VE Value	-VE Values					
13	0.10575	0.00001	10	0.0001	0.01	0.030549	7.762852615	0.24		0.22029	7.375296036	1.624704	7.569074326	47	71.57	0.105759008	0.10575	0.10575					
14	0.10193	0.00002	10	0.0002	0.014142136	0.04320281	7.668690999	0.33		0.311537106	6.862177182	2.1378228	7.265434091	47	71.27	0.101948921	0.10193	0.10193					
15	0.09927	0.00003	10	0.0003	0.017320508	0.05291242	7.597972868	0.40		0.381553472	6.514405834	2.4855942	7.056189351	47	71.06	0.099304359	0.09927	0.09927					
16	0.0972	0.00004	10	0.0004	0.02	0.061098	7.539360172	0.46		0.44058	6.247483652	2.7525163	6.893421912	47	70.89	0.097236411	0.0972	0.0972					
17	0.09547	0.00005	10	0.0005	0.02236068	0.068309641	7.488465606	0.51		0.492583415	6.029813752	2.9701862	6.759139679	47	70.76	0.095523203	0.09547	0.09547					
18	0.09399	0.00006	10	0.0006	0.024494897	0.074829462	7.443041228	0.56		0.539598095	5.845681432	3.1543186	6.64436133	47	70.64	0.094053668	0.09399	0.09399					
19	0.09269	0.00007	10	0.0007	0.026457513	0.080825057	7.401752901	0.60		0.582832556	5.686008898	3.3139911	6.5438809	47	70.54	0.092763268	0.09269	0.09269					
20	0.09153	0.00008	10	0.0008	0.028284271	0.08640562	7.363732156	0.64		0.623074211	5.545032961	3.454967	6.454382558	47	70.45	0.091610803	0.09153	0.09153					
21	0.09048	0.00009	10	0.0009	0.03	0.091647	7.328376297	0.67		0.66087	5.418846749	3.5811533	6.373611523	47	70.37	0.090568203	0.09048	0.09048					
22	0.08952	0.0001	10	0.001	0.031622777	0.09660442	7.295246902	0.70		0.696618146	5.304670366	3.6953296	6.299598634	47	70.30	0.089615396	0.08952	0.08952					
23	0.08863	0.00011	10	0.0011	0.033166248	0.101319571	7.264013292	0.74		0.730619275	5.200450573	3.7995494	6.232231932	47	70.23	0.088737489	0.08863	0.08863					
24	0.0878	0.00012	10	0.0012	0.034641016	0.10582484	7.234418788	0.77		0.763106945	5.10462512	3.8953749	6.169521954	47	70.17	0.087923101	0.0878	0.0878					
25	0.08703	0.00013	10	0.0013	0.036055513	0.110145986	7.206259448	0.79		0.79426689	5.015976189	3.9840238	6.111117819	47	70.11	0.08716332	0.08703	0.08703					
26	0.08631	0.00014	10	0.0014	0.037416574	0.114303892	7.179370063	0.82		0.824249706	4.933535125	4.0664649	6.056452594	47	70.06	0.086451032	0.08631	0.08631					
27	0.08563	0.00015	10	0.0015	0.038729833	0.118315768	7.153614594	0.85		0.853179501	4.856518213	4.1434818	6.005066403	47	70.01	0.085780454	0.08563	0.08563					
28	0.08499	0.00016	10	0.0016	0.04	0.122196	7.128879447	0.87		0.88116	4.784282039	4.215718	5.956580743	47	69.96	0.085146825	0.08499	0.08499					
29	0.08438	0.00017	10	0.0017	0.041231056	0.125956754	7.105068621	0.89		0.908278938	4.716291638	4.2837084	5.91068013	47	69.91	0.084546168	0.08438	0.08438					
30	0.0838	0.00018	10	0.0018	0.042426407	0.12960843	7.08210012	0.92		0.934611317	4.652097256	4.3479027	5.867098688	47	69.87	0.08397513	0.0838	0.0838					
31	0.08324	0.00019	10	0.0019	0.043588989	0.133160004	7.059903255	0.94		0.960221848	4.591317053	4.4086829	5.825610154	47	69.83	0.083430852	0.08324	0.08324					
32	0.08271	0.0002	10	0.002	0.04472136	0.136619281	7.038416585	0.96		0.98516683	4.533624009	4.466376	5.786020297	47	69.79	0.082910879	0.08271	0.08271					
33	0.0822	0.00021	10	0.0021	0.045825757	0.139993105	7.017586304	0.98		1.0094956	4.478735858	4.5212641	5.748161081	47	69.75	0.082413084	0.0822	0.0822					
34	0.08172	0.00022	10	0.0022	0.046904158	0.143287511	6.997364987	1.00		1.033251688	4.426407244	4.5735928	5.711886116	47	69.71	0.081935613	0.08172	0.08172					
35	0.08125	0.00023	10	0.0023	0.047958315	0.146507857	6.977710575	1.02		1.056473726	4.376423528	4.6235765	5.677067052	47	69.68	0.081476837	0.08125	0.08125					
36	0.0808	0.00024	10	0.0024	0.048989795	0.149658924	6.958585569	1.04		1.079196191	4.328595848	4.6714042	5.643590709	47	69.64	0.081035321	0.0808	0.0808					
37	0.08036	0.00025	10	0.0025	0.05	0.152745	6.939956365	1.06		1.10145	4.282757144	4.7172429	5.611356754	47	69.61	0.080609789	0.08036	0.08036					
38	0.07994	0.00026	10	0.0026	0.050990195	0.155769947	6.921792715	1.08		1.123263009	4.238758912	4.7612411	5.580275813	47	69.58	0.080199105	0.07994	0.07994					
39	0.07953	0.00027	10	0.0027	0.051961524	0.15873726	6.904067275	1.10		1.144660417	4.196468554	4.8035314	5.550267915	47	69.55	0.079802251	0.07953	0.07953					
	Totals		10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100		

Figure 2-3 - Dissociation Spreadsheet Layout in MS Excel

[illegible]

[illegible]

Figure 2-5 - Rich Results Using Initial Guess=0.0001

The Goodger method was implemented in a MATLAB script (Appendix E – MATLAB Script for Dissociation Model). It was believed that MATLAB's loop functions would require less processing power, thus relieving data collection of the bottleneck. One bar pressure increments could also now be calculated, for greater accuracy.

2.2.6 Results

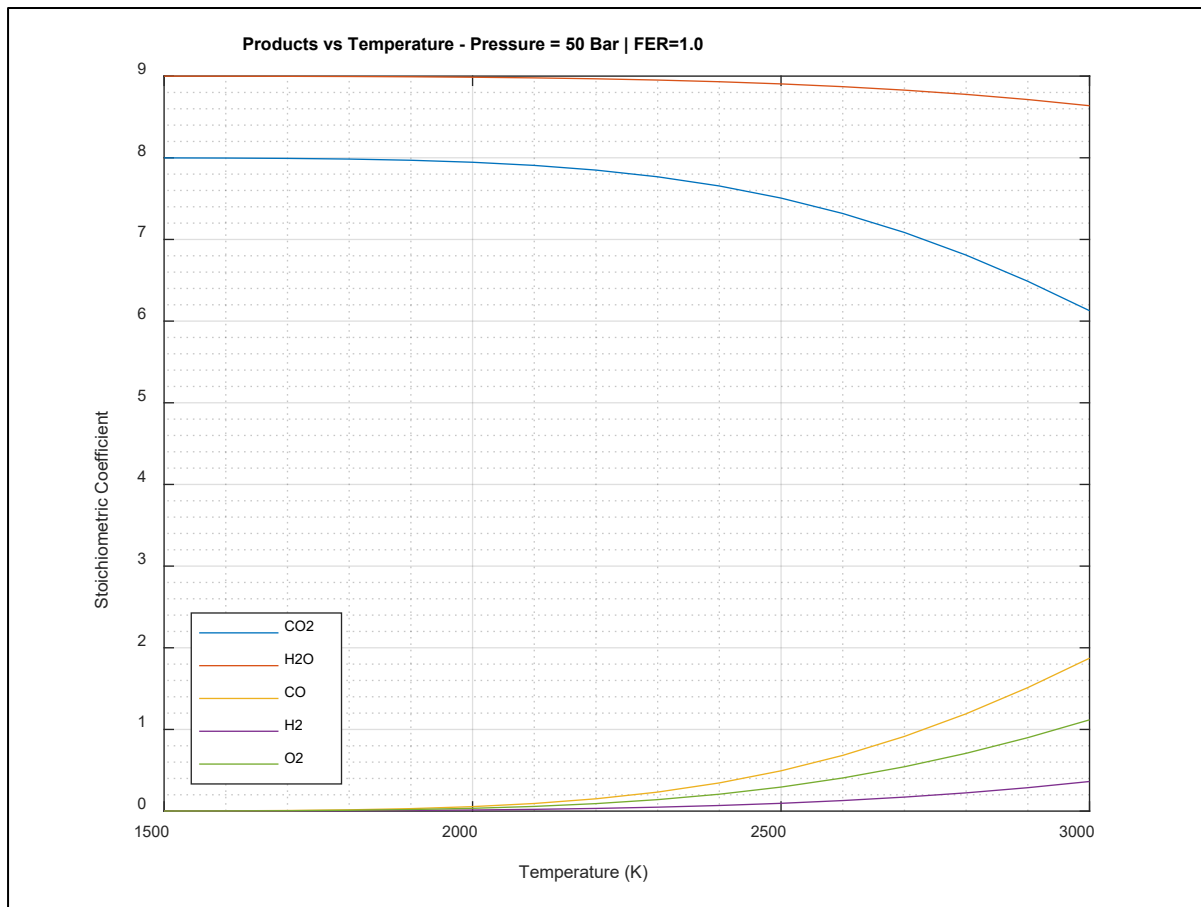


Figure 2-6 - Effect of temperature on dissociation of combustion products (moles)

Results for stoichiometric dissociation find agreement with Heywood's (1988) assertion of beginning at around 1800K. For CO_2 and H_2O , the relationship can now be seen between $-\Delta G$ and extent of dissociation: Higher Gibbs free energy (Table 2-2) represents a higher resistance to decomposition into constituent species. H_2O 's inversely proportional dissociation to H_2 (+O) is also evident in this six-species model.

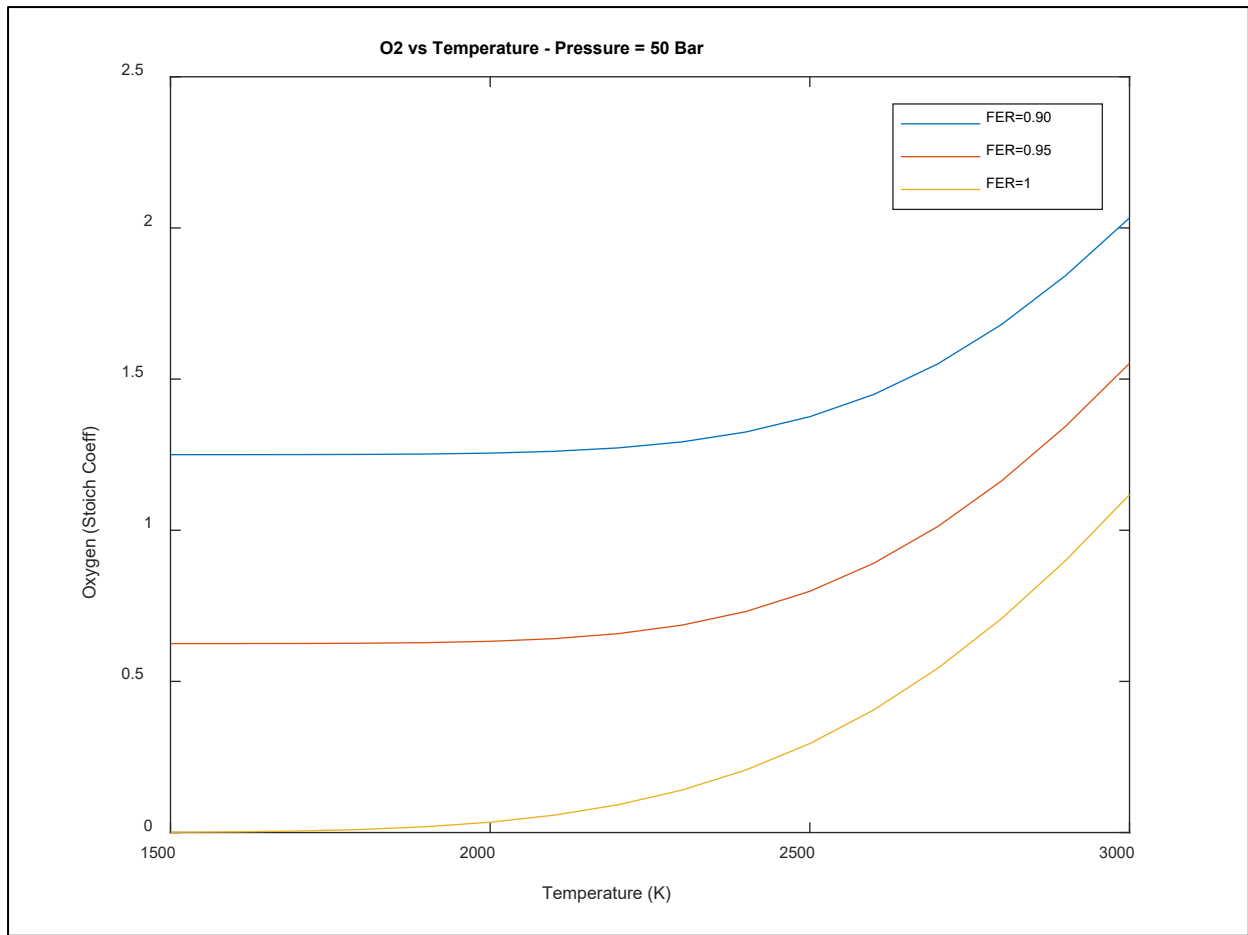


Figure 2-7 - Effect of temperature and FER on oxygen (moles) dissociation

As expected, leaner conditions result in higher levels of O_2 because combustion is performed with more air than required for complete oxidation of the fuel. This free O_2 is joined by products of dissociation from CO_2 and H_2O at temperatures $> 1800K$ – note the number of O_2 remains approximately constant up to this point.

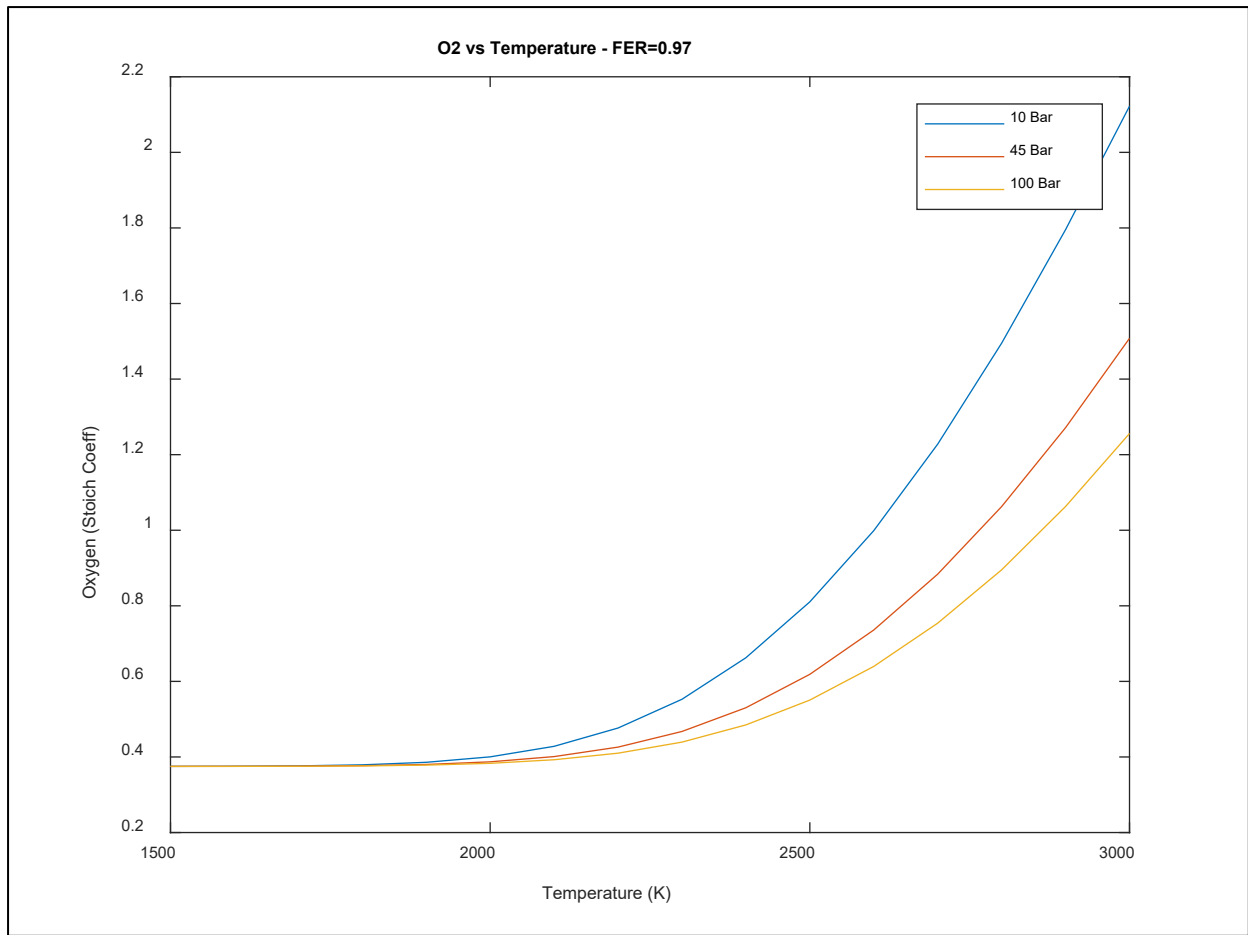
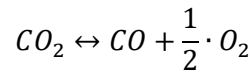


Figure 2-8 - Effect of pressure on dissociated oxygen levels (moles)

The effect of higher pressure on the decomposition of exhaust products that contribute to oxygen levels appears counter-intuitive. Oxygen levels decrease as pressure increase, revealing lower rates of dissociation. To make sense of this, the effect of pressure on product dissociation must be divorced from pressure effect on cylinder temperature and Le Chatelier's principle must be considered.

Chemical equilibrium describes the condition of a given reaction in which forward and reverse reaction rates equalize, resulting in no net change in species concentration for reactants and products. Clarification: concentrations of reactants do not have to be equal to concentrations of products for a reaction to be in equilibrium, and often differ.

Le Chatelier's principle states that the position of equilibrium in a system will move to minimize the effect of change if the system is disturbed. In the case of an increase in pressure, position of equilibrium will move in an attempt to reduce pressure. To illustrate this, the reaction describing dissociation of carbon dioxide is considered:



Equation 2-30

Cylinder pressure is described in equation Equation 2-31. Force in this equation is the result of gas particle collision – the same collision that can cause chemical bonds to form if conditions are favourable. It has been shown that total pressure in a system is the sum of partial pressures exerted by each species (Section 2.2.1).

$$p = \frac{\text{Force}}{\text{Area}} = \frac{F}{A}$$

Equation 2-31

Doubling cylinder pressure doubles the partial pressure of each species in Equation 2-30, causing collisions of CO and O to increase, thus increasing the reaction rate. Equilibrium shifts to the left of the equation as CO_2 concentration increases and CO and O_2 concentration decreases, because increasing pressure (divorced from temperature) has no direct effect on the decomposition of CO_2 (Myers, 2003). This effect seeks to reduce pressure once again to minimize the stress of the pressure increase, by turning two partial pressures into one.

As pressure increases, then, dissociation decreases. This relationship is most notable in stoichiometric mixture which experiences most dissociation (Glassman & Yetter, 2011) (which is also the reason adiabatic flame temperature peaks just rich of stoichiometric, as dissociation of products causes less heat release (Law, et al., 2006)). Given that pressure variances at instances of stoichiometric combustion are small (Table 2-3), it is unlikely the effect of Le Chatelier's principle will be observed in final NO_x predictions.

Table 2-3 - Maximum pressures from Ford Fox data

RPM	FER	Max Pressure
1000	0.9	54.16
1500	0.91	74.32
1750	0.97	79.50
2000	1.01	81.47
2500	1.01	81.17
3000	1	81.29
3500	1.06	86.48
4000	1.03	82.17
4500	1.05	82.30
5000	1.09	75.41
5500	1.09	77.58
6000	1.11	78.73
6500	1.11	82.48

2.2.7 Excel Integration

In-built functions were used to organise the data (Table 2-5) and create pressure-based polynomial fits as a function of temperature, to be used with the Ford Fox data. These polynomials were imported back into the original Excel combustion model, where lookup functions were used to ensure correct use, based on FER, temperature and pressure for every degree, using simple interpolation between single bar pressure values.

Table 2-4 - Comparison of practical methods for modelling dissociation

Features	Excel Method	MATLAB Method
Pressure range	10-100 bar	10-100 bar
Pressure increments	5 bar	1 bar
Temperature range	1500-3000K	2000-3000K
Temperature increments	Varied: 100K to 500K	100K
FER Range	$\phi = 0.9$ to $\phi = 1.02$	$\phi = 0.9$ to $\phi = 1.11$
K_p values source	Goodger (2001)	Calculated (Glassman & Yetter, 2011)

Table 2-5 - Showing result format used for polynomial Fit ($\phi = 1$)

Temp/ ^{Pressure}	10	11	12	13	14	15	...	100
2000	0.064473	0.062476	0.060705	0.05912	0.057689	0.056388	...	0.030072
2100	0.107816	0.104498	0.101553	0.098916	0.096537	0.094371	...	0.050465
2200	0.171386	0.166158	0.161517	0.157363	0.153611	0.150194	...	0.080626
2300	0.260288	0.252454	0.2455	0.239264	0.233629	0.228497	...	0.12335
2400	0.379228	0.368032	0.358079	0.349148	0.341068	0.333704	...	0.181543
2500	0.531689	0.516391	0.502778	0.490542	0.479457	0.469342	...	0.257987
2600	0.719123	0.699139	0.681316	0.665269	0.650708	0.6374	...	0.355095
2700	0.940143	0.915173	0.892845	0.872694	0.854367	0.837589	...	0.474587
2800	1.189945	1.160128	1.133373	1.109154	1.087069	1.066802	...	0.617177
2900	1.46024	1.426259	1.395636	1.367811	1.342349	1.318912	...	0.782353
3000	1.739635	1.702764	1.669354	1.638852	1.610822	1.584918	...	0.968161

3 Thermal NO_x Model

3.1 O₂ and N₂ Concentrations

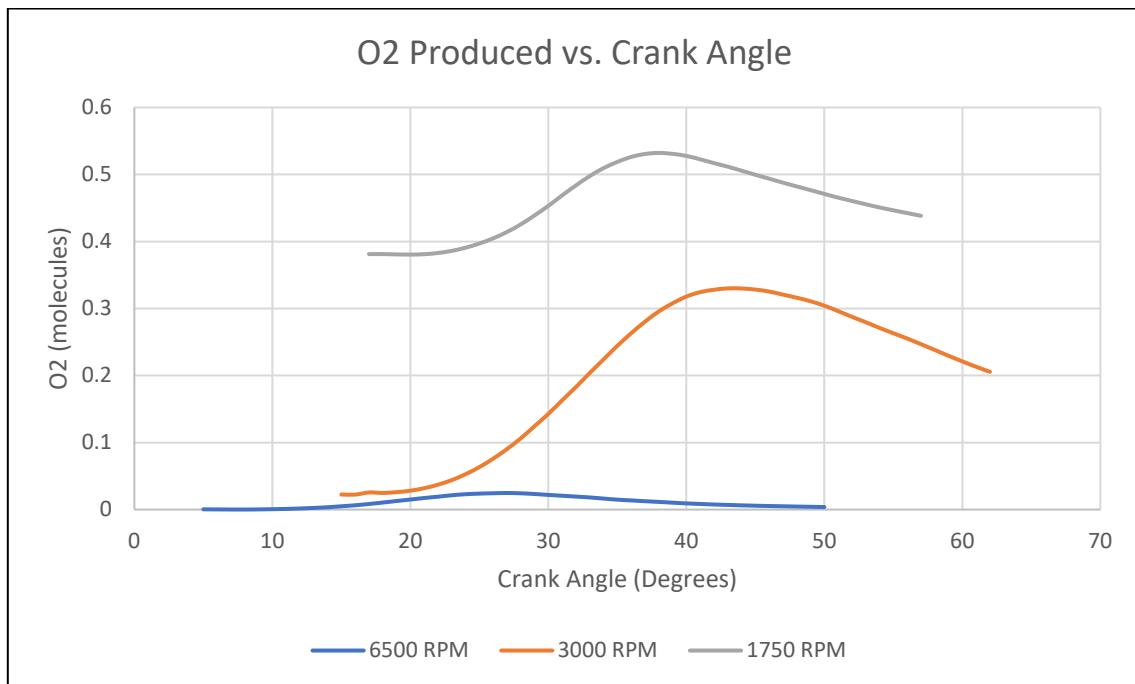


Figure 3-1 - O₂ produced at every degree (polynomial calculation)

The n_5 polynomials can now be used with the Ford Fox temperature and pressure data to calculate how much oxygen is produced at every degree. Figure 3-1 shows three such results, illustrating (along with Table 3-1) the significant impact FER has on oxygen formation (showing observance with Figure 3-7) during combustion with Figure 3-2 showing very similar temperature ranges for the three results.

Table 3-1 - Companion FER data for chart

RPM	ϕ
6500	1.11
3000	1.0
1750	0.97

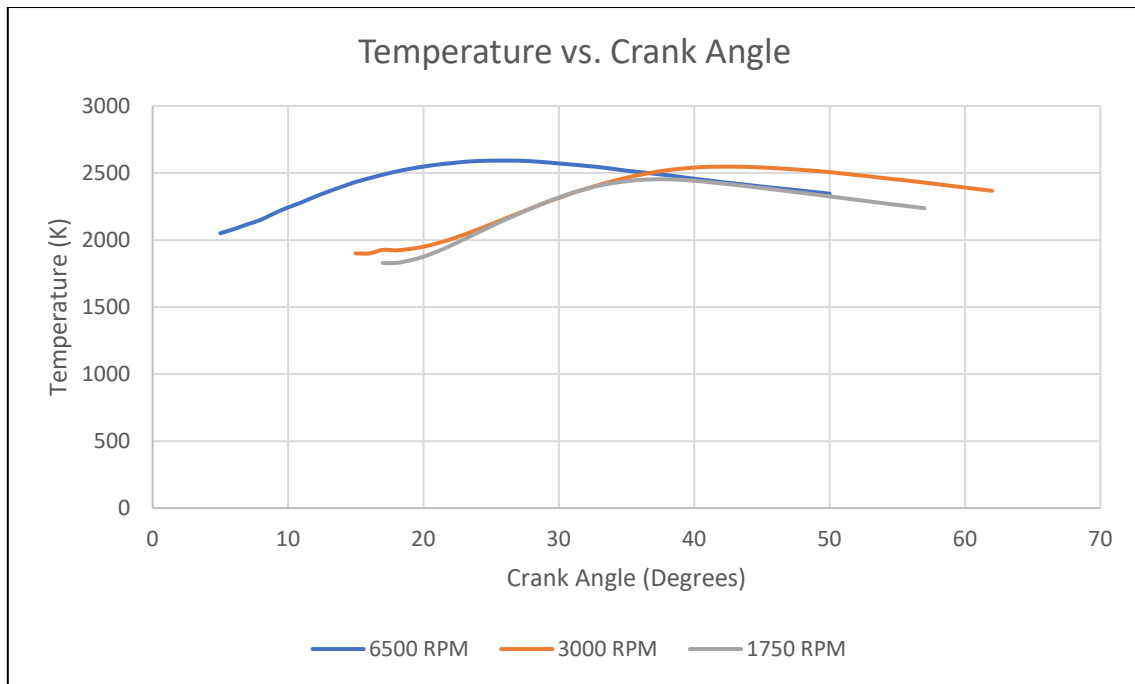


Figure 3-2 - Matching temperature data

The calculated oxygen produced so far does not take the actual cylinder mass into consideration. A multiplying constant must be constructed:

$$C_8H_{18} = \phi \cdot 114 \text{ kg/kmol}$$

Equation 3-1

$$\alpha \cdot \phi \cdot 114 = m_{fuel} \text{ kg}$$

Equation 3-2

Where m_{fuel} describes the mass of fuel in the cylinder. Since m_{fuel} is known for each RPM, α can be calculated:

$$\alpha = \frac{m_{fuel}}{\phi \cdot 114} = \frac{\text{kg}}{\text{kg/kmol}} = \text{kmol}$$

Equation 3-3

That the units of α become kmol allows use of the concentration equation:

$$c = \frac{n}{V}$$

Where c is molar concentration, n is the number of moles of substance and V is volume.

Concentrations of O_2 and N_2 can now be calculated at every degree, through inclusion of α , x (MFB) and the calculated coefficient values, which are unitless (Fuller & Harb, 2016):

$$[O_2] = \frac{\alpha \cdot x \cdot O_2}{V_B} = \frac{kmol}{m^3}$$

Equation 3-4

$$[N_2] = \frac{\alpha \cdot x \cdot N_2}{V_B} = \frac{kmol}{m^3}$$

Equation 3-5

Now that concentration values have been established, the extended Zeldovich mechanism must be reviewed.

3.2 Extended Zeldovich Mechanism

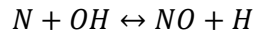
It has been seen (p27) that rates of reactant or product formation can be described in terms of instantaneous concentration of contributing species and experimentally determined reaction rates. For the extended Zeldovich mechanism:



Reaction 8



Reaction 9



Reaction 10

Rate of NO formation can therefore be written as:

$$\begin{aligned} \frac{d[NO]}{dt} = & k_{f(1)}[O][N_2] + k_{f(2)}[N][O_2] + k_{f(3)}[N][OH] - k_{r(1)}[NO][N] - k_{r(2)}[NO][O] \\ & - k_{r(3)}[NO][H] \end{aligned}$$

Equation 3-6

A companion equation can be created for the formation of atomic N:

$$\begin{aligned} \frac{d[N]}{dt} = & k_{f(1)}[O][N_2] - k_{f(2)}[N][O_2] - k_{f(3)}[N][OH] - k_{r(1)}[NO][N] + k_{r(2)}[NO][O] \\ & + k_{r(3)}[NO][H] \end{aligned}$$

Equation 3-7

A quasi-steady state is said to occur when a highly reactive species is consumed at approximately the same rate it is produced (Warnatz, et al., 2006). Figure 2-1 shows that forward reaction rates for Reaction 2 and Reaction 3 are several orders of magnitude faster than Reaction 1, immediately converting the atomic nitrogen (Merker, et al., 2006). In such a state, net change of the species production over time is approximately zero after an initial interval to reach steady state:

$$\frac{d[N]}{dt} \approx 0$$

Equation 3-8

Zeldovich's (1946) mechanism calls for combustion and NO formation to be decoupled since the process is slow when compared to fuel oxidation reactions, even at the high temperatures that speed up NO production (Weisser, 2001). Now decoupled from

combustion, a state of equilibrium in the burned zone can be assumed, allowing product species concentrations to be calculated using partial pressure equilibrium coefficients. Table 2-2 goes some way to validating the assumption of equilibrium in the burned zone, revealing the low activation energy required to decompose H_2O and CO_2 into constituent species'.

Work undertaken by Miller and Bowman (1989), however, found that superequilibrium concentrations of the O radical could be formed at the flame front and would not immediately reconstitute, due to entry into the lower temperature burned zone. This resulted in a significant rise in the rate of NO formation over calculated methods involving the equilibrium assumption, suggesting that the two processes are not so easily separated. Warnatz et al (2006) expands on this, suggesting that use of equilibrium-based concentrations for the O radical – obtained from thermodynamic data – under predicts $[O]$ by a factor of up to 10.

From this point, methods from Heywood (1988) and Merker et al (2006) diverge in acquiring what is, by the end, the same expression for rate of NO production. Merker's method is favoured here for the sake of brevity. From Equation 3-7 and the quasi-steady state approximation of $\frac{d[N]}{dt}$:

$$0 = k_{f(1)}[O][N_2] - k_{f(2)}[N][O_2] - k_{f(3)}[N][OH] - k_{r(1)}[NO][N] + k_{r(2)}[NO][O] + k_{r(3)}[NO][H]$$

Equation 3-9

This is added to Equation 3-6 resulting elimination of all terms except:

$$\frac{d[NO]}{dt} = 2k_{f(1)}[O][N_2] - 2k_{r(1)}[NO][N]$$

Equation 3-10

Reverse reactions are now neglected because instantaneous NO concentration is said to be less than the equilibrium concentration for a given temperature during the clear majority of the combustion cycle. Only when instantaneous NO concentration is greater than the equilibrium concentration does the reverse reaction have significant effect on rate of NO production. This occurs only near the end of the expansion stroke, when temperatures are too low for NO formation to take place (Merker, et al., 2006).

The expression for rate of NO production, shared by Merker and Heywood, is then:

$$\frac{d[NO]}{dt} = 2k_{f(1)}[O][N_2]$$

Equation 3-11

Rate reactions and O radical concentrations are expressed differently for both methods, due to variance in experimental data used and unit conversions:

Table 3-2 - NO formation attributes

Attribute	Merker et al	Heywood
$[O] =$	$0.13[O_2]^{0.5} \cdot \exp\left(-\frac{29468}{T}\right)$	$\frac{K_{p(o)}[O_2]^{0.5}}{(\tilde{R}T)^{0.5}}$
$k_{f(1)} =$	$1.8 \cdot 10^{11} \cdot \exp\left(-\frac{38370}{T}\right)$	$7.6 \cdot 10^{13} \cdot \exp\left(-\frac{38000}{T}\right)$
$K_{p(o)} =$	--	$3.6 \cdot 10^3 \cdot \exp\left(-\frac{31090}{T}\right)$

Finally, from Merker et al (kmol/m³):

$$\frac{d[NO]}{dt} = 4.7 \cdot 10^{13} \cdot [N_2][O_2]^{0.5} \cdot \exp\left(-\frac{67837}{T}\right)$$

Equation 3-12

Merker et al points out that this equation is expected to overpredict NO concentration since reverse reactions of the mechanism act to slow down NO production, perhaps contradicting earlier observations. Alternatively, Heywood derives a very different expression prior to Equation 3-11 to nullify all other expressions by assuming instantaneous NO concentration is much smaller than equilibrium NO concentration, so that $\frac{[NO]}{[NO]_e}$ is approximately zero for the initial value of rate of NO production.

Heywood, unlike Merker, is not apologetic in neglecting reverse reactions and cites validation of the method with measured data collected from high pressure combustion bomb experiments, akin to the ICE.

Heywood's final equation (mol/cm³):

$$\frac{d[NO]}{dt} = \frac{6 \cdot 10^{16}}{T^{0.5}} \cdot \exp\left(-\frac{69090}{T}\right) \cdot [O_2]^{0.5}[N_2]$$

Equation 3-13

Heywood disregards Warnatz's caution against assuming equilibrium of the O radical (Table 3-2) to calculate atomic oxygen concentration and cites validation with measured data, albeit not from an ICE. Merker's method of calculating $[O]$ is more difficult to approach without access to the source (Bockhorn, 1997).

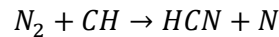
With equations now derived, it is prudent to discuss findings in the literature associated with implementation of the extended Zeldovich mechanism before continuing.

3.3 Literature Review

Three mechanisms of NO formation are agreed upon in the literature:

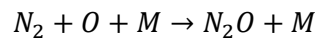
Thermal NO: Already discussed.

Prompt NO: Produced at the flame front as a result of atmospheric nitrogen reacting with the CH radical (a transient species created during fuel oxidation):

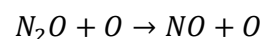


(Fenimore, 1979)

NO via N₂O: The result of oxidation of atmospheric nitrogen, but differs from thermal NO through the presence of any third molecule (*M*):



Nitrous oxide can then further react with O radicals:



(Malte & Pratt, 1974)

NO formation is dominant only in the absence of thermal and prompt NO. I.e., when low combustion temperatures do not allow thermal NO to form and lean combustion reduces occurrences of the CH radical. Three body reactions are characterized by sensitivity to increased pressures (which increases molecule collision rates) and low activation energy (allowing reactions to take place at low temperatures) (Warnatz, et al., 2006).

Several authors outline limitations in sole use of the Zeldovich model for NO_x prediction because of its inadequacy at low temperatures, where NO formed from other mechanisms is prevalent.

Andersson et al (2009) presents a semi-empirical Zeldovich model that shows compliance against real data, except in situations of high EGR percentage. EGR used in the test engine reduces combustion temperatures, resulting in low prediction of NO (Figure 3-3). In these conditions, prompt and N₂O NO are dominant but not accounted for in the model, hence underprediction.

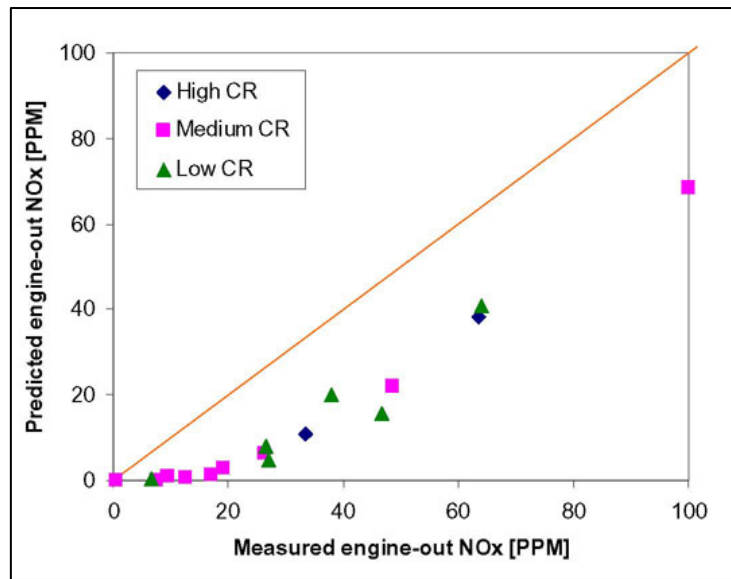


Figure 3-3 - Predicted vs Measured NO_x at high EGR% sites (Andersson et al, 2006)

A previous study from the same authors outlines a simple curve fit method by plotting measured NO_x against modelled NO_x. From this, a correction algorithm is extracted that is employed to burned zone calculations at the end of a cycle when modelled NO_x < 60 PPM:

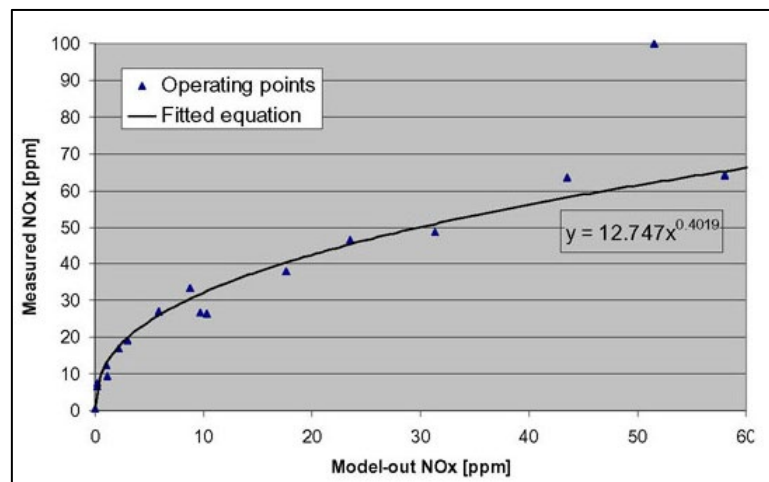


Figure 3-4 - Correction algorithm (Andersson et al, 2006)

Correction is then described for the previous study in the form:

$$ppmNO_{corr} = 12.747 \cdot (ppmNO_{uncorr})^{0.4019}$$

Employment of the same method for the newer study shows good correction (Figure 3-5) but requires engine specific calibration.

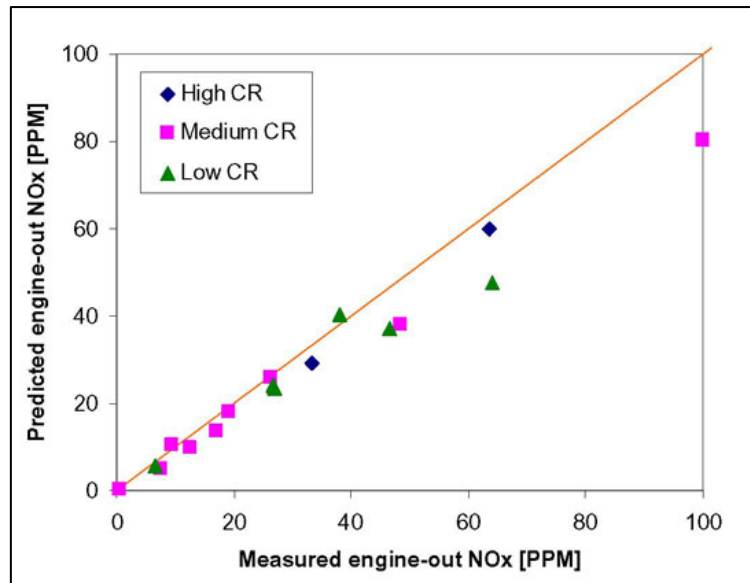


Figure 3-5 - Predicted vs Measured NO_x at high EGR% sites, employing correction algorithm (Andersson et al, 2006)

The study goes on to show good agreement of the Zeldovich model down to temperatures of 2300K, at which point deviation from measured NO occurs as another mechanism becomes prevalent. The exact mechanism is, unfortunately, difficult to identify.

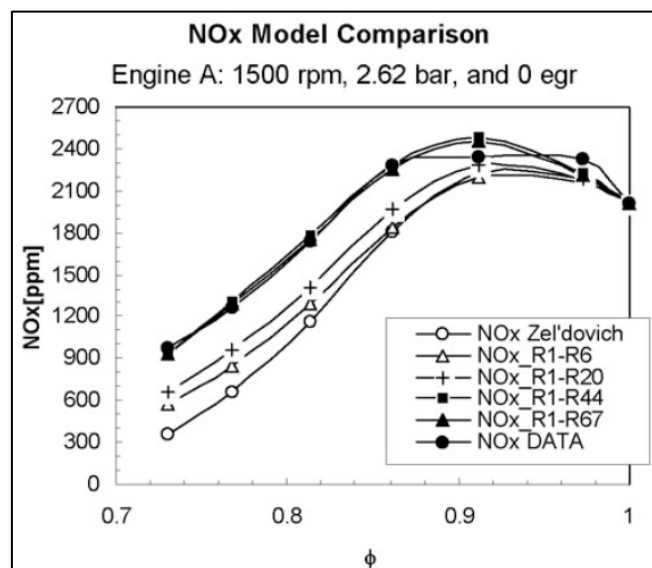


Figure 3-6 - NO_x vs FER for vary degrees of SEZM implementation, where *R* denotes which reactions out of the 67 were used (Miller et al, 1998)

Miller et al (1997) argues that the extended Zeldovich mechanism is not comprehensive enough in its identification of reactions that contribute to NO formation, citing errors of more than 50% when compared to measured engine data. In response to these failings, Miller et al expanded the mechanism to include 13 chemical species and 67 reactions that either directly or indirectly lead to the formation of NO and are said to dominate NO_x production. This is dubbed the super extended Zeldovich mechanism (SEZM).

Figure 3-6 shows increasing accuracy as more of the 67 reactions are considered to produce NO_x . Of particular note in this study is the extended Zeldovich mechanism inability to accurately predict NO levels in lean conditions, where temperatures are typically lower and less capable of dissociating the stable N_2 molecule. Research so far would indicate that prompt or N_2O mechanisms are responsible for this discrepancy. The increased accuracy of the SEZM seems to find some agreement with this observation, including reactions that form NO by way of N_2O in the 67 used.

Also of note in this study is the required initial calibration of NO production for the SEZM using measured data. Reasons given relate to agreement for reaction rates used and, more notably, to overcome the assumption of mixture homogeneity inherent in the calculations.

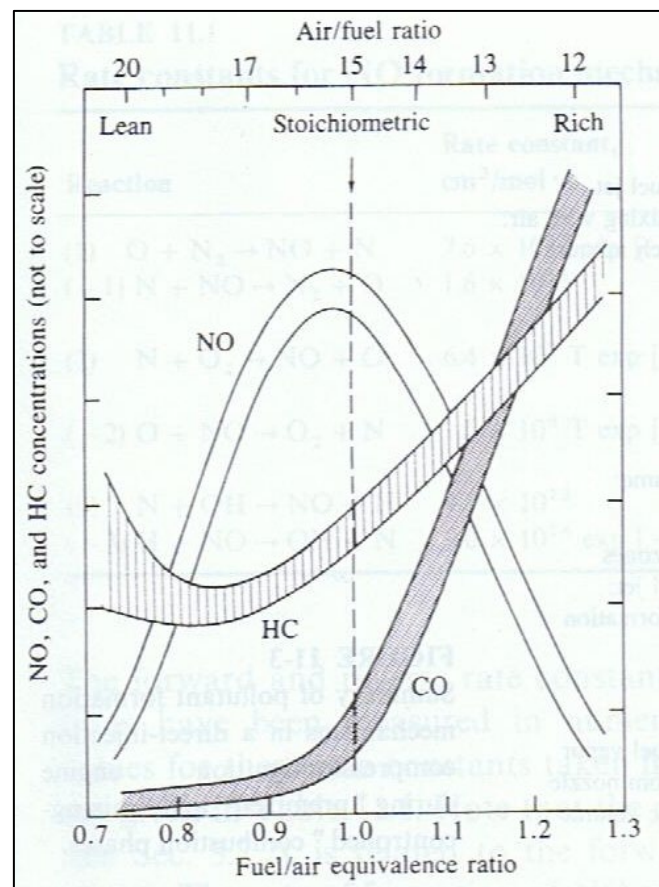


Figure 3-7 - Variation of combustion products in terms of FER (Heywood, 1988, p571)

An investigation carried out to model NO_x in a stratified charge spark ignition engine by Stone et al (2002) sheds further light on this, noting a GDI engine is capable of unintentionally producing a non-homogenous air/fuel mixture by its very design. Figure 3-7 shows peak NO_x concentration is experienced at slightly leaner than stoichiometric FER, where the trade-off between temperature and available O_2 is optimum for production. By dividing the burned zone into 10 zones of varying FER that equate to an overall average

FER, the study finds that NO_x produced locally in stratified regions does not equate to levels that would be indicative of the average FER.

For the case of homogenous mixture at FER where maximum NO_x is generated ($\phi \approx 0.9$), for example, any stratification of the mixture leads to reduced NO_x as can be observed by Figure 3-7.

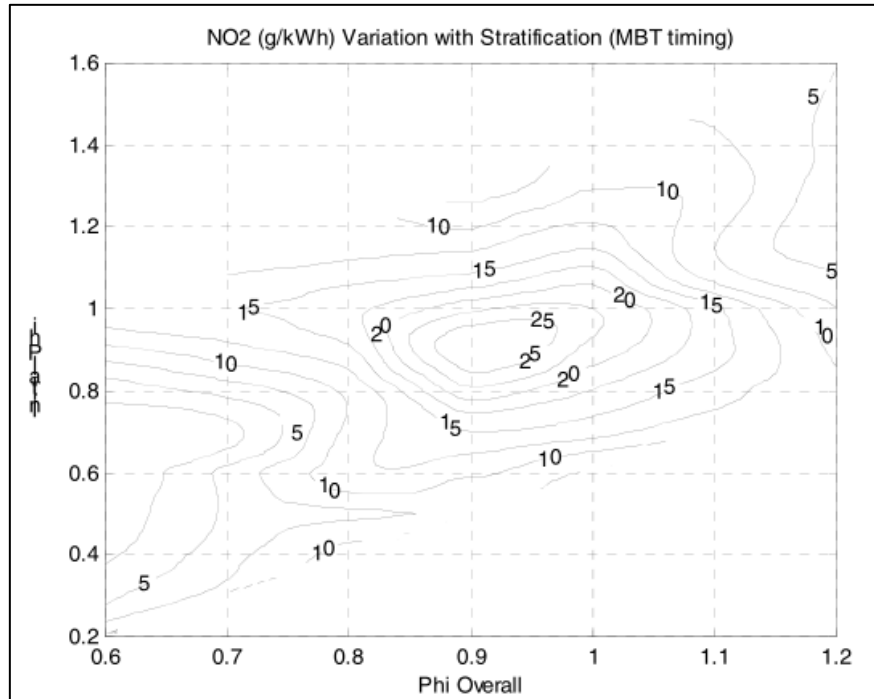


Figure 3-8 - NO_2 plotted as initial FER (badly labelled y-axis) vs overall FER (Stone et al, 2002)

The findings are more interesting for a non-homogenous charge at $\phi = 1$ as the NO_x mechanism becomes contingent on early burned zones, which have more time during the cycle to approach equilibrium levels than zones burned later on. This is further compounded by the slope of Figure 3-7, looking from $\phi = 1$: Lean early burned zones not only have more time to reach equilibrium, but can attain high values of equilibrium than rich zones. The relationship can be observed by following $\phi_o = 1$ vertically on Figure 3-8.

That the homogeneity, or lack thereof, in GDI engines has substantial effect on NO_x production is evident. Quantifying and modelling the effect for NO_x prediction would be the base for interesting CFD work but is unfortunately outside the scope of this model.

CFD simulations carried out based around a high performance GDI engine by Costa et al (2015) give some insight into the extent of non-homogeneity (Figure 3-9). The piston bowl is shaped to receive spray directly from the injector to encourage turbulence and direct a stratified charge directly at the spark plug as part of a single or multi-spray strategy.

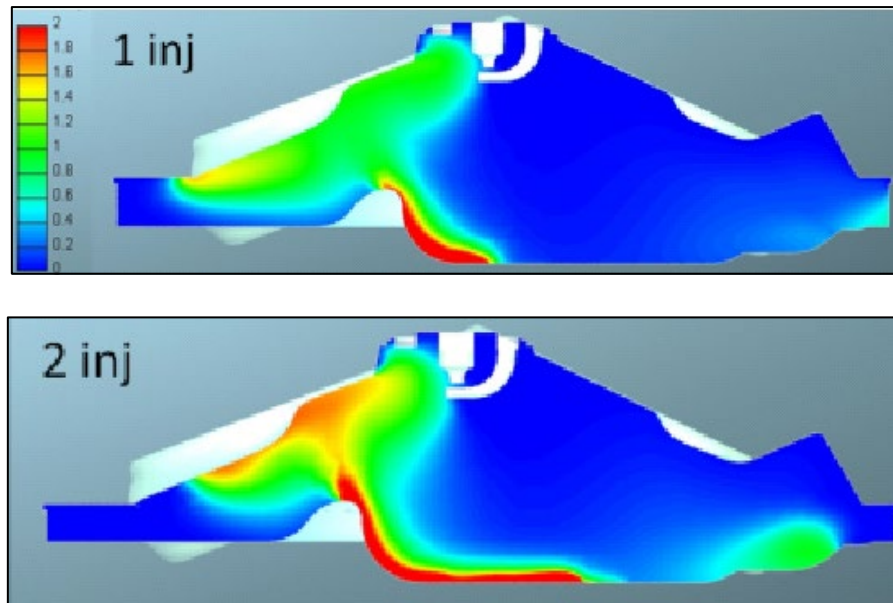


Figure 3-9 - Comparison of optimal single and double injection events. A scale for ϕ is top left. (Costa et al, 2015)

The findings are based upon stratified rather than homogenous GDI - where the latter typically has more advanced injection timings to facilitate mixture – correlations can be made between the strategies. For example, late injection is required for certain load sites to avoid knock problems in homogenous GDI and the argument could be made that such a case is comparable to a stratified injection event, as mixture time is reduced further. Ergo, Figure 3-9 may well be a representation of homogenous GDI injection in these conditions.

Other authors identify mixture composition as a problem in NO prediction for diesel engines.

Referring to diesel engines, Weisser (2001) attributes precision of the extended Zeldovich mechanism to the accuracy employed when modelling mixture formation and combustion parameters rather than the method itself. Also on diesel engines, in creation of a multi-zone model Diotallevi (2007) deems it necessary to distinguish a combustion AFR from a global AFR. The latter refers to the target AFR determined by the amount of fuel injected, while the former refers to the local AFR value present when spray reacts with air.

3.4 Results

With equations in terms of two knowns, calculated from the dissociation model and engine data, rate of NO can be calculated per degree using both methods, though it should be noted that additional unit conversions are required for Heywood's equation. Next, a simple calculation is used to convert $\frac{kmols}{m^3 \cdot sec}$ to $\frac{kmols}{^\circ}$, using seconds per 1 degree and burned volume:

$$\frac{kmols}{^\circ} = \frac{\frac{kmols}{m^3}}{sec} \cdot \frac{1}{6RPM} \cdot V_B$$

Equation 3-14

$\frac{kmols}{^\circ}$ can now be summed:

$$Total\ NO\ Produced = \sum \frac{kmols}{^\circ}$$

Equation 3-15

To calculate values for NO production in terms of parts per million (PPM), exhaust products must first be expressed in terms of kmol. From the dissociation model:

$$\sum Products = n_t$$

Equation 3-16

n_t is found to vary from around 64 to 66 so an approximation is made for ease of calculation:

$$n_t \approx 65$$

Equation 3-17

Equation 3-18

The multiplying constant (Equation 3-6) is reintroduced to account for actual cylinder mass:

$$Products = \alpha \cdot n_t = kmol$$

Equation 3-19

NO PPM can now be expressed as the fraction of NO present in one million parts of product:

$$NO_{PPM} = \frac{Total\ NO}{Total\ Products} \cdot 10^6$$

Equation 3-20

Results of both methods are presented in Figure 3-11.

Figure 3-10 shows the Merker method plotted with different n_t values, showing low sensitivity to the usable range. Largest deviation occurs in areas around stoichiometric sites and is around 60 PPM, rendering the approximation satisfactory.

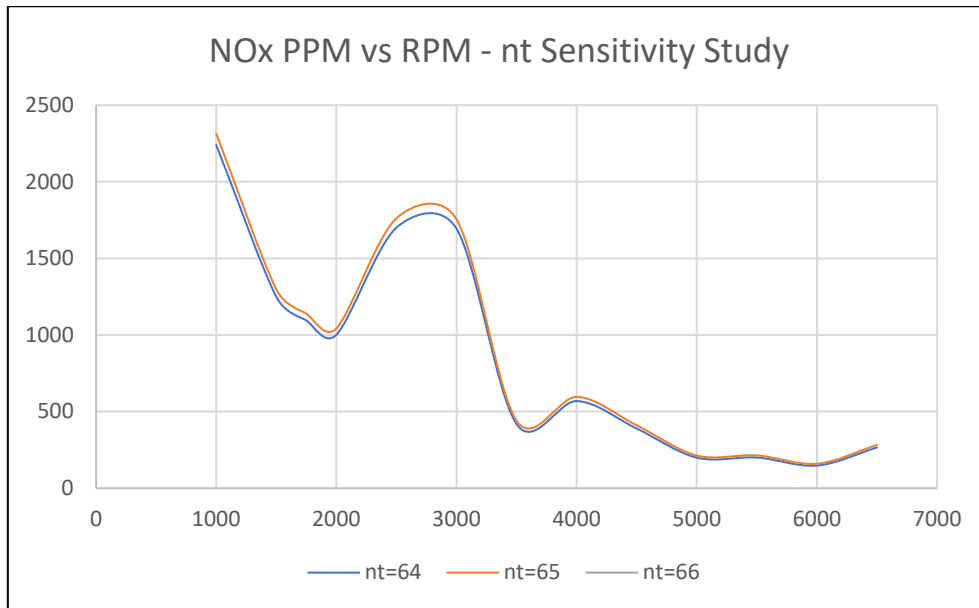


Figure 3-10 - The Merker method with varying values for n_t

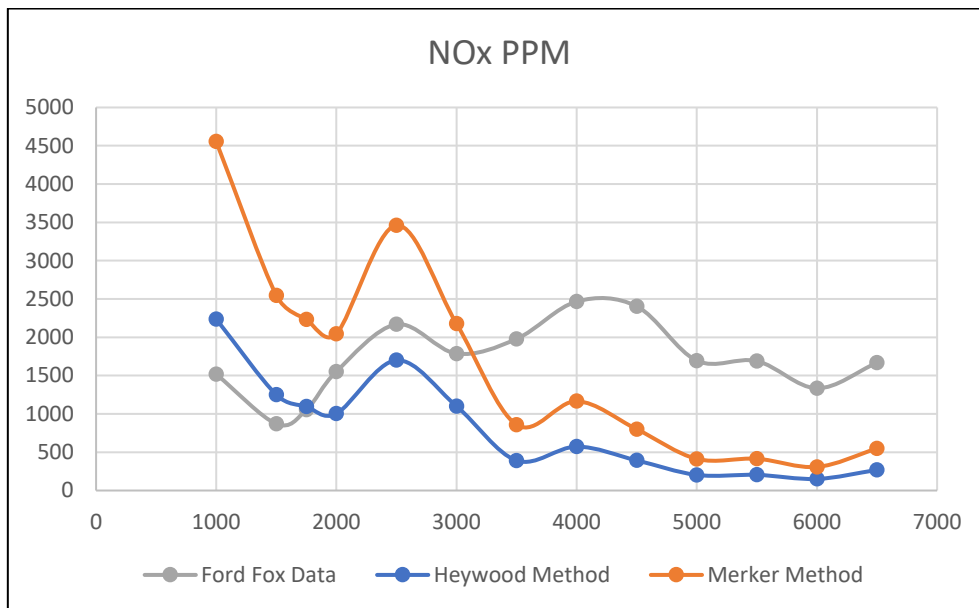


Figure 3-11- Results from extended Zeldovich mechanism, using Heywood and Merker

Results from both methods approach actual data values, though with different degrees of accuracy. Heywood's equation is more accurate overall, particularly at lean sites where Merker's equation overpredicts by over 200% compared to Heywood's 50% (more in-line with findings from Miller et al (1997)).

Table 3-3 - Select engine data and NOx Results

RPM	ϕ	Max Temp	Fox Data	Merker	Error	Heywood	Error
		K	NOx PPM	NOx PPM	%	NOx PPM	%
1000	0.90	2457.25	1518	4554.29	200.02	2237.14	47.37
1500	0.91	2416.96	871	2547.89	192.53	1250.50	43.57
1750	0.97	2453.21	1053	2235.00	112.25	1097.20	4.20
2000	1.01	2503.03	1553	2044.44	31.64	1003.18	-35.40
2500	1.01	2553.44	2169	3463.22	59.67	1700.45	-21.60
3000	1.00	2547.10	1787	3449.16	93.01	1696.51	-5.06
3500	1.06	2528.52	1979	858.19	-56.63	391.22	-80.23
4000	1.03	2525.03	2467	1165.72	-52.75	572.87	-76.78
4500	1.05	2532.60	2404	799.89	-66.73	392.96	-83.65
5000	1.09	2528.27	1695	412.72	-75.65	203.43	-88.00
5500	1.09	2537.16	1690	414.58	-75.47	204.24	-87.91
6000	1.11	2539.31	1335	308.23	-76.91	151.36	-88.66
6500	1.11	2592.31	1668	548.45	-67.12	268.61	-83.90

3.4.1 Analysis

3.4.1.1 Reaction Rate Comparative Study

Error diverges at around stoichiometric at 3000 RPM: Heywood's equation produces results of slight underprediction with Merker's equation continuing to overpredict (Table 3-3). Both methods accurately follow measured data trends with pleasing accuracy. Low RPM sites do not seem in agreement with the high RPM sites – This is in line with the temperature inconsistencies discovered in section 1.1.2.5.

Given that Heywood's equation shows better overall distribution and that the two methods show better correlation with each other in rich conditions suggest a sensitivity issue with Merker's equation for the low RPM, lean condition temperatures. It is recalled that rate constant and source of [O] substitution differs for both methods.

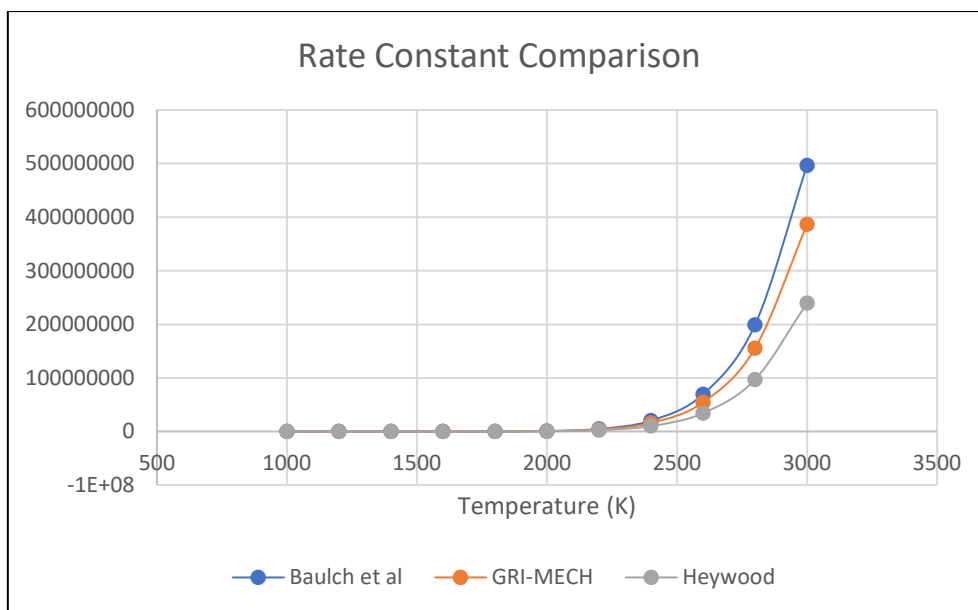


Figure 3-12 - Rate constants as function of temperature

Forward reaction rate constants from Figure 2-1 – for reaction 1 – were plotted with temperatures from 1000-3000K at 200K increments (Figure 3-12). As expected, deviation increases with temperature, becoming visually apparent at the point of maximum calculated temperatures for the combustion model. The lower values for the reaction rate constant used in Heywood's method compared to Merker's show better alignment with NO_x data from the Ford Fox.

A similar sensitivity comparison is made for techniques of including $[O]$ (section 4) resulting in little change to each result when a shared method is used. Overall the rate constant used in Heywood's equation finds better agreement with measured results, making Equation 3-13 the focus of comparison.

3.4.1.2 Temperature Sensitivity

Unsurprisingly, results in the literature vary as much as deployment of the method. Many authors agree on underprediction at low temperature conditions as the mechanism fails to account for other forms of NO production as discussed (E.g. Miller et al, 1997. Andersson et al, 2006). This phenomenon is not present in Figure 3-11 (Heywood) with overprediction present in very lean zones.

Speaking on thermal fluctuations, Raine et al (1995) reports the last burned regions of a 5-zone model to be up to 240K cooler than first burned. Modelling of this temperature differential predicts lower NO_x than single zone results. This investigation also drew attention to the uncertainty present in selecting the correct rate constant, already discussed.

Narayan and Rajan (2004) formed a modified Zeldovich-based model that incorporates both turbulence and temperature fluctuations in the burned zone, arguing that underprediction of NO_x in the Zeldovich mechanism at lean AFRs is also down to these principles. It was found when these turbulence and temperature fluctuations are included in calculations of a multizone model, a knock-on effect of increased NO concentrations is observed (up to 40%) from zone to zone. Results then more accurately approach measured data when the burned zone is modelled as a series of sequential temperature events than an all-encompassing single temperature element (Figure 3-13).

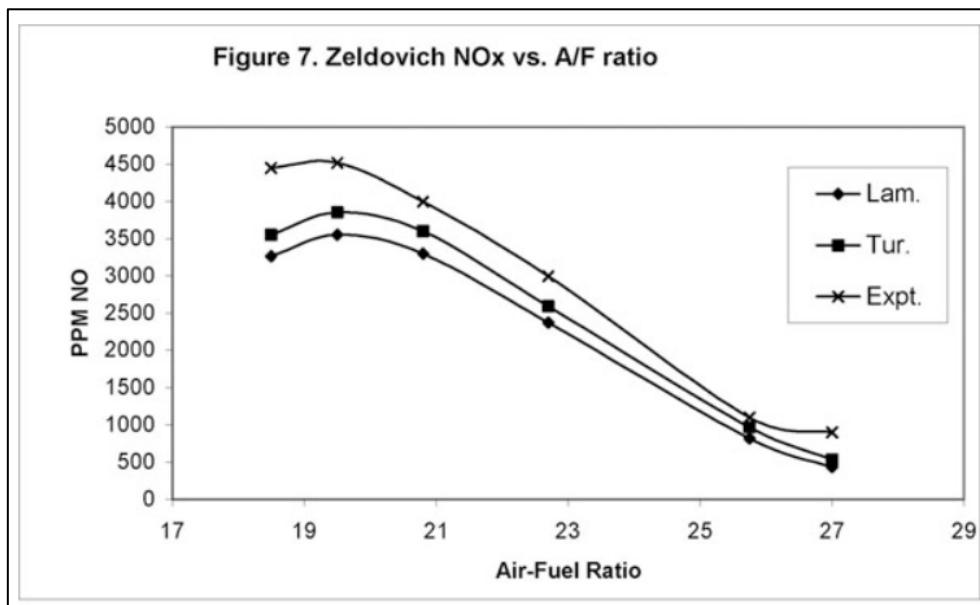


Figure 3-13 - NO_x model inclusive of turbulence for natural gas engine (AFR: $\sim 17:1$). Results plotted for laminar, turbulence model and experimental data. Discrepancy between turbulent and experiment could be explained through other mechanisms.

Results of the calculated model certainly benefits from the previous study's findings: The effects of turbulence on NO_x would hold truer for the Ford Fox GDI engine, given the turbocharger's role in increasing air velocity gradients as it passes through the intake port, thereby increasing in-cylinder turbulence (Heywood, 1988). It is also reasonable to assume significant design consideration to related parameters (such as valve shape or squish region) for the turbocharged Ford Fox because of turbulence's role in increasing combustion speed and thereby increasing knock limit.

Temperature significance has already been discussed. Temperature sensitivity was investigated by generating NO_x results for artificially increased and decreased temperatures (Figure 3-14).

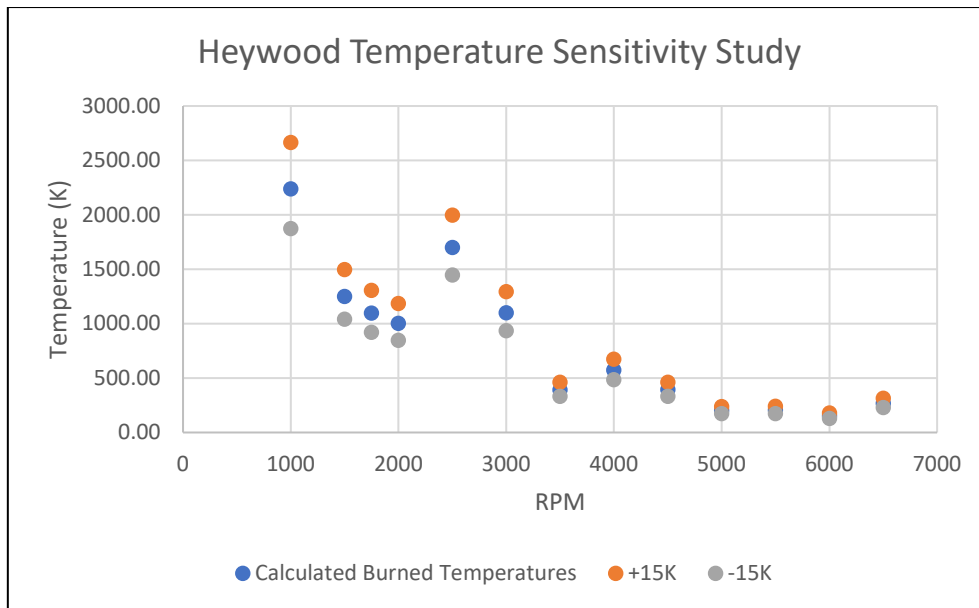


Figure 3-14- Temperature sensitivity of extended Zeldovich mechanism

It is tempting to conclude that low RPM lean conditions are most sensitive to temperature fluctuations. However, no linear relationship is observed between either temperature and RPM or temperature and FER.

Instead, Figure 3-14 and FER values in Table 3-3 iterate that temperature sensitivity is not so easily decoupled from oxygen concentration ($[O_2]$), itself determined in part by temperature. What can be concluded is temperature fluctuation has more consequence at conditions that generate peak NO_x – i.e., when temperature and $[O_2]$ align.

This is significant when considering total NO_x PPM at every RPM site for the Ford Fox model is determined largely by NO produced at peak temperatures, which contributes overwhelming to the total. Variations at these peak temperatures (whether this is down to lack of accuracy in calculating T_B or the relevance of multi-zone sequential combustion temperatures) have substantial effect on the model's prediction.

3.4.1.3 Evaluation

Discussion of less quantifiable factors may be key to understand failings of the prediction model at high RPM, rich conditions, if low RPM, lean-to-stoichiometric conditions are taken to be either:

- Approximately correct through use of Heywood's reaction rate.
- Overpredicted due to failings in burned zone temperature calculation during late ignition events.

The latter is, regrettably, far more likely. However, further discussion is warranted nonetheless.

Strong agreement is found in the literature regarding the assumption of equilibrium for product decompositions that contribute to NO. Many authors cite research carried by Heywood (1971) and Westenberg (1971). To summarize: The thin flame front is followed by a thicker zone responsible for the creation of CO₂ through oxidation of CO. Levels of radicals in this zone is greater than equilibrium (superequilibrium event, already discussed) leading to the formation of NO. Heywood states if NO in this zone is much less than that present in the burned zone proper, the equilibrium assumption holds true. This is at odds with work by Miller and Bowman (1989), previously discussed, who found that this phenomenon can contribute significantly to NO.

The issue of equilibrium is very relevant to the accuracy of this model, where high RPM, rich conditions sees a gross underprediction of NO. Combustion at high RPM is faster, leaving less time for equilibrium to be reached in the burned zone. These conditions may lead to the superequilibrium mechanism becoming more significant the final NO levels, leading to underprediction when excluded.

A similar issue arises for homogeneity at high RPM, which results in decreased mixture time in addition to decreased combustion time. Fuel injection is typically a longer process at high RPM, resulting in less mixing time for the fuel introduced at latter stages of the compression stroke before spark.

It has been demonstrated by various authors (section 3.3) that use of a global FER is far from accurate in absence of a homogenous mixture. These multi-zone studies concluded that combustion does not equal the sum of its parts: that local FER and its position in the combustion process is more determinant in a non-homogenous environment. This indeliberate stratified effect is a very real concern for GDI engines (Blair, 1999).

Assumption of the global FER governing NO production at high RPM could well be the failing of the model, due to decreased mixture time despite increased turbulence. Similar increases in NO_x production have resulted from the inclusion of turbulence mechanics in Zeldovich-based models. A lesser case could be made for the same issue at low RPM, where time prevails but piston-induced turbulence is relatively low.

Burned zone temperatures at low RPM are unreliable, as discussed. Lower temperatures in these areas – more in line with expectation – would reduce NO_x prediction at these sites to, feasibly, levels of underprediction more in-line with the upper end of the RPM scale. It is reasonable to conclude that the result would be a more consistent– albeit under predictive –

model that imitated trends of the measured data. Such a model would align with the underprediction that crops up in much of the literature for these simple models, that are uncoupled from the various (undeniably significant) parameters that contribute to NO formation.

4 Extended Goodger Dissociation Method

To simplify implementation of the extended Zeldovich mechanism (Equation 3-11), the Goodger dissociation model can be expanded to include additional species:

$$\begin{aligned} &\phi \cdot C_8H_{18} + 12.5 \cdot (O_2 + 3.76N_2) \\ &= \\ &n_1 \cdot CO_2 + n_2 \cdot H_2O + n_3 \cdot CO + n_4 \cdot H_2 + n_5 \cdot O_2 + n_6 \cdot N_2 + n_7 \cdot O + n_8 \cdot N + n_9 \cdot NO \end{aligned}$$

Equation 4-1

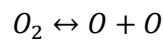
This section will detail the inclusion of the O radical, followed by a discussion regarding N and NO .

4.1 Methodology

$$\begin{aligned} &\phi \cdot C_8H_{18} + 12.5 \cdot (O_2 + 3.76N_2) \\ &= \\ &n_1 \cdot CO_2 + n_2 \cdot H_2O + n_3 \cdot CO + n_4 \cdot H_2 + n_5 \cdot O_2 + n_6 \cdot N_2 + n_7 \cdot O \end{aligned}$$

Equation 4-2

With an additional unknown a new equation must be introduced, using the partial pressure equilibrium constant for the decomposition of O_2 to O :



Equation 4-3

Partial pressure equilibrium constant calculation followed the same method outlined in section 2.2.3.

Table 4-1 - Change in Gibbs free energy and partial pressure equilibrium constants including O_2

T	CO_2		H_2O		Goodger		O_2	
(Kelvin)	$-\Delta G^\circ$ (KJ/mol)	K_p	$-\Delta G^\circ$ (KJ/mol)	K_p	$K_{p(CO_2)}$	$K_{p(H_2O)}$	$-\Delta G^\circ$ (KJ/mol)	K_p
2000	110.299	760.0083	135.528	3465.421	765.6	3467	243.10	
2100	101.932	343.1715	129.721	1685.581	345.94	1686.6	229.72	
2200	93.59	166.7974	123.905	874.9534	168.27	874.98	216.32	
2300	85.273	86.4332	118.082	480.6437	87.097	480.84	202.90	
2400	76.978	47.36342	112.252	277.453	47.753	277.43	189.47	
2500	68.706	27.26372	106.416	167.3096	27.543	167.49	176.02	
2600	60.455	16.39078	100.575	104.8696	-	-	162.57	
2700	52.224	10.24166	94.729	68.03131	10.351	68.077	149.10	
2800	44.013	6.623724	88.878	45.50921	-	-	135.63	
2900	35.82	4.417852	83.023	31.2939	-	-	122.14	
3000	27.645	3.029421	77.163	22.05849	3.0549	22.029	108.65	

An expressed must be derived for the partial pressure of n_7 in terms of knowns. Starting with the definition for partial pressure equilibrium constant:

$$K_{p(O_2)} = \frac{p_{O_2}}{p_O \cdot p_O}$$

Equation 4-4

Where:

$$p_{O_2} = p \cdot \frac{n_{O_2}}{n_t} = p \cdot \frac{n_5}{n_t}$$

Equation 4-5

$$p_O = p \cdot \frac{n_O}{n_t} = p \cdot \frac{n_7}{n_t}$$

Equation 4-6

Substitution of Equation 4-5 and Equation 4-6 into Equation 4-4 gives:

$$K_{p(O_2)} = \frac{p \cdot \frac{n_5}{n_t}}{p \cdot \frac{n_7}{n_t} + p \cdot \frac{n_7}{n_t}} = \frac{n_5}{n_7 \cdot \left(p \cdot \frac{n_7}{n_t}\right)} = \frac{n_5}{n_7^2 \cdot \left(\frac{p}{n_t}\right)}$$

Equation 4-7

Because:

$$\frac{n_5}{n_7^2 \cdot \left(\frac{p}{n_t}\right)} = \frac{n_5}{\frac{n_7^2}{n_5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)}$$

Equation 4-8

Then:

$$K_{p(O_2)} = \frac{n_5}{\frac{n_7^2}{n_5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)} = \frac{n_5^2}{n_7^2 \cdot \left(p \cdot \frac{n_5}{n_t}\right)} = \left(\frac{n_5}{n_7}\right)^2 \cdot \frac{1}{\left(p \cdot \frac{n_5}{n_t}\right)}$$

Equation 4-9

$$\left(\frac{n_5}{n_7}\right)^2 = K_{p(O_2)} \cdot \left(p \cdot \frac{n_5}{n_t}\right)$$

Equation 4-10

$$\frac{n_5}{n_7} = K_{p(O_2)}^{0.5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)^{0.5}$$

Equation 4-11

$$n_7 = \frac{n_5}{K_{p(O_2)}^{0.5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)^{0.5}}$$

Equation 4-12

(McDonald, 2018)

The final expression gives an equation for n_7 in terms of:

- $K_{p(O_2)}$ – Values are worked out via the partial pressure equilibrium constant above.
- $\left(p \cdot \frac{n_5}{n_t}\right)^{0.5}$ – The initial Goodger method guess, multiplied by selected pressure increments.
- n_5 – Total atmospheric oxygen (O_2) present in the products.

Atomic oxygen (n_7) must be included in the total of atmospheric oxygen (n_5). This creates a circular reference – with each species calculation dependant on the other – which is solved by substituting the new expression for n_7 . Recalling Equation 2-3, the oxygen balance for the extended Goodger method becomes:

$$25 = 2 \cdot n_1 + n_2 + n_3 + 2 \cdot n_5 + n_7$$

Equation 4-13

Substituting for n_7 and manipulating:

$$25 = 2 \cdot n_1 + n_2 + n_3 + 2 \cdot n_5 + \frac{n_5}{K_{p(O_2)}^{0.5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)^{0.5}}$$

$$25 = 2 \cdot n_1 + n_2 + n_3 + n_5 \cdot \left(2 + \frac{1}{K_{p(O_2)}^{0.5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)^{0.5}}\right)$$

Equation 4-14

$$n_5 = \frac{25 - 2 \cdot n_1 - n_2 - n_3}{\left(2 + \frac{1}{K_{p(O_2)}^{0.5} \cdot \left(p \cdot \frac{n_5}{n_t}\right)^{0.5}}\right)}$$

Equation 4-15

Steps for the Goodger method remain the same. The new n_5 expression replaces the existing one and Equation 4-12 is added after the calculation of n_6 in the previous method.

4.2 Results

n_7 is calculated at every degree through the same method of using polynomials for 1 bar pressure intervals from 10-100 across temperature ranges, and then interpolating locally for all decimal pressure points of the Ford Fox data (section 2.2.6). Concentrations were also reached using the same methodology as previous (section 3.1).

With $[O]$ calculated, a reverse step can be taken from the previous method of calculating rate of NO production as no conversion substitution is required from $[O]$ to $[O_2]$. Equation 3-11 is recalled for this section:

$$\frac{d[NO]}{dt} = 2 \cdot k_{f(1)}[O][N_2]$$

Equation 4-16

Using Heywood's (1988) reaction rate, Equation 4-16 becomes:

$$\frac{d[NO]}{dt} = 15.2 \cdot 10^{13} \cdot \exp\left(-\frac{38000}{T}\right)[O][N_2]$$

Equation 4-17

And using Merker's reaction rate, Equation 4-16 becomes:

$$\frac{d[NO]}{dt} = 3.6 \cdot 10^{11} \cdot \exp\left(-\frac{38370}{T}\right)[O][N_2]$$

Equation 4-18

NO_x prediction (section 3.4) is repeated for both methods and compared with both data and original prediction.

Use of $[O]$ for both methods show an increase in NO_x prediction at lean and stoichiometric FERs that tapers to no difference in progressively rich conditions. Once again, the discrepancy in reaction rates becomes apparent with Merker's method suffering more of an increase in the lean zone that uses inconsistent temperatures. The upper RPM zone shows good agreement, though this is most likely down to the lesser oxygen content, ergo change in oxygen has little effect.

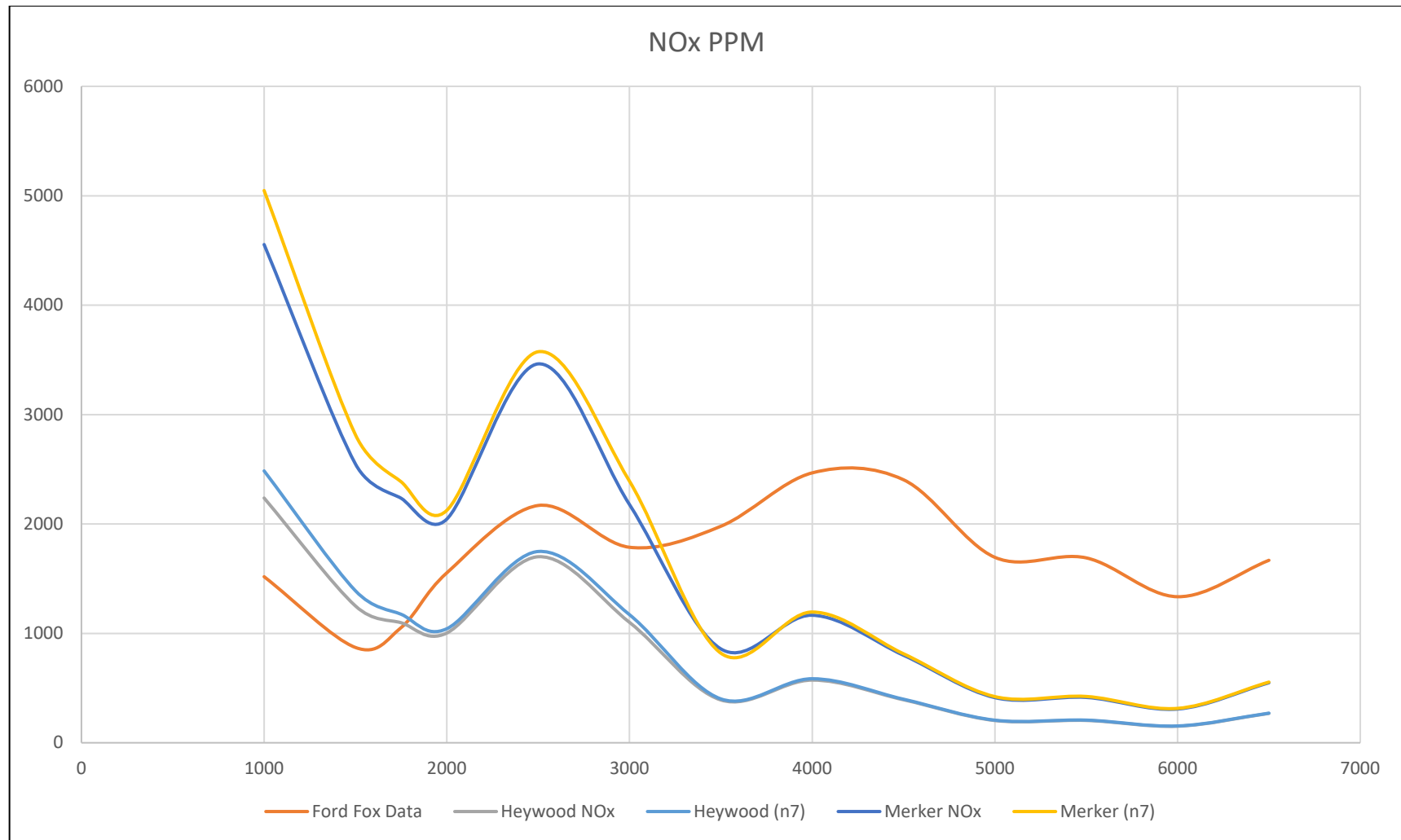


Figure 4-1- A comparison of methods

4.3 Further Work

Early calculations indicate it's possible to add n_8 (atomic nitrogen) and n_9 (NO) to the dissociation model, extending to a nine species-model that predicts equilibrium levels of NO. An interesting comparison could be drawn between the partial pressure, kinetically controlled NO of this work and a new model that assumes equilibrium concentrations for the governing NO reaction.

5 Conclusion

A model was created using a combination of measured engine data and calculations from first principles to predict NO_x emissions for a Ford Fox 1.0 turbocharged GDI engine. Inconsistencies with calculated burned zone temperature in retarded spark ignition zones have resulted in final predictions that find little agreement with either measured engine data, or the literature. Reasons behind the model's failings have been explored at length and accuracy within the landscape of the literature has been met with these in mind.

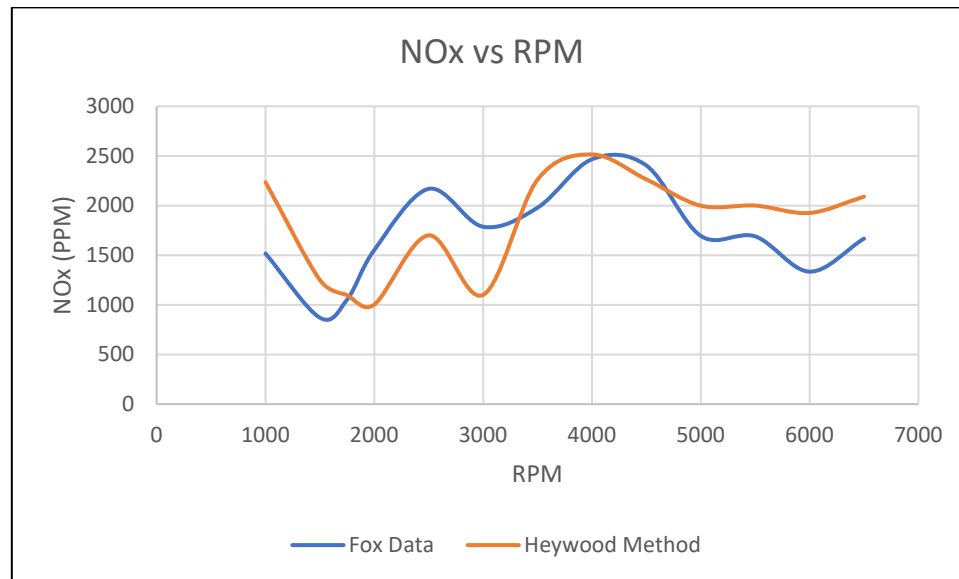


Figure 2 - Curve fit Heywood Method

Research conducted in section 3.3 outlines a simple correction method that can be applied to the results but doing so is akin to masking the fundamental problem rather than solving it. Additional engine data would be required to determine whether this curve fit method would hold up to a change in conditions and find more accurate prediction by using the correction.

The potential for further work has been discussed and it is believed understanding of the model and the underlying assumptions that govern it would be well advanced by it.

- Find sources where NO under predicts or overpredicts (change sentence about literature)
- Global FER may not reflect local FER (sources already listed), especially at high RPM where the mixture has less time to homogenize. Even if injections happen sooner to get the fuel in at high RPM, there are still knock limits that prevent fuel getting injected early enough to mix.
- Temperature sensitivity
 -
- Smith, J., "Temperature and Density Measurements in an Engine by Pulsed Raman Spectroscopy," SAE Technical Paper 800137, 1980, <https://doi-org.ezproxy.uwtsd.ac.uk/10.4271/800137>.
 - The fluctuations of temperature are important to pollutant formation. In fact, both Gouldin¹⁸ and Jones¹⁹ calculate an order of magnitude higher nitric oxide concentrations from use of fluctuating temperatures compared to using mean temperatures
- Aceves, S., Martinez-Frias, J., Flowers, D., Smith, J. et al., "A Computer Generated Reduced Iso-Octane Chemical Kinetic Mechanism Applied to Simulation of HCCI Combustion," SAE Technical Paper 2002-01-2870, 2002, <https://doi-org.ezproxy.uwtsd.ac.uk/10.4271/2002-01-2870>.

- Small temperature differences inside the cylinder have a considerable effect on (HCCI) combustion because chemical kinetics is very sensitive to temperature.

Table 4-1 - Reaction rates according to Heywood (1988, p573), via a collective review.

Reaction	Rate Constant
1) $O + N_2 \leftrightarrow NO + N$	$7.6 \cdot 10^{13} \cdot \exp\left(-\frac{38000}{T}\right)$
-1) $NO + N \leftrightarrow O + N_2$	$1.6 \cdot 10^{13}$
2) $N + O_2 \leftrightarrow NO + O$	$6.4 \cdot 10^9 \cdot \exp\left(-\frac{3150}{T}\right)$
-2) $NO + O \leftrightarrow N + O_2$	$1.5 \cdot 10^9 \cdot \exp\left(-\frac{19500}{T}\right)$
3) $NO + H \leftrightarrow N + OH$	$4.1 \cdot 10^{13}$
-3) $NO + H \leftrightarrow N + OH$	$2.0 \cdot 10^{14} \cdot \exp\left(-\frac{23650}{T}\right)$

6 References

- Air Quality Expert Group, 2004. *Nitrogen Dioxide in the UK*, London: DEFRA.
- al, B. e., 2005. Evaluated Kinetic Data for Combustion Modeling. *J. Phys. Chem. Ref. Data*, Vol. 34, No. 3.
- Alkidas, A. C., 1999. Combustion-chamber crevices: the major source of engine-out hydrocarbon emissions under fully warmed conditions. *Progress in energy and combustion science*, 25(3).
- Andersson, M., Johansson, B., Hultqvist, A. & Noehre, C., 2006. A predictive real time NO_x model for conventional and partially premixed diesel combustion. *SAE Technical Paper 2006-01-3329*.
- Attard, W. P. & Parsons, P., 2010. Flame Kernel Development for a Spark Initiated Pre-Chamber Combustion System Capable of High Load, High Efficiency and Near Zero NO_x Emissions. *SAE Int. J. Engines* 3(2):408-427.
- Battigelli, M., 1999. Biological Significant of Nitrogen Oxides. *SAE Technical paper*.
- Baulch, D. L. et al., 1991. Compilation of rate data for combustion modelling. *Supplement I. J Phys. Chem. Ref. Data* 22, p. 847.
- Blair, G. P., 1999. *Design and simulation of four-stroke engines*. Warrendale: SAE.
- Boningari, T. & Smirniotis, P. G., 2016. Impact of nitrogen oxides on the environment and human health: Mn-based materials for the NO_x abatement. *Current Opinion in Chemical Engineering*, pp. 133-141.
- Borman, G. L. & Ragland, K. W., 1998. *Combustion Engineering*. Michigan: McGraw-Hill.
- Brunt, M. F. J. & Platts, K. C., 1999. Calculation of Heat Release in Direct Injection Diesel Engines. *SAE Technical Paper 1999-01-0187*.
- Brunt, M. F. J. & Rai, H., 1998. The Calculation of Heat Release Energy from Engine Cylinder Pressure Data. *SAE Technical Paper 981052*.
- Camporeale, S. M., Ciliberti, P. D., Carlucci, A. & Ingrosso, D., 2017. Dynamic Validation and Sensitivity Analysis of a NO_x Estimation Model Based on In-Cylinder Pressure Measurement. *SAE Technical Paper 2017-24-0131*.

Costa, M. et al., 2015. Experimental and Numerical Investigation of the Effect of Split Injections on the Performance of a GDI Engine Under Lean Operation. *SAE Technical Paper 2015-24-2413*.

Diotallevi, F., 2007. Development of a Multi-Zone Model. *Master of Science Thesis MMK 2007:55 MFM109*.

EEA, 2017. *Air Quality in Europe - 2017 Report*, Luxembourg: Publications Office of the European Union.

EPA, n.d. *Overview of Greenhouse Gas Emissions*. [Online]

Available at: <https://www.epa.gov/ghgemissions/overview-greenhouse-gases#nitrous-oxide>

EPA, U., 2016. *Integrated Science Assessment (ISA) for Oxides of Nitrogen - Health Criteria (Final Report, 2016) EPA/600R-15/068,2016*, Washington DC: United States Environmental Protection Agency .

Fenimore, C. P., 1979. Studies of fuel-nitrogen in rich flame gases. *Proc Comb Inst 17:661*.

Fuller, T. F. & Harb, J. N., 2016. *Electrochemical Engineering*. Hoboken: John Wiley & Sons.

Glassman, I. & Yetter, R. A., 2011. *Combustion 4th Ed.*. San Diego: Elsevier.

Goodger, E. M., 2001. *Transport Fuels Technology*. s.l.:Landfall Press.

Gordon, S. & McBride, B. J., n.d. *CEA Software*. s.l.:s.n.

Grau, J. H. et al., 2002. Modelling Methodology of a Spark-Ignition Engine and Experimental Validation: Part I: Single-Zone Combustion Model. *SAE Technical Paper 2002-01-2193*.

Guezennec , Y. G. & Hamama, W., 1999. Two-Zone Heat Release Analysis of Combustion Data and Calibration of Heat Transfer Correlation in an I. C. Engine. *SAE Technical Paper 1999-01-0218*.

Heywood, J. B., 1971. Gas turbine combustor modelling for calculation nitric oxide emissions. *AIAA Paper No. 71-712, AIAA/SAE 7th Propulsion Join Specialist Conference*.

Heywood, J. B., 1988. *Internal Combustion Engine Fundamentals*. s.l.:McGraw-Hill Education.

Heywood, J. B. & Mikus, T., 1973. *Parameters controlling nitric oxide emissions from gas turbine combustors*. London, s.n.

- Karvountzis-Kontakiotis, A. T. & Ntziachristos, L., 2012. A detailed chemical mechanism to predict NO cycle-to-cycle variation in homogeneous engine combustion. *IFAC Proceedings Volumes, Volume 45, Issue 30*, pp. 408-415.
- Kihas, D. & Et al, 2016. Concept Analysis and Initial Results of Engine-Out NO_x Estimator Suitable for on ECM Implementation. *SAE Technical Paper 2016-01-0611*.
- Koenig, J. Q., 1999. Air pollution and asthma. *Journal of Allergy and Clinical Immunology*, pp. 717-722.
- Kojima, S. et al., 2018. Development of a New 2L Gasoline VC-Turbo Engine with the World's First Variable Compression Ratio Technology. *SAE Technical Paper 2018-01-0371*.
- Lancaster, D. R., Krieger, R. B. & Lienesch, J. H., 1975. Measurement and Analysis of Engine Pressure Data. *SAE Technical Paper 750026*.
- Lavoie, G. A., Heywood, J. B. & Keck, J. C., 1970. Experimental and Theoretical Study of Nitric Oxide Formation in Internal Combustion Engines. *Combustion Science & Technology, Vol 1*, pp. 313-326.
- Law, C. K., Makino, A. & Lu, T. F., 2006. On the off-stoichiometric peaking of adiabatic flame temperature. *Combustion and Flame, Volume 145, Issue 4*, pp. 808-819.
- Malte, P. C. & Pratt, D. T., 1974. Measurement of atomic oxygen and nitrogen oxides in jet-stirred combustion. *Proc Comb Inst 15:1061*.
- McDonald, M., 2018. *Lecture & supervisor notes*. Swansea: University of Wales Trinity Saint David.
- Merker, G. P., Schwarz, C., Stiesch, G. & Otto, F., 2006. *Simulating Combustion*, Berlin: Springer-Verlag.
- Miller, J. A. & Bowman, C. T., 1989. Mechanism and modeling of nitrogen chemistry in combustion. *Progress in Energy and Combustion Science, Volume 15, Issue 4*, pp. 287-338.
- Miller, R., Davis, G. & Lavoie, G., 1997. Dilute and Fuel Rich NO_x Chemistry: An Engine Simulation. *Combustion science technology*.
- Miller, R. et al., 1998. A Super-Extended Zel'dovich Mechanism for No_x Modeling and Engine Calibration. *SAE Technical Paper 980781*.
- Myers, R. L., 2003. *The Basics of Chemistry*. Westport: Greenwood.
- Narayan, S. & Rajan, S., 2004. Turbulent Temperature Fluctuations in Spark Ignition Engines and Their Effect on NO_x Emissions. *SAE Technical Paper 2004-01-0623*.

- Pilling, M. J. & Seaking, P. W., 2007. *Reaction Kinetics*, Oxford: Oxford University Press.
- Raine, R. R., Stone, C. R. & Gould, J., 1995. Modeling of Nitric Oxide Formation in Spark Ignition Engines with a Multizone Burned Gas. *Combustion and Flame*, Vol. 102.
- Rassweiler, G. M. & Withrow, L., 1938. Motion Pictures of Engine Flames Correlated with Pressure Cards. *SAE Technical Paper 380139*.
- Rogers, G. & Mayhew, Y., 1992. *Engineering Thermodynamics - Work & Heat Transfer*. Essex: Pearson Education Limited.
- Schwerdt, C., 2006. Modelling NO_x-Formation in Combustion Processes. *Master Thesis, Department of Automatic Control, Lund University*.
- Stone, C., Wysynski, L. & Raine, R., 2002. Prediction of NO Emissions from Stratified Charge Spark-Ignition Engines. *SAE Technical Paper 2002-01-1139*.
- Stone, R. & Ball, J. K., 2004. *Automotive Engineering Fundamentals*. Warrendale: SAE.
- Turns, S., 2006. *Thermal-Fluid Sciences: An Integrated Approach*. s.l.:Cambridge University Press.
- Von Mutius, E., 1998. Epidemiological studies on pollution and allergy. *Revue Française d'Allergologie et d'Immunologie Clinique*, pp. S76-S79.
- Warnatz, J., Maas, U. & Dibble, R. W., 2006. *Combustion*. Berlin: Springer-Verlag.
- Weisser, G. A., 2001. Modelling of combustion and nitric oxide formation for medium-speed DI diesel engines. *Doctoral Thesis*.
- Westenberg, A. A., 1971. Unkown title. *Combustion Science Technology, Volume , Issue 2*.
- Yeh, J. T. & Chen, W.-Y., 2005. *Handbook of Environmental Engineering Volume 2 - Advanced Air and Noise Pollution Control*. New Jersey: Humana Press Inc.
- Zeldovich, J., 1946. The oxidation of nitrogen in combustion and explosions. *European Physical Journal A. Hadrons and Nuclei Vol 21*, pp. 577-628.
- Zhao, F., Lai, M. & Harrington, D. L., 1999. Automotive spark-ignited direct-injection gasoline engines. *Progress in energy and combustion science*.

7 Appendix A – Stoichiometry

Stoichiometry refers to the relationship between reactants and products as they undergo reaction. Conservation of mass states that mass cannot be created or destroyed, only transformed. Therefore, it is possible to balance a reaction through use of stoichiometric coefficients – amount of species denoted by the number preceding a species – combined with atomic subscripts.

Combustion of automotive fuels is achieved through the oxidation of hydrocarbons. With the products of combustion known, AFR can be calculated by performing an atomic balance on each species.

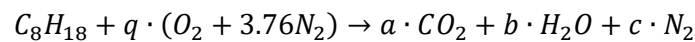
Air content is approximated to: 21% oxygen (O_2), 79% nitrogen (N_2). A ratio of nitrogen to oxygen can be calculated:

$$\frac{N_2}{O_2} = \frac{79}{21} = 3.76$$

Equation 6-1

There are 3.76 times the amount of N_2 than O_2 present in the air.

For a idealized combustion process with only three products:



Equation 6-2

Where q is the stoichiometric coefficient of air that would produce complete combustion of all fuel – I.e.: a stoichiometric combustion value. Recalling conservation of mass, balances for carbon and hydrogen can be made. Reactant atoms must equal product atoms:

$$a = C_8 = 8 \cdot CO_2$$

Equation 6-3

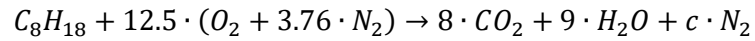
$$b = \frac{H_{18}}{2} = 9 \cdot H_2$$

Equation 6-4

These results can be used to calculate total product O atoms:

$$O = 8 \cdot 2 + 9 = 25$$

Products must equal reactants, so Equation 7-2 becomes:

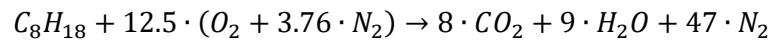


Equation 6-5

Finally, N_2 can be worked out:

$$c = 12.5 \cdot 3.76 \cdot N_2 = 47$$

Equation 6-6



Equation 6-7

AFR by mass can now be calculated by dividing the mass of air by the mass of fuel for chemically complete combustion, by introducing atomic masses for each element:

Table 6-1 - Atomic Mass

Species	ATOMIC WT
Carbon	12
Hydrogen	1
Oxygen	16
Nitrogen	14

$$AFR = \frac{12.5 \cdot [(16 \cdot 2) + (3.76 \cdot 14 \cdot 2)]}{(12 \cdot 8) + (18 \cdot 1)} = 15.05$$

Equation 6-8

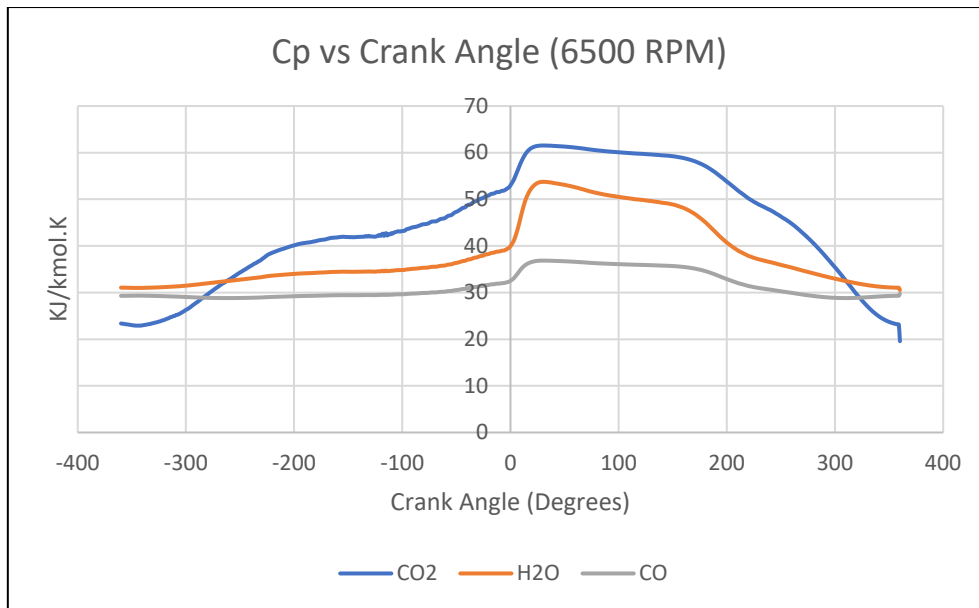


Figure 8-2 - C_p vs Crank Angle

$\tilde{C}_p$ for fuel (modelled as isooctane) is approached using values from the Chemical Equilibrium Product software developed by NASA (Gordon & McBride, n.d.) and delivered via Heywood (1988). An archetypal equation is given:

$$\frac{\tilde{C}_{p,i}}{\bar{R}} = a_{i1} + a_{i2} \cdot T + a_{i3} \cdot T^2 + a_{i4} \cdot T^3 + a_{i5} \cdot T^4$$

Where $\tilde{C}_{p,i}$ denotes specific heat in standard state at temperature (T) and a_{ix} denotes species-specific coefficients supplied:

Table 7-1 - C_p polynomial coefficient (Heywood, 1988, p133)

Fuels							
Fuels	Af1	Af2	Af3	Af4	Af5	Af6	Af8
Isooctane	-0.55313	181.62	-97.787	20.402	-0.03095	-60.751	20.232

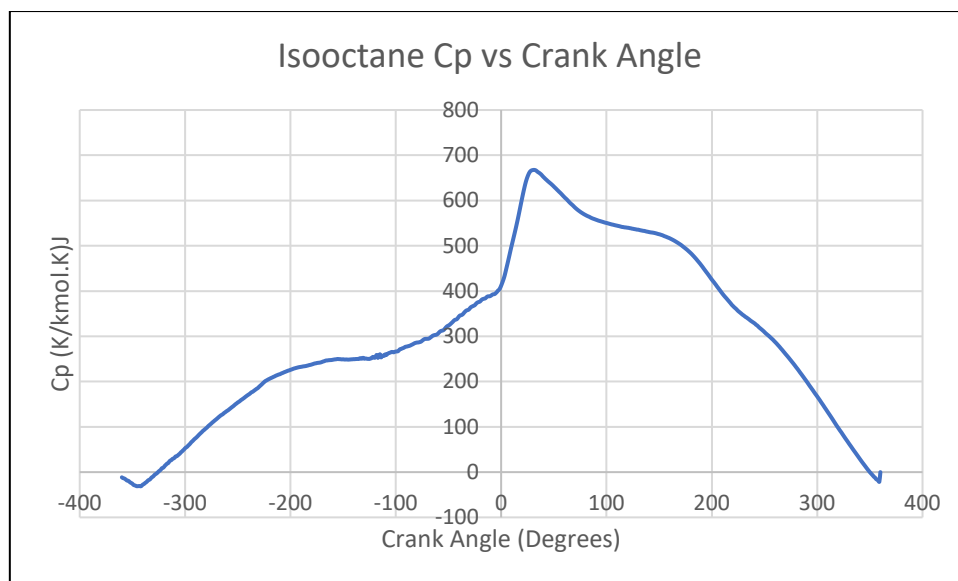


Figure 8-3 - C_p Isooctane vs crank angle

It should be noted that the NASA coefficient method can be applied to calculate product and reactant species. These calculations were performed in the interests of comparison and, unsurprisingly, found good agreement with tabulated polynomial values calculated since the coefficients are routed in polynomials. The tabulated method was favoured in this approach in keeping with first principle goals of the work.

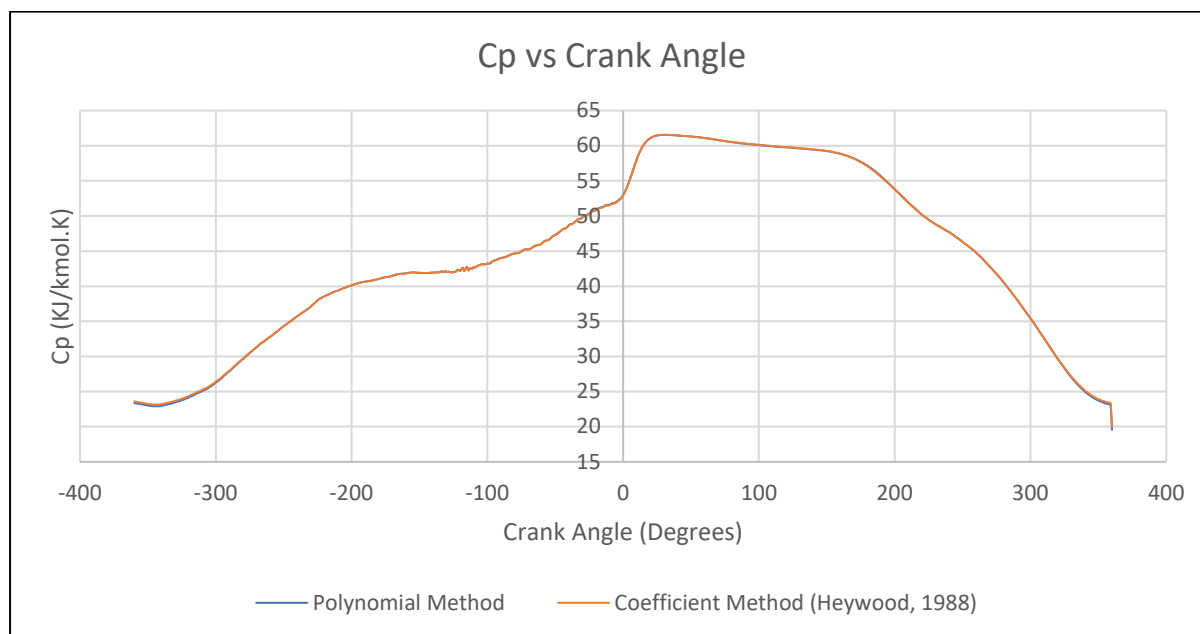
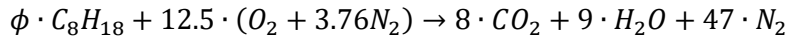


Figure 8-4 - Comparison of polynomial and coefficient

An average $\tilde{C}_p$ must be calculated for both the burned and unburned zone by multiplying species $\tilde{C}_p$ with stoichiometric coefficients and then dividing by stoichiometric coefficient total. EGR is also included in the calculation. For the stoichiometric case:



Equation 7-2

The calculation is:

$$\begin{aligned} & \tilde{C}_{p(UBavg)} \\ &= \frac{\phi \cdot \tilde{C}_{p(isooctane)} + 12.5 \cdot (\tilde{C}_{p(O_2)} + 3.76 \cdot \tilde{C}_{p(N_2)}) + z \cdot (8 \cdot \tilde{C}_{p(CO_2)} + 9 \cdot \tilde{C}_{p(H_2O)} + 47 \cdot \tilde{C}_{p(N_2)})}{\phi + 12.5(1 + 3.76) + z \cdot (8 + 9 + 47)} \end{aligned}$$

Equation 7-3

$$\tilde{C}_{p(Bavg)} = \frac{8 \cdot \tilde{C}_{p(CO_2)} + 9 \cdot \tilde{C}_{p(H_2O)} + 47 \cdot \tilde{C}_{p(N_2)} + z \cdot (8 \cdot \tilde{C}_{p(CO_2)} + 9 \cdot \tilde{C}_{p(H_2O)} + 47 \cdot \tilde{C}_{p(N_2)})}{(8 + 9 + 47) + z \cdot (8 + 9 + 47)}$$

Equation 7-4

Where z is the percentage of EGR, specific to each RPM.

With $\tilde{C}_p$ plotted for every degree, $\tilde{C}_v$ can also be worked out from the universal gas constant ($\tilde{R}$) relationship:

$$\tilde{C}_v = \tilde{C}_p - \tilde{R}$$

Equation 7-5

γ can now be calculated for every degree. MFB (x) is used to proportionally determine γ during combustion:

$$\gamma_\theta = (x \cdot \tilde{C}_{p(Bavg)}) + [(1 - x) \cdot \tilde{C}_{p(UBavg)}]$$

CP IS AFFECTED BY DISSOCIATION

9 Appendix C – Useful Derivations

Derivation of required equations greatly aids in understanding thermodynamic relationships and analysing results.

9.1 Polytropic Process Relationship

The first law of thermodynamics:

$$dQ = dU + dW$$

Equation 8-1

Where Q =heat, U =internal energy and W =work. Because:

$$dU = m \cdot C_V \cdot dT$$

Equation 8-2

$$dW = P \cdot dV$$

Equation 8-3

For a fixed mass of gas with adiabatic compression, the first law becomes:

$$0 = m \cdot C_V \cdot dT + P \cdot dV$$

Equation 8-4

Introducing the Ideal Gas Law:

$$PV = mRT$$

$$d(PV) = d(mRT)$$

$$P \cdot dV + V \cdot dP = mRdT$$

$$\frac{P \cdot dV + V \cdot dP}{R} = mdT$$

Equation 8-5

Re-arrange Equation 8-4 to better observe the substitution:

$$0 = C_V \cdot mdT + P \cdot dV$$

Equation 8-6

Substitute Equation 8-5 into Equation 8-6 and manipulate:

$$0 = \frac{C_V \cdot (P \cdot dV + V \cdot dP)}{R} + P \cdot dV$$

$$0 = \frac{C_V}{R} \cdot (P \cdot dV + V \cdot dP) + P \cdot dV$$

Equation 8-7

Deriving a useful relationship for substitution of $\frac{C_V}{R}$:

$$C_P = C_V + R$$

Equation 8-8

$$\frac{C_P}{C_V} = \frac{C_V}{C_V} + \frac{R}{C_V}$$

Equation 8-9

$$\gamma = 1 + \frac{R}{C_V}$$

$$\gamma - 1 = \frac{R}{C_V}$$

$$\frac{1}{\gamma - 1} = \frac{C_V}{R}$$

Equation 8-10

Substitute Equation 8-10 into Equation 8-7:

$$0 = \frac{(P \cdot dV + V \cdot dP)}{\gamma - 1} + P \cdot dV$$

Equation 8-11

Multiply equation by $\gamma - 1$ and collect terms:

$$0 = P \cdot dV + V \cdot dP + P \cdot dV \cdot (\gamma - 1)$$

$$0 = P \cdot dV + V \cdot dP + \gamma \cdot P \cdot dV - P \cdot dV$$

$$0 = V \cdot dP + \gamma \cdot P \cdot dV$$

Equation 8-12

Divide equation by PV :

$$\frac{0}{PV} = \frac{V \cdot dP}{PV} + \frac{\gamma \cdot P \cdot dV}{PV}$$

Equation 8-13

Multiply by PV , re-arrange and the integrate:

$$0 = \frac{dP}{P} + \frac{\gamma \cdot dV}{V}$$

$$\frac{dP}{P} = -\frac{\gamma \cdot dV}{V}$$

$$\int \frac{dP}{P} = -\int \frac{\gamma \cdot dV}{V}$$

Equation 8-14

Because:

$$\left[\int \frac{dx}{x} = \ln x \right]$$

Then:

$$\ln P = -\gamma \ln V + \text{Constant}$$

Equation 8-15

Re-arrange:

$$\ln P = -\ln V^\gamma + \text{Constant}$$

$$\ln P + \ln V^\gamma = \text{Constant}$$

$$\ln PV^\gamma = \text{Constant}$$

Equation 8-16

If

$$\ln PV^\gamma = \text{Constant}$$

Then:

$$PV^\gamma = \text{Constant}$$

Equation 8-17

And finally:

$$\gamma = n$$

If the process is polytropic but not adiabatic.

9.2 T_{UB2}

Polytropic process equation:

$$PV^n = \text{Constant}$$

Equation 8-18

Applying the equation to two states, factorising and rearranging:

$$\therefore P_1 V_1^n = P_2 V_2^n$$

Equation 8-19

$$\left(\frac{V_1}{V_2}\right)^n = \frac{p_2}{p_1}$$

$$\frac{V_1}{V_2} = \left(\frac{p_2}{p_1}\right)^{\frac{1}{n}}$$

Equation 8-20

Introducing the ideal gas law by first manipulating Equation 8-19:

$$p_1 V_1^n = p_2 V_2^n$$

$$(p_1 V_1) V_1^{n-1} = (p_2 V_2) V_2^{n-1}$$

Equation 8-21

This works because:

$$V_1^1 \cdot V_1^{n-1} = V_1^n$$

The ideal gas law written for two states:

$$p_1 V_1 = mRT_1$$

Equation 8-22

$$p_2 V_2 = mRT_2$$

Equation 8-23

Equation 8-22 and Equation 8-23 are now substituted into Equation 8-21 and cancelling terms:

$$(mRT_1) V_1^{n-1} = (mRT_2) V_2^{n-1}$$

$$T_1 V_1^{n-1} = T_2 V_2^{n-1}$$

Equation 8-24

Re-arranging:

$$\left(\frac{V_1}{V_2}\right)^{n-1} = \frac{T_2}{T_1}$$

$$\frac{V_1}{V_2} = \left(\frac{T_2}{T_1}\right)^{\frac{1}{n-1}}$$

Equation 8-25

Equation 8-20 is now substituted into Equation 8-25:

$$\left(\frac{T_2}{T_1}\right)^{\frac{1}{n-1}} = \left(\frac{P_2}{P_1}\right)^{\frac{1}{n}}$$

Equation 8-26

The new expression is now re-arranged to make T_2 the subject.

$$\frac{T_2}{T_1} = \left(\frac{P_2}{P_1}\right)^{\frac{n-1}{n}}$$

$$T_2 = T_1 \left(\frac{P_2}{P_1}\right)^{\frac{n-1}{n}}$$

Equation 8-27

Or in terms of unburned zone temperatures:

$$T_{UB2} = T_{UB1} \left(\frac{P_2}{P_1}\right)^{\frac{n-1}{n}}$$

For an adiabatic process:

$$T_{UB2} = T_{UB1} \left(\frac{P_2}{P_1}\right)^{\frac{\gamma-1}{\gamma}}$$

9.3 $C_p - C_v = R$

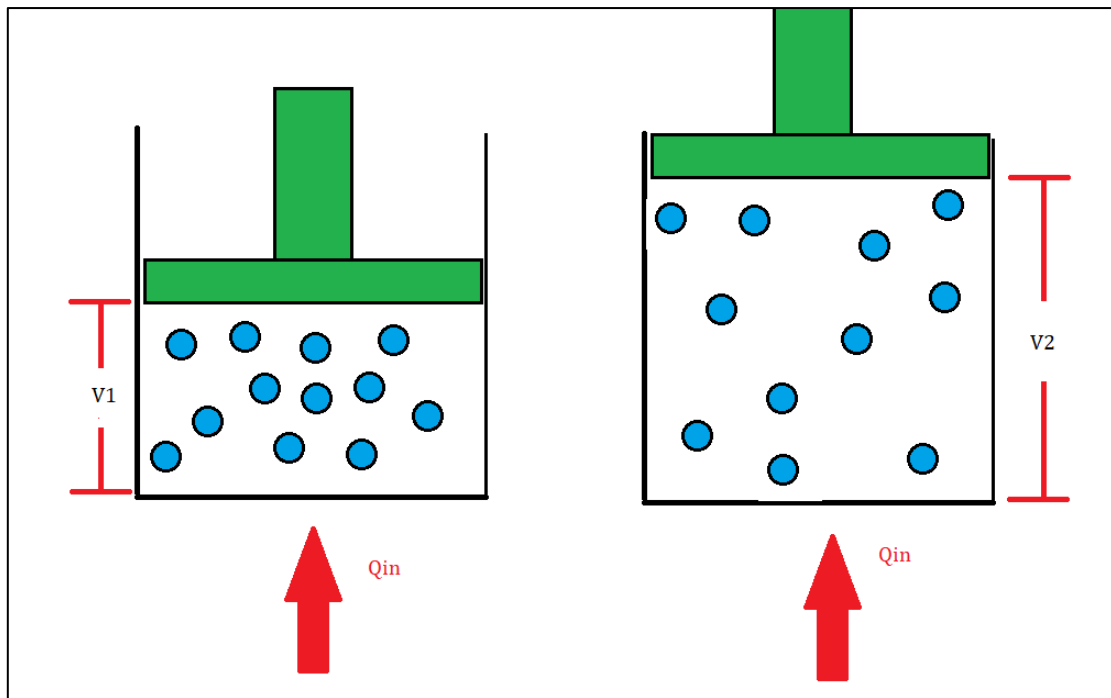


Figure 9-1 - Constant volume process (left), constant pressure process (right)

Fixed piston on left (constant volume), moving piston on right (constant pressure).

The first law:

$$dQ = dU + dW$$

Equation 8-28

$$dU = m \cdot C_v \cdot dT$$

Equation 8-29

Heat capacity at constant volume, with fixed piston, for 1 mole:

$$dQ = C_v \cdot dT + 0$$

Equation 8-30

Where Q is heat, C_v is specific heat at constant volume. Re-arranging:

$$C_v = \frac{dQ}{dT}$$

Equation 8-31

Re-arranging Equation 8-27 :

$$dQ = dU + dW$$

Equation 8-32

Substituting Equation 8-31 into Equation 8-30:

$$C_V = \frac{dU + W}{dT}$$

Equation 8-33

For constant volume, work is zero:

$$C_V = \frac{dU}{dT}$$

Equation 8-34

For a monatomic gas that has 3 degrees of translational freedom, per mole:

$$dU = \frac{3}{2} \tilde{R} \cdot dT$$

Equation 8-35

Where: $\tilde{R}$ =Universal gas constant. Substituting Equation 8-34 into Equation 8-33 and cancelling.

$$C_V = \frac{\frac{3}{2} \tilde{R} \cdot dT}{dT}$$

$$C_V = \frac{3}{2} \tilde{R}$$

Equation 8-36

Or:

$$C_V = \frac{3}{2} R$$

Equation 8-37

Heat capacity at constant pressure:

$$C_P = \frac{dQ}{dT} = \frac{dU + dW}{dT}$$

Equation 8-38

$$dW = p \cdot dV$$

Substituting Equation 8-34 and the expression for work:

$$C_P = \frac{\frac{3}{2}\tilde{R} \cdot dT + p \cdot dV}{dT}$$

Equation 8-39

Ideal Gas Law for 1 mole:

$$pV = \tilde{R}T$$

$$pdV = \tilde{R} \cdot dT$$

Equation 8-40

Substitute Equation 8-39 into Equation 8-38, cancel and collect terms:

$$C_P = \frac{\frac{3}{2}\tilde{R} \cdot dT + \tilde{R} \cdot dT}{dT}$$

$$C_P = \frac{3}{2}\tilde{R} + \tilde{R}$$

$$C_P = \frac{5}{2}\tilde{R}$$

Or:

$$C_P = \frac{5}{2}R$$

Equation 8-41

Subtract Equation 8-36 from Equation 8-40

$$C_P - C_V = R$$

Equation 8-42

$$\gamma = \frac{C_P}{C_V}$$

Equation 8-43

Divide Equation 8-41 by C_V :

$$\frac{C_P}{C_V} - \frac{C_V}{C_V} = \frac{R}{C_V}$$

$$\gamma = 1 + \frac{R}{C_V}$$

Equation 8-44

9.4 Heat Release

$$dQ = dU + PdV$$

$$dQ = mC_V dT + pDV$$

Equation 8-45

Re-arrange the ideal gas law:

$$pV = mRT$$

$$d(PV) = d(mRT)$$

$$pdV + Vdp = mRdT$$

$$mdT = \frac{pdV + Vdp}{R}$$

Equation 8-46

Substitute Equation 8-46 into Equation 8-45:

$$dQ = \frac{C_V}{R}(pdV + Vdp) + pdV$$

Equation 8-47

Manipulating Equation 8-44:

$$\frac{C_V}{R} = \frac{1}{\gamma - 1}$$

Equation 8-48

Substitute Equation 8-48 into Equation 8-47:

$$dQ = \left(\frac{pdV + Vdp}{\gamma - 1} \right) + pdV$$

Equation 8-49

Multiply all terms by $\gamma - 1$ and collect terms:

$$(\gamma - 1)dQ = pdV + Vdp + (\gamma - 1)pdV$$

$$(\gamma - 1)dQ = pdV + Vdp + \gamma pdV - pdV$$

$$(\gamma - 1)dQ = Vdp + \gamma pdV$$

$$dQ = \frac{Vdp + \gamma pdV}{(\gamma - 1)}$$

Equation 8-50

If:

$$pV^n = \text{Constant}$$

Equation 8-51

Then:

$$d(pV^n) = 0$$

$$pnV^{n-1}dv + V^n dp = 0$$

Equation 8-52

Divide by V^{n-1} :

$$pndV + Vdp = 0$$

$$Vdp = -npdV$$

Equation 8-53

Substitute Equation 8-53 into Equation 8-50:

$$dQ = \frac{-npdV + \gamma pdV}{(\gamma - 1)}$$

Equation 8-54

Manipulate:

$$dQ = \frac{(\gamma - n)}{(\gamma - 1)} \cdot pdV$$

Equation 8-55

10 Appendix E – MATLAB Script for Dissociation Model

10.1 Lean Script

[illegible]

```

    dG = G_co(c,1) - G_co2(c,1);
    dG_co2(c,1) = dG;
    c = c + 1;
end

c = 1;
while c <= 11
    Kp = exp((dG_co2(c,1)*(10^3))/(ru*t(c,1)));
    Kp_co2(c,1) = Kp;
    c = c + 1;
end

%%%%%%%%%%%%%% H2O
c = 1;
while c <= 11
    dG = 0 - G_h2o(c,1);
    dG_h2o(c,1) = dG;
    c = c + 1;
end

c = 1;
while c <= 11
    Kp = exp((dG_h2o(c,1)*(10^3))/(ru*t(c,1)));
    Kp_h2o(c,1) = Kp;
    c = c + 1;
end

%%%%%%%%%%%%%% O2
c = 1;
while c <= 11
    dG = G_o(c,1) + G_o(c,1);
    dG_o2(c,1) = dG;

```

```

        c = c + 1;
end

c = 1;

while c <= 11

    Kp = exp((dG_o2(c,1)*(10^3))/(ru*t(c,1)));

    Kp_o2(c,1) = Kp;

    c = c + 1;
end

%Kp_co2;
%Kp_h2o;
%Kp_o2;

ppco = zeros(11,4);

ppco(:,1) = t;
ppco(:,2) = Kp_co2;
ppco(:,3) = Kp_h2o;
ppco(:,4) = Kp_o2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% END OF Partial
Pressure Coefficients

n1t = zeros(8,19);
n2t = zeros(8,19);
n3t = zeros(8,19);
n4t = zeros(8,19);
n5t = zeros(8,19);
n7t = zeros(8,19);

%fer=input('Enter FER:\n');

carbon_stoich = 8;
hydrogen_stoich = 18;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DISSOCIATION 'GUESS'
ITERATIVE METHOD

g = 0.0000001; %Guess
g1 = 0.0000001; %Amount to add to guess

```

```

k = 1; %Fuel Equiv Counter

p = 10; %Pressure

fer = 0.9; %Fuel Equivalence Ratio
c_fer = 0.01; %Amount to add to FER

c = 1; %Counter

    carb = carbon_stoich*fer;
    hyd = hydrogen_stoich*fer;

while fer <= 1

    temp = ppco(k,1);
    kco2 = ppco(k,2);
    kh2o = ppco(k,3);
    ko2 = ppco(k,4);

    gp = g*p; %(n5/nt)*p

    gpp = gp^0.5; %((n5/nt)*p)^1/2

    n1n3 = kco2*gpp; %n1/n3

    n3 = carb/(n1n3+1);

    n1 = carb-n3;

    n2n4 = kh2o*gpp;

    n4 = (hyd/2)/(n2n4+1);

    n2 = (hyd/2)-n4;

    n5 = ((25-(2*n1)-n2-
n3))/(2+(1/((ko2^0.5)*gpp)));

    n6 = 47;

    n7 = n5/((ko2^0.5)*gpp);

    nt = n1+n2+n3+n4+n5+n6+n7;

```

```

n5nt = n5/nt;

err = n5nt-g;

g=g+g1 ;

v=[n1,n2,n3,n4,n5,n6,n7];

if err < 0 %0.00000 default
    p = p + 1;
    result(c,:) = v;
    c=c+1;
    g = 0.00000001; %n5/nt g = 0.000000001
else

end

if p > 100
    p = 10;
    k = k + 1;

    g = 0.00000001;
else
end

if k > 11
    k = 1;
    fer = fer + c_fer;
    carb = carbon_stoich*fer;
    hyd = hydrogen_stoich*fer;
else

end

end

result
size(result,1)

%CHANGE THESE TO size of result divided by number of FER
points. Second
%number is number of columns
m = mat2cell(result,[1001 1001 1001 1001 1001 1001 1001
1001 1001 1001 1001], [7]);

```

```

celldisp(m);

%%%%%%%%%% FER 0.9
r_09 = m{1}
n1_09 = [r_09(1:91,1)'; r_09(92:182,1)';
r_09(183:273,1)'; r_09(274:364,1)'; r_09(365:455,1)';
r_09(456:546,1)';
        r_09(547:637,1)';r_09(638:728,1)'; r_09(729:819,1)';
r_09(820:910,1)'; r_09(911:1001,1)'];

n2_09 = [r_09(1:91,2)'; r_09(92:182,2)';
r_09(183:273,2)'; r_09(274:364,2)'; r_09(365:455,2)';
r_09(456:546,2)';
        r_09(547:637,2)';r_09(638:728,2)'; r_09(729:819,2)';
r_09(820:910,2)'; r_09(911:1001,2)'];

n3_09 = [r_09(1:91,3)'; r_09(92:182,3)';
r_09(183:273,3)'; r_09(274:364,3)'; r_09(365:455,3)';
r_09(456:546,3)';
        r_09(547:637,3)';r_09(638:728,3)'; r_09(729:819,3)';
r_09(820:910,3)'; r_09(911:1001,3)'];

n4_09 = [r_09(1:91,4)'; r_09(92:182,4)';
r_09(183:273,4)'; r_09(274:364,4)'; r_09(365:455,4)';
r_09(456:546,4)';
        r_09(547:637,4)';r_09(638:728,4)'; r_09(729:819,4)';
r_09(820:910,4)'; r_09(911:1001,4)'];

n5_09 = [r_09(1:91,5)'; r_09(92:182,5)';
r_09(183:273,5)'; r_09(274:364,5)'; r_09(365:455,5)';
r_09(456:546,5)';
        r_09(547:637,5)';r_09(638:728,5)'; r_09(729:819,5)';
r_09(820:910,5)'; r_09(911:1001,5)'];

n7_09 = [r_09(1:91,7)'; r_09(92:182,7)';
r_09(183:273,7)'; r_09(274:364,7)'; r_09(365:455,7)';
r_09(456:546,7)';
        r_09(547:637,7)';r_09(638:728,7)'; r_09(729:819,7)';
r_09(820:910,7)'; r_09(911:1001,7)'];

n5_09_poly = zeros(91,8); % 91 rows, 7 columns
n7_09_poly = zeros(91,8);

c = 1;
while c < 92

    n5_09_poly(c,:) = [ polyfit(t,n5_09(:,c),7) ];

```

```

        c = c+1;
end

c = 1;
while c < 92 %%%% Making n7 polynomials

    n7_09_poly(c,:) = [ polyfit(t,n7_09(:,c),7) ];

    c = c+1;
end

% n5_09_poly

%%%%%%%%%%%% FER 0.91

r_91 = m{2};
n1_91 = [r_91(1:91,1)'; r_91(92:182,1)';
r_91(183:273,1)'; r_91(274:364,1)'; r_91(365:455,1)';
r_91(456:546,1)';
    r_91(547:637,1)';r_91(638:728,1)'; r_91(729:819,1)';
r_91(820:910,1)'; r_91(911:1001,1)'];

n2_91 = [r_91(1:91,2)'; r_91(92:182,2)';
r_91(183:273,2)'; r_91(274:364,2)'; r_91(365:455,2)';
r_91(456:546,2)';
    r_91(547:637,2)';r_91(638:728,2)'; r_91(729:819,2)';
r_91(820:910,2)'; r_91(911:1001,2)'];

n3_91 = [r_91(1:91,3)'; r_91(92:182,3)';
r_91(183:273,3)'; r_91(274:364,3)'; r_91(365:455,3)';
r_91(456:546,3)';
    r_91(547:637,3)';r_91(638:728,3)'; r_91(729:819,3)';
r_91(820:910,3)'; r_91(911:1001,3)'];

n4_91 = [r_91(1:91,4)'; r_91(92:182,4)';
r_91(183:273,4)'; r_91(274:364,4)'; r_91(365:455,4)';
r_91(456:546,4)';
    r_91(547:637,4)';r_91(638:728,4)'; r_91(729:819,4)';
r_91(820:910,4)'; r_91(911:1001,4)'];

n5_91 = [r_91(1:91,5)'; r_91(92:182,5)';
r_91(183:273,5)'; r_91(274:364,5)'; r_91(365:455,5)';
r_91(456:546,5)';
    r_91(547:637,5)';r_91(638:728,5)'; r_91(729:819,5)';
r_91(820:910,5)'; r_91(911:1001,5)'];

```

```

n7_91 = [r_91(1:91,7)'; r_91(92:182,7)';
r_91(183:273,7)'; r_91(274:364,7)'; r_91(365:455,7)';
r_91(456:546,7)';
r_91(547:637,7)';r_91(638:728,7)'; r_91(729:819,7)';
r_91(820:910,7)'; r_91(911:1001,7)'];

n5_91_poly = zeros(91,8); % 91 rows, 7 columns
n7_91_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_91_poly(c,:) = [ polyfit(t,n5_91(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_91_poly(c,:) = [ polyfit(t,n7_91(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 0.92
r_92 = m{3};
n1_92 = [r_92(1:91,1)'; r_92(92:182,1)';
r_92(183:273,1)'; r_92(274:364,1)'; r_92(365:455,1)';
r_92(456:546,1)';
r_92(547:637,1)';r_92(638:728,1)'; r_92(729:819,1)';
r_92(820:910,1)'; r_92(911:1001,1)'];

n2_92 = [r_92(1:91,2)'; r_92(92:182,2)';
r_92(183:273,2)'; r_92(274:364,2)'; r_92(365:455,2)';
r_92(456:546,2)';
r_92(547:637,2)';r_92(638:728,2)'; r_92(729:819,2)';
r_92(820:910,2)'; r_92(911:1001,2)'];

n3_92 = [r_92(1:91,3)'; r_92(92:182,3)';
r_92(183:273,3)'; r_92(274:364,3)'; r_92(365:455,3)';
r_92(456:546,3)';
r_92(547:637,3)';r_92(638:728,3)'; r_92(729:819,3)';
r_92(820:910,3)'; r_92(911:1001,3)'];

n4_92 = [r_92(1:91,4)'; r_92(92:182,4)';
r_92(183:273,4)'; r_92(274:364,4)'; r_92(365:455,4)';
r_92(456:546,4)'];

```

```

    r_92(547:637,4)';r_92(638:728,4)'; r_92(729:819,4)';
r_92(820:910,4)'; r_92(911:1001,4)'];

n5_92 = [r_92(1:91,5)'; r_92(92:182,5)';
r_92(183:273,5)'; r_92(274:364,5)'; r_92(365:455,5)';
r_92(456:546,5)';
    r_92(547:637,5)';r_92(638:728,5)'; r_92(729:819,5)';
r_92(820:910,5)'; r_92(911:1001,5)'];

n7_92 = [r_92(1:91,7)'; r_92(92:182,7)';
r_92(183:273,7)'; r_92(274:364,7)'; r_92(365:455,7)';
r_92(456:546,7)';
    r_92(547:637,7)';r_92(638:728,7)'; r_92(729:819,7)';
r_92(820:910,7)'; r_92(911:1001,7)'];

n5_92_poly = zeros(91,8); % 91 rows, 7 columns
n7_92_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_92_poly(c,:) = [ polyfit(t,n5_92(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_92_poly(c,:) = [ polyfit(t,n7_92(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 0.93
r_93 = m{4};

n1_93 = [r_93(1:91,1)'; r_93(92:182,1)';
r_93(183:273,1)'; r_93(274:364,1)'; r_93(365:455,1)';
r_93(456:546,1)';
    r_93(547:637,1)';r_93(638:728,1)'; r_93(729:819,1)';
r_93(820:910,1)'; r_93(911:1001,1)'];

n2_93 = [r_93(1:91,2)'; r_93(92:182,2)';
r_93(183:273,2)'; r_93(274:364,2)'; r_93(365:455,2)';
r_93(456:546,2)';
    r_93(547:637,2)';r_93(638:728,2)'; r_93(729:819,2)';
r_93(820:910,2)'; r_93(911:1001,2)'];

```

```
n3_93 = [r_93(1:91,3)'; r_93(92:182,3)';  
r_93(183:273,3)'; r_93(274:364,3)'; r_93(365:455,3)';  
r_93(456:546,3)';  
r_93(547:637,3)';r_93(638:728,3)'; r_93(729:819,3)';  
r_93(820:910,3)'; r_93(911:1001,3)'];  
  
n4_93 = [r_93(1:91,4)'; r_93(92:182,4)';  
r_93(183:273,4)'; r_93(274:364,4)'; r_93(365:455,4)';  
r_93(456:546,4)';  
r_93(547:637,4)';r_93(638:728,4)'; r_93(729:819,4)';  
r_93(820:910,4)'; r_93(911:1001,4)'];  
  
n5_93 = [r_93(1:91,5)'; r_93(92:182,5)';  
r_93(183:273,5)'; r_93(274:364,5)'; r_93(365:455,5)';  
r_93(456:546,5)';  
r_93(547:637,5)';r_93(638:728,5)'; r_93(729:819,5)';  
r_93(820:910,5)'; r_93(911:1001,5)'];  
  
n7_93 = [r_93(1:91,7)'; r_93(92:182,7)';  
r_93(183:273,7)'; r_93(274:364,7)'; r_93(365:455,7)';  
r_93(456:546,7)';  
r_93(547:637,7)';r_93(638:728,7)'; r_93(729:819,7)';  
r_93(820:910,7)'; r_93(911:1001,7)'];  
  
n5_93_poly = zeros(91,8); % 91 rows, 7 columns  
n7_93_poly = zeros(91,8);  
  
c = 1;  
while c < 92 %%% Making n5 polynomials  
  
    n5_93_poly(c,:) = [ polyfit(t,n5_93(:,c),7) ];  
  
    c = c+1;  
end  
  
c = 1;  
while c < 92 %%% Making n7 polynomials  
  
    n7_93_poly(c,:) = [ polyfit(t,n7_93(:,c),7) ];  
  
    c = c+1;  
end  
  
%%%%%%%%%% FER 0.94  
r_94 = m{5};
```

```

n1_94 = [r_94(1:91,1)'; r_94(92:182,1)';
r_94(183:273,1)'; r_94(274:364,1)'; r_94(365:455,1)';
r_94(456:546,1)';
r_94(547:637,1)'; r_94(638:728,1)'; r_94(729:819,1)';
r_94(820:910,1)'; r_94(911:1001,1)'];

n2_94 = [r_94(1:91,2)'; r_94(92:182,2)';
r_94(183:273,2)'; r_94(274:364,2)'; r_94(365:455,2)';
r_94(456:546,2)';
r_94(547:637,2)'; r_94(638:728,2)'; r_94(729:819,2)';
r_94(820:910,2)'; r_94(911:1001,2)'];

n3_94 = [r_94(1:91,3)'; r_94(92:182,3)';
r_94(183:273,3)'; r_94(274:364,3)'; r_94(365:455,3)';
r_94(456:546,3)';
r_94(547:637,3)'; r_94(638:728,3)'; r_94(729:819,3)';
r_94(820:910,3)'; r_94(911:1001,3)'];

n4_94 = [r_94(1:91,4)'; r_94(92:182,4)';
r_94(183:273,4)'; r_94(274:364,4)'; r_94(365:455,4)';
r_94(456:546,4)';
r_94(547:637,4)'; r_94(638:728,4)'; r_94(729:819,4)';
r_94(820:910,4)'; r_94(911:1001,4)'];

n5_94 = [r_94(1:91,5)'; r_94(92:182,5)';
r_94(183:273,5)'; r_94(274:364,5)'; r_94(365:455,5)';
r_94(456:546,5)';
r_94(547:637,5)'; r_94(638:728,5)'; r_94(729:819,5)';
r_94(820:910,5)'; r_94(911:1001,5)'];

n7_94 = [r_94(1:91,7)'; r_94(92:182,7)';
r_94(183:273,7)'; r_94(274:364,7)'; r_94(365:455,7)';
r_94(456:546,7)';
r_94(547:637,7)'; r_94(638:728,7)'; r_94(729:819,7)';
r_94(820:910,7)'; r_94(911:1001,7)'];

n5_94_poly = zeros(91,8); % 91 rows, 7 columns
n7_94_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_94_poly(c,:) = [ polyfit(t,n5_94(:,c),7) ];

    c = c+1;
end

c = 1;

```

```

while c < 92 %%% Making n7 polynomials

    n7_94_poly(c,:) = [ polyfit(t,n7_94(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 0.95
r_95 = m{6};

n1_95 = [r_95(1:91,1)'; r_95(92:182,1)';
r_95(183:273,1)'; r_95(274:364,1)'; r_95(365:455,1)';
r_95(456:546,1)';
r_95(547:637,1)';r_95(638:728,1)'; r_95(729:819,1)';
r_95(820:910,1)'; r_95(911:1001,1)'];

n2_95 = [r_95(1:91,2)'; r_95(92:182,2)';
r_95(183:273,2)'; r_95(274:364,2)'; r_95(365:455,2)';
r_95(456:546,2)';
r_95(547:637,2)';r_95(638:728,2)'; r_95(729:819,2)';
r_95(820:910,2)'; r_95(911:1001,2)'];

n3_95 = [r_95(1:91,3)'; r_95(92:182,3)';
r_95(183:273,3)'; r_95(274:364,3)'; r_95(365:455,3)';
r_95(456:546,3)';
r_95(547:637,3)';r_95(638:728,3)'; r_95(729:819,3)';
r_95(820:910,3)'; r_95(911:1001,3)'];

n4_95 = [r_95(1:91,4)'; r_95(92:182,4)';
r_95(183:273,4)'; r_95(274:364,4)'; r_95(365:455,4)';
r_95(456:546,4)';
r_95(547:637,4)';r_95(638:728,4)'; r_95(729:819,4)';
r_95(820:910,4)'; r_95(911:1001,4)'];

n5_95 = [r_95(1:91,5)'; r_95(92:182,5)';
r_95(183:273,5)'; r_95(274:364,5)'; r_95(365:455,5)';
r_95(456:546,5)';
r_95(547:637,5)';r_95(638:728,5)'; r_95(729:819,5)';
r_95(820:910,5)'; r_95(911:1001,5)'];

n7_95 = [r_95(1:91,7)'; r_95(92:182,7)';
r_95(183:273,7)'; r_95(274:364,7)'; r_95(365:455,7)';
r_95(456:546,7)';
r_95(547:637,7)';r_95(638:728,7)'; r_95(729:819,7)';
r_95(820:910,7)'; r_95(911:1001,7)'];

n5_95_poly = zeros(91,8); % 91 rows, 7 columns
n7_95_poly = zeros(91,8);

```

```

c = 1;
while c < 92 %%%% Making n5 polynomials

    n5_95_poly(c,:) = [ polyfit(t,n5_95(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%%% Making n7 polynomials

    n7_95_poly(c,:) = [ polyfit(t,n7_95(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%% FER 0.96
r_96 = m{7};
n1_96 = [r_96(1:91,1)'; r_96(92:182,1)';
r_96(183:273,1)'; r_96(274:364,1)'; r_96(365:455,1)';
r_96(456:546,1)';
    r_96(547:637,1)';r_96(638:728,1)'; r_96(729:819,1)';
r_96(820:910,1)'; r_96(911:1001,1)'];

n2_96 = [r_96(1:91,2)'; r_96(92:182,2)';
r_96(183:273,2)'; r_96(274:364,2)'; r_96(365:455,2)';
r_96(456:546,2)';
    r_96(547:637,2)';r_96(638:728,2)'; r_96(729:819,2)';
r_96(820:910,2)'; r_96(911:1001,2)'];

n3_96 = [r_96(1:91,3)'; r_96(92:182,3)';
r_96(183:273,3)'; r_96(274:364,3)'; r_96(365:455,3)';
r_96(456:546,3)';
    r_96(547:637,3)';r_96(638:728,3)'; r_96(729:819,3)';
r_96(820:910,3)'; r_96(911:1001,3)'];

n4_96 = [r_96(1:91,4)'; r_96(92:182,4)';
r_96(183:273,4)'; r_96(274:364,4)'; r_96(365:455,4)';
r_96(456:546,4)';
    r_96(547:637,4)';r_96(638:728,4)'; r_96(729:819,4)';
r_96(820:910,4)'; r_96(911:1001,4)'];

n5_96 = [r_96(1:91,5)'; r_96(92:182,5)';
r_96(183:273,5)'; r_96(274:364,5)'; r_96(365:455,5)';
r_96(456:546,5)';
    r_96(547:637,5)';r_96(638:728,5)'; r_96(729:819,5)';
r_96(820:910,5)'; r_96(911:1001,5)'];

```

```

n7_96 = [r_96(1:91,7)'; r_96(92:182,7)';
r_96(183:273,7)'; r_96(274:364,7)'; r_96(365:455,7)';
r_96(456:546,7)';
r_96(547:637,7)';r_96(638:728,7)'; r_96(729:819,7)';
r_96(820:910,7)'; r_96(911:1001,7)'];

n5_96_poly = zeros(91,8); % 91 rows, 7 columns
n7_96_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_96_poly(c,:) = [ polyfit(t,n5_96(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_96_poly(c,:) = [ polyfit(t,n7_96(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 0.97
r_97 = m{8};
n1_97 = [r_97(1:91,1)'; r_97(92:182,1)';
r_97(183:273,1)'; r_97(274:364,1)'; r_97(365:455,1)';
r_97(456:546,1)';
r_97(547:637,1)';r_97(638:728,1)'; r_97(729:819,1)';
r_97(820:910,1)'; r_97(911:1001,1)'];

n2_97 = [r_97(1:91,2)'; r_97(92:182,2)';
r_97(183:273,2)'; r_97(274:364,2)'; r_97(365:455,2)';
r_97(456:546,2)';
r_97(547:637,2)';r_97(638:728,2)'; r_97(729:819,2)';
r_97(820:910,2)'; r_97(911:1001,2)'];

n3_97 = [r_97(1:91,3)'; r_97(92:182,3)';
r_97(183:273,3)'; r_97(274:364,3)'; r_97(365:455,3)';
r_97(456:546,3)';
r_97(547:637,3)';r_97(638:728,3)'; r_97(729:819,3)';
r_97(820:910,3)'; r_97(911:1001,3)'];

```

```

n4_97 = [r_97(1:91,4)'; r_97(92:182,4)';
r_97(183:273,4)'; r_97(274:364,4)'; r_97(365:455,4)';
r_97(456:546,4)';
r_97(547:637,4)';r_97(638:728,4)'; r_97(729:819,4)';
r_97(820:910,4)'; r_97(911:1001,4)'];

n5_97 = [r_97(1:91,5)'; r_97(92:182,5)';
r_97(183:273,5)'; r_97(274:364,5)'; r_97(365:455,5)';
r_97(456:546,5)';
r_97(547:637,5)';r_97(638:728,5)'; r_97(729:819,5)';
r_97(820:910,5)'; r_97(911:1001,5)'];

n7_97 = [r_97(1:91,7)'; r_97(92:182,7)';
r_97(183:273,7)'; r_97(274:364,7)'; r_97(365:455,7)';
r_97(456:546,7)';
r_97(547:637,7)';r_97(638:728,7)'; r_97(729:819,7)';
r_97(820:910,7)'; r_97(911:1001,7)'];

n5_97_poly = zeros(91,8); % 91 rows, 7 columns
n7_97_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_97_poly(c,:) = [ polyfit(t,n5_97(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_97_poly(c,:) = [ polyfit(t,n7_97(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 0.98
r_98 = m{9};
n1_98 = [r_98(1:91,1)'; r_98(92:182,1)';
r_98(183:273,1)'; r_98(274:364,1)'; r_98(365:455,1)';
r_98(456:546,1)';
r_98(547:637,1)';r_98(638:728,1)'; r_98(729:819,1)';
r_98(820:910,1)'; r_98(911:1001,1)'];

n2_98 = [r_98(1:91,2)'; r_98(92:182,2)';
r_98(183:273,2)'; r_98(274:364,2)'; r_98(365:455,2)';
r_98(456:546,2)'];

```

```

    r_98(547:637,2)';r_98(638:728,2)'; r_98(729:819,2)';
r_98(820:910,2)'; r_98(911:1001,2)'];

n3_98 = [r_98(1:91,3)'; r_98(92:182,3)';
r_98(183:273,3)'; r_98(274:364,3)'; r_98(365:455,3)';
r_98(456:546,3)';
    r_98(547:637,3)';r_98(638:728,3)'; r_98(729:819,3)';
r_98(820:910,3)'; r_98(911:1001,3)'];

n4_98 = [r_98(1:91,4)'; r_98(92:182,4)';
r_98(183:273,4)'; r_98(274:364,4)'; r_98(365:455,4)';
r_98(456:546,4)';
    r_98(547:637,4)';r_98(638:728,4)'; r_98(729:819,4)';
r_98(820:910,4)'; r_98(911:1001,4)'];

n5_98 = [r_98(1:91,5)'; r_98(92:182,5)';
r_98(183:273,5)'; r_98(274:364,5)'; r_98(365:455,5)';
r_98(456:546,5)';
    r_98(547:637,5)';r_98(638:728,5)'; r_98(729:819,5)';
r_98(820:910,5)'; r_98(911:1001,5)'];

n7_98 = [r_98(1:91,7)'; r_98(92:182,7)';
r_98(183:273,7)'; r_98(274:364,7)'; r_98(365:455,7)';
r_98(456:546,7)';
    r_98(547:637,7)';r_98(638:728,7)'; r_98(729:819,7)';
r_98(820:910,7)'; r_98(911:1001,7)'];

n5_98_poly = zeros(91,8); % 91 rows, 7 columns
n7_98_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_98_poly(c,:) = [ polyfit(t,n5_98(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_98_poly(c,:) = [ polyfit(t,n7_98(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%% FER 0.99
r_99 = m{10};

```

```

n1_99 = [r_99(1:91,1)'; r_99(92:182,1)';
r_99(183:273,1)'; r_99(274:364,1)'; r_99(365:455,1)';
r_99(456:546,1)';
r_99(547:637,1)';r_99(638:728,1)'; r_99(729:819,1)';
r_99(820:910,1)'; r_99(911:1001,1)'];

n2_99 = [r_99(1:91,2)'; r_99(92:182,2)';
r_99(183:273,2)'; r_99(274:364,2)'; r_99(365:455,2)';
r_99(456:546,2)';
r_99(547:637,2)';r_99(638:728,2)'; r_99(729:819,2)';
r_99(820:910,2)'; r_99(911:1001,2)'];

n3_99 = [r_99(1:91,3)'; r_99(92:182,3)';
r_99(183:273,3)'; r_99(274:364,3)'; r_99(365:455,3)';
r_99(456:546,3)';
r_99(547:637,3)';r_99(638:728,3)'; r_99(729:819,3)';
r_99(820:910,3)'; r_99(911:1001,3)'];

n4_99 = [r_99(1:91,4)'; r_99(92:182,4)';
r_99(183:273,4)'; r_99(274:364,4)'; r_99(365:455,4)';
r_99(456:546,4)';
r_99(547:637,4)';r_99(638:728,4)'; r_99(729:819,4)';
r_99(820:910,4)'; r_99(911:1001,4)'];

n5_99 = [r_99(1:91,5)'; r_99(92:182,5)';
r_99(183:273,5)'; r_99(274:364,5)'; r_99(365:455,5)';
r_99(456:546,5)';
r_99(547:637,5)';r_99(638:728,5)'; r_99(729:819,5)';
r_99(820:910,5)'; r_99(911:1001,5)'];

n7_99 = [r_99(1:91,7)'; r_99(92:182,7)';
r_99(183:273,7)'; r_99(274:364,7)'; r_99(365:455,7)';
r_99(456:546,7)';
r_99(547:637,7)';r_99(638:728,7)'; r_99(729:819,7)';
r_99(820:910,7)'; r_99(911:1001,7)'];

n5_99_poly = zeros(91,8); % 91 rows, 7 columns
n7_99_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_99_poly(c,:) = [ polyfit(t,n5_99(:,c),7) ];

    c = c+1;
end

c = 1;

```

```

while c < 92 %%% Making n7 polynomials

    n7_99_poly(c,:) = [ polyfit(t,n7_99(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 1
r_1 = m{11};
n1_1 = [r_1(1:91,1)'; r_1(92:182,1)'; r_1(183:273,1)';
        r_1(274:364,1)'; r_1(365:455,1)'; r_1(456:546,1)';
        r_1(547:637,1)'; r_1(638:728,1)'; r_1(729:819,1)';
        r_1(820:910,1)'; r_1(911:1001,1)'];

n2_1 = [r_1(1:91,2)'; r_1(92:182,2)'; r_1(183:273,2)';
        r_1(274:364,2)'; r_1(365:455,2)'; r_1(456:546,2)';
        r_1(547:637,2)'; r_1(638:728,2)'; r_1(729:819,2)';
        r_1(820:910,2)'; r_1(911:1001,2)'];

n3_1 = [r_1(1:91,3)'; r_1(92:182,3)'; r_1(183:273,3)';
        r_1(274:364,3)'; r_1(365:455,3)'; r_1(456:546,3)';
        r_1(547:637,3)'; r_1(638:728,3)'; r_1(729:819,3)';
        r_1(820:910,3)'; r_1(911:1001,3)'];

n4_1 = [r_1(1:91,4)'; r_1(92:182,4)'; r_1(183:273,4)';
        r_1(274:364,4)'; r_1(365:455,4)'; r_1(456:546,4)';
        r_1(547:637,4)'; r_1(638:728,4)'; r_1(729:819,4)';
        r_1(820:910,4)'; r_1(911:1001,4)'];

n5_1 = [r_1(1:91,5)'; r_1(92:182,5)'; r_1(183:273,5)';
        r_1(274:364,5)'; r_1(365:455,5)'; r_1(456:546,5)';
        r_1(547:637,5)'; r_1(638:728,5)'; r_1(729:819,5)';
        r_1(820:910,5)'; r_1(911:1001,5)'];

n7_1 = [r_1(1:91,7)'; r_1(92:182,7)'; r_1(183:273,7)';
        r_1(274:364,7)'; r_1(365:455,7)'; r_1(456:546,7)';
        r_1(547:637,7)'; r_1(638:728,7)'; r_1(729:819,7)';
        r_1(820:910,7)'; r_1(911:1001,7)'];

n5_1_poly = zeros(91,8); % 91 rows, 7 columns
n7_1_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_1_poly(c,:) = [ polyfit(t,n5_1(:,c),7) ];

    c = c+1;

```

```
end

c = 1;
while c < 92 %%% Making n7 polynomials
    n7_1_poly(c,:) = [ polyfit(t,n7_1(:,c),7) ];
    c = c+1;
end
```

10.2 Rich Script

```

clc
clear all

format shortg
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% START OF
Partial Pressure Coefficients
t = [2000:100:3000];

t = t(:)

dG_co2 = zeros(11,1);
Kp_co2 = zeros(11,1);

dG_h2o = zeros(11,1);
Kp_h2o = zeros(11,1);

dG_o2 = zeros(11,1);
Kp_o2 = zeros(11,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gibbs Formation taken from Glassman & Yetter

G_co2 = [-396.333; -396.304; -396.232; -396.209; -
396.142; -396.062; -395.969; -395.862; -395.742; -
395.609; -395.461];

G_co = [-286.034; -294.372; -302.672; -310.936; -319.164;
-327.356; -335.514; -343.638; -351.729; -359.789; -
367.816];

G_h2o = [-135.528; -129.721; -123.905; -118.082; -
112.252; -106.416; -100.575; -94.729; -88.878; -83.023; -
77.163];

G_o = [121.552; 114.860; 108.159; 101.450; 94.734;
88.012; 81.284; 74.551; 67.814; 61.072; 54.327];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CO2
c = 1;

while c <= 11

    dG = G_co(c,1) - G_co2(c,1);

    dG_co2(c,1) = dG;

    c = c + 1;

```

```

end

c = 1;

while c <= 11

    Kp = exp((dG_co2(c,1)*(10^6))/(8314*t(c,1)));

    Kp_co2(c,1) = Kp;

    c = c + 1;
end

%%%%%%%%%%%%%% H2O
c = 1;

while c <= 11

    dG = 0 - G_h2o(c,1);

    dG_h2o(c,1) = dG;

    c = c + 1;
end

c = 1;

while c <= 11

    Kp = exp((dG_h2o(c,1)*(10^6))/(8314*t(c,1)));

    Kp_h2o(c,1) = Kp;

    c = c + 1;
end

%%%%%%%%%%%%%% O2
c = 1;

while c <= 11

    dG = G_o(c,1) + G_o(c,1);

    dG_o2(c,1) = dG;

    c = c + 1;
end

c = 1;

```

```

while c <= 11

    Kp = exp((dG_o2(c,1)*(10^6))/(8314*t(c,1)));

    Kp_o2(c,1) = Kp;

    c = c + 1;
end

%Kp_co2;
%Kp_h2o;
%Kp_o2;

ppco = zeros(11,4);

ppco(:,1) = t;
ppco(:,2) = Kp_co2;
ppco(:,3) = Kp_h2o;
ppco(:,4) = Kp_o2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% END OF Partial
Pressure Coefficients

n1t = zeros(8,19);
n2t = zeros(8,19);
n3t = zeros(8,19);
n4t = zeros(8,19);
n5t = zeros(8,19);
n7t = zeros(8,19);

%fer=input('Enter FER:\n');

carbon_stoich = 8;
hydrogen_stoich = 18;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DISSOCIATION 'GUESS'
ITERATIVE METHOD

g = 0.0000000001; %Guess
g1 = 0.0000000001; %Amount to add to guess

k = 1; %Fuel Equiv Counter

p = 10; %Pressure

```

```

fer = 1.01; %Fuel Equivalence Ratio
c_fer = 0.01; %Amount to add to FER

c = 1; %Counter

    carb = carbon_stoich*fer;
    hyd = hydrogen_stoich*fer;

while fer <= 1.11

    temp = ppco(k,1);
    kco2 = ppco(k,2);
    kh2o = ppco(k,3);
    ko2 = ppco(k,4);

        gp = g*p; %(n5/nt)*p

        gpp = gp^0.5; %((n5/nt)*p)^1/2

        n1n3 = kco2*gpp; %n1/n3

        n3 = carb/(n1n3+1);

        n1 = carb-n3;

        n2n4 = kh2o*gpp;

        n4 = (hyd/2)/(n2n4+1);

        n2 = (hyd/2)-n4;

        n5 = ((25-(2*n1)-n2-
n3))/(2+(1/((ko2^0.5)*gpp)));

        n6 = 47;

        n7 = n5/((ko2^0.5)*gpp);

        nt = n1+n2+n3+n4+n5+n6+n7;

        n5nt = n5/nt;

        err = n5nt-g;

        g=g+g1 ;

```

```

v=[n1,n2,n3,n4,n5,n6,n7];

if err < 0 %0.00000 default
    p = p + 1;
    result(c,:) = v;
    c=c+1;
    g = 0.0000000001; %n5/nt g = 0.0000000001
else

end

if p > 100
    p = 10;
    k = k + 1;

    g = 0.0000000001;
else
end

if k > 11
    k = 1;
    fer = fer + c_fer;
    carb = carbon_stoich*fer;
    hyd = hydrogen_stoich*fer;
else

end

end

result
size(result,1)

%CHANGE THESE TO size of result divided by number of FER
points. Second
%number is number of columns
m = mat2cell(result,[1001 1001 1001 1001 1001 1001 1001
1001 1001 1001 1001], [7]);

celldisp(m);
%%%%%%%%%% FER 1.01
r_01 = m{1}

```

```

n1_01 = [r_01(1:91,1)'; r_01(92:182,1)';
r_01(183:273,1)'; r_01(274:364,1)'; r_01(365:455,1)';
r_01(456:546,1)';
r_01(547:637,1)'; r_01(638:728,1)'; r_01(729:819,1)';
r_01(820:910,1)'; r_01(911:1001,1)'];

n2_01 = [r_01(1:91,2)'; r_01(92:182,2)';
r_01(183:273,2)'; r_01(274:364,2)'; r_01(365:455,2)';
r_01(456:546,2)';
r_01(547:637,2)'; r_01(638:728,2)'; r_01(729:819,2)';
r_01(820:910,2)'; r_01(911:1001,2)'];

n3_01 = [r_01(1:91,3)'; r_01(92:182,3)';
r_01(183:273,3)'; r_01(274:364,3)'; r_01(365:455,3)';
r_01(456:546,3)';
r_01(547:637,3)'; r_01(638:728,3)'; r_01(729:819,3)';
r_01(820:910,3)'; r_01(911:1001,3)'];

n4_01 = [r_01(1:91,4)'; r_01(92:182,4)';
r_01(183:273,4)'; r_01(274:364,4)'; r_01(365:455,4)';
r_01(456:546,4)';
r_01(547:637,4)'; r_01(638:728,4)'; r_01(729:819,4)';
r_01(820:910,4)'; r_01(911:1001,4)'];

n5_01 = [r_01(1:91,5)'; r_01(92:182,5)';
r_01(183:273,5)'; r_01(274:364,5)'; r_01(365:455,5)';
r_01(456:546,5)';
r_01(547:637,5)'; r_01(638:728,5)'; r_01(729:819,5)';
r_01(820:910,5)'; r_01(911:1001,5)'];

n7_01 = [r_01(1:91,7)'; r_01(92:182,7)';
r_01(183:273,7)'; r_01(274:364,7)'; r_01(365:455,7)';
r_01(456:546,7)';
r_01(547:637,7)'; r_01(638:728,7)'; r_01(729:819,7)';
r_01(820:910,7)'; r_01(911:1001,7)'];

n5_01_poly = zeros(91,8); % 91 rows, 7 columns
n7_01_poly = zeros(91,8);

c = 1;
while c < 92

    n5_01_poly(c,:) = [ polyfit(t,n5_01(:,c),7) ];

    c = c+1;
end

c = 1;

```

```

while c < 92 %%% Making n7 polynomials

    n7_01_poly(c,:) = [ polyfit(t,n7_01(:,c),7) ];

    c = c+1;
end

% n5_01_poly

%%%%%%%%%% FER 1.02

r_02 = m{2};
n1_02 = [r_02(1:91,1)'; r_02(92:182,1)';
r_02(183:273,1)'; r_02(274:364,1)'; r_02(365:455,1)';
r_02(456:546,1)';
    r_02(547:637,1)';r_02(638:728,1)'; r_02(729:819,1)';
r_02(820:910,1)'; r_02(911:1001,1)'];

n2_02 = [r_02(1:91,2)'; r_02(92:182,2)';
r_02(183:273,2)'; r_02(274:364,2)'; r_02(365:455,2)';
r_02(456:546,2)';
    r_02(547:637,2)';r_02(638:728,2)'; r_02(729:819,2)';
r_02(820:910,2)'; r_02(911:1001,2)'];

n3_02 = [r_02(1:91,3)'; r_02(92:182,3)';
r_02(183:273,3)'; r_02(274:364,3)'; r_02(365:455,3)';
r_02(456:546,3)';
    r_02(547:637,3)';r_02(638:728,3)'; r_02(729:819,3)';
r_02(820:910,3)'; r_02(911:1001,3)'];

n4_02 = [r_02(1:91,4)'; r_02(92:182,4)';
r_02(183:273,4)'; r_02(274:364,4)'; r_02(365:455,4)';
r_02(456:546,4)';
    r_02(547:637,4)';r_02(638:728,4)'; r_02(729:819,4)';
r_02(820:910,4)'; r_02(911:1001,4)'];

n5_02 = [r_02(1:91,5)'; r_02(92:182,5)';
r_02(183:273,5)'; r_02(274:364,5)'; r_02(365:455,5)';
r_02(456:546,5)';
    r_02(547:637,5)';r_02(638:728,5)'; r_02(729:819,5)';
r_02(820:910,5)'; r_02(911:1001,5)'];

n7_02 = [r_02(1:91,7)'; r_02(92:182,7)';
r_02(183:273,7)'; r_02(274:364,7)'; r_02(365:455,7)';
r_02(456:546,7)';
    r_02(547:637,7)';r_02(638:728,7)'; r_02(729:819,7)';
r_02(820:910,7)'; r_02(911:1001,7)'];

```

```

n5_02_poly = zeros(91,8); % 91 rows, 7 columns
n7_02_poly = zeros(91,8);

c = 1;
while c < 92 %%%% Making n5 polynomials

    n5_02_poly(c,:) = [ polyfit(t,n5_02(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%%% Making n7 polynomials

    n7_02_poly(c,:) = [ polyfit(t,n7_02(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%% FER 1.03
r_03 = m{3};
n1_03 = [r_03(1:91,1)'; r_03(92:182,1)';
r_03(183:273,1)'; r_03(274:364,1)'; r_03(365:455,1)';
r_03(456:546,1)';
    r_03(547:637,1)';r_03(638:728,1)'; r_03(729:819,1)';
r_03(820:910,1)'; r_03(911:1001,1)'];

n2_03 = [r_03(1:91,2)'; r_03(92:182,2)';
r_03(183:273,2)'; r_03(274:364,2)'; r_03(365:455,2)';
r_03(456:546,2)';
    r_03(547:637,2)';r_03(638:728,2)'; r_03(729:819,2)';
r_03(820:910,2)'; r_03(911:1001,2)'];

n3_03 = [r_03(1:91,3)'; r_03(92:182,3)';
r_03(183:273,3)'; r_03(274:364,3)'; r_03(365:455,3)';
r_03(456:546,3)';
    r_03(547:637,3)';r_03(638:728,3)'; r_03(729:819,3)';
r_03(820:910,3)'; r_03(911:1001,3)'];

n4_03 = [r_03(1:91,4)'; r_03(92:182,4)';
r_03(183:273,4)'; r_03(274:364,4)'; r_03(365:455,4)';
r_03(456:546,4)';
    r_03(547:637,4)';r_03(638:728,4)'; r_03(729:819,4)';
r_03(820:910,4)'; r_03(911:1001,4)'];

```

```

n5_03 = [r_03(1:91,5)'; r_03(92:182,5)';
r_03(183:273,5)'; r_03(274:364,5)'; r_03(365:455,5)';
r_03(456:546,5)';
r_03(547:637,5)';r_03(638:728,5)'; r_03(729:819,5)';
r_03(820:910,5)'; r_03(911:1001,5)'];

n7_03 = [r_03(1:91,7)'; r_03(92:182,7)';
r_03(183:273,7)'; r_03(274:364,7)'; r_03(365:455,7)';
r_03(456:546,7)';
r_03(547:637,7)';r_03(638:728,7)'; r_03(729:819,7)';
r_03(820:910,7)'; r_03(911:1001,7)'];

n5_03_poly = zeros(91,8); % 91 rows, 7 columns
n7_03_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_03_poly(c,:) = [ polyfit(t,n5_03(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_03_poly(c,:) = [ polyfit(t,n7_03(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%% FER 1.04
r_04 = m{4};

n1_04 = [r_04(1:91,1)'; r_04(92:182,1)';
r_04(183:273,1)'; r_04(274:364,1)'; r_04(365:455,1)';
r_04(456:546,1)';
r_04(547:637,1)';r_04(638:728,1)'; r_04(729:819,1)';
r_04(820:910,1)'; r_04(911:1001,1)'];

n2_04 = [r_04(1:91,2)'; r_04(92:182,2)';
r_04(183:273,2)'; r_04(274:364,2)'; r_04(365:455,2)';
r_04(456:546,2)';
r_04(547:637,2)';r_04(638:728,2)'; r_04(729:819,2)';
r_04(820:910,2)'; r_04(911:1001,2)'];

```

```

n3_04 = [r_04(1:91,3)'; r_04(92:182,3)';
r_04(183:273,3)'; r_04(274:364,3)'; r_04(365:455,3)';
r_04(456:546,3)';
r_04(547:637,3)';r_04(638:728,3)'; r_04(729:819,3)';
r_04(820:910,3)'; r_04(911:1001,3)'];

n4_04 = [r_04(1:91,4)'; r_04(92:182,4)';
r_04(183:273,4)'; r_04(274:364,4)'; r_04(365:455,4)';
r_04(456:546,4)';
r_04(547:637,4)';r_04(638:728,4)'; r_04(729:819,4)';
r_04(820:910,4)'; r_04(911:1001,4)'];

n5_04 = [r_04(1:91,5)'; r_04(92:182,5)';
r_04(183:273,5)'; r_04(274:364,5)'; r_04(365:455,5)';
r_04(456:546,5)';
r_04(547:637,5)';r_04(638:728,5)'; r_04(729:819,5)';
r_04(820:910,5)'; r_04(911:1001,5)'];

n7_04 = [r_04(1:91,7)'; r_04(92:182,7)';
r_04(183:273,7)'; r_04(274:364,7)'; r_04(365:455,7)';
r_04(456:546,7)';
r_04(547:637,7)';r_04(638:728,7)'; r_04(729:819,7)';
r_04(820:910,7)'; r_04(911:1001,7)'];

n5_04_poly = zeros(91,8); % 91 rows, 7 columns
n7_04_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_04_poly(c,:) = [ polyfit(t,n5_04(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_04_poly(c,:) = [ polyfit(t,n7_04(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 1.05
r_05 = m{5};
n1_05 = [r_05(1:91,1)'; r_05(92:182,1)';
r_05(183:273,1)'; r_05(274:364,1)'; r_05(365:455,1)';
r_05(456:546,1)'];

```

```

    r_05(547:637,1)';r_05(638:728,1)'; r_05(729:819,1)';
r_05(820:910,1)'; r_05(911:1001,1)'];

n2_05 = [r_05(1:91,2)'; r_05(92:182,2)';
r_05(183:273,2)'; r_05(274:364,2)'; r_05(365:455,2)';
r_05(456:546,2)';
    r_05(547:637,2)';r_05(638:728,2)'; r_05(729:819,2)';
r_05(820:910,2)'; r_05(911:1001,2)'];

n3_05 = [r_05(1:91,3)'; r_05(92:182,3)';
r_05(183:273,3)'; r_05(274:364,3)'; r_05(365:455,3)';
r_05(456:546,3)';
    r_05(547:637,3)';r_05(638:728,3)'; r_05(729:819,3)';
r_05(820:910,3)'; r_05(911:1001,3)'];

n4_05 = [r_05(1:91,4)'; r_05(92:182,4)';
r_05(183:273,4)'; r_05(274:364,4)'; r_05(365:455,4)';
r_05(456:546,4)';
    r_05(547:637,4)';r_05(638:728,4)'; r_05(729:819,4)';
r_05(820:910,4)'; r_05(911:1001,4)'];

n5_05 = [r_05(1:91,5)'; r_05(92:182,5)';
r_05(183:273,5)'; r_05(274:364,5)'; r_05(365:455,5)';
r_05(456:546,5)';
    r_05(547:637,5)';r_05(638:728,5)'; r_05(729:819,5)';
r_05(820:910,5)'; r_05(911:1001,5)'];

n7_05 = [r_05(1:91,7)'; r_05(92:182,7)';
r_05(183:273,7)'; r_05(274:364,7)'; r_05(365:455,7)';
r_05(456:546,7)';
    r_05(547:637,7)';r_05(638:728,7)'; r_05(729:819,7)';
r_05(820:910,7)'; r_05(911:1001,7)'];

n5_05_poly = zeros(91,8); % 91 rows, 7 columns
n7_05_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_05_poly(c,:) = [ polyfit(t,n5_05(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_05_poly(c,:) = [ polyfit(t,n7_05(:,c),7) ];

```

```

        c = c+1;
end

%%%%%%%%%% FER 1.06
r_06 = m{6};

n1_06 = [r_06(1:91,1)'; r_06(92:182,1)';
r_06(183:273,1)'; r_06(274:364,1)'; r_06(365:455,1)';
r_06(456:546,1)';
        r_06(547:637,1)';r_06(638:728,1)'; r_06(729:819,1)';
r_06(820:910,1)'; r_06(911:1001,1)'];

n2_06 = [r_06(1:91,2)'; r_06(92:182,2)';
r_06(183:273,2)'; r_06(274:364,2)'; r_06(365:455,2)';
r_06(456:546,2)';
        r_06(547:637,2)';r_06(638:728,2)'; r_06(729:819,2)';
r_06(820:910,2)'; r_06(911:1001,2)'];

n3_06 = [r_06(1:91,3)'; r_06(92:182,3)';
r_06(183:273,3)'; r_06(274:364,3)'; r_06(365:455,3)';
r_06(456:546,3)';
        r_06(547:637,3)';r_06(638:728,3)'; r_06(729:819,3)';
r_06(820:910,3)'; r_06(911:1001,3)'];

n4_06 = [r_06(1:91,4)'; r_06(92:182,4)';
r_06(183:273,4)'; r_06(274:364,4)'; r_06(365:455,4)';
r_06(456:546,4)';
        r_06(547:637,4)';r_06(638:728,4)'; r_06(729:819,4)';
r_06(820:910,4)'; r_06(911:1001,4)'];

n5_06 = [r_06(1:91,5)'; r_06(92:182,5)';
r_06(183:273,5)'; r_06(274:364,5)'; r_06(365:455,5)';
r_06(456:546,5)';
        r_06(547:637,5)';r_06(638:728,5)'; r_06(729:819,5)';
r_06(820:910,5)'; r_06(911:1001,5)'];

n7_06 = [r_06(1:91,7)'; r_06(92:182,7)';
r_06(183:273,7)'; r_06(274:364,7)'; r_06(365:455,7)';
r_06(456:546,7)';
        r_06(547:637,7)';r_06(638:728,7)'; r_06(729:819,7)';
r_06(820:910,7)'; r_06(911:1001,7)'];

n5_06_poly = zeros(91,8); % 91 rows, 7 columns
n7_06_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

```

```

n5_06_poly(c,:) = [ polyfit(t,n5_06(:,c),7) ];

c = c+1;
end

c = 1;
while c < 92 %%%% Making n7 polynomials

    n7_06_poly(c,:) = [ polyfit(t,n7_06(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%% FER 1.07
r_07 = m{7};
n1_07 = [r_07(1:91,1)'; r_07(92:182,1)';
r_07(183:273,1)'; r_07(274:364,1)'; r_07(365:455,1)';
r_07(456:546,1)';
r_07(547:637,1)';r_07(638:728,1)'; r_07(729:819,1)';
r_07(820:910,1)'; r_07(911:1001,1)'];

n2_07 = [r_07(1:91,2)'; r_07(92:182,2)';
r_07(183:273,2)'; r_07(274:364,2)'; r_07(365:455,2)';
r_07(456:546,2)';
r_07(547:637,2)';r_07(638:728,2)'; r_07(729:819,2)';
r_07(820:910,2)'; r_07(911:1001,2)'];

n3_07 = [r_07(1:91,3)'; r_07(92:182,3)';
r_07(183:273,3)'; r_07(274:364,3)'; r_07(365:455,3)';
r_07(456:546,3)';
r_07(547:637,3)';r_07(638:728,3)'; r_07(729:819,3)';
r_07(820:910,3)'; r_07(911:1001,3)'];

n4_07 = [r_07(1:91,4)'; r_07(92:182,4)';
r_07(183:273,4)'; r_07(274:364,4)'; r_07(365:455,4)';
r_07(456:546,4)';
r_07(547:637,4)';r_07(638:728,4)'; r_07(729:819,4)';
r_07(820:910,4)'; r_07(911:1001,4)'];

n5_07 = [r_07(1:91,5)'; r_07(92:182,5)';
r_07(183:273,5)'; r_07(274:364,5)'; r_07(365:455,5)';
r_07(456:546,5)';
r_07(547:637,5)';r_07(638:728,5)'; r_07(729:819,5)';
r_07(820:910,5)'; r_07(911:1001,5)'];

```

```

n7_07 = [r_07(1:91,7)'; r_07(92:182,7)';
r_07(183:273,7)'; r_07(274:364,7)'; r_07(365:455,7)';
r_07(456:546,7)';
r_07(547:637,7)';r_07(638:728,7)'; r_07(729:819,7)';
r_07(820:910,7)'; r_07(911:1001,7)'];

n5_07_poly = zeros(91,8); % 91 rows, 7 columns
n7_07_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_07_poly(c,:) = [ polyfit(t,n5_07(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_07_poly(c,:) = [ polyfit(t,n7_07(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 1.08

r_08 = m{8};
n1_08 = [r_08(1:91,1)'; r_08(92:182,1)';
r_08(183:273,1)'; r_08(274:364,1)'; r_08(365:455,1)';
r_08(456:546,1)';
r_08(547:637,1)';r_08(638:728,1)'; r_08(729:819,1)';
r_08(820:910,1)'; r_08(911:1001,1)'];

n2_08 = [r_08(1:91,2)'; r_08(92:182,2)';
r_08(183:273,2)'; r_08(274:364,2)'; r_08(365:455,2)';
r_08(456:546,2)';
r_08(547:637,2)';r_08(638:728,2)'; r_08(729:819,2)';
r_08(820:910,2)'; r_08(911:1001,2)'];

n3_08 = [r_08(1:91,3)'; r_08(92:182,3)';
r_08(183:273,3)'; r_08(274:364,3)'; r_08(365:455,3)';
r_08(456:546,3)';
r_08(547:637,3)';r_08(638:728,3)'; r_08(729:819,3)';
r_08(820:910,3)'; r_08(911:1001,3)'];

```

```

n4_08 = [r_08(1:91,4)'; r_08(92:182,4)';
r_08(183:273,4)'; r_08(274:364,4)'; r_08(365:455,4)';
r_08(456:546,4)';
r_08(547:637,4)';r_08(638:728,4)'; r_08(729:819,4)';
r_08(820:910,4)'; r_08(911:1001,4)'];

n5_08 = [r_08(1:91,5)'; r_08(92:182,5)';
r_08(183:273,5)'; r_08(274:364,5)'; r_08(365:455,5)';
r_08(456:546,5)';
r_08(547:637,5)';r_08(638:728,5)'; r_08(729:819,5)';
r_08(820:910,5)'; r_08(911:1001,5)'];

n7_08 = [r_08(1:91,7)'; r_08(92:182,7)';
r_08(183:273,7)'; r_08(274:364,7)'; r_08(365:455,7)';
r_08(456:546,7)';
r_08(547:637,7)';r_08(638:728,7)'; r_08(729:819,7)';
r_08(820:910,7)'; r_08(911:1001,7)'];

n5_08_poly = zeros(91,8); % 91 rows, 7 columns
n7_08_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_08_poly(c,:) = [ polyfit(t,n5_08(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_08_poly(c,:) = [ polyfit(t,n7_08(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 1.09
r_109 = m{9};
n1_109 = [r_109(1:91,1)'; r_109(92:182,1)';
r_109(183:273,1)'; r_109(274:364,1)'; r_109(365:455,1)';
r_109(456:546,1)';
r_109(547:637,1)';r_109(638:728,1)';
r_109(729:819,1)'; r_109(820:910,1)';
r_109(911:1001,1)'];

```

```

n2_109 = [r_109(1:91,2)'; r_109(92:182,2)';
r_109(183:273,2)'; r_109(274:364,2)'; r_109(365:455,2)';
r_109(456:546,2)';
        r_109(547:637,2)';r_109(638:728,2)';
r_109(729:819,2)'; r_109(820:910,2)';
r_109(911:1001,2)'];

```

```

n3_109 = [r_109(1:91,3)'; r_109(92:182,3)';
r_109(183:273,3)'; r_109(274:364,3)'; r_109(365:455,3)';
r_109(456:546,3)';
        r_109(547:637,3)';r_109(638:728,3)';
r_109(729:819,3)'; r_109(820:910,3)';
r_109(911:1001,3)'];

```

```

n4_109 = [r_109(1:91,4)'; r_109(92:182,4)';
r_109(183:273,4)'; r_109(274:364,4)'; r_109(365:455,4)';
r_109(456:546,4)';
        r_109(547:637,4)';r_109(638:728,4)';
r_109(729:819,4)'; r_109(820:910,4)';
r_109(911:1001,4)'];

```

```

n5_109 = [r_109(1:91,5)'; r_109(92:182,5)';
r_109(183:273,5)'; r_109(274:364,5)'; r_109(365:455,5)';
r_109(456:546,5)';
        r_109(547:637,5)';r_109(638:728,5)';
r_109(729:819,5)'; r_109(820:910,5)';
r_109(911:1001,5)'];

```

```

n7_109 = [r_109(1:91,7)'; r_109(92:182,7)';
r_109(183:273,7)'; r_109(274:364,7)'; r_109(365:455,7)';
r_109(456:546,7)';
        r_109(547:637,7)';r_109(638:728,7)';
r_109(729:819,7)'; r_109(820:910,7)';
r_109(911:1001,7)'];

```

```

n5_109_poly = zeros(91,8); % 91 rows, 7 columns
n7_109_poly = zeros(91,8);

```

```

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_109_poly(c,:) = [ polyfit(t,n5_109(:,c),7) ];

    c = c+1;
end

```

```

c = 1;
while c < 92 %%% Making n7 polynomials

```

```

n7_109_poly(c,:) = [ polyfit(t,n7_109(:,c),7) ];

c = c+1;
end

%%%%%%%%%% FER 1.10
r_110 = m{10};
n1_110 = [r_110(1:91,1)'; r_110(92:182,1)';
r_110(183:273,1)'; r_110(274:364,1)'; r_110(365:455,1)';
r_110(456:546,1)';
r_110(547:637,1)';r_110(638:728,1)';
r_110(729:819,1)'; r_110(820:910,1)';
r_110(911:1001,1)'];

n2_110 = [r_110(1:91,2)'; r_110(92:182,2)';
r_110(183:273,2)'; r_110(274:364,2)'; r_110(365:455,2)';
r_110(456:546,2)';
r_110(547:637,2)';r_110(638:728,2)';
r_110(729:819,2)'; r_110(820:910,2)';
r_110(911:1001,2)'];

n3_110 = [r_110(1:91,3)'; r_110(92:182,3)';
r_110(183:273,3)'; r_110(274:364,3)'; r_110(365:455,3)';
r_110(456:546,3)';
r_110(547:637,3)';r_110(638:728,3)';
r_110(729:819,3)'; r_110(820:910,3)';
r_110(911:1001,3)'];

n4_110 = [r_110(1:91,4)'; r_110(92:182,4)';
r_110(183:273,4)'; r_110(274:364,4)'; r_110(365:455,4)';
r_110(456:546,4)';
r_110(547:637,4)';r_110(638:728,4)';
r_110(729:819,4)'; r_110(820:910,4)';
r_110(911:1001,4)'];

n5_110 = [r_110(1:91,5)'; r_110(92:182,5)';
r_110(183:273,5)'; r_110(274:364,5)'; r_110(365:455,5)';
r_110(456:546,5)';
r_110(547:637,5)';r_110(638:728,5)';
r_110(729:819,5)'; r_110(820:910,5)';
r_110(911:1001,5)'];

n7_110 = [r_110(1:91,7)'; r_110(92:182,7)';
r_110(183:273,7)'; r_110(274:364,7)'; r_110(365:455,7)';
r_110(456:546,7)'];

```

```

        r_110(547:637,7)';r_110(638:728,7)';
r_110(729:819,7)'; r_110(820:910,7)';
r_110(911:1001,7)']];

n5_110_poly = zeros(91,8); % 91 rows, 7 columns
n7_110_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_110_poly(c,:) = [ polyfit(t,n5_110(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_110_poly(c,:) = [ polyfit(t,n7_110(:,c),7) ];

    c = c+1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FER 1
r_111 = m{11};
n1_111 = [r_111(1:91,1)'; r_111(92:182,1)';
r_111(183:273,1)'; r_111(274:364,1)'; r_111(365:455,1)';
r_111(456:546,1)';
        r_111(547:637,1)';r_111(638:728,1)';
r_111(729:819,1)'; r_111(820:910,1)';
r_111(911:1001,1)']];

n2_111 = [r_111(1:91,2)'; r_111(92:182,2)';
r_111(183:273,2)'; r_111(274:364,2)'; r_111(365:455,2)';
r_111(456:546,2)';
        r_111(547:637,2)';r_111(638:728,2)';
r_111(729:819,2)'; r_111(820:910,2)';
r_111(911:1001,2)']];

n3_111 = [r_111(1:91,3)'; r_111(92:182,3)';
r_111(183:273,3)'; r_111(274:364,3)'; r_111(365:455,3)';
r_111(456:546,3)';
        r_111(547:637,3)';r_111(638:728,3)';
r_111(729:819,3)'; r_111(820:910,3)';
r_111(911:1001,3)']];

```

```

n4_111 = [r_111(1:91,4)'; r_111(92:182,4)';
r_111(183:273,4)'; r_111(274:364,4)'; r_111(365:455,4)';
r_111(456:546,4)';
        r_111(547:637,4)';r_111(638:728,4)';
r_111(729:819,4)'; r_111(820:910,4)';
r_111(911:1001,4)'];

n5_111 = [r_111(1:91,5)'; r_111(92:182,5)';
r_111(183:273,5)'; r_111(274:364,5)'; r_111(365:455,5)';
r_111(456:546,5)';
        r_111(547:637,5)';r_111(638:728,5)';
r_111(729:819,5)'; r_111(820:910,5)';
r_111(911:1001,5)'];

n7_111 = [r_111(1:91,7)'; r_111(92:182,7)';
r_111(183:273,7)'; r_111(274:364,7)'; r_111(365:455,7)';
r_111(456:546,7)';
        r_111(547:637,7)';r_111(638:728,7)';
r_111(729:819,7)'; r_111(820:910,7)';
r_111(911:1001,7)'];

n5_111_poly = zeros(91,8); % 91 rows, 7 columns
n7_111_poly = zeros(91,8);

c = 1;
while c < 92 %%% Making n5 polynomials

    n5_111_poly(c,:) = [ polyfit(t,n5_111(:,c),7) ];

    c = c+1;
end

c = 1;
while c < 92 %%% Making n7 polynomials

    n7_111_poly(c,:) = [ polyfit(t,n7_111(:,c),7) ];

    c = c+1;
end

```

11 Appendix F - Dissociation Model: Tabulated Results

Select oxygen results from the Excel version (pressure increments of 5 bar, K_p supplied from Goodger (2001)) of the dissociation model are collected here. A far more expensive tabulation of results from the MATLAB model (1 bar increments, calculated K_p values and includes n_7) is offered in Supplement 1: Atmospheric Oxygen (O_2) Tables and Supplement 2: Monatomic Oxygen (O) Tables. Note that any negative numbers illustrate an error in initial guess estimation and requires one of smaller magnitude.

$\phi = 0.99$																			
n1																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920	7.920
2000.000	7.856	7.867	7.873	7.878	7.881	7.884	7.886	7.888	7.889	7.891	7.892	7.893	7.894	7.895	7.895	7.896	7.897	7.897	7.898
2100.000	7.793	7.813	7.825	7.834	7.841	7.846	7.850	7.854	7.857	7.859	7.861	7.863	7.865	7.867	7.869	7.870	7.871	7.872	7.874
2200.000	7.694	7.727	7.748	7.763	7.774	7.783	7.790	7.796	7.802	7.806	7.810	7.814	7.817	7.820	7.823	7.826	7.828	7.830	7.832
2300.000	7.549	7.600	7.633	7.656	7.674	7.688	7.700	7.710	7.718	7.725	7.732	7.738	7.743	7.748	7.752	7.756	7.760	7.764	7.767
2400.000	7.350	7.425	7.473	7.507	7.534	7.555	7.572	7.587	7.600	7.611	7.621	7.630	7.638	7.645	7.651	7.658	7.663	7.668	7.674
2500.000	7.092	7.196	7.263	7.311	7.348	7.378	7.403	7.424	7.442	7.458	7.472	7.485	7.496	7.506	7.516	7.525	7.533	7.540	7.547
2700.000	6.386	6.561	6.676	6.759	6.824	6.876	6.919	6.957	6.989	7.018	7.043	7.065	7.086	7.105	7.122	7.138	7.152	7.166	7.179
3000.000	4.948	5.228	5.418	5.559	5.670	5.762	5.839	5.905	5.963	6.015	6.061	6.103	6.141	6.176	6.208	6.238	6.266	6.292	6.316
n2																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910	8.910
2000.000	8.894	8.897	8.898	8.899	8.900	8.901	8.901	8.902	8.902	8.903	8.903	8.903	8.903	8.904	8.904	8.904	8.904	8.904	8.904
2100.000	8.880	8.885	8.888	8.890	8.892	8.893	8.894	8.895	8.895	8.896	8.896	8.897	8.897	8.898	8.898	8.898	8.899	8.899	8.899
2200.000	8.860	8.867	8.872	8.875	8.878	8.880	8.882	8.883	8.884	8.885	8.886	8.887	8.888	8.888	8.889	8.889	8.890	8.890	8.891
2300.000	8.831	8.843	8.850	8.855	8.859	8.862	8.864	8.866	8.868	8.870	8.871	8.872	8.873	8.874	8.875	8.876	8.877	8.878	8.878
2400.000	8.793	8.809	8.819	8.826	8.832	8.836	8.840	8.843	8.846	8.848	8.850	8.852	8.854	8.855	8.856	8.858	8.859	8.860	8.861
2500.000	8.742	8.765	8.779	8.790	8.797	8.804	8.809	8.813	8.817	8.820	8.823	8.826	8.828	8.830	8.832	8.834	8.835	8.837	8.838
2700.000	8.596	8.638	8.665	8.683	8.698	8.709	8.718	8.726	8.733	8.739	8.744	8.749	8.753	8.757	8.761	8.764	8.767	8.770	8.772
3000.000	8.225	8.316	8.374	8.414	8.445	8.470	8.490	8.508	8.522	8.535	8.547	8.557	8.566	8.574	8.582	8.589	8.595	8.601	8.607
n3																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2000.000	0.064	0.053	0.047	0.042	0.039	0.036	0.034	0.032	0.031	0.029	0.028	0.027	0.026	0.025	0.025	0.024	0.023	0.023	0.022
2100.000	0.127	0.107	0.095	0.086	0.079	0.074	0.070	0.066	0.063	0.061	0.059	0.057	0.055	0.053	0.051	0.050	0.049	0.048	0.046
2200.000	0.226	0.193	0.172	0.157	0.146	0.137	0.130	0.124	0.118	0.114	0.110	0.106	0.103	0.100	0.097	0.094	0.092	0.090	0.088
2300.000	0.371	0.320	0.287	0.264	0.246	0.232	0.220	0.210	0.202	0.195	0.188	0.182	0.177	0.172	0.168	0.164	0.160	0.156	0.153
2400.000	0.570	0.495	0.447	0.413	0.386	0.365	0.348	0.333	0.320	0.309	0.299	0.290	0.282	0.275	0.269	0.262	0.257	0.252	0.246
2500.000	0.828	0.724	0.657	0.609	0.572	0.542	0.517	0.496	0.478	0.462	0.448	0.435	0.424	0.414	0.404	0.395	0.387	0.380	0.373
2700.000	1.534	1.359	1.244	1.161	1.096	1.044	1.001	0.963	0.931	0.902	0.877	0.855	0.834	0.815	0.798	0.782	0.768	0.754	0.741
3000.000	2.972	2.692	2.502	2.361	2.250	2.158	2.081	2.015	1.957	1.905	1.859	1.817	1.779	1.744	1.712	1.682	1.654	1.628	1.604

n4																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2000.000	0.016	0.013	0.012	0.011	0.010	0.009	0.009	0.008	0.008	0.007	0.007	0.007	0.007	0.006	0.006	0.006	0.006	0.006	0.006
2100.000	0.030	0.025	0.022	0.020	0.018	0.017	0.016	0.015	0.015	0.014	0.014	0.013	0.013	0.012	0.012	0.012	0.011	0.011	0.011
2200.000	0.050	0.043	0.038	0.035	0.032	0.030	0.028	0.027	0.026	0.025	0.024	0.023	0.022	0.022	0.021	0.021	0.020	0.020	0.019
2300.000	0.079	0.067	0.060	0.055	0.051	0.048	0.046	0.044	0.042	0.040	0.039	0.038	0.037	0.036	0.035	0.034	0.033	0.032	0.032
2400.000	0.117	0.101	0.091	0.084	0.078	0.074	0.070	0.067	0.064	0.062	0.060	0.058	0.056	0.055	0.054	0.052	0.051	0.050	0.049
2500.000	0.168	0.145	0.131	0.120	0.113	0.106	0.101	0.097	0.093	0.090	0.087	0.084	0.082	0.080	0.078	0.076	0.075	0.073	0.072
2700.000	0.314	0.272	0.245	0.227	0.212	0.201	0.192	0.184	0.177	0.171	0.166	0.161	0.157	0.153	0.149	0.146	0.143	0.140	0.138
3000.000	0.685	0.594	0.536	0.496	0.465	0.440	0.420	0.402	0.388	0.375	0.363	0.353	0.344	0.336	0.328	0.321	0.315	0.309	0.303
n5																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125
2000.000	0.165	0.158	0.154	0.151	0.149	0.148	0.146	0.145	0.144	0.143	0.143	0.142	0.141	0.141	0.140	0.140	0.140	0.139	0.139
2100.000	0.203	0.191	0.183	0.178	0.174	0.171	0.168	0.166	0.164	0.163	0.161	0.160	0.159	0.158	0.157	0.156	0.155	0.154	0.154
2200.000	0.263	0.243	0.230	0.221	0.214	0.209	0.204	0.200	0.197	0.194	0.192	0.190	0.188	0.186	0.184	0.183	0.181	0.180	0.179
2300.000	0.350	0.319	0.299	0.285	0.274	0.265	0.258	0.252	0.247	0.242	0.238	0.235	0.232	0.229	0.226	0.224	0.222	0.219	0.217
2400.000	0.469	0.423	0.394	0.373	0.357	0.344	0.334	0.325	0.317	0.311	0.305	0.299	0.294	0.290	0.286	0.282	0.279	0.276	0.273
2500.000	0.623	0.560	0.519	0.490	0.467	0.449	0.434	0.422	0.411	0.401	0.393	0.385	0.378	0.372	0.366	0.361	0.356	0.351	0.347
2700.000	1.049	0.940	0.870	0.819	0.779	0.747	0.721	0.698	0.679	0.662	0.647	0.633	0.620	0.609	0.599	0.589	0.580	0.572	0.564
3000.000	1.954	1.768	1.644	1.554	1.482	1.424	1.375	1.334	1.297	1.265	1.236	1.210	1.187	1.165	1.145	1.127	1.109	1.094	1.079

STOICHIOMETRIC																			
n1																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	7.997	7.998	7.998	7.998	7.998	7.998	7.998	7.998	7.998	7.998	7.998	7.998	7.999	7.999	7.999	7.999	7.999	7.999	7.999
2000.000	7.897	7.910	7.918	7.924	7.928	7.932	7.935	7.937	7.939	7.941	7.943	7.945	7.946	7.947	7.948	7.949	7.950	7.951	7.952
2100.000	7.825	7.847	7.861	7.871	7.878	7.884	7.889	7.893	7.897	7.900	7.903	7.906	7.908	7.910	7.912	7.913	7.915	7.916	7.918
2200.000	7.719	7.754	7.776	7.792	7.804	7.813	7.821	7.828	7.834	7.839	7.844	7.848	7.851	7.855	7.858	7.861	7.863	7.865	7.868
2300.000	7.569	7.621	7.655	7.679	7.697	7.712	7.724	7.735	7.744	7.751	7.759	7.765	7.771	7.776	7.780	7.784	7.789	7.792	7.796
2400.000	7.367	7.442	7.492	7.526	7.553	7.574	7.593	7.607	7.621	7.632	7.642	7.651	7.659	7.667	7.674	7.680	7.687	7.692	7.697
2500.000	7.106	7.211	7.279	7.327	7.364	7.395	7.420	7.441	7.459	7.476	7.490	7.502	7.514	7.525	7.535	7.543	7.552	7.560	7.567
2700.000	6.397	6.573	6.688	6.772	6.836	6.889	6.932	6.970	7.002	7.031	7.056	7.079	7.100	7.118	7.136	7.151	7.167	7.180	7.193
3000.000	4.958	5.238	5.428	5.569	5.680	5.772	5.849	5.916	5.974	6.025	6.072	6.113	6.151	6.187	6.219	6.249	6.277	6.302	6.327
n2																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999	8.999
2000.000	8.974	8.977	8.979	8.981	8.982	8.983	8.984	8.984	8.985	8.985	8.986	8.986	8.986	8.987	8.987	8.987	8.988	8.988	8.988
2100.000	8.959	8.964	8.967	8.970	8.972	8.973	8.974	8.975	8.976	8.977	8.977	8.978	8.979	8.979	8.979	8.980	8.980	8.981	8.981
2200.000	8.937	8.945	8.950	8.954	8.957	8.959	8.961	8.962	8.963	8.965	8.966	8.967	8.967	8.968	8.969	8.969	8.970	8.970	8.971
2300.000	8.908	8.920	8.927	8.932	8.936	8.940	8.942	8.944	8.946	8.948	8.950	8.951	8.952	8.953	8.954	8.955	8.956	8.957	8.957
2400.000	8.869	8.885	8.896	8.904	8.909	8.914	8.918	8.921	8.924	8.926	8.928	8.930	8.932	8.933	8.935	8.936	8.937	8.938	8.939
2500.000	8.818	8.841	8.856	8.866	8.874	8.880	8.886	8.890	8.894	8.897	8.900	8.903	8.905	8.908	8.910	8.911	8.913	8.915	8.916
2700.000	8.670	8.712	8.739	8.758	8.773	8.785	8.794	8.802	8.809	8.815	8.821	8.825	8.830	8.834	8.837	8.841	8.844	8.846	8.849
3000.000	8.294	8.387	8.445	8.486	8.518	8.543	8.563	8.581	8.596	8.609	8.620	8.631	8.640	8.648	8.656	8.663	8.670	8.676	8.682
n3																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.003	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.001	0.001	0.001	0.001	0.001	0.001	0.001
2000.000	0.103	0.090	0.082	0.076	0.072	0.068	0.065	0.063	0.061	0.059	0.057	0.055	0.054	0.053	0.052	0.051	0.050	0.049	0.048
2100.000	0.175	0.153	0.139	0.129	0.122	0.116	0.111	0.107	0.103	0.100	0.097	0.094	0.092	0.090	0.088	0.087	0.085	0.084	0.082
2200.000	0.281	0.246	0.224	0.208	0.196	0.187	0.179	0.172	0.166	0.161	0.156	0.152	0.149	0.145	0.142	0.139	0.137	0.135	0.132
2300.000	0.431	0.379	0.345	0.321	0.303	0.288	0.276	0.265	0.256	0.249	0.241	0.235	0.229	0.224	0.220	0.216	0.211	0.208	0.204
2400.000	0.633	0.558	0.508	0.474	0.447	0.426	0.407	0.393	0.379	0.368	0.358	0.349	0.341	0.333	0.326	0.320	0.313	0.308	0.303
2500.000	0.894	0.789	0.721	0.673	0.636	0.605	0.580	0.559	0.541	0.524	0.510	0.498	0.486	0.475	0.465	0.457	0.448	0.440	0.433
2700.000	1.603	1.427	1.312	1.228	1.164	1.111	1.068	1.030	0.998	0.969	0.944	0.921	0.900	0.882	0.864	0.849	0.833	0.820	0.807
3000.000	3.042	2.762	2.572	2.431	2.320	2.228	2.151	2.084	2.026	1.975	1.928	1.887	1.849	1.813	1.781	1.751	1.723	1.698	1.673
n4																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
2000.000	0.026	0.023	0.021	0.019	0.018	0.017	0.016	0.016	0.015	0.015	0.014	0.014	0.014	0.013	0.013	0.013	0.012	0.012	0.012
2100.000	0.041	0.036	0.033	0.030	0.028	0.027	0.026	0.025	0.024	0.023	0.023	0.022	0.021	0.021	0.021	0.020	0.020	0.019	0.019
2200.000	0.063	0.055	0.050	0.046	0.043	0.041	0.039	0.038	0.037	0.035	0.034	0.033	0.033	0.032	0.031	0.031	0.030	0.030	0.029
2300.000	0.092	0.080	0.073	0.068	0.064	0.060	0.058	0.056	0.054	0.052	0.050	0.049	0.048	0.047	0.046	0.045	0.044	0.043	0.043
2400.000	0.131	0.115	0.104	0.096	0.091	0.086	0.082	0.079	0.076	0.074	0.072	0.070	0.068	0.067	0.065	0.064	0.063	0.062	0.061
2500.000	0.182	0.159	0.144	0.134	0.126	0.120	0.114	0.110	0.106	0.103	0.100	0.097	0.095	0.092	0.090	0.089	0.087	0.085	0.084
2700.000	0.330	0.288	0.261	0.242	0.227	0.215	0.206	0.198	0.191	0.185	0.179	0.175	0.170	0.166	0.163	0.159	0.156	0.154	0.151
3000.000	0.706	0.613	0.555	0.514	0.482	0.457	0.437	0.419	0.404	0.391	0.380	0.369	0.360	0.352	0.344	0.337	0.330	0.324	0.318

n5																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.002	0.002	0.001	0.002	0.002	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
2000.000	0.065	0.057	0.051	0.048	0.045	0.043	0.041	0.039	0.038	0.037	0.036	0.035	0.034	0.033	0.032	0.032	0.031	0.031	0.030
2100.000	0.108	0.094	0.086	0.080	0.075	0.071	0.068	0.066	0.064	0.062	0.060	0.058	0.057	0.055	0.054	0.053	0.052	0.051	0.051
2200.000	0.172	0.150	0.137	0.127	0.120	0.114	0.109	0.105	0.101	0.098	0.095	0.093	0.091	0.088	0.087	0.085	0.084	0.082	0.081
2300.000	0.262	0.229	0.209	0.194	0.183	0.174	0.167	0.160	0.155	0.150	0.146	0.142	0.139	0.136	0.133	0.130	0.128	0.126	0.123
2400.000	0.382	0.336	0.306	0.285	0.269	0.256	0.245	0.236	0.228	0.221	0.215	0.209	0.204	0.200	0.196	0.192	0.188	0.185	0.182
2500.000	0.538	0.474	0.433	0.403	0.381	0.362	0.347	0.334	0.323	0.314	0.305	0.297	0.290	0.284	0.278	0.273	0.268	0.263	0.259
2700.000	0.966	0.857	0.786	0.735	0.695	0.663	0.637	0.614	0.594	0.577	0.562	0.548	0.535	0.524	0.514	0.504	0.495	0.487	0.479
3000.000	1.874	1.688	1.564	1.473	1.401	1.343	1.294	1.252	1.215	1.183	1.154	1.128	1.104	1.082	1.063	1.044	1.027	1.011	0.996

$\phi = 1.01$																			
n1																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	8.076	8.077	8.077	8.078	8.078	8.078	8.078	8.078	8.078	8.078	8.078	8.078	8.079	8.079	8.079	8.079	8.079	8.079	8.079
2000.000	7.861	7.870	7.870	7.874	7.882	7.885	7.885	7.882	7.892	7.884	7.892	7.878	7.885	7.892	7.898	7.903	7.882	7.887	7.892
2100.000	7.808	7.824	7.836	7.841	7.848	7.851	7.853	7.857	7.858	7.863	7.861	7.864	7.865	7.866	7.865	7.871	7.870	7.867	7.872
2200.000	7.713	7.745	7.765	7.778	7.788	7.796	7.803	7.809	7.814	7.818	7.822	7.825	7.826	7.829	7.832	7.834	7.835	7.838	7.838
2300.000	7.569	7.620	7.651	7.675	7.692	7.706	7.717	7.727	7.734	7.741	7.748	7.753	7.759	7.762	7.766	7.771	7.775	7.778	7.781
2400.000	7.370	7.445	7.493	7.527	7.553	7.573	7.591	7.605	7.618	7.629	7.639	7.648	7.656	7.663	7.669	7.675	7.680	7.685	7.691
2500.000	7.112	7.216	7.283	7.331	7.368	7.397	7.422	7.442	7.461	7.477	7.491	7.503	7.514	7.525	7.534	7.543	7.551	7.558	7.565
2700.000	6.405	6.580	6.695	6.778	6.842	6.895	6.938	6.975	7.008	7.036	7.061	7.084	7.104	7.123	7.140	7.156	7.171	7.184	7.197
3000.000	4.965	5.246	5.435	5.576	5.688	5.779	5.856	5.923	5.981	6.032	6.078	6.120	6.158	6.193	6.225	6.255	6.283	6.309	6.333
n2																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	9.088	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089	9.089
2000.000	9.034	9.037	9.037	9.038	9.040	9.041	9.041	9.040	9.042	9.040	9.042	9.039	9.041	9.042	9.044	9.045	9.040	9.041	9.042
2100.000	9.026	9.029	9.032	9.034	9.035	9.036	9.036	9.037	9.038	9.039	9.038	9.039	9.039	9.039	9.039	9.041	9.040	9.040	9.041
2200.000	9.008	9.015	9.020	9.023	9.025	9.027	9.028	9.030	9.031	9.032	9.033	9.033	9.034	9.034	9.035	9.035	9.036	9.036	9.036
2300.000	8.980	8.992	8.999	9.004	9.008	9.011	9.013	9.015	9.017	9.018	9.020	9.021	9.022	9.023	9.024	9.025	9.026	9.027	9.027
2400.000	8.942	8.959	8.969	8.976	8.982	8.987	8.990	8.993	8.996	8.998	9.001	9.002	9.004	9.006	9.007	9.008	9.009	9.010	9.011
2500.000	8.891	8.914	8.929	8.940	8.948	8.954	8.959	8.964	8.968	8.971	8.974	8.976	8.979	8.981	8.983	8.985	8.986	8.988	8.989
2700.000	8.742	8.785	8.813	8.832	8.847	8.858	8.868	8.876	8.883	8.889	8.895	8.900	8.904	8.908	8.912	8.915	8.918	8.921	8.924
3000.000	8.363	8.456	8.515	8.557	8.589	8.614	8.635	8.653	8.668	8.681	8.693	8.704	8.713	8.721	8.729	8.737	8.743	8.749	8.755
n3																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.004	0.003	0.003	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.001	0.001	0.001	0.001	0.001	0.001	0.001
2000.000	0.219	0.210	0.210	0.206	0.198	0.195	0.195	0.198	0.188	0.196	0.188	0.202	0.195	0.188	0.182	0.177	0.198	0.193	0.188
2100.000	0.272	0.256	0.244	0.239	0.232	0.229	0.227	0.223	0.222	0.217	0.219	0.216	0.215	0.214	0.215	0.209	0.210	0.213	0.208
2200.000	0.367	0.335	0.315	0.302	0.292	0.284	0.277	0.271	0.266	0.262	0.258	0.255	0.254	0.251	0.248	0.246	0.245	0.242	0.242
2300.000	0.511	0.460	0.429	0.405	0.388	0.374	0.363	0.353	0.346	0.339	0.332	0.327	0.321	0.318	0.314	0.309	0.305	0.302	0.299
2400.000	0.710	0.635	0.587	0.553	0.527	0.507	0.489	0.475	0.462	0.451	0.441	0.432	0.424	0.417	0.411	0.405	0.400	0.395	0.389
2500.000	0.968	0.864	0.797	0.749	0.712	0.683	0.658	0.638	0.619	0.603	0.589	0.577	0.566	0.555	0.546	0.537	0.529	0.522	0.515
2700.000	1.675	1.500	1.385	1.302	1.238	1.185	1.142	1.105	1.072	1.044	1.019	0.996	0.976	0.957	0.940	0.924	0.909	0.896	0.883
3000.000	3.115	2.834	2.645	2.504	2.392	2.301	2.224	2.157	2.099	2.048	2.002	1.960	1.922	1.887	1.855	1.825	1.797	1.771	1.747
n4																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	0.002	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
2000.000	0.056	0.053	0.053	0.052	0.050	0.049	0.049	0.050	0.048	0.050	0.048	0.051	0.049	0.048	0.046	0.045	0.050	0.049	0.048
2100.000	0.064	0.061	0.058	0.056	0.055	0.054	0.054	0.053	0.052	0.051	0.052	0.051	0.051	0.051	0.051	0.049	0.050	0.050	0.049
2200.000	0.082	0.075	0.070	0.067	0.065	0.063	0.062	0.060	0.059	0.058	0.057	0.057	0.056	0.056	0.055	0.055	0.054	0.054	0.054
2300.000	0.110	0.098	0.091	0.086	0.082	0.079	0.077	0.075	0.073	0.072	0.070	0.069	0.068	0.067	0.066	0.065	0.064	0.063	0.063
2400.000	0.148	0.131	0.121	0.114	0.108	0.103	0.100	0.097	0.094	0.092	0.089	0.088	0.086	0.084	0.083	0.082	0.081	0.080	0.079
2500.000	0.199	0.176	0.161	0.150	0.142	0.136	0.131	0.126	0.122	0.119	0.116	0.114	0.111	0.109	0.107	0.105	0.104	0.102	0.101
2700.000	0.348	0.305	0.277	0.258	0.243	0.232	0.222	0.214	0.207	0.201	0.195	0.190	0.186	0.182	0.178	0.175	0.172	0.169	0.166
3000.000	0.727	0.634	0.575	0.533	0.501	0.476	0.455	0.437	0.422	0.409	0.397	0.386	0.377	0.369	0.361	0.353	0.347	0.341	0.335

n5																			
	10.000	15.000	20.000	25.000	30.000	35.000	40.000	45.000	50.000	55.000	60.000	65.000	70.000	75.000	80.000	85.000	90.000	95.000	100.000
1500.000	-0.122	-0.123	-0.123	-0.123	-0.123	-0.123	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124	-0.124
2000.000	0.012	0.007	0.007	0.004	-0.001	-0.003	-0.003	-0.001	-0.007	-0.002	-0.007	0.001	-0.003	-0.007	-0.011	-0.014	-0.001	-0.004	-0.007
2100.000	0.043	0.033	0.026	0.023	0.018	0.017	0.015	0.013	0.012	0.009	0.010	0.009	0.008	0.007	0.008	0.004	0.005	0.007	0.003
2200.000	0.100	0.080	0.068	0.059	0.053	0.049	0.044	0.041	0.038	0.035	0.033	0.031	0.030	0.028	0.027	0.026	0.025	0.023	0.023
2300.000	0.186	0.154	0.135	0.121	0.110	0.102	0.095	0.089	0.084	0.080	0.076	0.073	0.069	0.067	0.065	0.062	0.060	0.058	0.056
2400.000	0.304	0.258	0.229	0.208	0.193	0.180	0.169	0.161	0.153	0.146	0.140	0.135	0.130	0.126	0.122	0.119	0.115	0.112	0.109
2500.000	0.459	0.395	0.354	0.325	0.302	0.284	0.269	0.257	0.246	0.236	0.228	0.220	0.214	0.207	0.201	0.196	0.191	0.187	0.183
2700.000	0.887	0.777	0.706	0.655	0.615	0.584	0.557	0.534	0.515	0.498	0.482	0.468	0.456	0.445	0.434	0.425	0.416	0.407	0.400
3000.000	1.796	1.609	1.485	1.393	1.322	1.263	1.214	1.172	1.135	1.103	1.074	1.048	1.024	1.003	0.983	0.964	0.947	0.931	0.916