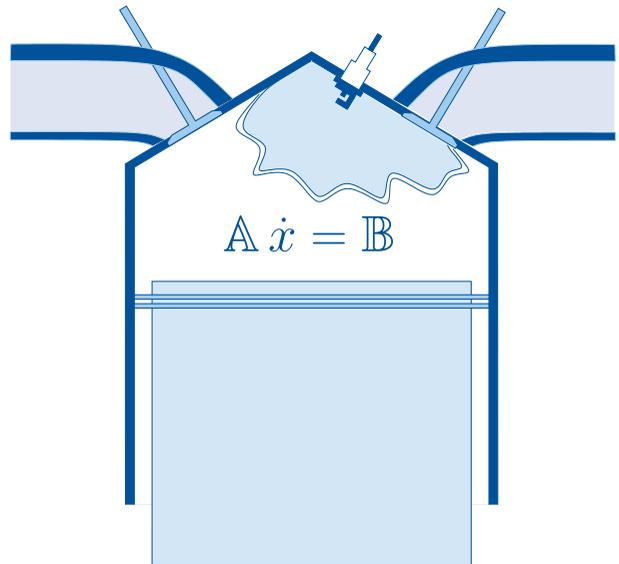


Linköping Studies in Science and Technology
Dissertations No. 1257

A DAE Formulation for Multi-Zone Thermodynamic Models and its Application to CVCP Engines

Per Öberg



Linköping University
INSTITUTE OF TECHNOLOGY

Department of Electrical Engineering
Linköping 2009

Linköping Studies in Science and Technology
Dissertations No. 1257

A DAE Formulation for Multi-Zone Thermodynamic Models and its Application to CVCP Engines

Per Öberg



Linköping University
INSTITUTE OF TECHNOLOGY

Department of Electrical Engineering
Linköping 2009

**A DAE Formulation for Multi-Zone Thermodynamic Models and its Application
to CVCP Engines**

© 2009 Per Öberg

per.oberg@liu.se
www.vehicular.isy.liu.se
Division of Vehicular Systems
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping
Sweden

ISBN 978-91-7393-607-1

ISSN 0345-7524

Printed by LiU-Tryck, Linköping, Sweden 2009

To Karin and Moa

Abstract

In the automotive area there are ever increasing demands from legislators and customers on low emissions and good fuel economy. In the process of developing and investigating new technologies, that can meet these demands, modeling and simulation have become important as standard engineering tools. To improve the modeling process new concepts and tools are also being developed.

A formulation of a differential algebraic equation (DAE) that can be used for simulation of multi-zone in-cylinder models is extended and analyzed. Special emphasis is placed on the separation between thermodynamic state equations and the thermodynamic properties. This enables implementations with easy reuse of model components and analysis of simulation results in a structured manner which gives the possibility to use the formulation in a large number of applications. The introduction and depletion of zones are handled and it is shown that the DAE formulation has a unique solution as long as the gas model fulfills a number of basic criteria. Further, an example setup is used to validate that energy, mass, and volume are preserved when using the formulation in computer simulations. In other words, the numerical solution obeys the thermodynamic state equation and the first law of thermodynamics, and the results are consistent and converge as tolerances are tightened. As example applications, the DAE formulation is used to simulate spark ignited SI and Diesel engines as well as simple control volumes and 1-dimensional pipes. It is thus shown that the DAE formulation is able to adapt to the different requirements of the SI and Diesel engine models.

An interesting application is the SI engine with continuously variable cam phasing (CVCP), which is a technology that reduces the fuel consumption. It influences the amount of air and residual gases in the engine in a non trivial manner and this SI application is used to evaluate three control oriented models for cylinder air charge and residual mass fraction for a CVCP-engine both for static and transient conditions. The models are: a simple generalized flow restriction model created with physical insight and two variants of a model that is based on an energy balance at intake valve closing (IVC). The two latter models require measurement of cylinder pressure and one also requires an air mass flow measurement. Using the SI model as reference it is shown that transients in cam positions have a large impact on air charge and residual mass fraction, and the ability of the models to capture these effects is evaluated. The main advantages of the generalized flow restriction model are that it is simple and does not require measurement of the cylinder pressure but it is also the model with the largest errors for static operating points and highest sensitivity in transients. The two models that use an energy balance at IVC both handle the transient cycles well. They are, however, sensitive to the temperature at IVC. For static cycles it is therefore advantageous to use the model with air mass flow measurement since it is less sensitive to input data. During transients however, if the external measurement is delayed, it is better to use the model that does not require the air mass flow.

The conclusion is that the DAE formulation is a flexible, robust, tool, and that it is well suited for multi-zone in-cylinder models as well as models for manifolds and pipes outside the cylinder.

Populärvetenskaplig Sammanfattning

Kraven på styrsystemet i en motor är stora. Styrsystemet ansvarar för att förbränningen sker på ett kontrollerat sätt och är centralt när det gäller att uppnå miljö och lagkrav. En kontrollerad förbränning ger låga utsläpp av miljö- och hälsovådliga ämnen, hög effektivitet och bra körkomfort. Allt eftersom kraven på våra transportmedel hårdnar blir det nödvändigt att introducera mer och mer avancerade motorer på marknaden. För att utnyttja de nya koncept som introduceras på ett effektivt sätt måste styrsystemet följa med i utvecklingen så att avvägningen mellan miljö- och lagkrav samt effektivitet och körkomfort blir bästa möjliga. Även om det i de flesta fall handlar om att försöka få ut mesta möjliga ur nya koncept som introduceras så kan en bristfällig reglering faktiskt försämra egenskaperna. Två exempel på tekniker som blivit vanliga på senare år är variabel kamfasning för bensinmotorer och multi-jet insprutare för diesel motorer.

Variabel kamfasning gör att styrsystemet kan ändra öppnings- och stängningstider för insugs- och avgasventilerna beroende på arbetspunkt istället för att som förut använda fasta värden som ger den bästa kompromissen över hela arbetsområdet. Kamfasningen ändrar mängden luft och mängden restgaser i motorn på ett icke uppenbart sätt och därför måste styrsystemet anpassas så att det tar hänsyn till detta. Att känna till mängden luft är centralt för att uppnå miljökraven eftersom det avgör hur mycket bränsle som skall sprutas in. När det gäller mängden restgaser så är det i vissa arbetspunkter en målsättning att ha så mycket så möjligt utan att för den skull påverka förbränningen negativt medan det i andra fall är mest effektivt att ha så lite så möjligt. Mängden restgas är också en av de faktorer som har störst påverkan på mängden luft som kommer in och är även därför ett viktigt delsteg i beräkningen av mängden luft.

För diesel exemplet med multi-jet insprutningssystemet så ger det en möjlighet att spruta in bränsle i cylindern flera gånger under en och samma förbränningscykel. På så sätt kan mängden partiklar och andra föroreningar i avgaserna minska samtidigt som förbränningseffektiviteten ökar. Förbränningen i en diesel-motor är dock komplex och det är svårt att förutsäga hur bränslet kommer att brinna. Styrsystemet måste alltså, under de förutsättningar som gäller i det aktuella fallet, försöka beräkna hur insprutningarna ska ske för att få ut önskat arbete med användning av minimalt bränsle utan att det bildas för mycket partiklar och andra föroreningar.

I de två exemplen som givits berörs det faktum att styrsystemet måste känna till, eller kunna beräkna, olika storheter för att sköta styrningen korrekt. Ofta kan man mäta dessa storheter, som t.ex. tryck och temperaturer eller luftflöde. P.g.a. fördröjningar i systemen samt det faktum att vi inte kan mäta riktigt allt, åtminstone inte i en bil som är ute på väg, så behövs dynamiska modeller, d.v.s. modeller som tar hänsyn till förändringar över tid. Dessa modeller måste dessutom vara tillräckligt enkla för att kunna användas i ett styrsystem men samtidigt ge tillräcklig noggrannhet för ändamålet.

Vid utvecklingen av de modeller som ska användas i ett styrsystem så behövs förståelse för hur systemet fungerar samt vad som är viktigt och inte. På så sätt kan man skala bort sådant som är onödigt. Dessutom behöver man kunna verifiera att modellen stämmer överens med verkliga förhållanden och det kan ibland vara svårt eftersom allt inte går att mäta. I fallet med restgaser som blir kvar i motorn så går dessa t.ex. inte att mäta på en motor som sitter i bil utan att modifiera den. Då kan istället en referensmodell användas. I referensmodellen ligger fokus mer på noggrannhet än på enkelhet eftersom den bara används under utvecklings- och utvärderingsfasen för styrsystemet.

De vetenskapliga bidragen består av att en differentialalgebraisk uppställning av en multizonmodell för simulering av tryck och temperaturer i en förbränningsmotor utökas och analyseras. Den speciella uppställningen ger större frihet att introducera nya zoner och kan på så sätt enkelt anpassas till olika förhållanden. Användningen av flera zoner ger möjligheten att dela upp förbränningsrummet i mindre delar och på så sätt hålla reda på vad som brann när. Det kan i sin tur användas för att förbättra modeller för t.ex. mängden kväveoxider, NO_x , eller storleken på de jon-strömmar som ibland mäts för att avgöra hur förbränningen har gått. I utökningen av den differentialalgebraiska modellen så läggs tonvikten på åtskillnaden av tillståndsekvationer och gasegenskaper. Detta möjliggör att modellkomponenter enkelt går att återanvända på ett strukturerat sätt. De numeriska egenskaperna för uppställningen utreds och det visas att uppställningen har en unik lösning givet att ett antal villkor är uppfyllda. Introduktionen och utsläckningen av zoner hanteras av uppställningen och en exempelmodell används för att visa att numeriska simuleringsresultat av uppställningen uppfyller fysikens grundlagar så som mass och energibevarande.

Som tillämpningsexempel används uppställningen för att simulera en bensinmotor utrustad med variabel kamfasning där s.k. endimensionella modeller använts för insugs- och avgasrör samt för att simulera förbränning i en dieselmotor med möjlighet till flera insprutningar. Det visas därmed att uppställningen är tillräckligt flexibel för att kunna anpassas till dessa tillämpningar. Bensinmotorexemplet används sedan för att utvärdera tre olika reglermodeller för luftmassa och restgasandel. De tre modellerna består av en generaliserad flödesrestriktionsmodell samt två varianter av en modell som använder en energibalans vid insugsventilens stängning. Båda de sistnämnda varianterna kräver mätning av cylindertryck varav en även luftmassflöde in i motorn. Referensmodellen från tillämpningsexemplet används först för att visa att snabba förändringar i kamfasningsposition har stor påverkan på mängden luft som kommer in i motorn och de tre regleranpassade modellernas förmåga att fånga detta utvärderas.

Slutsatserna av experimentet är att den enklaste modellen, d.v.s. den generaliserade flödesrestriktionsmodellen, ger störst fel under statiska förhållanden och är den modell med störst känslighet under snabba förändringar. Båda modellerna som använder sig av energibalans vid insugsventilens stängning klarar de snabba förändringarna bra. Modellerna är dock känsliga för uppskattningen av temperaturen vid ventilstängning. Känsligheten är störst för modellen där luftmassan inte mäts och det är därför under statiska förhållanden bättre att använda modellen som behöver mätning av luftmassflöde. Under snabba förändringar, där mätvärdet från luftmassflödesgivaren kan vara felaktigt p.g.a. fördröjningar, är det dock bättre att använda modellen som klarar sig utan luftmassflöde.

Acknowledgments

This work has been carried out at the Division of Vehicular Systems at the Department of Electrical Engineering, Linköpings Universitet. The work has been a part of the VISIMOD and MOVIII projects funded by the Swedish Foundation for Strategic Research.

First of all I would like to thank my supervisors Lars Eriksson and Lars Nielsen for their guidance and for believing in me. Without them this would not have been possible. I would also like to thank them for being there when it really counted. Per Andersson, Markus Klein and Ylva Nilsson are acknowledged for interesting and fruitful research discussions. Martin Gunnarsson is also acknowledged for helping with the measurements and keeping the engine lab running.

I would also like to thank Gustaf Hendeby, Erik Frisk, and Erik Hellström for always taking time to help me sort out the details in certain computer programs. Your help has been very appreciated and it's comforting to know that I'm not alone in the open source swamp. I guess we're even, but a special thanks goes to Gustaf. You must have read every manual ever written.

All the employees and masters students that have contributed to the atmosphere at Vehicular Systems are also thanked. A special thanks to Martin Gunnarsson for all the interesting discussions over the years is, however, at place. Of all the people at Vehicular Systems I never managed to bore you with my furnace control system discussions.

Finally I would like to thank my family, Karin and Moa, who have supported me throughout the process. Your support has been invaluable and I don't know what I'd do without you.

Contents

1	Introduction	1
1.1	Computer Simulation	2
1.2	In-Cylinder Models	2
1.2.1	DAE Formulation	3
1.2.2	Application of the DAE Formulation	4
1.2.3	Control Oriented Modeling of CVCP Engine	4
1.2.4	A Diesel Combustion Application	5
1.3	Thesis Outline	5
1.4	Contributions	6
I	Simulation Methodology	9
2	Thermodynamics of Open and Closed Systems	11
2.1	Outline of the Chapter	11
2.2	Thermodynamic Summary	12
2.2.1	Thermodynamic Properties	12
2.2.2	Thermodynamic Laws	14
2.2.3	The Fundamental Equation and The Maxwell Relations	15
2.3	Introducing Ideal and Non Inert Gases	16

2.3.1	Mixture of Reacting Ideal Gases, the <i>Well Stirred Mixer</i> and the <i>Well Stirred Reactor</i>	16
2.4	Open and Closed Systems with Non Inert Gases	18
2.4.1	Keeping Track of the Gas Composition	20
2.5	The Equation of State	20
2.5.1	State Equation for the Well Stirred Reactor	21
2.5.2	State Equation for the Well Stirred Mixer	21
2.6	Energy Preservation for Open Systems	22
2.6.1	Energy for the Well Stirred Reactor	23
2.6.2	Energy for the Well Stirred Mixer	25
2.7	Thermochemical Properties	26
2.8	Collecting the Equations and Introducing Combustion	26
2.8.1	Combustion	27
2.8.2	The Control Volume Model	29
2.9	A First Example and Validation	30
2.10	Concluding Remarks about the The Control Volume Model	34
3	Gas Models for Simulation of Thermodynamic Systems	35
3.1	Outline of the Chapter	36
3.2	Characterizing Ideal and non Inert Gases	37
3.2.1	Examples of Gas Descriptions	37
3.3	Calculation of Gas Properties using a Chemical Equilibrium Program	42
3.3.1	Interfacing with CHEPP	43
3.3.2	Calculation of Mass Specific Thermodynamic Properties for the Well Stirred Reactor	45
3.3.3	Calculation of Mass Specific Thermodynamic Properties for the Well Stirred Mixer	47
3.4	Implementation Choices and Simple Gas Models	47
3.4.1	Full Equilibrium Gas Model	49
3.4.2	Four Component Gas using Chemical Equilibrium	49
3.4.3	Tables	51
3.4.4	A Simple Equilibrium Model	52
3.5	Concluding Remarks about the Modeling of Gases	57
4	A DAE Formulation for Simulation of Thermodynamic Systems	59

4.1	Outline of the Chapter	59
4.2	The Multi-Zone DAE Formulation	60
4.2.1	A Single Zone Model in the DAE Formulation	62
4.3	Adding and Removing Zones	63
4.4	Existence and Uniqueness of the DAE $\mathbb{A} \dot{\mathbf{x}} = \mathbb{B}$	63
4.4.1	Determinant of the \mathbb{A} -matrix	64
4.4.2	Condition Number of the \mathbb{A} -matrix	66
4.4.3	Initialization and Depletion of Zones	71
4.5	Numerical Properties of the DAE Formulation	74
4.5.1	Investigation Setup	74
4.5.2	Short Introduction to ODE Solvers	75
4.5.3	Drift and Convergence for Different Solvers and Tolerances	76
4.5.4	Drift and Convergence for Different Gas Models	81
4.5.5	Influence of Initialization	89
4.6	Concluding Remarks about the DAE Formulation	91
5	Implementation of the DAE Framework	93
5.1	Outline of the Chapter	93
5.2	PSPACK, a Process Simulation Package, Basic Idea and Overview of the Simulation Package	93
5.2.1	Overview of PSPACK modules	94
5.2.2	Structure – <code>psSimStruct</code>	96
5.3	Detailed Descriptions of the Modules	97
5.3.1	Module – <code>psKernel</code> – PSPACK Kernel	97
5.3.2	Module – <code>psGeometry</code> – Engine Geometry	99
5.3.3	Module – <code>psHeat</code> – Heat Transfer	101
5.3.4	Module – <code>psVibe</code> – Burned Mass Fraction	104
5.3.5	Module – <code>psSimPar</code> – Simulation Parameters	105
5.3.6	Module – <code>psThermProp</code> – Thermochemical Properties	105
5.3.7	Module – <code>psValve</code> – Valve Areas	107
5.3.8	Frequently Used Model Components	108
5.3.9	Module – <code>psModels</code> – Cylinder/Engine Models	112

II	Applications of Theory	119
6	Modeling of SI and Diesel Engines	121
6.1	Outline of the Chapter	122
6.2	Control Oriented Modeling of the Gas Exchange Process in VCT Engines	122
6.2.1	Gas Model	124
6.3	Control Oriented Gas Exchange Models for CVCP Engines and	124
6.4	Diesel Engine Modeling	125
6.4.1	Adding a Liquid Incompressible Zone	126
6.4.2	Evaluation of Mass, Energy, and Volume Conservation	126
6.4.3	Investigation of Gas Model Dependence	133
7	Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines	135
7.1	Introduction	136
7.1.1	The Models	136
7.2	Experimental Setup	137
7.3	Reference Model	137
7.3.1	Model Parameters	138
7.3.2	Accuracy of Reference Model	138
7.4	The Evaluated Models	139
7.4.1	Model A	139
7.4.2	Model B	140
7.4.3	Model C	141
7.5	Results	146
7.5.1	Model A	146
7.5.2	Model B	146
7.5.3	Model C	148
7.6	Conclusions	151
7.A	Nomenclature	152
7.B	Reference Model	153
7.C	Tuning of Model Parameters	155
8	Control Oriented Gas Exchange Models for CVCP Engines and their Transient Sensitivity	159
8.1	Reference Model	162

8.2	The Evaluated Models	163
8.2.1	Model A	163
8.2.2	Model B	164
8.2.3	Model C	168
8.3	Investigation Setup	171
8.4	Deviations for Transient Cycles	172
8.5	Evaluation of Model Performance	172
8.A	Nomenclature	178
8.B	Parameters for the Reference Model	179
8.C	Model Parameters	180
Bibliography		183
A Notational Conventions		187
B PSPACK Functions and Variables		193
B.1	Basic Idea and Overview	193
B.1.1	Structure – psSimStruct	193
B.2	Module – psKernel – PSPACK Kernel	195
B.2.1	Function – psDX	195
B.2.2	Function – psDX_SingleZone_ConstV	196
B.2.3	Function – psAdiabaticMix	197
B.3	Module – psGeometry – Engine Geometry	198
B.3.1	Global Parameter – psGeometry	198
B.3.2	Function – psVolume	198
B.3.3	Function – psDVolume	199
B.3.4	Function – psArea	199
B.4	Module – psHeat – Heat Transfer	200
B.4.1	Function – psDQ	200
B.4.2	Function – psDQlp	201
B.5	Module – psVibe – Burned Mass Fraction	202
B.5.1	Function – psVibe	202
B.5.2	Function – psVibeEvents	203
B.5.3	Function – psDVibe	203
B.6	Module – psSimPar – Simulation Parameters	204

B.7	Module – psThermProp – Thermochemical Properties	205
B.7.1	Global Parameter – psThermProp	205
B.7.2	Function – psThermProp	206
B.7.3	Function – psCalcMixXgc	207
B.7.4	Function – psCalcReactXgc	207
B.7.5	Function – psCalcPhi	208
B.7.6	Function – psCalcXgc	208
B.7.7	Available Alternatives	209
B.8	Module – psValve – Valve Areas	211
B.8.1	Function – psValve	211
B.8.2	Function – psValveEvents	212
B.9	Module – psModels – Cylinder/Engine Models	213
B.9.1	Function – psConvertSimOutput	213
B.9.2	Available Alternatives	214
C	Example S1 Engine Setup	217

Introduction

There are high demands on the engine control units (ECU) since they are central for achieving good performance like stable combustion, good torque response, good fuel economy, and low emissions. New technologies are introduced to meet the increasing demands from legislators and customers, and these must also be handled and properly controlled by the ECU. For example, continuously variable cam phasing is a technology that provides an interesting possibility to reduce the fuel consumption in spark ignited (SI) engines but it also influences the amount of air and residual gases in the engine in a non trivial manner. The cylinder air charge is important for fuel control and torque control, while residual mass fraction is a crucial factor that limits stable engine operation since it influences the combustion variability. Therefore it is essential for the control system to know the cylinder air charge and residual mass fraction. For Diesel engines, as another example, the common rail multi-jet system is a technology that allows for multiple injections for each combustion which provides a possibility to reduce particulate emissions and fuel consumption. The complexity of Diesel engine combustion makes the control of a multi-injection system a non trivial problem, and understanding the effects that are involved can help in the design of such control systems.

The central topic in this work is modeling and simulation of processes close to and in the cylinder. The focus has been twofold; models that are used for control and in the design process of control systems as well as models that increase the understanding of the system that is being controlled. The models that are used for analysis are founded upon a robust thermodynamic framework while the control oriented models are simpler. The models that are studied are zero- or one-dimensional, i.e. it is assumed that there is no spatial variation or that the spatial variation is captured by one single dimension. The systems that are studied are a turbocharged SI engine fitted with a continuously variable cam phasing (CVCP) mechanism and a multi-jet Diesel engine. A common attribute for the models are that they lack analytical solutions and computer simulation is therefore needed.

1.1 Computer Simulation

In 1961 Dr. Edward N. Lorenz accidentally rediscovered chaos theory. When running simulations of a simple weather model on his computer he wanted to reuse a result from a previous simulation. According to the tale, he was expecting the same results but with higher accuracy. Instead he found out that his simulation results were completely different. Obviously he was surprised and at first he thought there was something wrong with the computer. After a while he realised that he had entered the numbers of the second simulation manually and with a reduced accuracy. The reduction of accuracy was less than 1 promille but the results were nevertheless completely different. Lorenz's discovery came to be the foundation of modern chaos theory and greatly affected the meteorology community. His discovery gave rise to the question if a flap of a butterfly's wings is enough to set off a tornado.

We may draw two conclusions from this story. Firstly, if you make a mistake, make an interesting one. Secondly and most important, computer simulation can, if used correctly, be a great tool in the design, analysis, or control of complex systems. A good model can also give understanding of the system by helping to answer the questions "Why is it that this happens?" and "Is this effect or phenomena important for the end result?". It is, however, dangerous to blindly trust results from simulations without considering the traps and pitfalls involved. Therefore it is important to study how the numerical solution to the equations in the model behave for different solvers as well as different relative tolerances and parametrizations. It is also interesting to assess how simplifications in the models affect the results.

There is no analytic solution available for comparing the models that are used in this work, and therefore other approaches for validating the models have to be taken instead. The closest to an exact solution that we can get is a simulation where the simulation accuracy is as high as possible. It is, however, necessary to note that even if tighter tolerances perform better on average, random occurrence of events can affect the step length control for one particular simulation with a low accuracy so that it yields better results for a particular simulation. Because of this the simulation that is used as reference is not necessarily the closest to the exact solution and this needs to be considered when comparing results from simulations with different tolerances.

1.2 In-Cylinder Models

Computer simulation of engine processes was introduced in the late 50's and there are several papers in the early 60's, see e.g. [10, 41, 52]. In the late 60's cylinder pressure analysis by use of the first law of thermodynamics was investigated in [33]. In [16] the same modeling approach is extended to cover heat transfer and crevice effects. The models can be used both in heat release analysis where the burn rate is calculated given the pressure in the cylinder or in simulation where the burn rate is given as input while the pressure and temperature traces are calculated. The early models that were used were single zone models, i.e. they assume that the gas in the cylinder is homogeneous in temperature and composition. During the combustion, however, the flame front divides the

cylinder in two parts which are more or less separate yielding a combustion chamber that is not homogeneous. The gas in front of the flame is cold while the gas after the flame is hot. In fact, the assumption that the unburned and burned elements do not mix at all is more realistic than the assumption that the cylinder content is homogeneous [25]. However, there are many occasions where the single zone models are adequate, for example if pressure and mean gas temperature traces are the only required outputs from the simulations. There are other situations where the desired accuracy cannot be obtained by assuming that the mixture is homogeneous in temperature and composition and it is important keep track of the temperature distribution. This can be done by introducing multiple zones in the control volume model. One way to think of the zones is as N sub-volumes of the control volume. The sub-volumes have flexible barriers, that can allow mass transfer, in between them. The zones can have a geometrical interpretation or they can be used for bookkeeping of the mass that have burned at a specific instance. In [29] two zones are used to model NO_x emissions for an engine with exhaust gas recirculation and in [42] and [11] a two zone layout is used for modeling of diesel combustion. Other examples of when multiple zones can be beneficial is modeling of knock, as in [23] where three zones are used, and modeling of ion currents, as in [1] where two zones are used to estimate the pressure from ion current measurements.

1.2.1 DAE Formulation

A multi-zone approach is proposed in [37, 36] where a differential algebraic equation (DAE) formulation of a multi-zone model is developed and tested for a two zone setup. In [19] the DAE formulation is used in an object oriented multi-zone simulation environment implemented in JAVA. In the testing of the environment up to 35 zones were used. The DAE formulation is also used in for example [6] in which NO_x formation is studied and in [9] and [8] where it is used to model ion currents by introducing multiple zones.

In Chapter 2 a thermodynamic framework is developed where special emphasis is placed on the separation between thermodynamic state equations and the thermodynamic properties, and in Chapter 3 a thermodynamic property framework is developed. The separation between thermodynamic state equations and thermodynamic properties yields the possibility to model both *Well Stirred Mixers* and *Reactors* using the same set of equations. In Chapter 4 the DAE formulation that is presented in [37] and that can be used for simulation of multi-zone in-cylinder models is extended with the ability to have composition changes using the thermodynamic property framework. The introduction and depletion of zones are handled by the framework which gives the possibility to use the formulation in a large number of applications. It is shown that using the DAE formulation in computer simulations is viable, i.e the DAE formulation will have a unique solution as long as the gas model fulfills a number of basic criteria. Further, an example setup is used to validate that energy, mass, and volume are preserved when using the formulation in computer simulations. In other words, the numerical solution obeys the thermodynamic state equation and the first law of thermodynamics, and the results converge as tolerances are tightened.

1.2.2 Application of the DAE Formulation

Computer simulation of engines encompasses a number of different model types where the models range from complex combustion, or heat release, models to simple zero dimensional manifold filling models. Models of different complexity are required depending on the aspect that is studied and a flexible framework with interchangeable parts is therefore desirable. In Chapter 5 an implementation, that is able to simulate different zone layouts and engine setups using the DAE formulation, is described. The implementation of the framework relies on results from Chapters 2–4. In the thermodynamic framework that is founded in Chapters 2 and 3 special emphasis is placed on the separation between thermodynamic state equations and the thermodynamic properties, and this is done to support the demands from the application in Chapter 5 where it enables easy reuse of model components and analysis of simulation results in a structured manner. Being able to reuse model components does not only save time, it also helps reducing implementation errors because models that are previously debugged and validated can be used.

The application framework is used to design and implement a reference model that is used in the publications [38, 39] that make up Chapters 7 and 8 respectively. It is also used in two masters theses [24, 28].

1.2.3 Control Oriented Modeling of CVCP Engine

As mentioned earlier, continuously variable cam phasing (CVCP) is a technology that provides an interesting possibility to reduce the fuel consumption but it also influences the amount of air and residual gases in the engine in a non trivial manner. The introduction of this technology requires that new control strategies are developed and these strategies require that the cylinder air charge and the residual mass fraction are either measured or estimated using a model. Measuring the cylinder air charge is fairly easy, at least when steady state operating points are considered. Directly measuring the residual mass fraction is, however, harder and requires special equipment. A possible solution is to use a reference model during the development and validation process of the control oriented model. Using a reference model can also be beneficial because it can give physical insights about the processes that are of importance when constructing the control oriented model. There are commercially available packages that can be used as a reference model, e.g. [48] and [50], but with the drawback that valuable insight into the details of the models are lost.

There are other publications, e.g. [27, 34, 17], that propose control or estimation algorithms for different types of variable valve train systems, but the focus there is on the cylinder air charge and not on the residual gas fraction. In [27] and [34] cylinder air charge for *dual equal* and *intake only* CVCP systems, that have moderate valve overlap, is studied. In [17] the focus is on fuel injection for the same type of engines as mentioned earlier. In [18] the focus is models for, amongst other things, air charge and residual mass fraction, of a supercharged engine.

In [38, 39] that make up Chapters 7 and 8, three control oriented models for cylinder air charge and residual mass fraction for a *dual independent* CVCP-engine with a turbocharger are analysed both for transient and static conditions.

1.2.4 A Diesel Combustion Application

For Diesel engines, the common rail multi-jet system is a technology that allows for multiple injections for each combustion which provides a possibility to reduce particulate emissions and fuel consumption. The complexity of Diesel engine combustion makes the control of a multi-injection system a non trivial problem, and understanding the effects that are involved can help in the design of such control systems.

Diesel combustion is complex because it is governed by turbulent fuel-air mixing. The combustion process depends on the temperature, pressure, and composition close to and in the fuel spray. Diesel combustion therefore puts high demands on the thermodynamic model because the composition in proximity to the fuel spray changes rapidly as air is entrained in the fuel spray and when the mixture is finally combusted. The combustion process in Diesel engines is therefore an excellent test application for the thermodynamic framework. A complicating factor is that the liquid Diesel spray is usually modeled as an incompressible volume and therefore has to be treated in a different way compared to a gas.

In Chapter 6 the DAE framework is applied to Diesel engine modeling. The model is based on [42], and an implementation from a masters thesis, [28], that uses the framework is used in the validation of the framework.

1.3 Thesis Outline

Throughout this thesis an example setup of a two-zone in-cylinder model is used to illustrate different phenomena and effects that different simplifications has. The model setup and parametrization is described in Appendix C and it is mainly used in Chapters 3 and 4. As a help to the reader Appendix A lists the notation that is used together with frequently referenced equations. Note also that all references, both from chapters and publications, are listed in the back of the thesis.

In Chapter 2, **Thermodynamics of Open and Closed Systems**, the foundation for the model in Chapter 4 is built. It introduces and derives the equations and gives an overview of the model structure. The chapter starts with a thermodynamic summary and concludes with a model for simulation of control volumes and a first validation that serves as a first proof of concept for the chosen formulation.

In Chapter 3, **Gas Models for Simulation of Thermodynamic Systems**, the focus is on the calculation of thermodynamic properties and thermodynamic property models. The concept of composition parameters is introduced and two simplified gas models are given.

In Chapter 4, **A DAE Formulation for Simulation of Thermodynamic Systems**, a DAE formulation for simulation of multi-zone in-cylinder models is extended and analysed. The formulation builds on the framework in Chapter 2 and Chapter 3.

In Chapter 5, **Implementation of the DAE Framework**, the package PSPACK is described. The package implements the DAE framework applied to in-cylinder models for SI and Diesel engines as well as components such as manifolds and pipes. The structure of the package and its sub-division into modules together with the available choices for

the different modules are described. The chapter also documents the interfaces that have been developed, and highlights some important implementation choices.

Chapter 6, **Modeling of SI and Diesel Engines**, introduces the two publications that are included as Chapter 7, [38], and Chapter 8, [39]. It discusses the differences between the models that were used in these publications and the model that is described in Chapter 5. The reason for differences is that [38, 39] were written while the package was still under development. Furthermore, the application of the DAE model to a Diesel engine is also described. The model that is used was implemented in a masters thesis, [28], and the results from simulations of this model are used to show the flexibility of the framework.

Chapter 7, **Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines**, corresponds to [38], where an early version of the implementation in Chapter 5 is used to evaluate three control oriented models that predict cylinder air charge and/or residual mass fraction. The three models all predict residual mass fraction, and one of the models also predicts the air charge.

Chapter 8, **Control Oriented Gas Exchange Models for CVCP Engines and their Transient Sensitivity**, corresponds to [39], in which another early version of the implementation in Chapter 5 is used to analyze the same set of control oriented models that were used in Chapter 7. There are two main additions made. The first addition is the investigation of how much transients in cam phasing affect the air charge as well as residual mass fraction, and the models ability to capture these effects. Additions have also been made to the models of the intake- and exhaust ducts of the reference model which are here one-dimensional models, that capture the ram effects in the manifolds, instead of the static pressure models used in the earlier publication.

1.4 Contributions

- A thermodynamic framework where special emphasis is placed on the separation between thermodynamic state equations and the thermodynamic properties is developed. The separation between thermodynamic state equations and thermodynamic properties yields the possibility to model both *Well Stirred Mixers* and *Reactors* using the same set of equations.
- A differential algebraic equation (DAE) formulation of a multi-zone in-cylinder model is extended with ability to have composition changes using the thermodynamic property framework.
- Existence and uniqueness of solution for the DAE formulation is analysed. It is shown that using the DAE formulation in computer simulations is viable, i.e the DAE formulation will have a unique solution as long as the gas model fulfills a number of basic criteria.
- It is shown how the DAE framework based on compressible gas models can be extended to also handle incompressible sub-components.

- A factorization of the $\mathbb{A}(x, y)$ -matrix is given. The numerical properties related to the conversion of the DAE to an ODE are analysed using this factorization. Finally it is shown that the numerical solutions are consistent with the state equation and the energy equation.
- The initialization of a zone is analyzed, and the effect that an error in the initial volume has is investigated.
- Four gas models are suggested and analysed. The effect of different gas models and tolerances on the simulation result is investigated and advice on how to choose relative tolerance is given.
- An application of the framework has been developed and implemented. The implementation can handle for example CVCP and Diesel engines and has been used in a number of publications which proves its usability. The application is also used to gain knowledge about three control oriented models applied to a CVCP-engine. The models sensitivity to inputs as well as noise both for static and transient conditions are evaluated and it is shown that
 - The simplest of the evaluated models has the advantage that it doesn't require cylinder pressure to be measured. It is however the model with the largest errors for static operating points and highest sensitivity in transients.
 - The two models that require the cylinder pressure to be measured and use an energy balance at IVC both handle the transient cycles well. They are, however, sensitive to the temperature at IVC.
 - For static cycles it is advantageous to use the model with air mass flow measurement since it is less sensitive to input data. During transients however, if the external measurement is delayed, it is better to use the model that does not require the air mass flow.

Part I

Simulation Methodology

2

Thermodynamics of Open and Closed Systems

Thermodynamically an engine is a series of open systems of which some are closed during parts of the engine cycle. Heat, work, and mass can be transferred to or from the systems and reactions, that change the number of moles of different molecules in the gases, can take place. The transfer of heat, work and mass affects the temperature and pressure of the systems and knowledge of thermodynamics is therefore crucial when building an engine model. In this chapter the foundation for the model in Chapter 4 is built. The highlight of the chapter is the model in (2.20) which is a differential equation for the pressure and temperature of a control volume. The chapter introduces the equations that are necessary for the model in (2.20) and serves as an overview of the notation that is adopted throughout this work.

2.1 Outline of the Chapter

In Section 2.2, **Thermodynamic Summary**, the foundation is built. Energy, work, and heat as well as some thermodynamic basics are discussed. Thermodynamics is a big subject, that is not possible to cover in a thesis, and therefore this section summarizes only the fundamentals that are central and used in the modeling, analysis and applications.

Section 2.3, **Introducing Ideal and Non Inert Gases**, introduces a multi-molecule gas, where the molecules are allowed to react with each other. Reacting and frozen mixtures where the number of moles do or do not depend on temperature and pressure are discussed. It is noted that in case of a frozen mixture it's the molecules that should be bookkept and in the equilibrium case it's the atoms. Bookkeeping equations are introduced as well as nomenclature for mole and mass fractions.

Section 2.4, **Open and Closed Systems with Non Inert Gases**, is a short introduction that discusses how to model open and closed systems. Pressure and temperature as well as composition are chosen as state variables and it is then shown how the thermodynamic state equation together with the first law of thermodynamics can be used to derive expressions for the temperature and pressure derivatives.

In Section 2.5, **The Equation of State**, the ideal gas law is used as a foundation for a state equation for a non inert multi-molecule gas. The state equation is then differentiated to obtain an explicit expression that only involves quantities which are available as gas properties together with the sought pressure and temperature differentials.

In Section 2.6, **Energy Preservation for Open Systems**, the first law of thermodynamics is used to obtain a second expression for the temperature and pressure differentials. The energy in the beginning and end of a small time interval Δt is used in the deduction to keep track of, as well as give understanding of, the processes that take place.

In Section 2.7, **Thermochemical Properties**, a number of gas properties that is needed by the model in (2.20) are collected.

In Section 2.8, **Collecting the Equations and Introducing Combustion**, the equations that are needed to simulate an open system are collected and how to model combustion in the framework is discussed.

In Section 2.9, **A First Example and Validation**, a bomb calorimeter experiment is used as an example system that serves as proof of concept and a first validation. Heating values, q_{HV} , are calculated using the example system and the results are compared with data from [25].

2.2 Thermodynamic Summary

A thermodynamic system is a portion of space with a boundary. If the boundary allows mass, heat, and work transfer it's called an open system and when the boundary only allows heat and work transfer it's called a closed system.

There are many ways to describe an open thermodynamic system, or control volume. In this work the focus is on models that describe the evolution of the thermodynamic state in the control volume. At a minimum there should therefore be some initial state and a set of equations that describe how the system state evolves. In this text only the essentials for simulation of an engine are summarized. For a more thorough description of thermodynamics see for example [7] for an engineering approach or [31] for a statistical approach. The subject of physical chemistry is covered in [3].

2.2.1 Thermodynamic Properties

The state in a system can be described by a number of variables. Some variables are directly measurable while others are introduced as intermediate quantities that are useful when analysing thermodynamic systems. A few examples are temperature T , pressure p , volume V , internal energy U , and entropy S , where the three first can be measured. It

is postulated that the state of a thermodynamic system in equilibrium can be completely described by two *independent* state variables and the system size, for example the masses of the different molecule species of the system. If for example the temperature, pressure, and masses are known then we can calculate all other variables and properties, as long as pressure and temperature are independent variables. For a single phase system temperature and pressure are independent [7]. The above postulate will be used later in the text when building a model for open and closed systems.

Intensive and Extensive properties

Thermodynamic properties are either *intensive* or *extensive*. Extensive properties vary with the size of the system while intensive do not. For example temperature T and pressure p are intensive properties while volume V and entropy S are extensive properties. For every extensive property it is possible to define an intensive property by dividing with mass. These so called *mass specific* properties are usually written using lower-case version of the letter denoting the property. For example we have enthalpy and internal energy $h = \frac{H}{m}$ and $u = \frac{U}{m}$

In some cases it is advantageous to use *mole specific* properties, for example when using the ideal gas law, $pV = n\bar{R}T$. The reason is that the number of moles, n , occurs directly in the equation. Mole specific properties are denoted with a tilde over the lower-case property. For example we have, once again, enthalpy and internal energy

$$H = mh = n\tilde{h} \quad \text{and} \quad U = mu = n\tilde{u}$$

where m is the mass while n is the number of moles. For a gas made up of molecules with the molar mass M we therefore have

$$h = \frac{n\tilde{h}}{m} = \frac{\tilde{h}}{M} \quad \text{and} \quad u = \frac{\tilde{u}}{M}$$

Energy, Internal Energy, Enthalpy, Work, and Heat

The total amount of energy for a system is denoted E and it is composed by *internal*, *kinetic*, and *potential* energy.

$$E = U + KE + PE$$

In the present analysis kinetic and potential energy are neglected, which leaves the internal energy. The concept of *internal energy*, U , is central in the analysis of thermodynamic systems. It is the sum of all microscopic forms of energy, i.e. those that are related to the molecular structure of the system. It is also a measure of the amount of energy in the system that is not kinetic or potential.

In some applications the *enthalpy* H is used. Enthalpy is defined as $H = U + pV$ and is introduced because it simplifies the equations and the analysis of open systems. In particular the enthalpy encapsulates both the internal energy and the mechanical work that is done on the system by a flowing gas when it enters a system.

Heat Q and *Work* W represent energy in transition and do not accumulate as such. Energy that is transferred due to a temperature difference is heat while other forms of energy transfer is work. Work in thermodynamics can be the same as the traditional concept of mechanical work but it can also mean other things like for example electrical work. Only mechanical work is, however, considered here.

A distinction is made between *reversible* and *irreversible* processes. For a *reversible* change in a closed system the work is $dW = -p dV$ while the heat transfer is $dQ = T dS$. This is later used in Section 2.2.3 in the *fundamental equation*.

Specific Heats

The specific heats are frequently used in thermodynamics to capture the amount of energy per unit mass that it takes to raise the temperature of a system one degree. There are two of them, one at constant pressure and one at constant volume. More specifically they are defined as

$$c_v = \left(\frac{\partial q}{\partial T} \right)_V \quad \text{and} \quad c_p = \left(\frac{\partial q}{\partial T} \right)_p$$

An interesting detail is the subscript for constant volume or pressure. They are needed because a system can be described by a number of variable combinations. For the constant volume case it is assumed that the system is described by the volume temperature pair $\{V, T\}$ and in the constant pressure case it is assumed that the system is described by the pressure temperature pair $\{p, T\}$. It is easy to show that

$$c_v = \left(\frac{\partial u}{\partial T} \right)_V \quad \text{and} \quad c_p = \left(\frac{\partial h}{\partial T} \right)_p$$

and therefore these quantities are often used in the analysis of open systems where both internal energy, u , and enthalpy, h are commonly used energy measures.

2.2.2 Thermodynamic Laws

There are a number of thermodynamic laws available. Depending on the author they are either postulated or deduced statistically. Most well known are probably the first and second laws of thermodynamics. The first law of thermodynamics states that energy is preserved while the second specifies the direction of spontaneous heat flow.

The first law of thermodynamics can be written as $dE = dQ + dW$, or with $dU = dE$ due to the fact that kinetic and potential energy of the system are assumed to be constant

$$dE = dU = dQ + dW \quad (2.1)$$

where E is the total energy of the system, U is the internal energy, Q is the heat, and W is the work.

From the second law of thermodynamics it is possible to formulate the Clausius inequality

$$dS \leq \frac{dQ}{T} \quad (2.2)$$

that holds with equality for a *reversible* process.

2.2.3 The Fundamental Equation and The Maxwell Relations

Using the first law (2.1) together with the Clausius inequality (2.2) for a reversible process gives the *fundamental equation* of thermodynamics

$$dU = T dS - p dV \quad (2.3)$$

Even if a reversible process was assumed to obtain (2.3) the equation is valid for *irreversible* processes too, see [3] for a longer discussion.

The result in (2.3) can be used to derive a number of relationships between quantities that do not, at a first glance, look related. The postulate that any two independent variables together with the system size may be chosen to describe a system completely yields the possibility to write U as a function of S and V . We then have that

$$dU(S, V) = \left(\frac{\partial U}{\partial S} \right)_V dS + \left(\frac{\partial U}{\partial V} \right)_S dV$$

and hence that $T = \left(\frac{\partial U}{\partial S} \right)_V$ while $p = - \left(\frac{\partial U}{\partial V} \right)_S$.

Because $U(S, V)$ is a function that is completely described by its arguments S and V we have that

$$\left(\frac{\partial}{\partial V} \left(\frac{\partial U(S, V)}{\partial S} \right) \right)_S = \left(\frac{\partial}{\partial S} \left(\frac{\partial U(S, V)}{\partial V} \right) \right)_V$$

and therefore also that

$$\left(\frac{\partial T}{\partial V} \right)_V = - \left(\frac{\partial p}{\partial S} \right)_V \quad (2.4)$$

This equation is an example of a *Maxwell relation* and by the definition of enthalpy, $H = U + pV$, Helmholtz Free Energy, $F = U - TS$, and Gibbs Function $G = H - TS$ it is possible to derive three more.

The four well known Maxwell Relations are summarized in Table 2.1 below. They are practical when manipulating and simplifying thermodynamic expressions.

Table 2.1: The four Maxwell Relations derived from the internal energy, the enthalpy, the Helmholtz energy, and Gibbs function.

$$\begin{aligned} \left(\frac{\partial T}{\partial V} \right)_V &= - \left(\frac{\partial p}{\partial S} \right)_V & \left(\frac{\partial T}{\partial p} \right)_S &= \left(\frac{\partial V}{\partial S} \right)_p \\ \left(\frac{\partial p}{\partial T} \right)_V &= \left(\frac{\partial S}{\partial V} \right)_T & \left(\frac{\partial V}{\partial T} \right)_p &= - \left(\frac{\partial S}{\partial p} \right)_T \end{aligned}$$

2.3 Introducing Ideal and Non Inert Gases

An ideal gas is by definition a gas that obeys the ideal gas state equation (2.5). In the engineering approach in [7] the ideal state equation is formulated as a model capturing results from measurements while in statistical thermodynamics this equation is deduced using the assumption that there are no other forces acting on the molecules than those of the confining box [31].

For commonly available gases it is a good approximation to assume that the ideal gas law is valid. The ideal gas law is formulated as

$$pV = n\tilde{R}T \quad (2.5)$$

where p is pressure, V is volume, n is the number of moles of molecules, $\tilde{R} = 8.315$ [$\text{J}_{\text{mol}}\text{K}$] is the molar gas constant, and T is the temperature. \tilde{R} is used as the molar gas constant to conform with the notation that is used for *mass specific* and *mole specific* properties.

It is easy to show that the internal energy of an ideal gas is a function of temperature only. This is for example done by using the fundamental equation (2.3) together with the ideal gas law (2.5) as well as one of the Maxwell relations from Table 2.1 and rewriting dS and dV in terms of dp and dT

$$\begin{aligned} dU &= T dS - p dV = T \left(\left(\frac{\partial S}{\partial T} \right)_p dT + \left(\frac{\partial S}{\partial p} \right)_T dp \right) - \\ & p \left(\left(\frac{\partial V}{\partial T} \right)_p dT + \left(\frac{\partial V}{\partial p} \right)_T dp \right) = \left/ \left(\frac{\partial V}{\partial T} \right)_p = - \left(\frac{\partial S}{\partial p} \right)_T \text{ From Table 2.1} \right/ \\ &= T \left(\left(\frac{\partial S}{\partial T} \right)_p dT - \left(\frac{\partial V}{\partial T} \right)_p dp \right) - p \left(\left(\frac{\partial V}{\partial T} \right)_p dT + \left(\frac{\partial V}{\partial p} \right)_T dp \right) = \\ &= \left/ V = \frac{n\tilde{R}T}{p} \Rightarrow \left(\frac{\partial V}{\partial T} \right)_p = \frac{n\tilde{R}}{p} \text{ and } \left(\frac{\partial V}{\partial p} \right)_T = -\frac{n\tilde{R}T}{p^2} \right/ = \\ &= \left(T \left(\frac{\partial S}{\partial T} \right)_p - p \left(\frac{\partial V}{\partial T} \right)_p \right) dT + 0 dp \end{aligned}$$

which shows the result. As mentioned before, two state variables are enough to completely describe a system in equilibrium and therefore p can be replaced with any other variable. The result is always the same, if the temperature is constant the internal energy does not change for an ideal gas. This property of ideal gases is used later when choosing state variables for a system.

2.3.1 Mixture of Reacting Ideal Gases, the *Well Stirred Mixer* and the *Well Stirred Reactor*

For a closed thermodynamic system with inert gases the number of molecules are constant over time. In an engine however, the working medium is composed of molecules

that are sometimes allowed to react with each other but always follow a path of equilibrium. The kinetics of the reactions between the molecules is important when trying to calculate things like flame development and knock tendency of a mixture. In the following presentation, however, the distribution of species is either assumed to be static, in the sense that no **reactions** that change the distribution occur, or the reactions are assumed to be a quasi-static process, more specifically they are assumed to follow a path of chemical equilibrium. As will be seen later, combustion is modeled as an instantaneous conversion of a mass element between two states, one unburned and one burned.

The two cases, static or quasi-static distribution are a fundamentally different and the choice between the assumptions is a property of the thermodynamic system. A *Well Stirred Mixer* is a control volume filled with a gas where the species do not react with each other, i.e. a frozen mixture, while a *Well Stirred Reactor* is a control volume filled with a gas with quasi-static distribution of species. The assumptions can be interpreted as follows: for a frozen gas the reactions are much slower than the studied time frame and for an equilibrium gas the reactions happens so fast that they reach equilibrium instantaneously.

Implications of the Mixer/Reactor Distinction

The amount of substance can be specified in terms of the number of molecules, moles, or by mass. If the total number of molecules and the relative fractions of respective molecule in a gas are known we can calculate gas properties as a sum of properties of the components in a straight forward manner. For example, the internal energy of a gas can be written with *mole specific* properties as follows

$$U = n(p, T, \tilde{\mathbf{x}}_r) \sum_k \tilde{x}_k(p, T, \tilde{\mathbf{x}}_r) \tilde{u}_k(T) = n(p, T, \tilde{\mathbf{x}}_r) \tilde{\mathbf{x}}(p, T, \tilde{\mathbf{x}}_r)^T \tilde{\mathbf{u}}(T)$$

where $\tilde{\mathbf{x}}$ and \tilde{x}_i are mole fractions of respective molecule specie, $\tilde{\mathbf{x}}_r$ is the mole fractions of respective reactant atom, $\tilde{\mathbf{u}}$ and \tilde{u}_k are the mole specific energies, and n is the total number of moles. Using *mass specific* properties instead yields

$$U = m \sum_k x_k(p, T, \mathbf{x}_r) u_k(T) = m \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T)$$

where \mathbf{x} and x_i are mass fractions of respective molecule specie, \mathbf{x}_r is the mass fractions of respective reactant atom, \mathbf{u} and u_k are the mole specific energies, and m is the total mass. The notation is summarized in Table 2.2. The relation between the mass fractions and the mole fractions of the species is $x_k = \frac{\tilde{x}_k M_k}{M}$ where M_k is the molar masses of specie k while $M = \sum \tilde{x}_k M_k$ is the average molar mass.

In the above expressions the number of moles, $n(p, T, \tilde{\mathbf{x}}_r)$, the mole fractions, $\tilde{\mathbf{x}}(p, T, \tilde{\mathbf{x}}_r)$, and the mass fractions, $\mathbf{x}(p, T, \mathbf{x}_r)$, all depend on pressure, p , temperature, T , and available amount of atoms, $\tilde{\mathbf{x}}_r$ or \mathbf{x}_r . The difference between the reactor and mixer cases is how these functions depend on temperature, pressure, and available atoms. In the reactor case they are truly functions of pressure, temperature, and reactant atoms. In the mixer case they are instead constants and $n(p, T, \tilde{\mathbf{x}}_r)$, $\tilde{\mathbf{x}}(p, T, \tilde{\mathbf{x}}_r)$, and $\mathbf{x}(p, T, \mathbf{x}_r)$ are therefore

Table 2.2: The notation used to keep track of the amount of substances in a control volume and their properties.

Symbol	Property
\mathbf{x}	Vector of mass fractions of molecule species
\mathbf{x}_r	Vector of mass fractions of reactant atoms
x_k	Mass fraction of molecule specie k
\mathbf{u}	Vector of mass specific internal energy of species
u_k	Mass specific internal energy of molecule specie k
$\tilde{\mathbf{x}}$	Vector of mole fractions of molecule species
$\tilde{\mathbf{x}}_r$	Vector of mole fractions of reactant atoms
\tilde{x}_i	Mole fraction of molecule specie i
$\tilde{\mathbf{u}}$	Vector of mole specific internal energy of species
\tilde{u}_k	Mole specific internal energy of specie k

replaced with n , $\tilde{\mathbf{x}}$, and \mathbf{x} in the mixer case. As a consequence it is therefore in the **Reactor** necessary to keep track of the number of **atoms** of different species while for the **Mixer** it is necessary to keep track of the **molecules** of different species.

While the gas in a *Well Stirred Mixer* is indeed ideal if the components are ideal, the gas in a *Well Stirred Reactor* is not. It is however possible to use the ideal gas state equation as a foundation for a state equation for the reactor by regarding the number of molecules as a function of pressure, temperature, and composition instead of as being constant. The state equation for the *Reactor* is then expressed as follows when using *mole specific* properties

$$pV = n(p, T, \mathbf{x}_r) \tilde{R}T$$

and with *mass specific* properties

$$pV = m \sum_k \frac{x_k(p, T, \mathbf{x}_r) \tilde{R}}{M_k} T = m R(\mathbf{x}(p, T, \mathbf{x}_r)) T$$

Even if the state equation for a *Reactor* is very similar to the ideal gas law many properties of ideal gases do not apply and therefore a mixture of reacting ideal gases needs to be treated with care.

2.4 Open and Closed Systems with Non Inert Gases

As discussed in Section 2.2.1 a thermodynamic system in equilibrium can be completely described by two independent intensive properties together with the system size and its composition. Two properties that are easily understood and at the same time frequently available as measurements are temperature, T , and pressure, p . Therefore they are chosen as state variables in the following presentation. As system size the volume, V , is used. As mentioned in Section 2.3 another important factor when choosing temperature is that for an ideal gas the internal energy is only a function of temperature. This simplifies the

equations for the *Mixer* but it is also practical to only have one dimension when tabulating or making polynomials for internal energy or enthalpy of different molecule species. In fact, polynomials for enthalpy as a function of temperature are readily available, e.g. as in [20] and [47].

A consequence of deciding once and for all to use temperature and pressure as state variables is that it is not necessary to use subscripts of the thermodynamic differentials any more. For example $\frac{\partial u}{\partial T}$ always means *while pressure is constant* because $u = u(p, T)$ is a function of temperature and pressure only and because of the definition of partial derivative. Therefore, in the following text the following convention is used to save space

$$\frac{\partial u}{\partial T} = \left(\frac{\partial u}{\partial T} \right)_p \quad \text{and} \quad \frac{\partial u}{\partial p} = \left(\frac{\partial u}{\partial p} \right)_T$$

Together with the thermodynamic state variables we need to keep track of the contents of the control volume. Therefore the mass fractions of reactant atoms, \mathbf{x}_r , or the mass fractions of molecules, \mathbf{x} , are included in the state variables depending on the system type.

The choices of state variables made for the open system or control volume model are summarized in Figure 2.1. Note especially that the temperature and composition of the out flows are not indexed with the destination index. This is because the flowing gas has the same properties as the system for these flows.

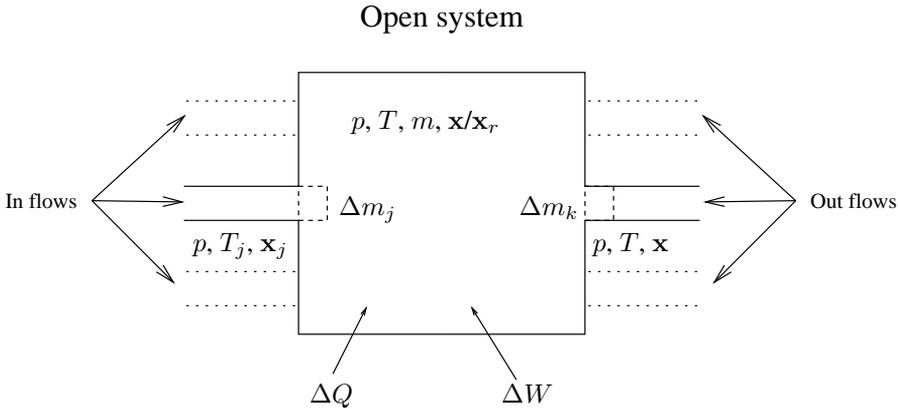


Figure 2.1: State variables and sign conventions for an open system. Pressure, p , and temperature, T , are chosen as state variables together with the gas composition, \mathbf{x} or \mathbf{x}_r , and system size m . During the time interval Δt the mass element Δm_j flows across the system boundary from source j and is thus **added** to the system while Δm_k flows across the boundary and is **removed** from the system. At the same time the heat ΔQ is **added** to the system and the work ΔW is done **on** the system.

To simulate an open or closed system with non inert gases we need equations for the derivatives of the state variables, i.e. we need equations for the temperature, pressure, and

gas composition derivatives as a function of known properties. Gas composition is easily handled using simple bookkeeping of molecules or atoms as is shown below. Using a state equation, as for example the ideal gas law, together with the first law of thermodynamics gives two equations that can be used to obtain expressions for the temperature and pressure differentials.

2.4.1 Keeping Track of the Gas Composition

As mentioned earlier different bookkeepings are needed for the reactor and mixer cases. In the reactor case the important quantity to keep track of is the reactant atoms. The governing equation for the gas composition in a *Reactor* is

$$d\mathbf{x}_r = \sum_j \frac{\hat{\mathbf{x}}_{r,j} - \mathbf{x}_r}{m} dm_j \quad (2.6)$$

where properties with index j represent the gas that is flowing across the system boundary. For a *Well Stirred Mixer* \mathbf{x}_r is replaced with \mathbf{x} . The gas composition only changes with **in-flow** to the system and it is important to choose the right properties of the flowing gas. We have that

$$\hat{\mathbf{x}}_{r,j} = \begin{cases} \mathbf{x}_{r,j} & \text{When flow is from the outside } (dm_j > 0) \\ \mathbf{x}_r & \text{When flow is from the inside } (dm_j \leq 0) \end{cases}$$

The bookkeeping equation in (2.6) will be used later used in Section 2.8 in the equations for the control volume model (2.20).

2.5 The Equation of State

In the quest for the expressions that describe the temperature and pressure derivatives the state equation can be used to obtain a relationship between the sought derivatives and known quantities. As mentioned in Section 2.3.1 the ideal gas law can be used as a foundation for a state equation for the *Well Stirred Reactor* and the *Well Stirred Mixer* cases. By letting the number of moles depend on pressure, temperature and gas composition we obtain

$$pV = n(p, T, \mathbf{x}_r) \tilde{R}T$$

which with $n = \sum_k n_k(p, T, \mathbf{x}_r) = \sum_k x_k(p, T, \mathbf{x}_r) \frac{m}{M_k}$ becomes

$$pV = m \sum_k \frac{x_k(p, T, \mathbf{x}_r) \tilde{R}}{M_k} T = m R(\mathbf{x}(p, T, \mathbf{x}_r)) T \quad (2.7)$$

To obtain an expression for the temperature and pressure differentials (2.7) is differentiated, giving

$$p dV + V dp = R(\mathbf{x}(p, T, \mathbf{x}_r)) T \sum_{j=1}^J dm_j + m R(\mathbf{x}(p, T, \mathbf{x}_r)) dT + m T dR \quad (2.8)$$

where J mass flows dm_j have been assumed.

In (2.8) all differentials except dR are either sought for or assumed to be available, i.e. dp and dT are sought for and dm_j and dV are assumed to be available. Because R is a function of $\mathbf{x}(p, T, \mathbf{x}_r)$, dR can also be expressed in terms of dp , dT , and dm_j or $d\mathbf{x}_r/dx$ but the expressions are different for the *Mixer* and *Reactor* cases and are therefore treated separately.

2.5.1 State Equation for the Well Stirred Reactor

In (2.8) dR has been left as it is with no information about how to calculate it. As mentioned earlier, R is a function of pressure, temperature, and composition, i.e. $R(\mathbf{x}(p, T, \mathbf{x}_r))$, and for the *reactor* case we have with the bookkeeping equation (2.6)

$$\begin{aligned} dR &= \frac{\partial R}{\partial p} dp + \frac{\partial R}{\partial T} dT + (\nabla_{\mathbf{x}_r} R)^T d\mathbf{x}_r = \\ &= \frac{\partial R}{\partial p} dp + \frac{\partial R}{\partial T} dT + (\nabla_{\mathbf{x}_r} R)^T \frac{1}{m} (\hat{\mathbf{x}}_{r,j} - \mathbf{x}_r) dm_j \end{aligned}$$

where

$$\hat{\mathbf{x}}_{r,j} = \begin{cases} \mathbf{x}_{r,j} & \text{When flow is from the outside } (dm_j > 0) \\ \mathbf{x}_r & \text{When flow is from the inside } (dm_j \leq 0) \end{cases}$$

Inserting this into (2.8) yields

$$\begin{aligned} p dV + V dp &= RT \sum_j dm_j + m R dT + \\ &+ T \left(m \frac{\partial R}{\partial p} dp + m \frac{\partial R}{\partial T} dT + \sum_j (\nabla_{\mathbf{x}_r} R)^T (\hat{\mathbf{x}}_{r,j} - \mathbf{x}_r) dm_j \right) \end{aligned} \quad (2.9)$$

where all properties are functions of the state only. This equation is later used in Section 2.8 for the *Reactor* part of the control volume model in (2.20).

2.5.2 State Equation for the Well Stirred Mixer

Since no reactions that form new molecules occur when the composition of the mix changes, the bookkeeping of reactant atoms, \mathbf{x}_r , is replaced with bookkeeping of molecules, \mathbf{x} for the *mixer* case. Therefore we have

$$p dV + V dp = RT \sum_j dm_j + m R dT + \sum_j (\nabla_{\mathbf{x}} R)^T (\hat{\mathbf{x}}_j - \mathbf{x}) T dm_j \quad (2.10)$$

where all properties are functions of the state only. This equation is later used in Section 2.8 for the *Mixer* part of the control volume model in (2.20).

2.6 Energy Preservation for Open Systems

As mentioned earlier the first law of thermodynamics (2.1) states that energy is conserved. An energy balance equation can therefore be used to get a relation between pressure and temperature differentials as a function of work and heat transfer as well as mass exchange. There are many ways of deducing the equations. The following presentation is, however, tailored to fit the rest of the framework and at the same time give insight to what the different contributions the final equations represent.

Using the sign convention in Figure 2.1 the differential form of the first law of thermodynamics is

$$dU = dQ + dW \quad (2.11)$$

where the change in internal energy, dU , can come from internal energy brought to or from the system by the flowing gas as well as changes in temperature and pressure while the work done on the system, dW , is both the work from volume change and from the work done by the flowing gas occupying space.

The idea is to deduce expressions for dU and dW and use these together with the first law of thermodynamics as formulated in (2.11) to get a relation between the pressure and temperature differentials. The heat transfer dQ is assumed to be a function of pressure and temperature and not their differentials and is therefore left as it is.

Internal Energy differential, dU

Because all thermodynamic variables can be regarded as functions of time, or for the engine case crank angle, we can see U as a function of a single variable and thus we have that

$$dU = \lim_{\Delta t \rightarrow 0} \left(\frac{\Delta U}{\Delta t} \right) dt \quad (2.12)$$

where ΔU represents the change in internal energy during a short time interval Δt . During that time interval the mass elements Δm_j have been added to or removed from the system. In the following equations only one positive mass flow from source j is assumed but the equations are, however, later on modified to account for multiple positive and negative mass flows. Negative flows are handled by using different compositions depending on the sign of the mass flow.

It is possible to directly use that

$$\begin{aligned} dU = & \lim_{h_p \rightarrow 0} \frac{U(p + h_p, T, m, \mathbf{x})}{h_p} dp + \lim_{h_T \rightarrow 0} \frac{U(p, T + h_T, m, \mathbf{x})}{h_T} dT + \\ & + \lim_{h_m \rightarrow 0} \frac{U(p, T, m + h_m, \mathbf{x})}{h_m} dm + \lim_{h_{\mathbf{x}} \rightarrow 0} \frac{U(p, T, m, \mathbf{x} + h_{\mathbf{x}})}{h_{\mathbf{x}}} d\mathbf{x} \end{aligned}$$

but introducing the time interval Δt makes it easy to keep track of what is really happening. It is assumed that the pressure is the same outside the system but that the temperature and gas compositions may differ as in Figure 2.1. The internal energy in the beginning

and at the end of the time interval Δt is described by the following equations for the *Well Stirred Reactor* case

$$\begin{aligned} U_b &= m \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \mathbf{u}(T_b) + \Delta m_j \mathbf{x}_j(p_b, T_j, \mathbf{x}_{r,j})^T \mathbf{u}(T_j) \\ U_e &= (m + \Delta m_j) \mathbf{x}(p_e, T_e, \mathbf{x}_{r,e})^T \mathbf{u}(T_e) \end{aligned} \quad (2.13)$$

and for the *Well Stirred Mixer* case

$$\begin{aligned} U_b &= m \mathbf{x}_b^T \mathbf{u}(T_b) + \Delta m_j \mathbf{x}_j^T \mathbf{u}(T_j) \\ U_e &= (m + \Delta m_j) \mathbf{x}_e^T \mathbf{u}(T_e) \end{aligned} \quad (2.14)$$

Here indices b and e represent the properties at the beginning and end of the time interval while the index j represents properties from source number j . Note that the only difference between the two cases is how \mathbf{x} depend on pressure, temperature, and composition.

Using $\Delta U = U_e - U_b$ expressions for dU can now be deduced. It will be shown in Section 2.8 that the two cases *Well Stirred Mixer* and *Well Stirred Reactor* can be described by the same set of equations. In the following sections they are, however, treated separately.

Work differential, dW

The gas that is flowing does work on the system and this work has to be added to any other work in the dW term. It is assumed that all other work that is done on the system is reversible work due to volume change so that we have

$$dW = -p dV + dW_j$$

where dW_j is the contribution from the j 'th flow. The amount of work, dW_j , that needs to be added can be calculated in a manner similar to the calculation of the internal energy differential, dU .

Throughout a time interval Δt the work $\int_t^{t+\Delta t} p dV_j$ is done on the system where dV_j can be thought of as the volume that is occupied by the flowing gas. dW_j can then be calculated by first integrating this work during the time interval Δt , as shown, and then calculating the limit as Δt tends to zero. As shown in [12] this gives

$$dW_j = p v_j dm_j$$

and therefore we have that

$$dW = -p dV + p v_j dm_j \quad (2.15)$$

2.6.1 Energy for the Well Stirred Reactor

U_e can be expressed as a function of the state before the mixing and the difference between the states using a Taylor series expansion at $[p_b, T_b, \mathbf{x}_b]$. For the *Well Stirred Reactor*-case

we therefore have that

$$\begin{aligned}
 U_e &= (m + \Delta m_j) \left(\mathbf{x}(p_b, T_b, \mathbf{x}_{r,b}) + \frac{\partial \mathbf{x}}{\partial p} \Delta p + \frac{\partial \mathbf{x}}{\partial T} \Delta T + \right. \\
 &\quad \left. + \nabla_{\mathbf{x}_r} \mathbf{x} \Delta \mathbf{x}_r + \mathcal{O}(\Delta^2) \right)^T \left(\mathbf{u}(T_b) + \frac{\partial \mathbf{u}}{\partial T} \Delta T + \mathcal{O}(\Delta^2) \right) = \\
 &= m \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \mathbf{u}(T_b) + m \left(\frac{\partial \mathbf{x}}{\partial p} \Delta p + \frac{\partial \mathbf{x}}{\partial T} \Delta T + \nabla_{\mathbf{x}_r} \mathbf{x} \Delta \mathbf{x}_r \right)^T \mathbf{u}(T_b) + \\
 &\quad + m \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \frac{\partial \mathbf{u}}{\partial T} \Delta T + \Delta m_j \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \mathbf{u}(T_b) + \mathcal{O}(\Delta^2)
 \end{aligned}$$

where $\mathcal{O}(\Delta^2)$ captures the contributions from all second order terms, i.e. $(\Delta p)^2$, $\Delta p \Delta T$ etc. $\Delta \mathbf{x}$ is the vector $[\dots \Delta x_k \dots]^T$ and $\nabla_{\mathbf{x}_r} \mathbf{x}$ is thus a matrix. Therefore $\nabla_{\mathbf{x}_r} \mathbf{x} \Delta \mathbf{x}_r$ is a vector.

From (2.13) we have that

$$U_b = m \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \mathbf{u}(T_b) + \Delta m_j \mathbf{x}_j(p_b, T_j, \mathbf{x}_{r,j})^T \mathbf{u}(T_j)$$

and therefore

$$\begin{aligned}
 \Delta U &= m \left(\frac{\partial \mathbf{x}}{\partial p} \Delta p + \frac{\partial \mathbf{x}}{\partial T} \Delta T + \nabla_{\mathbf{x}_r} \mathbf{x} \Delta \mathbf{x}_r \right)^T \mathbf{u}(T_b) + m \mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \frac{\partial \mathbf{u}}{\partial T} \Delta T + \\
 &\quad + \Delta m_j (\mathbf{x}(p_b, T_b, \mathbf{x}_{r,b})^T \mathbf{u}(T_b) - \mathbf{x}_j(p_b, T_j, \mathbf{x}_{r,j})^T \mathbf{u}(T_j)) + \mathcal{O}(\Delta^2)
 \end{aligned}$$

Inserting this into the definition of the differential dU in (2.12) combined with the first law (2.11) and the expression for the work in (2.15) yields

$$\begin{aligned}
 dQ - p dV + p v_j dm_j &= \\
 &= m \left(\frac{\partial \mathbf{x}}{\partial p} dp + \frac{\partial \mathbf{x}}{\partial T} dT + \nabla_{\mathbf{x}_r} \mathbf{x} d\mathbf{x}_r \right)^T \mathbf{u}(T) + m \mathbf{x}(p, T, \mathbf{x}_r)^T \frac{\partial \mathbf{u}}{\partial T} dT + \\
 &\quad + (\mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) - \mathbf{x}_j(p, T_j, \mathbf{x}_{r,j})^T \mathbf{u}(T_j)) dm_j
 \end{aligned}$$

It is possible to simplify this expression by noting that $u(p, T, \mathbf{x}_r) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T)$ which implies that $c_v = \frac{\partial u}{\partial T} = \left(\frac{\partial \mathbf{x}}{\partial T} \right)^T \mathbf{u} + \mathbf{x}^T \left(\frac{\partial \mathbf{u}}{\partial T} \right)$ and $\frac{\partial u}{\partial p} = \left(\frac{\partial \mathbf{x}}{\partial p} \right)^T \mathbf{u}$. It can also be noted that the composition differential $d\mathbf{x}_r$ can be written as $d\mathbf{x}_r = \frac{\mathbf{x}_{r,j} - \mathbf{x}_r}{m} dm_j$. Using this together with $h = u + p v$ gives the following energy preservation equation for the *Well Stirred Reactor*

$$\begin{aligned}
 dQ - p dV &= \tag{2.16} \\
 &= m \left(c_v dT + \frac{\partial u}{\partial p} dp \right) + \left((\nabla_{\mathbf{x}_r} u)^T (\mathbf{x}_{r,j} - \mathbf{x}_r) + u - h(p, T_j, \mathbf{x}_{r,j}) \right) dm_j
 \end{aligned}$$

where $u = u(p, T, \mathbf{x}_r) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T)$.

The energy preservation equation in (2.16) is later used in Section 2.8 for the *Reactor* part of the control volume model in (2.20).

2.6.2 Energy for the Well Stirred Mixer

As mentioned earlier the only difference between the *Well Stirred Mixer* and *Well Stirred Reactor* is the \mathbf{x} function. In the *Mixer* \mathbf{x} does not depend on pressure or temperature but only on bookkeeping of in-flowing molecules.

A Taylor series expansion of U_e in (2.14) yields

$$\begin{aligned} U_e &= (m + \Delta m_j)(\mathbf{x}_b + \Delta \mathbf{x})^T \left(\mathbf{u}(T_b) + \frac{\partial \mathbf{u}}{\partial T} \Delta T + \mathcal{O}(\Delta^2) \right) = \\ &= m \mathbf{x}_b^T \mathbf{u}(T_b) + m \mathbf{x}_b^T \frac{\partial \mathbf{u}}{\partial T} \Delta T + m \Delta \mathbf{x}^T \mathbf{u}(T_b) + \Delta m_j \mathbf{x}_b^T \mathbf{u}(T_b) + \mathcal{O}(\Delta^2) \end{aligned}$$

where as before $\mathcal{O}(\Delta^2)$ captures the contributions from all second order terms, i.e. $(\Delta p)^2$, $\Delta p \Delta T$ etc. and $\Delta \mathbf{x}$ is the vector $[\dots \Delta x_k \dots]^T$.

With U_b from (2.14) ΔU now becomes

$$\Delta U = m \mathbf{x}_b^T \frac{\partial \mathbf{u}}{\partial T} \Delta T + m \Delta \mathbf{x}^T \mathbf{u}(T_b) + \Delta m_j (\mathbf{x}_b^T \mathbf{u}(T_b) - \mathbf{x}_j^T \mathbf{u}(T_j)) + \mathcal{O}(\Delta^2)$$

Inserting ΔU in (2.12), once again combined with the first law in (2.11) and the expression for the work in (2.15) yields

$$dQ - p dV + p v_j dm_j = m \mathbf{x}^T \frac{\partial \mathbf{u}}{\partial T} dT + d\mathbf{x}^T \mathbf{u}(T) + (\mathbf{x}^T \mathbf{u}(T) - \mathbf{x}_j^T \mathbf{u}(T_j)) dm_j$$

As with the *Reactor*-case it is possible to simplify this expression further. For the *Mixer*-case $u(T, \mathbf{x}) = \mathbf{x}^T \mathbf{u}(T)$ and therefore we have that $c_v = \frac{\partial u}{\partial T} = \mathbf{x}^T \left(\frac{\partial \mathbf{u}}{\partial T} \right)$ and $\frac{\partial u}{\partial p} = \mathbf{x}^T \left(\frac{\partial \mathbf{u}}{\partial p} \right)$. The composition differential, $d\mathbf{x}$, is still calculated by simple bookkeeping but of molecules instead of atoms. Therefore $d\mathbf{x} = \frac{\mathbf{x}_j - \mathbf{x}}{m} dm_j$. Using these equations together with $h = u + p v$ the energy preservation equation for the *Mixer*-case becomes

$$dQ - p dV = m c_v dT + (\nabla_{\mathbf{x}} u(T, \mathbf{x}) (\mathbf{x}_j - \mathbf{x}) + u(T, \mathbf{x}) - h(T_j, \mathbf{x}_j)) dm_j \quad (2.17)$$

Because of the simple relation between u and \mathbf{x} for the *Mixer*-case, i.e. $u(T, \mathbf{x}) = \mathbf{x}^T \mathbf{u}(T)$, it is possible to simplify the equation further using $\nabla_{\mathbf{x}} u(\mathbf{x}_j - \mathbf{x}) = u(T, \mathbf{x}_j) - u(T, \mathbf{x})$ which then yields

$$dQ - p dV = m c_v dT + (u(T, \mathbf{x}_j) - h(T_j, \mathbf{x}_j)) dm_j \quad (2.18)$$

In the *Well Stirred Reactor*-case this simplification is not possible because the relation is $u = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T)$ where it is \mathbf{x}_r that is used as state variable. While it is tempting to use the simplification that leads to (2.18) it is better to keep the more general form because it enables us to use the same implementation for the two cases. A flexible implementation that allows for model updates without making old models invalid is desirable even if the cost is an increased model complexity because it enables us to reuse model components.

The choice to keep (2.17) for the *Mixer* instead of simplifying it to (2.18) enables us to make a separation between the system model and the gas model which in turn is a key to be able to reuse model components. It is thus better to keep the equation on the form in (2.17) to minimize the differences between the two cases.

As a side note it might be remarked that (2.18) gives some valuable insight. No reactions that change the composition occur and therefore the only real contributions from the flowing gas are the differences that come from the temperature difference and the pv_j -work part of $(u(T, \mathbf{x}_j) - h(T_j, \mathbf{x}_j)) dm_j$.

2.7 Thermochemical Properties

To be able to simulate the equations for the *Well Stirred Reactor*-case in (2.9) and (2.16) the following properties are needed

$$\left. \begin{array}{l} c_p \quad c_v \quad R \quad u/h \\ \frac{\partial R}{\partial p} \quad \frac{\partial R}{\partial T} \quad \frac{\partial u}{\partial p} \end{array} \right\} \quad \text{Scalar properties}$$

$$\left. \begin{array}{l} \nabla_{\mathbf{x}_r} u \quad \nabla_{\mathbf{x}_r} R \end{array} \right\} \quad \text{Vector properties}$$

In the *Well Stirred Mixer*-case in (2.10) and (2.17), \mathbf{x}_r is replaced with \mathbf{x} and some properties are not necessary. More specific, $\frac{\partial \mathbf{x}}{\partial T} = \frac{\partial \mathbf{x}}{\partial p} = 0$ for a *Mixer*-gas. This simplifies the equations for the gas properties and for example we have that $\frac{\partial R}{\partial p} = \frac{\partial R}{\partial T} = \frac{\partial u}{\partial p} = 0$. Therefore only the following properties are needed

$$\left. \begin{array}{l} c_p \quad c_v \quad R \quad u/h \end{array} \right\} \quad \text{Scalar properties}$$

$$\left. \begin{array}{l} \nabla_{\mathbf{x}} u \quad \nabla_{\mathbf{x}} R \end{array} \right\} \quad \text{Vector properties}$$

The collections of properties are complete regardless if u or h is specified. How to obtain these properties is the subject of Chapter 3. For now, however, it is stated that the quality of these quantities have a profound effect on the quality of the simulation.

2.8 Collecting the Equations and Introducing Combustion

When the equations for the *Well Stirred Mixer* and the *Well Stirred Reactor*-cases are studied it is evident that it is enough to implement the *Well Stirred Reactor*-case. This is due to the fact that the gas properties for a gas composed of molecules that are not allowed to react takes a form where the difference between the two cases disappears. It is thus possible to hide the information about which case a system belongs to in the gas properties. By including the gas properties that are not needed, and that are all zero, in (2.10) and (2.17) the difference between the cases disappears.

When the difference between \mathbf{x}_r and \mathbf{x} is hidden in the calculation of gas properties, then \mathbf{x}_r can be replaced with \mathbf{x} in the equations for the *Well Stirred Reactor*-case which can then be used in the general case.

Up until now only one flow **into** the system has been considered for the energy equation in (2.16). To allow for many flows **in** and **out** of the system T_j and \mathbf{x}_j can be replaced with \hat{T}_j and $\hat{\mathbf{x}}_j$ together with a sum over the flows in the following manner

$$\begin{aligned} dQ - p dV &= \tag{2.19} \\ &= m \left(c_v dT + \frac{\partial u}{\partial p} dp \right) + \sum_j \left((\nabla_{\mathbf{x}} u)^T (\hat{\mathbf{x}}_j - \mathbf{x}) + u - h(p, \hat{T}_j, \hat{\mathbf{x}}_j) \right) dm_j \\ d\mathbf{x} &= \sum_j \frac{\hat{\mathbf{x}}_j - \mathbf{x}}{m} dm_j \end{aligned}$$

where

$$\begin{aligned} \hat{\mathbf{x}}_j &= \begin{cases} \mathbf{x}_j & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\ \hat{T}_j &= \begin{cases} T_j & \text{When flow is from outside } (dm_j > 0) \\ T & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\ u &= u(p, T, \mathbf{x}) \end{aligned}$$

Note that in (2.19) \mathbf{x} can represent both \mathbf{x} from the *Mixer* or \mathbf{x}_r from the *Reactor*. The distinction between the cases are then left to the gas property calculation. This separation between the thermodynamic state equation and the calculation of gas properties will be further explored in Chapter 3.

2.8.1 Combustion

An interesting, and essential, detail is that it is possible to introduce combustion in the framework by allowing the use of two different gas descriptions for a gas that is flowing from a source to a destination. When for example a gas is flowing from a *Well Stirred Mixer* to a *Well Stirred Reactor* \mathbf{x} is different in the different systems. In the first case it's the molecules, \mathbf{x} , that is being tracked while in the latter it's the reactant atoms, \mathbf{x}_r . The \mathbf{x} has thus to be translated from one representation to another on the way to it's destination.

The introduction of the \mathbf{x} to \mathbf{x}_r translation into (2.19) has, however, to be done carefully. If the translated \mathbf{x} is used in all parts of (2.19) the energy from the combustion of the mass element would be lost and no energy would, according to the equation, be released in the process. This is because the energy that represents combustion is accounted for by the $u(p, T, \mathbf{x}) - h(p, \hat{T}_j, \hat{\mathbf{x}}_j)$ part of (2.19) and because the same energy would be removed from one system and added to the other.

In the simplest case, when the temperatures are the same at the source and the destination, i.e. $T_j = T$, and the translated \mathbf{x}_j of the source is equal to \mathbf{x} , i.e. the composition at the source and destination is not changed by the mass transfer, the combustion energy is

not included in (2.19) because $u(p, T_j, g(\mathbf{x}_j)) = u(p, T, \mathbf{x})$. Here $g(\mathbf{x}_j)$ represents the translated \mathbf{x}_j and thus we have that $g(\mathbf{x}_j) = \mathbf{x}$.

However, if the composition from the source, \mathbf{x}_j , is used to calculate the energy that the gas brings to it's destination, i.e. the difference in (2.19) is $u(p, T, \mathbf{x}) - h(p, T_j, \mathbf{x}_j)$, this accounts for the energy that is released by combustion of the transported mass element. The situation is depicted in Figure 2.2 where new indices for $\hat{\mathbf{x}}$ have been introduced.

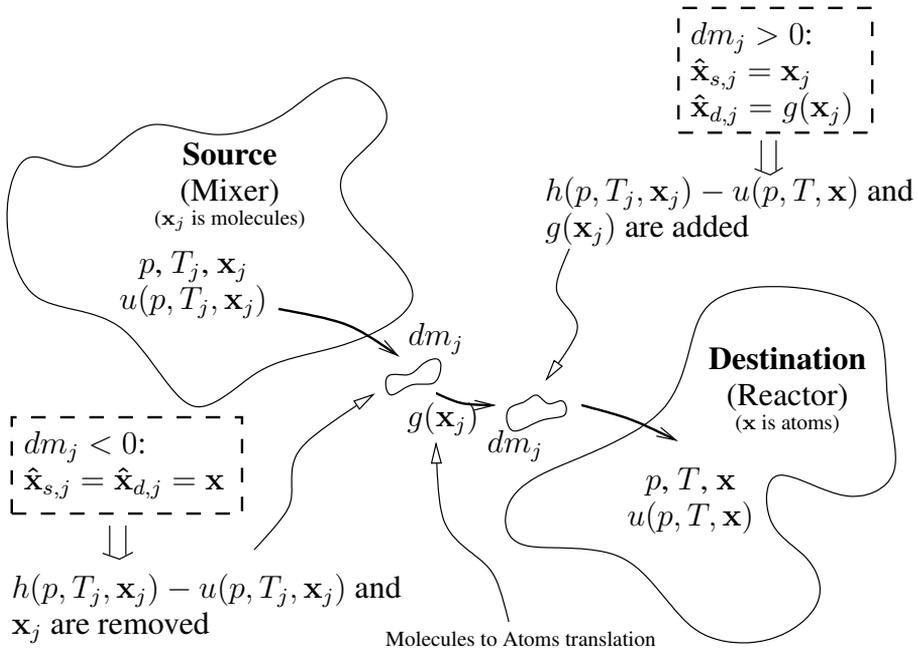


Figure 2.2: When a mass element dm_j is transported from the source system to the destination system, $dm_j x_j$ and $dm_j (h(p, T_j, \mathbf{x}_j) - u(p, T_j, \mathbf{x}_j))$ are removed from the source system. On it's way to the destination the combustion changes the composition of dm_j according to the conditions of the destination system but the energy it brings to the system is the same. The energy that is released by a combustion process is $(u(p, T_j, \mathbf{x}_j) - u(p, T, \mathbf{x})) dm_j$ because only $(h_j - u_j) dm_j = p v_j dm_j$ is removed from the source, while $(h(p, T_j, \mathbf{x}_j) - u(p, T, \mathbf{x})) dm_j$ is added to the destination. Note that it is assumed that the property functions knows from the context if the \mathbf{x} :s represent a Reactor or Mixer. This will be further discussed in Chapter 3.

2.8.2 The Control Volume Model

Thus, by introducing new indices, d and s , that represent destination and source to track which \mathbf{x} that should be used in the calculations it is possible to account for combustion. The equations for a general thermodynamic control volume that can model combustion can then be written in the following form

$$\begin{aligned}
 m c_v dT + m \frac{\partial u}{\partial p} dp + p dV &= dQ + \sum_j \left((\nabla_{\mathbf{x}} u)^T (\mathbf{x} - \hat{\mathbf{x}}_{d,j}) + h(p, \hat{T}_{s,j}, \hat{\mathbf{x}}_{s,j}) - u \right) dm_j \\
 p dV + \left(V - m T \frac{\partial R}{\partial p} \right) dp - m \left(R + T \frac{\partial R}{\partial T} \right) dT &= \sum_j \left(R + (\nabla_{\mathbf{x}} R)^T (\hat{\mathbf{x}}_{d,j} - \mathbf{x}) \right) T dm_j \\
 d\mathbf{x} &= \sum_j \frac{\hat{\mathbf{x}}_{d,j} - \mathbf{x}}{m} dm_j \tag{2.20}
 \end{aligned}$$

where

$$\begin{aligned}
 \hat{\mathbf{x}}_{d,j} &= \begin{cases} g(\mathbf{x}_j) & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 \hat{\mathbf{x}}_{s,j} &= \begin{cases} \mathbf{x}_j & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 \hat{T}_{s,j} &= \begin{cases} T_j & \text{When flow is from outside } (dm_j > 0) \\ T & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 u &= u(p, T, \mathbf{x}) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) \quad \text{or} \quad \mathbf{x}^T \mathbf{u}(T)
 \end{aligned}$$

Some Important Details

Note that (2.20) has been written in a form where all unknown quantities, i.e. dp , dV , and dT , are on the left hand side while all known quantities are on the right hand side. However, dV isn't unknown when a single control volume is considered but since the equations are to be used in a multi-zone model the volume change, dV , for a zone will be unknown. dQ and dm do not depend on dp , dT , or dV but only on p , V , T , and time and are therefore considered as known.

When using computer simulations to solve the differential equations that are obtained by differentiating other equations, as done with the state equation and the energy equation when deriving (2.20), it is possible that the solution diverges so that the original equations are not fulfilled. The simulation tolerances can normally be tightened to minimize this drift and it is interesting to see how this affects the model (2.20). In other words this means that it should be validated that the total mass, volume, and energy are preserved when simulating (2.20) and this is one of the subjects of Section 4.5.

2.9 A First Example and Validation

The essential part of this chapter is summarized in (2.20) and in Chapter 4, a multi-zone model will be constructed using (2.20) as foundation. It is interesting, however, to put the model to the test to see how it performs in a simple case. As a test system a calorimeter is used to calculate the heating values of some commonly used fuels. The heating value, q_{HV} , of a fuel refers to the energy released during complete combustion of a fuel. The combustion can take place at constant volume or at constant pressure and a formal definition is [25]

$$\begin{aligned} q_{HVp} &= -(\Delta h)_{p, T_0} \\ q_{HVV} &= -(\Delta u)_{V, T_0} \end{aligned} \quad (2.21)$$

where T_0 is usually at $298.15K$. If all the water from the reaction is allowed to condensate after the reaction the term higher heating value q_{HHV} is used, while if all the water is gaseous the term lower heating value q_{LHV} is used.

q_{HVV} can be measured using a bomb calorimeter which is a closed, constant volume, device for deciding the heating value of fuels. q_{HVp} on the other hand can be measured using a constant pressure calorimeter which is a device where the pressure instead of the volume is constant. The difference in constant volume or constant pressure heating values is small [25]. A rough sketch of idealized versions of the two devices is presented in Figure 2.3 below.

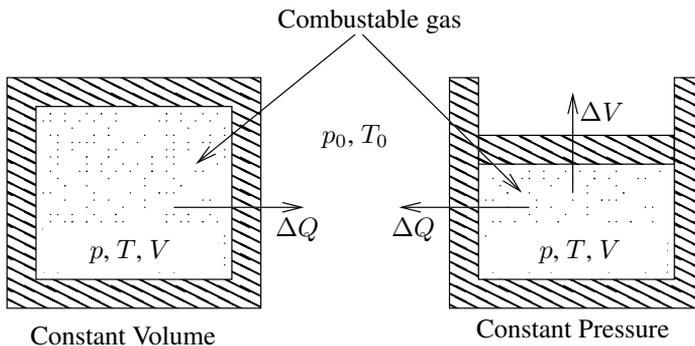


Figure 2.3: The constant volume heating value is measured using a closed, constant volume, vessel while the constant pressure heating value is measured using a constant pressure vessel. In both cases a gas is combusted and the amount of energy that is released is measured.

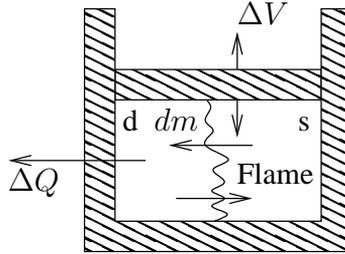
A Qualitative Experiment with a Constant Pressure Calorie-meter

We can think of a constant pressure calorimeter as two systems in the same confinement both having the same pressure as the outside atmosphere and sharing the volume amongst them. The first system is a mixer filled with unburned species while the second system is a reactor. The idea is to determine the heat that is generated when mass is moved from

the mixer to the reactor. Index s and d are used to denote source and destination system and dm is assumed to be positive when gas is flowing from the source to the destination. In Example 2.1 it is shown how a model of a constant pressure calorimeter can be set up.

Example 2.1

A model of a constant pressure calorimeter can be set up using (2.20) and defining two regions with the same pressure but with different temperatures and compositions. The source system is a mixer with unburnt fuel while the destination is a reactor.



For this simple case the bookkeeping part of (2.20) is unnecessary because the fuel/air-ratio is constant. In other words the change of composition at the destination, $d\mathbf{x}_d = 0$ because $\hat{\mathbf{x}}_{d,j} = g(\hat{\mathbf{x}}_j) = \mathbf{x}$, i.e. the first element that enters the burned zone/region has the same composition as all the others, and for the source $d\mathbf{x}_s = 0$ trivially. This will be discussed in greater detail in Chapters 3 and 4. Because the model system is a constant pressure calorimeter we have that $dp = 0$ and if we assume that the heat transfer is large enough we can also assume that $dT \approx 0$. These assumptions thus yield

$$\begin{aligned} p_0 dV_d &= dQ_d + (h(p_0, T_0, \mathbf{x}_s) - u(p_0, T_0, \mathbf{x}_d)) dm \\ p_0 dV_s &= dQ_s - (h(p_0, T_0, \mathbf{x}_s) - u(p_0, T_0, \mathbf{x}_s)) dm \\ p_0 dV_d &= R_d T_0 dm \\ p_0 dV_s &= -R_s T_0 dm \\ V_d + V_s &= V \\ d\mathbf{x}_s &= 0 \\ d\mathbf{x}_d &= 0 \end{aligned}$$

We have for this idealized case that $dp = dT = 0$ and since there is no change in composition, pressure, or temperature in the destination during the mass transfer we have that

$$p_0 dV_d = R_d T_0 dm$$

which when inserted in the equations for the destination yields

$$\begin{aligned} R_d T_0 dm &= dQ_d + (h(p_0, T_0, \mathbf{x}_s) - u(p_0, T_0, \mathbf{x}_d)) dm \implies \\ dQ_d &= -(h(p_0, T_0, \mathbf{x}_s) - h(p_0, T_0, \mathbf{x}_d)) dm \end{aligned}$$

The interpretation is that the amount of energy dQ that needs to be removed to preserve status quo in a vessel is $h(p_0, T_0, \mathbf{x}_s) - h(p_0, T_0, \mathbf{x}_d)$.

The conclusion is that looking at the definition of q_{HV_p} in (2.21) there are obvious similarities.

Simulation of Bomb Calorie-meter

While it is a crude assumption that the heat transfer is so fast that $dT \approx 0$ we can instead assume that all the energy is released so fast that the heat transfer during the combustion process is negligible or we can simply set up a heat transfer model as

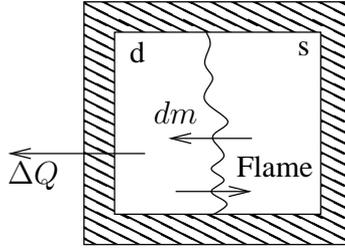
$$dQ = A h_{ht} (T_d - T_0) \quad (2.22)$$

where A is the area of the vessel, h_{ht} is the heat transfer coefficient and $T_d - T_0$ is the difference in temperature between the vessel and the surroundings.

A simulation model of (2.20) can be implemented for a bomb calorimeter in a similar manner as with the constant pressure calorimeter in Example 2.1 and with the heat transfer model (2.22).

Example 2.2

A model of a bomb calorimeter can be set up using (2.20) and once again defining two regions with the same pressure but with different temperatures and compositions. The source system is, as before, a mixer with unburnt fuel while the destination is a reactor.



As in Example 2.1 the bookkeeping part of (2.20) is unnecessary because the fuel/air-ratio is constant.

$$m_d c_{v,d} dT_d + m_d \frac{\partial u_d}{\partial p} dp + p dV_d = dQ_d + (h(p, T_s, \mathbf{x}_s) - u(p, T_d, \mathbf{x}_d)) dm$$

$$m_s c_{v,s} dT_s + m_s \frac{\partial u_s}{\partial p} dp + p dV_s = dQ_s - (h(p, T_s, \mathbf{x}_s) - u(p, T_s, \mathbf{x}_s)) dm$$

$$p dV_d + \left(V_d - m_d T_d \frac{\partial R_d}{\partial p} \right) dp - m_d \left(R_d + T_d \frac{\partial R_d}{\partial T} \right) dT_d = R_d T_d dm$$

$$p dV_s + \left(V_s - m_s T_s \frac{\partial R_s}{\partial p} \right) dp - m_s \left(R_s + T_s \frac{\partial R_s}{\partial T} \right) dT_s = R_s T_s dm$$

$$dV_d + dV_s = dV = 0$$

$$d\mathbf{x}_s = 0$$

$$d\mathbf{x}_d = 0$$

Two reaction paths are modeled, one with adiabatic combustion and one where the heat release rate, $\frac{dm}{dt}$, as well as the heat transfer coefficient, h_{ht} , and the area of the vessel, A ,

have been coarsely tuned to each other so that half the energy has been lost as heat at the time when the combustion is finished.

The setup of Example 2.2 has been simulated using gas properties from a chemical equilibrium program and the results are shown in Figure 2.4 and Table 2.3. Chemical equilibrium and models of gas properties are discussed in Chapter 3 and for these simulations a gas model with 10 molecule species has been used for the reactor while polynomials for air and fuel have been used for the mixer.

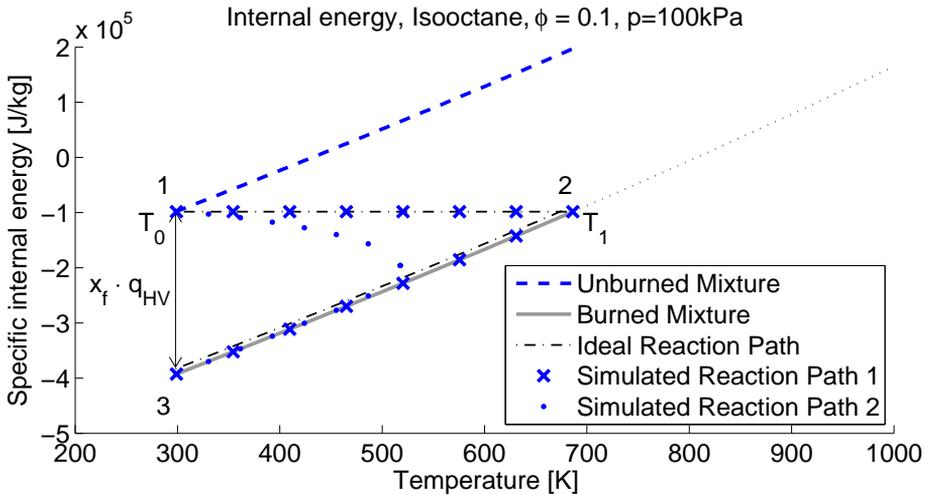


Figure 2.4: The lower heating value at constant pressure, q_{LHV_V} , of a fuel is the amount of energy that is released during the combustion. At first all species are unburnt at the temperature T_0 , i.e. at point 1, then they are burnt either adiabatically with $dQ = 0$ or starts to cool at once. The adiabatic process gives the mixture the end temperature T_1 , at point 2. The mixture is then cooled until it reaches the temperature T_0 again, 3, and the amount of energy that is released is recorded. There is no notable difference in the end result between the process with the adiabatic combustion and the one with heat transfer all the time. Finally q_{LHV_V} is then the difference in energy per unit mass of fuel in the mixture. Note that the ideal reaction path, dash dotted, is slightly adjusted to increase visibility.

As can be seen in Figure 2.4 the estimates of q_{LHV_V} do not depend on whether the heat transfer occurs during the combustion or not for this model. The q_{LHV_V} estimates are close to the tabulated reference data but there are differences. These differences come from the quality of the input data to the chemical equilibrium program, in forms of thermochemical tables, and the assumptions that are made during the setup. For this test case the most important result is that there are no differences between the two simulated reaction paths and the ideal reaction path.

Table 2.3: Calculated q_{LHV} for different fuels compared to tabulated values from [25]. The fuel denoted *Diesel* has two values, one for heavy and one for light diesel.

Fuel	Simulated Value	Tabulated Value
Isooctane	44.7 [MJ/kg]	44.3 [MJ/kg]
Methanol	21.1 [MJ/kg]	20.0 [MJ/kg]
Ethanol	27.8 [MJ/kg]	26.9 [MJ/kg]
Diesel	42.7 [MJ/kg]	41.4/42.5 [MJ/kg]

2.10 Concluding Remarks about the The Control Volume Model

In this chapter a simulation model for a control volume is deduced. An investigation is made of the mixer and the reactor cases and it is shown that the same model equations can be used for both cases if proper care is taken when the gas property model is implemented. It is also shown that the model can be used to model combustion in a straight forward manner.

A short summary of the highlights of the chapter is

- In (2.20) a control volume model capable of handling combustion is summarized.
- The model uses the same equations regardless of system type.
- The choice to keep (2.17) for the *Mixer* instead of simplifying it to (2.18) is made because it enables us to make a distinction between the system model and the gas model which in turn is a key to be able to reuse model components because it gives a clean cut between thermodynamic equations and gas properties.
- A bomb calorie experiment is used as an example which serves as a first validation and shows proof of concept of the model in (2.20).

The model which is summarized in (2.20) is a differential equation for the temperature and pressure in the control volume. It is derived by differentiating the thermodynamic state equation and the energy equation. Because (2.20) lacks an analytic solution it must be solved using numerical methods and this can introduce a drift in the solution. Therefore it should be validated that the total mass, volume, and energy are preserved when simulating (2.20) and this is the subject of Section 4.5.

3

Gas Models for Simulation of Thermodynamic Systems

When simulating thermodynamic systems the description of the gas has a central role. The choice of gas model affects both the simulation time as well as the accuracy of the simulation. Different applications have different demands on the gas model. For example it may sometimes be necessary to have a fraction of residuals together with the fresh mixture and in some cases this is unnecessary.

In Section 2.6.2 the choice to keep (2.17) for the *Mixer* instead of simplifying it to (2.18) was made because it enables us to make a distinction between the system model and the gas model which in turn is a key to be able to reuse model components. With this construction everything that has to do with the internal representation of the gas can be hidden from the user. The goal is now to develop a framework that encapsulates all possible cases, from gases that are modeled in a molecular level to gases that are modeled as fresh air with some residuals. The concept is illustrated in Figure 3.1.

In Section 2.7 a number of gas properties that are needed when simulating a control volume were listed. It was then stated that in the *Well Stirred Reactor* it is necessary to keep track of atoms while for the *Well Stirred Mixer* it is the molecules that need bookkeeping. However, as long as the needed properties are calculated correctly it doesn't matter what representation that is used for a gas. Moreover, we may sometimes choose to introduce simplifications that greatly reduce the model complexity, and hence primarily simulation speed, and in these cases sacrifice accuracy for speed.

For a *Well Stirred Reactor* a gas can for example be characterised by its relative air and fuel content without losing any information as long as there is only one fuel. For a *Well Stirred Mixer* however, this type of simplification can introduce an error when the gases that enter the mixer come from mixtures that have been frozen at different conditions. The implications of such a simplification have to be assessed from case to case.

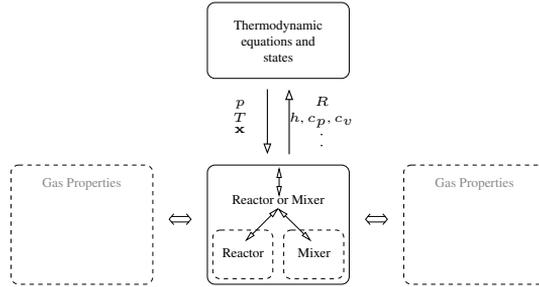


Figure 3.1: Separating the state equation for the control volume from the equations for the gas enables us to change gas properties without reimplementing the control volume model. Note that the composition part of the thermodynamic state is \mathbf{x} regardless of if it represents a Reactor or Mixer and that this enables us to use the same equations for the control volume regardless of if it is a Reactor or Mixer.

For reference the necessary quantities for the control volume model are repeated here. To be able to simulate (2.20) the following thermodynamic properties are needed

$$\left. \begin{array}{l} c_p \quad c_v \quad R \quad u/h \\ \frac{\partial R}{\partial p} \quad \frac{\partial R}{\partial T} \quad \frac{\partial u}{\partial p} \end{array} \right\} \text{Scalar properties}$$

$$\left. \begin{array}{l} \nabla_{\mathbf{x}} u \quad \nabla_{\mathbf{x}} R \end{array} \right\} \text{Vector properties}$$

where for the *Well Stirred Mixer*-case in (2.10) and (2.17), $\frac{\partial R}{\partial p} = \frac{\partial R}{\partial T} = \frac{\partial u}{\partial p} = 0$. The collection of properties is complete regardless if u or h is specified.

3.1 Outline of the Chapter

In Section 3.2, **Characterizing Ideal and non Inert Gases**, the concept of composition parameters is introduced. The composition parameters may for example represent the residual mass fraction x_{RG} , the fraction of air in the mixture x_{Air} , or the amount of different molecules $\{x_{\text{CO}_2}, x_{\text{H}_2\text{O}}, x_{\text{O}_2}, x_{\text{N}_2}, x_{\text{H}_2}\}$. It is shown that the model in (2.20) can be used together with the composition parameters regardless of if they represent molecules, atoms, collections of molecules, or collections of atoms. As examples three different gas representations are given, the molecule level gas, the two component gas, and the four component gas.

In Section 3.3, **Calculation of Gas Properties using a Chemical Equilibrium Program**, the concept of chemical equilibrium is discussed and expressions for the gas properties that are needed in (2.20) are given.

In Section 3.4, **Implementation Choices and Simple Gas Models**, a model that interpolates tabular data and a simplified equilibrium model are introduced and compared to the molecule level gas. The properties of the gas models are then evaluated using the example model-setup in Appendix C.

3.2 Characterizing Ideal and non Inert Gases

The properties of a gas depend on pressure, p , temperature, T , and the amount of molecules of each specie, x_k . In the thermodynamic analysis, that yields equation (2.20), only two assumptions are made about how these properties depend on the composition of the gas. The assumptions are that the internal energy and the state equation can be written as

$$U = m \sum_{k=1}^K x_k(p, T, \mathbf{x}_r) u_k(T)$$

$$pV = m \sum_{k=1}^K \frac{x_k(p, T, \mathbf{x}_r) \tilde{R}}{M_k} T = m R(\mathbf{x}(p, T, \mathbf{x}_r)) T$$

Here \mathbf{x}_r is the mass fractions of the reactant atoms, \mathbf{x} and x_k are the mass fractions of the molecule species, and M_k are the molar masses of the species. As long as these assumptions are valid it is possible to replace the atoms or molecules with a collection of more general composition parameters, $\{x_k\}$. The composition parameters can for example represent the residual mass fraction x_{RG} , the fraction of air in the mixture x_{Air} , or as before the amount of atoms or molecules, $\{x_H, x_C, x_N, x_O\}$ or $\{x_{CO_2}, x_{H_2O}, x_{O_2}, x_{N_2}, x_{H_2}, \dots\}$ respectively. This means that a gas in this definition can be built of K components where each represents only one molecule, but the only demand on a gas model is that the fraction of components always sums up to one, i.e. $\sum_{k=1}^K x_k = 1$.

Each x_k can thus represent different things such as air and fuel fractions or mass fractions of individual molecules or atoms. The control volume model only knows the fractions of composition properties while the information whether they internally represent actual molecule fractions or lumped components is hidden in thermodynamic properties together with the information whether the control volume is a *Mixer* or *Reactor*. The concept is illustrated in Figure 3.2.

Just as when keeping track of atoms or molecules, the collection of fractions of composition properties $\{x_k\}$ only change because of in-flow to the control volume. Keeping track of the composition properties is equivalent to the simple bookkeeping equation used in (2.6). Therefore, using the vector notation \mathbf{x} to represent composition properties gives

$$d\mathbf{x} = \sum_j \frac{\mathbf{x}_j - \mathbf{x}}{m} dm_j \quad (3.1)$$

where j represents the composition of **in** flow number j .

3.2.1 Examples of Gas Descriptions

The use of composition properties instead of molecules or atoms yields the possibility to define a number of different gas property models. A few examples of possible scenarios and the definition of the gas models are listed below. A more detailed study of different assumptions will be presented later in this chapter.

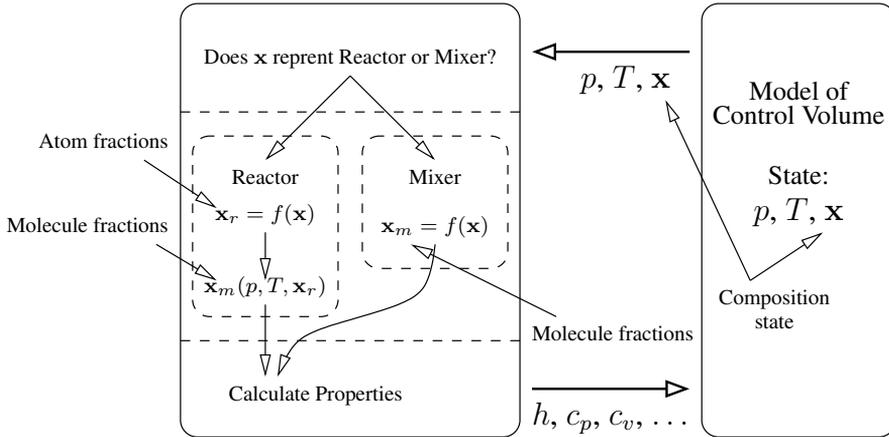


Figure 3.2: Overview of the gas property model. Composition parameters are used in the state of the control volume instead of atoms or molecules. The gas property model then translates the composition parameters into atom or molecule fractions depending on if \mathbf{x} represents a reactor or mixer. In the Reactor the atom representation is then used to calculate molecule fractions depending on pressure, temperature, and available atoms before calculating the gas properties while in the Mixer the molecule fractions are passed directly to the property calculation.

Gas at Atom/Molecule Level

A natural way of defining a gas in the framework is probably to keep track of all **molecules** that can exist in the mixture for the mixer case and all **atoms** that can exist in the mixture for the reactor case. However, *all molecules that can exist* are quite a few which makes a selection in order of importance necessary. This affects both the reactor and mixer cases because even if there in general are only a very limited number of atoms available in the reactor case one has to remember that the gas properties are calculated by first calculating the existing molecules, i.e. we have $\mathbf{x}(p, T, \mathbf{x}_r)$ and for example $u(p, T, \mathbf{x}_r) = \mathbf{x}^T(p, T, \mathbf{x}_r)\mathbf{u}(T)$ which implies that the fractions of respective molecule specie, $\mathbf{x}(p, T, \mathbf{x}_r)$, have to be calculated as a middle step.

Using a chemical equilibrium program it is possible to calculate which molecule species that contribute to the gas properties of a gas mixture. It is a common choice, e.g. as in [13] and [25], to assume that a combustible mixture contains what is regularly defined as air and hydrocarbons or alcohols. This implies that the reactant atoms consist of carbon C, hydrogen H, oxygen O, and nitrogen N. Therefore we generally have

$$\mathbf{x} = (x_C \quad x_H \quad x_O \quad x_N)^T$$

for a atom level reactor gas. An example of molecules that can be accounted for in a mixer, but also as the intermediate step in the reactor, is water H_2O , carbon dioxide CO_2 , nitrogen gas N_2 , oxygen gas O_2 , carbon oxide CO , nitrogen oxide NO , hydrogen oxide

OH, hydrogen H, hydrogen gas H_2 , oxygen O and the actual fuel that is used. This choice will be ratified later on, but for this choice we have

$$\mathbf{x} = \left(x_{uf} \quad x_{H_2O} \quad x_{CO_2} \quad x_{N_2} \quad x_{O_2} \quad x_{CO} \quad x_{NO} \quad x_{OH} \quad x_H \quad x_{H_2} \quad x_O \right)^T \quad (3.2)$$

where the amount of unburned fuel x_{uf} has been listed first. It is, however, easy to change the molecules or species that are accounted for to fit a specific purpose without affecting the structure of the model.

A Two Component Gas, (fixed ϕ , one fuel)

A spark ignited engine with a catalytic converter is normally run on a stoichiometric fuel/air mixture, i.e. $\phi = 1$. For such a case it might not be necessary to account for different mixture compositions. In the well stirred mixer, which is the natural choice when modeling an unburned mixture, there are however often some burned gases present with the fresh mixture and therefore we need at least two components in the gas if only one fuel is considered. One component that represents an unburned mixture and one component that represents a burned mixture.

Here the x_k 's represent the relative portions of two different gas mixtures. For a well stirred reactor there are, by definition, no unburned species because we always assume equilibrium and the need for a composition property therefore disappears. A sufficient state vector that covers both cases is then

$$\mathbf{x} = \left(x_u \quad x_b \right)^T$$

For this type of gas model it is necessary to decide what is meant by a burned mixture. In Figure 3.3 a chemical equilibrium program is used to calculate the molar fractions for a range of species at different temperatures and fuel/air ratios. It is evident that for lower temperatures, i.e. below 1700K, the only species that reach above 1% are nitrogen gas, water, carbon dioxide, carbon monoxide, hydrogen gas, and oxygen gas. Depending on the rate of cooling and the reaction rate a gas might freeze to different compositions for different processes. In [25] it is stated that the most accurate approach is to use an equilibrium program above 1700K and assume a frozen mixture below. It is then natural to postulate that a cooled down burned mixture has the same composition as it would at 1700K.

Postulating that a cooled down burned mixture has the same composition as it would at 1700K is, however, not always the best alternative. Firstly because there are still reactions that take place and secondly because the thermodynamic model can become inconsistent, i.e. for example

$$\lim_{\epsilon \rightarrow 0} c_p(p, T_{\text{frozen}} + \epsilon, \mathbf{x}) \neq \lim_{\epsilon \rightarrow 0} c_p(p, T_{\text{frozen}} - \epsilon, \mathbf{x})$$

One reason for this is that it is necessary to choose a pressure at which the mixture has frozen since the enthalpy also depends on pressure. Using a lower temperature limit can reduce this type of inconsistency because the pressure dependence is smaller at lower temperatures and T_{frozen} is thus a design parameter. The implications of this type of inconsistency is discussed in Section 4.5.

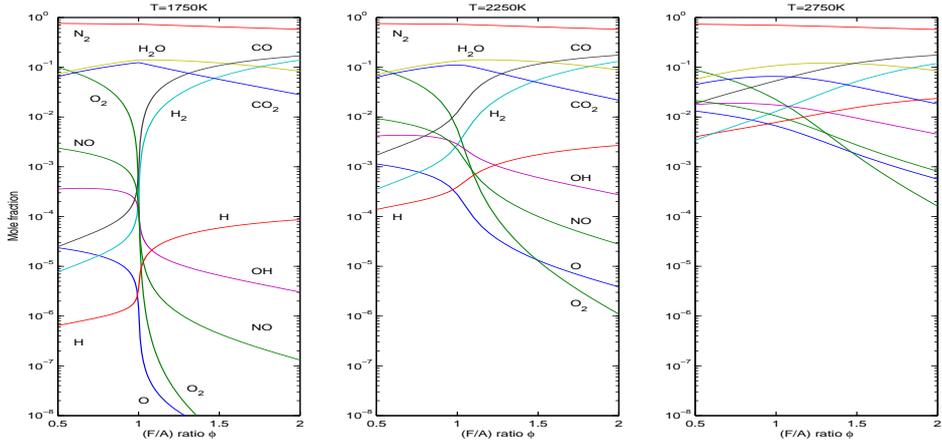


Figure 3.3: Concentrations of different species as a function of equivalence ratio for different temperatures. Equivalent of Figure 3.10 in [25], generated using CHEPP [13] with standard selection of species and isoctane as fuel. None of the species that are available to the equilibrium program but are not part of the standard selection shown in the figure reach above 10^{-5} in concentration. Therefore it is in most cases not necessary to include more species than those that are available in the standard selection.

A Four Component Gas, (varying ϕ , one fuel)

For an SI engine it is sometimes enough to use a gas with two components, i.e. differentiating between burned and unburned gases, but for modeling of compression ignited engines it is often necessary to include the fuel/air ratio. Diesel combustion is often modeled as fuel transport from a liquid zone via an unburned mixing zone and finally to a burned zone, e.g. as in [42]. The fuel/air ratio of the mixing zone changes during the combustion and the fuel air ratio thus needs to be included in the model. It is common practice to use the fuel/air equivalence ratio, $\phi = \frac{m_f (A/F)_s}{m_a}$, where (A/F) is stoichiometric fuel/air ratio, as describing parameter. A drawback of using ϕ as a state variable for the composition is however that ϕ tends to infinity as the amount of air in the mixture tends to zero, which for example is the case in the fuel spray. Therefore the fractions of fuel and air in the mixture, $x_f = \frac{m_f}{m_{\text{Tot}}}$ and $x_a = \frac{m_a}{m_{\text{Tot}}}$, are used instead. Because there are both burned and unburned gases that can have different compositions it is practical to have four components in the gas description if only one fuel is considered

$$\mathbf{x} = (x_{uf} \quad x_{ua} \quad x_{bf} \quad x_{ba})^T$$

where the indices u, b, f and a represents *unburned*, *burned*, *fuel*, and *air* respectively.

The four component gas is introduced as a first step towards simpler models. The benefits are that the small state vector opens up for simplifications in the gas model and that the number of calculations during the simulation is smaller. If, however, this modeling approach is used then one has to watch out for two cases. The first is that it is assumed that

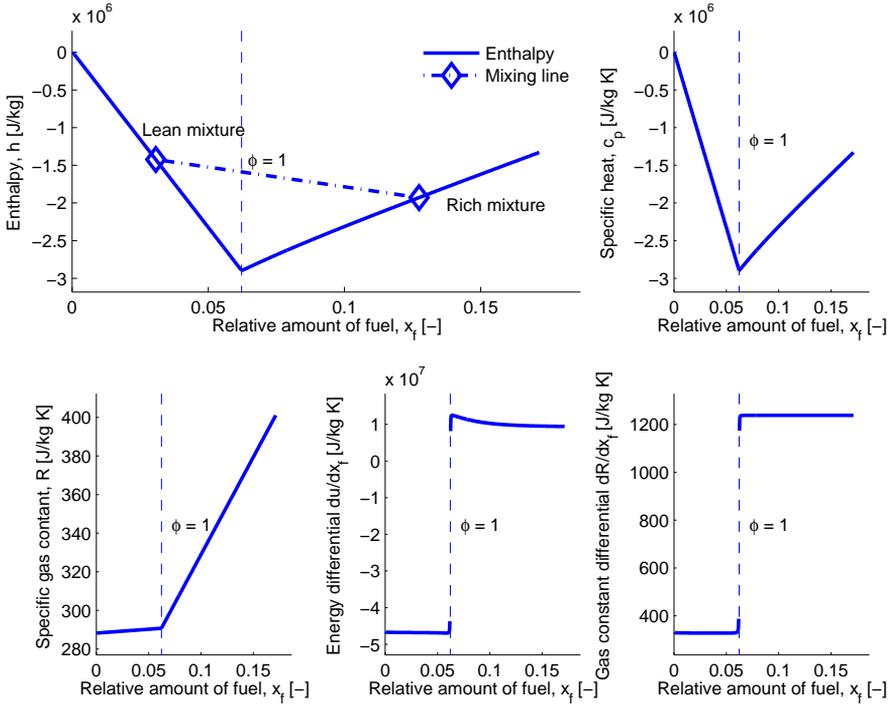


Figure 3.4: Gas properties of a burned mixture of isooctane and air at $T = 300$ K, $p = 100$ kPa. It is evident that the behaviour changes rapidly around $\phi = 1$ but outside a small region the properties are nearly linear in fraction of fuel.

all molecules in the mixer originates from combustion at the current fuel/air ratio. The second is that it is assumed that the thermodynamic properties $u(p, T, \mathbf{x})$ and $R(p, T, \mathbf{x})$ are linear in the burned air and fuel fractions, x_{ba} and x_{bf} , i.e. for the enthalpy we have that $h(p, T, x_{bf}, x_{ba}) = h_{bf}(p, T) x_{bf} + h_{ba}(p, T) x_{ba}$.

Rich burned mixtures or lean burned mixtures are quite linear in the fraction of fuel, x_{bf} . Close to $\phi = 1$ things changes rapidly though. In Example 3.1 the implications that this has when mixing a rich burned mixture and a lean burned mixture is illustrated.

Example 3.1

In Figure 3.4 the enthalpy, specific heat, specific gas constant, derivatives of energy and gas constant w.r.t. the amount of fuel are shown. In the top left plot two mixtures, one lean and one rich are illustrated. The difference between the dashed line and the solid line with markers in the top left figure represents the amount of energy that would be released if the lean burned and the rich burned mixtures in the figure were to be mixed.

The interpretation is that if the indicated lean burned and the rich burned mixtures were to be mixed in proportions that make the fuel/air equivalence ratio equal to one for the

resulting mixture, i.e. $\phi = 1$, then about 1.4 [MJ/kg] extra energy would be released. For the specific fuel this represents about 3% of the heating value, q_{HV} , for the fuel in the figure.

As the example shows, mixing a lean burned mixture with a rich burned mixture would release a substantial amount of energy if a model is set up in a way that would allow this to happen. If handled correctly the problem may never arise but the implications of this modeling approach has to be assessed from case to case.

3.3 Calculation of Gas Properties using a Chemical Equilibrium Program

There are many publications that deal with calculation of chemical equilibrium, most notable the NASA equilibrium program [20], but also [40] and [46]. In this presentation, however, a program package called CHEPP, c.f. [13], is used. Although all of the methods can calculate the necessary properties this presentation refers to the implementation used in CHEPP. The package can be used to calculate properties for burned and unburned fuel air mixtures for a number of fuels. The properties are calculated by first calculating the chemical equilibrium of a mixture and then calculating the thermochemical properties. Thermochemical property data is obtained from the Gas Research Institute, the so called *Gri-Mech* tables [47], for common combustion species as well as fuel data in polynomial form from [25].

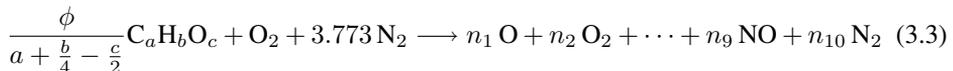
In the calculation of chemical equilibrium CHEPP assumes that a combustible mixture contains what is commonly defined as air and hydrocarbons or alcohols. The chemical equilibrium is calculated by setting up an atom conservation constraint using the reaction coefficients on the form

$$A\mathbf{n} = \mathbf{b}$$

and determining the \mathbf{n} that minimizes Gibbs free energy under the constraint. Here \mathbf{n} is a vector containing the number of moles of each combustion product specified per mole oxygen that is available in the air while \mathbf{b} is a vector of available atoms, also specified per each mole of oxygen that is available in the air. Example 3.2 shows a common situation.

Example 3.2

If the species are chosen as in Equation (3.2) and we have the model reaction



the \mathbf{b} vector is

$$\mathbf{b} = \begin{pmatrix} a \frac{\phi}{a + \frac{b}{4} - \frac{c}{2}} \\ b \frac{\phi}{a + \frac{b}{4} - \frac{c}{2}} \\ 2 + c \frac{\phi}{a + \frac{b}{4} - \frac{c}{2}} \\ 2 \cdot 3.773 \end{pmatrix}$$

while the A matrix takes the following form

	H ₂ O	CO ₂	N ₂	O ₂	CO	NO	OH	H	H ₂	O
C	0	1	0	0	1	0	0	0	0	0
H	2	0	0	0	0	0	1	1	2	0
O	1	2	0	2	1	1	1	0	0	1
N	0	0	2	0	0	1	0	0	0	0

Each field in the matrix specifies the number of atoms that makes up a certain molecule amongst the chosen species.

In Table 2.2 the notation that is used to keep track of the amount of substance is given. To conform with the notation in CHEPP the normalized vector of mole fractions, \mathbf{b} , is added to these. The following notation is thus used in this chapter

- $\tilde{\mathbf{x}}(p, T, \tilde{\mathbf{x}}_r(\mathbf{b}))$ Vector of molecule specie mole fractions.
- $\tilde{\mathbf{x}}_r$ Vector of atom fractions.
- \mathbf{x}_r Vector of mass specific atom fractions.
- \mathbf{b} Normalized vector of atom fractions.

and using this notation we have that

$$\mathbf{b} = \frac{2 \cdot 3.773}{\tilde{x}_N} \tilde{\mathbf{x}}_r \quad \text{and} \quad \tilde{\mathbf{x}}_r = \frac{\sum_i \mathbf{e}_i \frac{x_i}{M_i}}{\sum_j \frac{x_j}{M_j}}$$

where M_i is the mole weight of respective atom, \tilde{x}_N is fraction of nitrogen atoms while \mathbf{e}_i is unit vector i .

3.3.1 Interfacing with CHEPP

CHEPP has a number of functions that makes it possible to obtain the following properties as a function of pressure, temperature, and available atoms in the mixture

- $\tilde{\mathbf{x}}(p, T, \mathbf{b})$ Vector of specie mole fractions.
- $\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial T}$ Vector with temperature differentials of the mole fractions.
- $\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial p}$ Vector with pressure differentials of the mole fractions.
- $\nabla_{\mathbf{b}} \tilde{\mathbf{x}}(p, T, \mathbf{b})^T$ Matrix with differentials of the mole fractions with respect to the available atoms.
- $\tilde{\mathbf{c}}_p(T)$ Vector with mole specific heats for constant pressure
- $\tilde{\mathbf{h}}(T)$ Vector with mole specific enthalpies
- \mathbf{M} Vector with mole weights of species

Using these quantities it is possible to calculate mole specific properties of the gas as

$$\begin{aligned}
 \tilde{h}(p, T, \mathbf{b}) &= \tilde{\mathbf{x}}(p, T, \mathbf{b})^T \tilde{\mathbf{h}}(T) \\
 \tilde{u}(p, T, \mathbf{b}) &= \tilde{h}(p, T, \mathbf{b}) - \tilde{R}T \\
 \tilde{c}_p(p, T, \mathbf{b}) &= \frac{\partial \tilde{h}(p, T, \mathbf{b})}{\partial T} = \tilde{\mathbf{x}}(p, T, \mathbf{b})^T \tilde{\mathbf{c}}_p(T) + \left(\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial T} \right)^T \tilde{\mathbf{h}}(T) \\
 \tilde{c}_v(p, T, \mathbf{b}) &= \frac{\partial \tilde{u}(p, T, \mathbf{b})}{\partial T} = \frac{\partial \tilde{h}(p, T, \mathbf{b})}{\partial T} - \tilde{R} = \tilde{c}_p(p, T, \mathbf{b}) - \tilde{R} \\
 \frac{\partial \tilde{h}(p, T, \mathbf{b})}{\partial p} &= \left(\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial p} \right)^T \tilde{\mathbf{h}}(T) \\
 \nabla_{\mathbf{b}} \tilde{u}(p, T, \mathbf{b}) &= \nabla_{\mathbf{b}} \tilde{h}(p, T, \mathbf{b}) = (\nabla_{\mathbf{b}} \tilde{\mathbf{x}}(p, T, \mathbf{b})^T) \tilde{\mathbf{h}}(T) \\
 M(p, T, \mathbf{b}) &= \tilde{\mathbf{x}}(p, T, \mathbf{b})^T \mathbf{M} \\
 \frac{\partial M(p, T, \mathbf{b})}{\partial T} &= \frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})^T \mathbf{M}}{\partial T} = \left(\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial T} \right)^T \mathbf{M} \\
 \frac{\partial M(p, T, \mathbf{b})}{\partial p} &= \frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})^T \mathbf{M}}{\partial p} = \left(\frac{\partial \tilde{\mathbf{x}}(p, T, \mathbf{b})}{\partial p} \right)^T \mathbf{M} \\
 \nabla_{\mathbf{b}} M(p, T, \mathbf{b}) &= (\nabla_{\mathbf{b}} \tilde{\mathbf{x}}(p, T, \mathbf{b})^T) \mathbf{M}
 \end{aligned}$$

These properties are not available to the control volume model but they are used when calculating the mass specific properties as will be shown in the two following sections.

3.3.2 Calculation of Mass Specific Thermodynamic Properties for the Well Stirred Reactor

We may recall that the following gas properties were needed by the simulation model: c_p , c_v , R , h , $\frac{\partial R}{\partial p}$, $\frac{\partial R}{\partial T}$, $\frac{\partial u}{\partial p}$, $\nabla_{\mathbf{x}} u$ and $\nabla_{\mathbf{x}} R$. Remembering that \mathbf{x} is equivalent to \mathbf{x}_r for the *Well Stirred Reactor* while using the definition of these quantities and writing them on a CHEPP compatible form the following equations are obtained

$$\begin{aligned}
 h(p, T, \mathbf{x}_r) &= \frac{\tilde{h}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} \\
 u(p, T, \mathbf{x}_r) &= \frac{\tilde{u}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} \\
 R(p, T, \mathbf{x}_r) &= \frac{\tilde{R}}{M(p, T, \mathbf{b})} \\
 \frac{\partial R(p, T, \mathbf{x}_r)}{\partial T} &= \frac{\partial}{\partial T} \frac{\tilde{R}}{M(p, T, \mathbf{x}_r)} = -\frac{R(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} \frac{\partial M(p, T, \mathbf{b})}{\partial T} \\
 \frac{\partial R(p, T, \mathbf{x}_r)}{\partial p} &= \frac{\partial}{\partial p} \frac{\tilde{R}}{M(p, T, \mathbf{x}_r)} = -\frac{R(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} \frac{\partial M(p, T, \mathbf{b})}{\partial p} \\
 c_p(p, T, \mathbf{x}_r) &= \frac{\partial h(p, T, \mathbf{x}_r)}{\partial T} = \frac{\partial}{\partial T} \frac{\tilde{h}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} = \frac{\tilde{c}_p(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} - \frac{\tilde{h}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})^2} \frac{\partial M(p, T, \mathbf{b})}{\partial T} \\
 c_v(p, T, \mathbf{x}_r) &= \frac{\partial u(p, T, \mathbf{x}_r)}{\partial T} = c_p(p, T, \mathbf{x}_r) - R(p, T, \mathbf{x}_r) - \frac{R(p, T, \mathbf{x}_r) T}{M(p, T, \mathbf{b})} \frac{\partial M(p, T, \mathbf{b})}{\partial T} \\
 \frac{\partial u(p, T, \mathbf{x}_r)}{\partial p} &= \frac{\partial}{\partial p} \frac{\tilde{h}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} - \frac{\partial R(p, T, \mathbf{x}_r)}{\partial p} T = \frac{\frac{\partial \tilde{h}(p, T, \mathbf{b})}{\partial p} - u(p, T, \mathbf{x}_r) \frac{\partial M(p, T, \mathbf{b})}{\partial p}}{M(p, T, \mathbf{b})} \\
 \nabla_{\mathbf{x}_r} u(p, T, \mathbf{x}_r) &= (\nabla_{\mathbf{x}_r} \mathbf{b}^T) \left(\frac{\nabla_{\mathbf{b}} \tilde{u}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})} - \frac{\tilde{u}(p, T, \mathbf{b})}{M(p, T, \mathbf{b})^2} \nabla_{\mathbf{b}} M(p, T, \mathbf{b}) \right) \\
 \nabla_{\mathbf{x}_r} R(p, T, \mathbf{x}_r) &= (\nabla_{\mathbf{x}_r} \mathbf{b}^T) \nabla_{\mathbf{b}} \frac{\tilde{R}}{M(p, T, \mathbf{b})} = -(\nabla_{\mathbf{x}_r} \mathbf{b}^T) \frac{\tilde{R}}{M(p, T, \mathbf{b})^2} \nabla_{\mathbf{b}} M(p, T, \mathbf{b})
 \end{aligned}$$

These equations can be used directly when calculating gas properties for a *Reactor* or as a middle step when designing simpler gas models.

Calculation of $\nabla_{\mathbf{x}_r} \mathbf{b}^T$

When calculating $\nabla_{\mathbf{x}_r} \mathbf{b}^T$ in the above expressions a little trick can be used to simplify the calculations. The relationship between \mathbf{x}_r and \mathbf{b} is

$$\mathbf{b} = \frac{2 \cdot 3.773}{\tilde{x}_N} \tilde{\mathbf{x}}_r \quad \text{and} \quad \tilde{\mathbf{x}}_r = \frac{\sum_i \mathbf{e}_i \frac{x_{r,i}}{M_i}}{\sum_j \frac{x_{r,j}}{M_j}}$$

where M_i is the mole weight of respective atom, \tilde{x}_N is the fraction of nitrogen atoms and \mathbf{e}_i is unit vector i . Therefore we have that

$$\nabla_{\mathbf{x}_r} \mathbf{b}^T = 2 \cdot 3.773 \frac{M_N}{x_N} \begin{pmatrix} \frac{1}{M_1} & 0 & 0 & 0 \\ 0 & \frac{1}{M_3} & 0 & 0 \\ 0 & 0 & \frac{1}{M_3} & 0 \\ \frac{-1}{x_N} \frac{M_1}{x_{r,1}} & \frac{-1}{x_N} \frac{M_2}{x_{r,2}} & \frac{-1}{x_N} \frac{M_3}{x_{r,3}} & 0 \end{pmatrix}^T \quad (3.4)$$

assuming that x_N is the fourth element in the \mathbf{x}_r vector.

However, $\nabla_{\mathbf{x}_r} \mathbf{b}^T$ is only used to calculate $\nabla_{\mathbf{x}_r} u(p, T, \mathbf{x}_r)$ and $\nabla_{\mathbf{x}_r} R(p, T, \mathbf{x}_r)$ above. Therefore it is possible to use

$$\widehat{\nabla_{\mathbf{x}_r} \mathbf{b}^T} = 2 \cdot 3.773 \frac{M_N}{x_N} \begin{pmatrix} \frac{1}{M_1} & 0 & 0 & 0 \\ 0 & \frac{1}{M_3} & 0 & 0 \\ 0 & 0 & \frac{1}{M_3} & 0 \\ 0 & 0 & 0 & \frac{1}{M_4} \end{pmatrix}^T \quad (3.5)$$

instead of (3.4) with exactly the same results. The reason is that $\nabla_{\mathbf{x}_r} u(p, T, \mathbf{x}_r)$ and $\nabla_{\mathbf{x}_r} R(p, T, \mathbf{x}_r)$ implicitly contains $\nabla_{\mathbf{x}_r} \tilde{\mathbf{x}}^T$ which can be written as

$$\nabla_{\mathbf{x}_r} \tilde{\mathbf{x}}^T = \nabla_{\mathbf{b}} \tilde{\mathbf{x}}^T \nabla_{\mathbf{x}_r} \mathbf{b}^T = \nabla_{\mathbf{b}} \tilde{\mathbf{x}}^T \widehat{\nabla_{\mathbf{x}_r} \mathbf{b}^T}$$

This is because $\sum_i \tilde{x}_i = \sum_i x_i = \sum_i \tilde{x}_{r,i} = 1$ which implies that the value of $\nabla_{\mathbf{x}_r} \tilde{\mathbf{x}}^T$ does not change when $\nabla_{\mathbf{x}_r} \mathbf{b}^T$ from (3.4) is replaced with $\widehat{\nabla_{\mathbf{x}_r} \mathbf{b}^T}$ from (3.5).

3.3.3 Calculation of Mass Specific Thermodynamic Properties for the Well Stirred Mixer

In the *Mixer* case things are significantly simplified because $\tilde{\mathbf{x}}$ does neither depend on pressure, temperature, nor reactant atoms. As for the *Reactor* case we need to calculate c_p , c_v , R , h , $\frac{\partial R}{\partial p}$, $\frac{\partial R}{\partial T}$, $\frac{\partial u}{\partial p}$, $\nabla_{\mathbf{x}} u$, and $\nabla_{\mathbf{x}} R$. Using that $\tilde{\mathbf{x}}$ is only a function of \mathbf{x} we have

$$\begin{aligned}
 h(T, \mathbf{x}) &= \frac{\tilde{h}(T, \tilde{\mathbf{x}})}{M(\tilde{\mathbf{x}})} = \sum_k \frac{\tilde{x}_k M_k \tilde{h}_k(T)}{M(\tilde{\mathbf{x}}) M_k} = \sum_k x_k \frac{\tilde{h}_k(T)}{M_k} \\
 u(T, \mathbf{x}) &= \frac{\tilde{u}(T, \tilde{\mathbf{x}})}{M(\tilde{\mathbf{x}})} = \sum_k x_k \frac{\tilde{u}_k(T)}{M_k} = \sum_k x_k \frac{\tilde{h}_k(T) - \tilde{R} T}{M_k} \\
 R(\mathbf{x}) &= \frac{\tilde{R}}{M(\tilde{\mathbf{x}})} = \sum_k x_k \frac{\tilde{R}}{M_k} \\
 \frac{\partial R(\mathbf{x})}{\partial T} &= \mathbf{0} \\
 \frac{\partial R(\mathbf{x})}{\partial p} &= \mathbf{0} \\
 c_p(T, \mathbf{x}) &= \frac{\partial h(T, \mathbf{x})}{\partial T} = \frac{\partial \frac{\tilde{h}(T, \tilde{\mathbf{x}})}{M(\tilde{\mathbf{x}})}}{\partial T} = \frac{\tilde{c}_p(T, \tilde{\mathbf{x}})}{M(\tilde{\mathbf{x}})} \\
 c_v(T, \mathbf{x}) &= \frac{\partial u(T, \mathbf{x})}{\partial T} = c_p(T, \mathbf{x}) - R(\mathbf{x}) \\
 \frac{\partial u(T, \mathbf{x})}{\partial p} &= \mathbf{0} \\
 \nabla_{\mathbf{x}} u(T, \mathbf{x}) &= \sum_k \mathbf{e}_k \frac{\tilde{u}_k - \tilde{R} T}{M_k} \\
 \nabla_{\mathbf{x}} R(\mathbf{x}) &= \sum_k \mathbf{e}_k \frac{\tilde{R}}{M_k}
 \end{aligned}$$

The equations above can be used directly when calculating gas properties for a *Mixer* or as a middle step when designing simpler gas models.

3.4 Implementation Choices and Simple Gas Models

In Section 3.2.1 a number of different ways to characterize, or model, a gas were discussed. Three examples of choices of composition states were given and in Section 3.3 it was discussed how to calculate gas properties using a chemical equilibrium code.

It is of course possible to use a chemical equilibrium code to calculate the gas properties for all three example models. However, using a chemical equilibrium code to calculate thermochemical properties during a simulation is time consuming and having a small state vector, as for the two or four component gases, does not help that much since the molecule

fractions have to be calculated as a middle step. Even if the equilibrium code were to be optimized for the particular application a significant part of the simulation time is spent calculating gas properties.

The reason that the two and four component models are interesting is because they serve as a first step to further simplifications. If we choose to model a gas as a two or four component mixture of burned and unburned gases as in Section 3.2.1 the state vector is kept small. In this way the dimensions of the gas property functions are essentially reduced to two or three. For the mixer-case the thermochemical properties, like e.g. u , h , R , c_p , are functions of fuel/air ratio and temperature only, but for the reactor-case they also depend on pressure. This implies that simple gas models only have to be 2 or 3-dimensional.

As noted in Section 2.8 the same control volume model can be used regardless if it represents a *Reactor* or *Mixer* because this is taken care of by the gas description. In many cases we can even use the same state vector for mixers and reactors if we impose certain restrictions so that there is no confusion about which category a gas belongs to. It is easy to use a temperature limit, usually about 1700K, to differentiate between a reactor and a mixer. Another restriction is that a reactor may not contain any unburned species so for the four component example, on page 40, only two components would be used for a reactor. In Example 3.3 an example setup of a four component model is depicted.

Example 3.3

Property	$T \leq T_{\text{frozen}} \Rightarrow \text{Mixer}$ Interval	$T > T_{\text{frozen}} \Rightarrow \text{Reactor}$ Interval
x_{uf}	[0–1]	–
x_{ua}	[0–1]	–
x_{bf}	[0–1]	[0–1]
x_{ba}	[0–1]	[0–1]

Example of an implementation with a four component gas. A cold system is assumed to represent a mixer while a hot system is assumed to represent a reactor. This example uses the composition parameters as depicted in Figure 3.2 and translates the components into molecule fractions.

For $T > T_{\text{frozen}}$ we have that

$$\begin{pmatrix} x_{\text{H}} \\ x_{\text{C}} \\ x_{\text{N}} \\ x_{\text{O}} \end{pmatrix} = f \begin{pmatrix} 0 \\ 0 \\ x_{\text{bf}} \\ x_{\text{ba}} \end{pmatrix} \quad \text{and} \quad \mathbf{x}_m = g \left(p, T, \begin{pmatrix} x_{\text{H}} \\ x_{\text{C}} \\ x_{\text{N}} \\ x_{\text{O}} \end{pmatrix} \right)$$

where $\mathbf{x}_m = (x_{\text{H}_2\text{O}} \ x_{\text{CO}_2} \ x_{\text{N}_2} \ x_{\text{O}_2} \ x_{\text{CO}} \ x_{\text{NO}} \ x_{\text{OH}} \ x_{\text{H}} \ x_{\text{H}_2} \ x_{\text{O}})^T$ and g represents calculation of chemical equilibrium. The internal representation of the molecule fractions, \mathbf{x}_m , can then be used to calculate the thermochemical properties.

3.4.1 Full Equilibrium Gas Model

As stated earlier a natural way of defining a gas in the framework is to keep track of all molecules that can exist in the mixture for the mixer case and all atoms that can exist in the mixture for the reactor case. For this model two different state vectors are used for the *Reactor* and *Mixer* cases

$$\mathbf{x} = \begin{cases} (x_{uf} & x_{H_2O} & \cdots & x_{NO} & x_{OH} & x_H & x_{H_2} & x_O)^T & \text{Mixer} \\ (x_H & x_C & x_N & x_O)^T & \text{Reactor} \end{cases}$$

where the species of molecules in the mixers composition vector are allowed to change. The molecules that are accounted for in the mixer is also used as intermediate step for the reactor when calculating the gas properties. The full equilibrium model will later be used as reference for comparing with the other, simplified, gas models.

3.4.2 Four Component Gas using Chemical Equilibrium

As a first step towards simpler models the four component composition vector was introduced. For the gas model that is introduced here, and that use the four component composition vector, the chemical equilibrium is calculated in each simulation step by first translating the composition parameters to fuel/air ratio and then to molecules as shown in Example 3.3. As mentioned in Section 3.2.1 the treatment of burned gases for the mixer can suffer from problems with mixing of gases that are burned at different fuel/air ratios as well as different temperatures and it is interesting to have a model that can be used to capture this effect only. The reactor however does not suffer from the same problem because the amount of available atoms are fully described using only the fuel/air ratio.

Even if the large speed up would be to remove the call to the equilibrium code for each simulation step this four component gas implementation is an important middle step because it gives the user a possibility to evaluate the implications of choosing a composition vector with four components. A four component composition vector will be used in the models in Section 3.4.3 and Section 3.4.4 but there the call to the equilibrium code is replaced by simplified models. Using the four component composition vector while keeping the call to the equilibrium code it is possible to evaluate the implications of choosing a composition vector with four components without having other sources of error.

The composition vector for the gas is, as in Section 3.2.1,

$$\mathbf{x} = (x_{uf} \quad x_{ua} \quad x_{bf} \quad x_{ba})^T$$

where the indices u, b, f and a represent *unburned*, *burned*, *fuel* and *air* respectively. The temperature limit may be changed as needed to fit particular purposes but as stated earlier 1700K is a good trade off because the gas can be assumed to be frozen below 1700K. In Example 3.4 the full equilibrium model of Section 3.4.1 is compared with the four component model to validate that the use of four components in the composition vector does not influence simulation results for a SI engine model.

Example 3.4

The SI engine model setup that is described in Appendix C has been simulated using both the full equilibrium and the four component equilibrium gas models. In Figure 3.5 the pressure and temperature traces as well as the relative differences between the pressure and temperatures for the two models are shown. The simulations use a relative tolerance of 10^{-10} and as can be seen in the figure the differences in pressure and burned mixture temperature between the two simulations is of the same order of magnitude as the relative tolerance.

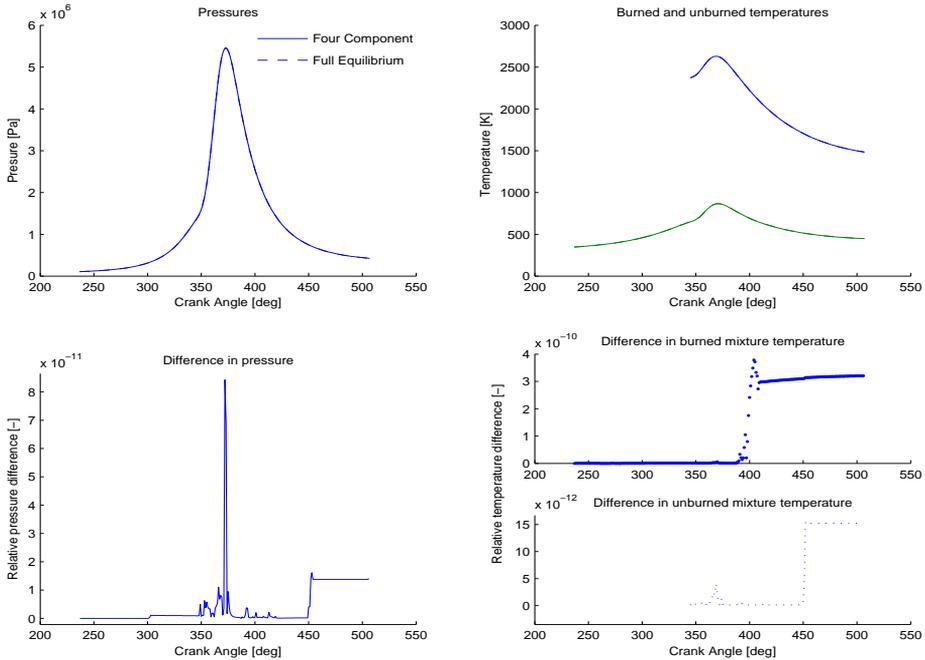


Figure 3.5: Results from simulation of one cylinder in a spark ignited engine using the DAE formulation in Chapter 4 and the four component gas as well as the full equilibrium gas. As can be seen there is no significant difference between the two simulations. The model setup that is used is described in Appendix C with the exception that the temperature of the burned zone is adjusted so that there is no steep gradient at start of combustion.

3.4.3 Tables

The amount of data, and hence memory, that is needed to tabulate a function grows exponentially with the dimensions of the function. It is therefore feasible to tabulate data only if the number of dimensions is low enough. As mentioned earlier a four component gas model can be used for both the *Reactor* and the *Mixer* as long as two burned mixtures that are combusted at rich and lean conditions are not mixed. The composition vector for the four component gas is

$$\mathbf{x} = (x_{uf} \quad x_{ua} \quad x_{bf} \quad x_{ba})^T$$

and as in Example 3.3 these properties need to be translated into atom or molecule fractions, depending on if the control volume is a *Reactor* or *Mixer*. Because the fuel and air of the unburned mixture does not react with the burned mixture in the *Mixer* the burned and unburned properties can be treated separately and then weighted together. For the enthalpy, for example, we therefore have that

$$h(p, T, \mathbf{x}) = \begin{cases} h_{ua}(T)x_{ua} + h_{uf}(T)x_{uf} + h_{b,mix}(p, T, \mathbf{x}_b) |\mathbf{x}_b| & T \leq T_{frozen} \\ h_{b,react}(p, T, \mathbf{x}_b) & T > T_{frozen} \end{cases}$$

where $\mathbf{x}_b = (x_{bf} \quad x_{ba})^T$. This indicates that the dimensions of the gas property functions are as low as three for the burned mixture and one for the unburned mixture. Therefore it is feasible to tabulate the data and use linear interpolation in between sample points. For burned mixtures it is necessary to, for each fuel, tabulate h , c_p , c_v , M , $\frac{\partial R}{\partial p}$, $\frac{\partial R}{\partial T}$, $\frac{\partial u}{\partial p}$, $\frac{\partial u}{\partial x_f}$, $\frac{\partial u}{\partial x_a}$, $\frac{\partial R}{\partial x_a}$, and $\frac{\partial R}{\partial x_a}$ as functions of fuel/air ratio, temperature, and for temperatures higher than T_{frozen} also pressure. For unburned mixtures only h_a , $c_{p,a}$, h_f , and $c_{p,f}$ are needed as functions of temperature.

A drawback of using tabular data and linear interpolation in this naive manner is that some properties become inconsistent with each other. For example we have that $c_p(T, \mathbf{x}) = \frac{\partial h(T, \mathbf{x})}{\partial T}$ for the mixer case and using linear interpolation of $h(T, \mathbf{x})$ implies that $c_p(T, \mathbf{x})$ is piecewise constant in temperature, but with the interpolation it also becomes linear in temperature. This gives a thermodynamic inconsistency as shown in Figure 3.6 where the difference in c_p is about 3-4% for the specific choice of table. This inconsistency can be removed by using a thermodynamically consistent interpolation as suggested in [49]. The drawback is, however, that this method is computationally expensive as the dimensions grow. In [49] the number of terms in the interpolation expression is 36 for a two dimensional case and 216 for a three dimensional case.

The numerical solution to a differential equation that is obtained by differentiating other equations, as done with the state equation and the energy equation when deriving (2.20), can diverge so that the original equations are not fulfilled. The simulation tolerances can then normally be tightened to minimize this drift but when the gas model itself is inconsistent the simulation tolerances are no longer effective. Another issue is that the thermodynamic inconsistency can interfere with the simulation routine causing a shortening of the step length or cause other phenomena. It is therefore necessary to investigate how, and if, this affects simulation performance and this will be investigated in Section 4.5. However, as a first test of the model, simulations of the table model and the full equilibrium model are obtained using the setup in Appendix C. The pressure and the temperatures

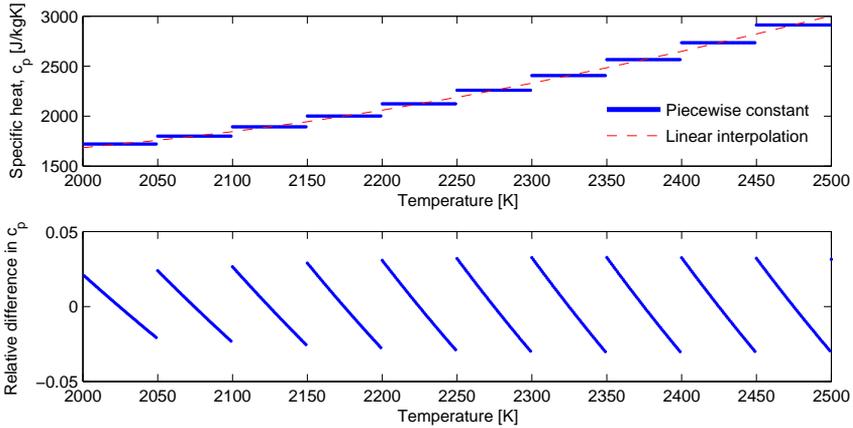


Figure 3.6: Linear interpolation of specific heat gives a difference between a thermodynamically correct c_p and an interpolation of the actual c_p . As can be seen the difference is as large as 3-4%, for this discretization.

of burned and unburned gases are shown in Figure 3.7 together with the differences between the two models. The maximal differences in pressure and temperatures are for this simulation 1 – 1.5% which shows proof of concept for the table model. The tabular data is sampled with 50 K between the sample points in temperature and between 10 kPa and 1000 kPa for the pressure. The grid layout is shown in Figure C.3 in Appendix.

3.4.4 A Simple Equilibrium Model

For a relatively cold, i.e. less than 2000K, burned fuel/air mixture the most common species are nitrogen gas, N_2 , water, H_2O , carbon monoxide, CO , carbon dioxide, CO_2 , oxygen gas, O_2 , and hydrogen gas, H_2 . As can be seen in Figure 3.3 there is less than one percent of all other common species at 1750K. It is evident from the figure that for lean mixtures CO and H_2 are negligible while for rich mixtures it is instead O_2 that is negligible. Because there are no nitrogen compounds other than N_2 amongst the most common species the nitrogen in the mixture can be considered as inert. A simple model can be built using the assumption that either the amount of CO and H_2 or the amount of O_2 is of negligible size. This model is presented in [25] and is summarized below.

Rich Mixtures

For a rich mixture it is assumed that the amount of O_2 is negligible and because the nitrogen is assumed to be inert we have for the rest of the species



where ϕ is the fuel/air equivalence ratio and ϵ depends on the H/C ratio. The equation is a simplified variant of the model reaction in (3.3).

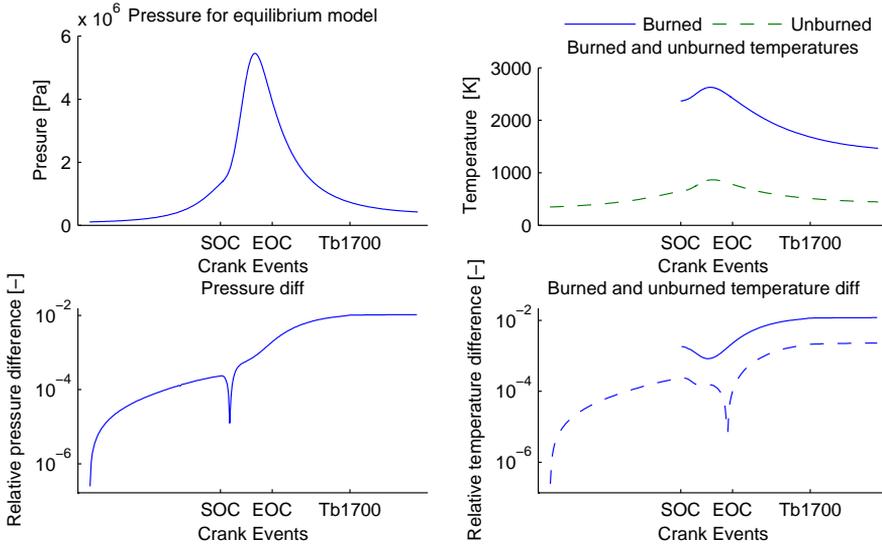


Figure 3.7: Comparison between pressure and temperatures for the full equilibrium model and the four component table model. The modeled engine cylinder is the same as in Figure 3.5 in Section 3.4.2. The maximal relative differences in pressure and temperatures is 1-1.5% for this specific choice of table grid.

If preservation equations are formulated for each reactant atom the following system of equations is obtained

$$\left. \begin{aligned} 2(1 - \epsilon)\phi &= n_{\text{H}_2\text{O}} + n_{\text{H}_2} \\ \epsilon\phi &= n_{\text{CO}_2} + n_{\text{CO}} \\ 2 &= 2n_{\text{CO}_2} + n_{\text{CO}} + n_{\text{H}_2\text{O}} \end{aligned} \right\} \quad (3.6)$$

Supposing that $n_{\text{CO}} = c$ is known it is possible to find a unique solution to the equation system in (3.6). The following relation is then obtained

$$\begin{pmatrix} n_{\text{H}_2} \\ n_{\text{CO}_2} \\ n_{\text{H}_2\text{O}} \\ n_{\text{CO}} \\ n_{\text{O}_2} \\ n_{\text{N}_2} \end{pmatrix} = \begin{pmatrix} 2(\phi - 1) - c \\ \epsilon\phi - c \\ 2(1 - \epsilon\phi) + c \\ c \\ 0 \\ 3.773 \end{pmatrix}$$

The carbon in the products can form either carbon monoxide, CO, or carbon dioxide, CO₂, which affects the amount of air that can form water. It is possible to formulate a reaction, called the water gas shift reaction, that describes the relationship between the species



and the equilibrium concentrations of this reaction gives the constant c . The equilibrium concentrations can be determined using the equilibrium equation for the reaction

$$K n_{\text{CO}_2} n_{\text{H}_2} = n_{\text{H}_2\text{O}} n_3 \Rightarrow K (\epsilon\phi - c) (2(\phi - 1) - c) = (2(1 - \epsilon\phi) + c) c \quad (3.7)$$

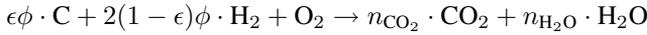
where the equilibrium constant K is a function of temperature. In [25] $K(T)$ is fitted to

$$\ln(K(T)) = 2.743 - \frac{1.761 \cdot 10^3}{T} - \frac{1.611 \cdot 10^6}{T^2} + \frac{0.2803 \cdot 10^9}{T^3}$$

where T is in kelvin.

Lean Mixtures

For lean mixtures, where CO and H_2 are negligible, all atoms from the fuel form water H_2O and carbon dioxide CO_2 . In the same manner as for the rich mixture we can simplify (3.3) to



This reaction gives, again as for the rich mixture, a number of equations that can be solved for the species concentrations as follows

$$\begin{pmatrix} n_{\text{H}_2} \\ n_{\text{CO}_2} \\ n_{\text{H}_2\text{O}} \\ n_{\text{CO}} \\ n_{\text{O}_2} \\ n_{\text{N}_2} \end{pmatrix} = \begin{pmatrix} 0 \\ \epsilon\phi \\ 2(1 - \epsilon\phi) \\ 0 \\ 1 - \phi \\ 3.773 \end{pmatrix}$$

Calculating Thermochemical Properties

The water gas shift reaction (3.7) is a second order equation except for $K = 1$ for which it is a first degree equation. It is therefore possible to solve the equation analytically. However, it is necessary to consider the possibility of numerical problems that can arise when solving the equation for c . Numerical problems can also be encountered when calculating $\frac{dc}{dK}$ and to avoid these problems the implicit function theorem is used

$$f(K, c) = 0 \Rightarrow f'_K dK + f'_c dc = 0 \Rightarrow \frac{dc}{dK} = -\frac{\frac{\partial f}{\partial K}}{\frac{\partial f}{\partial c}} = \frac{(\epsilon\phi - c) (2(\phi - 1) - c)}{K (\epsilon\phi - c + 2(\phi - 1)) + 2c + 2(1 - \epsilon\phi)}$$

Once c and $\frac{dc}{dT}$ are calculated they can be used to calculate gas properties such as h, u, c_p in the same straight forward manner as before and for example we have that

$$\tilde{c}_p = \frac{\partial \tilde{h}}{\partial T} = \sum \tilde{h}_i \frac{\partial \tilde{x}_i}{\partial T} + \sum \tilde{c}_{p,i} \tilde{x}_i$$

$$\tilde{x}_i = \frac{n_i}{\sum_j n_j}$$

$$\frac{\partial \tilde{x}_i}{\partial T} = \frac{\partial \tilde{x}_i}{\partial c} \frac{\partial c}{\partial K} \frac{\partial K}{\partial T}$$

A comparison with Chepp is made in Figure 3.8 for the molar fractions. As can be seen the model is able to predict the molar fractions reasonably well up to about 2000K. It is,

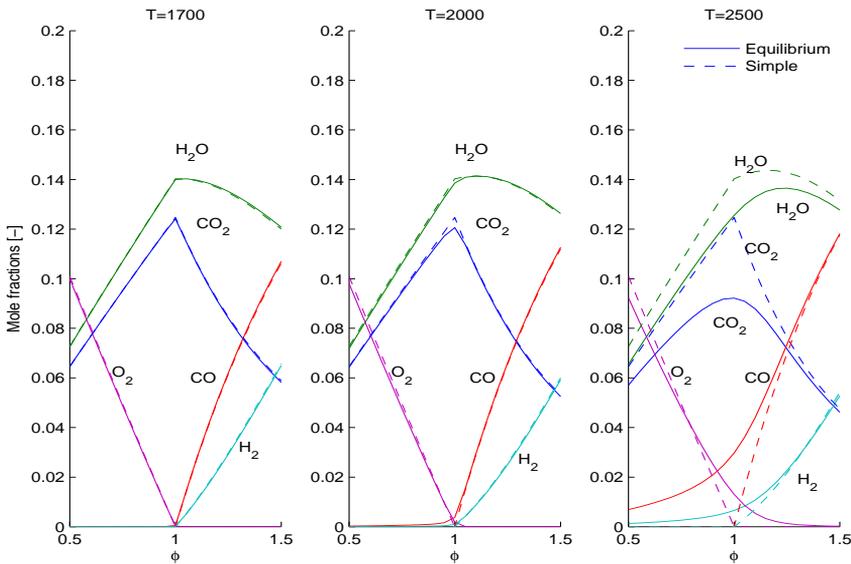


Figure 3.8: The amount of respective specie is described reasonably well up to about 2000K. Dashed molar fractions are from simple model while solid are from an equilibrium code.

however, more interesting to see how well the model is able to predict the necessary gas properties such as enthalpy, internal energy or specific heat capacities, or how it affects simulation performance measured by pressure and temperature errors. In Figure 3.9 the mole specific heat capacity $\tilde{c}_p(p, T, \tilde{x})$ is plotted against temperature for fuel/air ratio $\phi = 0.8, 1.0, \text{ and } 1.2$. The pressure for the calculation is obtained by matching the temperature in the figure to the pressure from a simulation of the model setup in Appendix C using the full equilibrium model. This illustrates that for high temperatures the fact that the molecules dissociate, is an important contribution to the heating value. In Figure 3.10 the model has then been used in an engine simulation where temperatures reach as high as 2700K. Although the errors in heat capacity are as large as 50% the error in pressure and temperature is only about 5-6%.

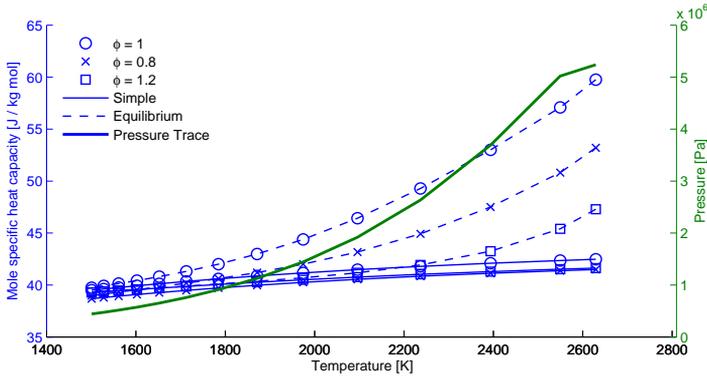


Figure 3.9: Heat capacity, c_p , as a function of temperature. For each sample in temperature the corresponding pressure that is obtained from a simulation of the setup in Appendix C is used. This pressure is also shown in the figure. The simple model gives a quite good fit for mole specific heat capacity for a temperature of 1500K but for temperatures as high as 2600K the error is as large as 50%. This illustrates that for high temperatures the fact that the molecules dissociate into smaller compounds is an important contribution to the heating value. It is, however, the actual errors in the pressure and temperature traces that are important.

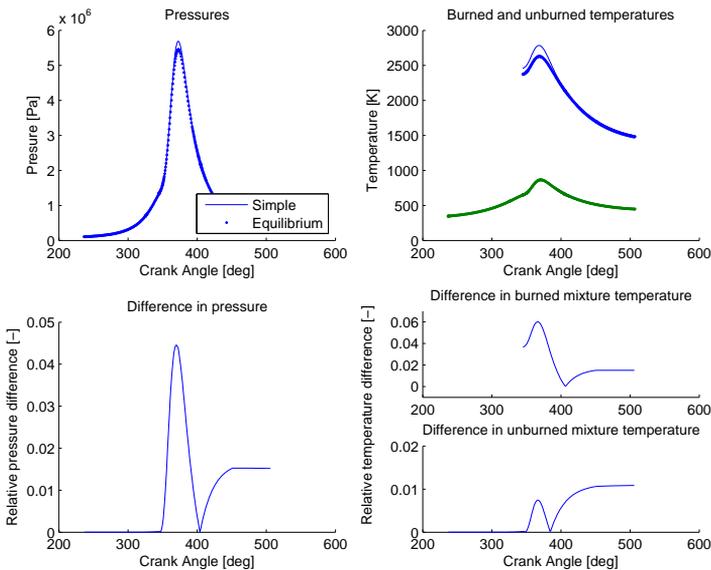


Figure 3.10: Simulation results from the simple gas model compared to the full equilibrium model using the DAE formulation from Chapter 4. The difference in pressure and burned temperature is about 5-6%. Given that the error in heat capacity at 2600K is as large as 50% this has to be considered as a small error. The modeled cylinder is the example setup from Appendix C.

3.5 Concluding Remarks about the Modeling of Gases

In this chapter the focus has been on the calculation of thermodynamic properties and thermodynamic property models.

The following list summarizes highlights of the chapter

- The concept of composition parameters, \mathbf{x} , is introduced. Each element x_k can represent different things such as air and fuel fractions or mass fractions of individual molecules or atoms.
- It is shown that the model in (2.20) can be used together with the composition parameters regardless of if they represent a frozen mixture of molecules, atoms that are allowed to react, or a collection of molecules or atoms.
- As examples three different gas representations are given, the molecule level gas, the two component gas and the four component gas.
- Two simplified gas models that use four components for the composition part of the state vector are given. The models are the table interpolation model and the simplified equilibrium model.
- A first validation of the table and simple equilibrium models gives that
 - Using the grid layout with 50 K between sample points, as in Figure C.3 in Appendix, for the table interpolation model gives an inconsistency in c_p of about 3-4% and an error of about 1-1.5% in the pressure and temperature traces.
 - For the simple equilibrium model the maximal error in pressure and temperature is about 6% which has to be considered as low since the error in heat capacity, c_p is about 50% at 2600K.

A DAE Formulation for Simulation of Thermodynamic Systems

In this Chapter the DAE formulation that is presented in [37, 36] for simulation of multi-zone cylinder models is extended with ability to have composition changes using the thermodynamic property framework. It is shown that using the DAE formulation in computer simulations is viable, i.e the DAE formulation will have a unique solution as long as the gas model fulfills a number of basic criteria and that formulating the equations as a DAE gives a better, more scalable formulation than an ODE formulation. Further, an example setup is used to validate that energy, mass, and volume are preserved when using the formulation in computer simulations. In other words, the numerical solution obeys the thermodynamic state equation and the first law of thermodynamics, and the results converge as tolerances are tightened.

4.1 Outline of the Chapter

In Section 4.2, **The Multi-Zone DAE Formulation**, the DAE formulation for simulation of thermodynamic systems is introduced. The formulation uses the result from (2.20) to derive equations for a flexible multi-zone model.

In Section 4.3, **Adding and Removing Zones**, the introduction and depletion of zones in the DAE formulation is discussed.

In Section 4.4, **Existence and Uniqueness of the DAE $\mathbb{A} \dot{\mathbf{x}} = \mathbb{B}$** , it is shown that as long as all the zones masses are non zero in a neighbourhood of the state then there is a unique solution to the simulation problem. The case of zero or very small mass is given special considerations.

In Section 4.5, **Numerical Properties of the DAE Formulation**, drift and convergence properties of the numerical solution of the DAE formulation are explored using different ODE solvers, gas models and tolerances.

4.2 The Multi-Zone DAE Formulation

The foundation for the multi-zone formulation is discussed in Chapter 2 where a model for an open thermodynamic system was deduced. In Chapter 3 it was discussed how to obtain thermochemical data for this model. The model is summarized in (2.20) which is repeated here for reference

$$\begin{aligned}
 m c_v dT + m \frac{\partial u}{\partial p} dp + p dV &= dQ + \sum_j \left((\nabla_{\mathbf{x}} u)^T (\mathbf{x} - \hat{\mathbf{x}}_{d,j}) + h(p, \hat{T}_{s,j}, \hat{\mathbf{x}}_{s,j}) - u \right) dm_j \\
 p dV + \left(V - m T \frac{\partial R}{\partial p} \right) dp - m \left(R + T \frac{\partial R}{\partial T} \right) dT &= \sum_j \left(R + (\nabla_{\mathbf{x}} R)^T (\hat{\mathbf{x}}_{d,j} - \mathbf{x}) \right) T dm_j \\
 d\mathbf{x} = \sum_j \frac{\hat{\mathbf{x}}_{d,j} - \mathbf{x}}{m} dm_j & \quad (2.20)
 \end{aligned}$$

where

$$\begin{aligned}
 \hat{\mathbf{x}}_{d,j} &= \begin{cases} g(\mathbf{x}_j) & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 \hat{\mathbf{x}}_{s,j} &= \begin{cases} \mathbf{x}_j & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 \hat{T}_{s,j} &= \begin{cases} T_j & \text{When flow is from outside } (dm_j > 0) \\ T & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\
 u &= u(p, T, \mathbf{x}) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) \quad \text{or} \quad \mathbf{x}^T \mathbf{u}(T)
 \end{aligned}$$

In the derivation of (2.20) differentials, dp , dT , and $d\mathbf{x}$, have been used instead of the time derivatives, $\frac{dp}{dt}$, $\frac{dT}{dt}$, and $\frac{d\mathbf{x}}{dt}$, which is standard in thermodynamic analysis. With the differential form there is also another possibility, namely that the independent variable can be chosen at a later time, e.g. time or crank angle can be chosen depending on what is needed in the analysis. However the same results can be obtained by using time derivatives. Note that when the independent variable has been selected then equation (2.20) is a differential algebraic equation (DAE). The difference between an ordinary differential equation (ODE) and a DAE is that there is no explicit expression for the derivatives in the DAE. It is, however, possible to solve (2.20) and get explicit expressions for the pressure and temperature derivatives and then simulate the equation as an ODE. However, the objective here is a multi-zone model and traditional ODE simulation of a thermodynamic system with multiple zones gives complex equations when the number of zones are large.

In the multi-zone model the modeled system is divided into a number of zones with a common pressure but with different temperatures, volumes, and gas compositions. Each zone in the multi-zone model is thus a control volume with a temperature, volume, and composition state but with a common state for pressure. One way to think of the zones is as N sub-volumes of the system with flexible barriers in between them. These barriers can allow mass transport between them and the zones can have a geometrical interpretation or they can be used to bookkeep mass that has burned at a specific instance.

Table 4.1: States in the general multi-zone model.

State	Description
p	Global Pressure
V_i	Volume of zone i
T_i	Temperature of zone i
\mathbf{x}_i	Composition of zone i

The model is formulated to fit the gas description and the states of the reformulated model are shown in Table 4.1. Assuming that there are k_i components in the composition state for respective zone, and that there are N zones, then $1 + \sum_{i=1}^N (2 + k_i)$ states, and hence state equations, are needed when simulating the model. Using the result from (2.20) all but one, i.e. $\sum_{i=1}^N (2 + k_i)$, equations are obtained. The last equation is obtained by observing that the sum of the volumes are equal to the total volume of the system, i.e. $\sum_{i=1}^N V_i = V$ which when differentiated yields

$$\sum_i dV_i = dV \quad (4.1)$$

When (2.20) and (4.1) are compiled to an entity we can write the resulting system of equations as follows

$$\begin{pmatrix} 0 & 1 & 0 & \cdots & 1 & 0 \\ a_1 & p & b_1 & \cdots & 0 & 0 \\ c_1 & p & d_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_N & 0 & 0 & \cdots & p & b_N \\ c_N & 0 & 0 & \cdots & p & d_N \end{pmatrix} \begin{pmatrix} dp \\ dV_1 \\ dT_1 \\ \vdots \\ dV_N \\ dT_N \end{pmatrix} = \begin{pmatrix} e_1 \\ f_1 \\ \vdots \\ e_N \\ f_N \end{pmatrix} \quad (4.2)$$

and

$$d\mathbf{x}_i = \sum_j \frac{\hat{\mathbf{x}}_{d,ij} - \mathbf{x}_i}{m_i} dm_{ij} \quad (4.3)$$

where

$$a_i = V_i - m_i T_i \frac{\partial R_i}{\partial p} = \left/ \begin{array}{l} \text{by Maxwell's relations} \\ \text{(c.f. Table 2.1)} \end{array} \right/ = m_i \left(\frac{\partial u_i}{\partial p} + \frac{T}{p} (c_{p,i} - c_{v,i}) \right)$$

$$b_i = -m_i \left(R_i + T_i \frac{\partial R_i}{\partial T} \right) = \left/ \text{by Maxwell's relations} \right/ = -m_i (c_{p,i} - c_{v,i})$$

$$c_i = m_i \frac{\partial u_i}{\partial p}$$

$$d_i = m_i c_{v,i}$$

$$e_i = \sum_j \left(R_i + (\nabla_{\mathbf{x}} R_i)^T (\hat{\mathbf{x}}_{d,ij} - \mathbf{x}_i) \right) T_i dm_{ij}$$

$$f_i = dQ_i + \sum_j \left((\nabla_{\mathbf{x}} u_i)^T (\mathbf{x}_i - \hat{\mathbf{x}}_{d,ij}) + h(p, \hat{T}_{s,ij}, \hat{\mathbf{x}}_{s,ij}) - u_i \right) dm_{ij}$$

and

$$\begin{aligned}\hat{\mathbf{x}}_{d,ij} &= \begin{cases} g(\mathbf{x}_{ij}) & \text{When flow is from outside } (dm_{ij} > 0) \\ \mathbf{x}_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases} \\ \hat{\mathbf{x}}_{s,ij} &= \begin{cases} \mathbf{x}_{ij} & \text{When flow is from outside } (dm_{ij} > 0) \\ \mathbf{x}_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases} \\ \hat{T}_{s,ij} &= \begin{cases} T_{ij} & \text{When flow is from outside } (dm_{ij} > 0) \\ T_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases} \\ u &= u(p, T, \mathbf{x}) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) \quad \text{or} \quad \mathbf{x}^T \mathbf{u}(T)\end{aligned}$$

As mentioned earlier (4.2) is a Differential Algebraic Equation, or a DAE and there is therefore no explicit expression for the state derivatives. While there are many results for stability, convergence and existence of solution for ODE's it is not obvious whether these properties are preserved in the given DAE formulation.

Using the notation

$$\mathbb{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 & 0 \\ a_1 & p & b_1 & & & \\ c_1 & p & d_1 & & & \\ \vdots & & & \ddots & & \\ a_N & & & & p & b_N \\ c_N & & & & p & d_N \end{pmatrix}, \quad \dot{\mathbf{x}} = \begin{pmatrix} dp \\ dV_1 \\ dT_1 \\ \vdots \\ dV_N \\ dT_N \end{pmatrix}, \quad \mathbb{B} = \begin{pmatrix} dV \\ e_1 \\ f_1 \\ \vdots \\ e_N \\ f_N \end{pmatrix},$$

$$\dot{\mathbf{y}} = \begin{pmatrix} d\mathbf{x}_1 \\ \vdots \\ d\mathbf{x}_N \end{pmatrix}, \quad \text{and} \quad \mathbb{B}_y = \begin{pmatrix} \sum_j \frac{\hat{\mathbf{x}}_{d,1j} - \mathbf{x}_1}{m_1} dm_{1j} \\ \vdots \\ \sum_j \frac{\hat{\mathbf{x}}_{d,Nj} - \mathbf{x}_N}{m_N} dm_{Nj} \end{pmatrix}$$

we can write the model, i.e. (4.2) and (4.3), as

$$\begin{pmatrix} \mathbb{A}(x, y) & \mathbf{0} \\ \mathbf{0} & I \end{pmatrix} \begin{pmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{y}} \end{pmatrix} = \begin{pmatrix} \mathbb{B}(x, y, z) \\ \mathbb{B}_y(x, y, z) \end{pmatrix} \quad (4.4)$$

where $\mathbb{A}(x, y)$, $\mathbb{B}(x, y, z)$, and $\mathbb{B}_y(x, y, z)$ are functions of the volume, pressure, and temperature part, x , of the state vector as well as the composition part, y . $\mathbb{B}(x, y, z)$ and $\mathbb{B}_y(x, y, z)$ may also depend on the independent variable z , i.e. time or crank angle. In the following the DAE part of the equation $\mathbb{A}(x, y)\dot{\mathbf{x}} = \mathbb{B}(x, y, z)$ will be given special attention.

4.2.1 A Single Zone Model in the DAE Formulation

From (4.2) it is evident that if we have only one zone the equations can be solved and get and ODE form, i.e. where none of the state differentials depend on other state differentials.

The equations are

$$\begin{aligned} dp &= \frac{p dV (b-d) + d e - b f}{a d - c b} \\ dT &= \frac{p dV (c-a) + a f - c e}{a d - c b} \end{aligned} \quad (4.5)$$

and for a closed well stirred mixer where the gases are thermally perfect we have $\frac{\partial R}{\partial p} = \frac{\partial R}{\partial T} = 0$ and therefore also that $a = V$, $b = -\frac{pV}{T}$, $c = 0$ and $d = \frac{pV}{RT}(c_p - R)$ in (4.5). For a single zone model of a closed system this yields

$$\begin{aligned} dp &= \frac{T}{V(c_p - R)} dQ - \frac{pT}{V} \left(\frac{1}{(c_p - R)} - \frac{1}{R} \right) dV \\ dT &= \left(\frac{RT}{pV} dQ - \frac{RT}{V} dV \right) \frac{1}{c_p - R} \end{aligned} \quad (4.6)$$

which is a common formulation of the temperature and pressure differentials. An example of the equations for a two zone approach is given in [5, p. 240]. When the number of zones are larger, however, analytic inversion of the matrix in (4.2) becomes cumbersome or even impossible.

4.3 Adding and Removing Zones

The DAE formulation has nice properties w.r.t. scalability. In fact, an $N - 1$ zone model is easily expanded to a N zone model by adding temperature, volume, and composition states, and then augmenting the \mathbb{B} -vector with two rows and the \mathbb{A} -matrix with two rows and two columns as follows

$$\mathbb{A}_N = \begin{pmatrix} & & & 1 & 0 \\ & \mathbb{A}_{N-1} & & \vdots & \vdots \\ & & & 0 & 0 \\ a_N & \cdots & 0 & p & b_N \\ c_N & \cdots & 0 & p & d_N \end{pmatrix} \quad (4.7)$$

Analogously, we can remove zones simply by removing the states and their corresponding rows and columns in \mathbb{B} and \mathbb{A} . Using the DAE formulation thus yields the possibility to add and remove zones as required by the application and is thus a better, more scalable, formulation than calculating the expression for ODE.

4.4 Existence and Uniqueness of the DAE $\mathbb{A} \dot{\mathbf{x}} = \mathbb{B}$

Now we turn to the existence and uniqueness of (4.4). The most important observation is that (4.4) is a quasi linear problem, c.f. [22, pp. 442-445]. For this type of problem we have that in a neighbourhood of (x, y) where $\mathbb{A}(x, y)$ is invertible it is possible to reformulate the problem to

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} \mathbb{A}(x, y)^{-1} \mathbb{B}(x, y, z) \\ \mathbb{B}_y(x, y, z) \end{pmatrix}$$

This equivalent problem is an ODE and all standard results for existence, uniqueness and stability of solution applies. It is therefore possible to use any ODE code as usual as long as \mathbb{A} is invertible [22]. An alternative is to use a DAE solver like the ones in [2] directly, however, this line is not pursued here. Note that this means that $\mathbb{B}(x, y, z)$ and

$\mathbb{B}_y(x, y, z)$ must still fulfill the Lipschitz condition to guarantee existence and uniqueness. This requirement is, however, not imposed by the DAE formulation or the fact that multiple zones are used.

The matrix \mathbb{A} is invertible if $\det(\mathbb{A}) \neq 0$ which means that the determinant of \mathbb{A} needs to be determined to ensure that it is possible to simulate the equations. An expression for the determinant of a perfect gas, or a gas in a well stirred mixer, is given in [37] but for a reactor the determinant looks a little different as is shown below. Another aspect is that if \mathbb{A} should be ill conditioned, numerical problems can arise and it is therefore essential to ensure that the matrix is well conditioned during a simulation or handle the problems that occur.

4.4.1 Determinant of the \mathbb{A} -matrix

It is easy to show that the determinant of the \mathbb{A} -matrix is

$$\det(\mathbb{A}) = p^{N-1} \sum_{i=1}^N (b_i c_i - a_i d_i) \prod_{n \neq i} (d_n - b_n) \quad (4.8)$$

This is for example done with induction by first observing that it is true for a single zone model with $N = 1$ and showing that if it is true for an $N - 1$ zone model it is true for an N zone model. In (4.7) an $N - 1$ zone model was expanded to an N zone model by adding two rows and two columns to the \mathbb{A}_{N-1} -matrix. The determinant of this new \mathbb{A}_N -matrix is by cofactor expansion

$$\begin{aligned} \det(\mathbb{A}_N) &= \det(\mathbb{A}_{N-1}) p (d_N - b_N) + p^{N-1} \prod_{i=1}^{N-1} (b_i - d_i) (b_N c_N - a_N d_N) \\ &= \left(p^{N-2} \sum_{i=1}^{N-1} (b_i c_i - a_i d_i) \prod_{n \neq i} (d_n - b_n) \right) p (d_N - b_N) + \\ &\quad p^{N-1} \prod_{i=1}^{N-1} (b_i - d_i) (b_N c_N - a_N d_N) \\ &= p^{N-1} \sum_{i=1}^N (b_i c_i - a_i d_i) \prod_{n \neq i} (d_n - b_n) \end{aligned}$$

which shows (4.8).

The Determinant of the \mathbb{A} -matrix for a Well Stirred Mixer

For a *Well Stirred Mixer* we have that $a_i = V_i$, $b_i = \frac{-pV_i}{T_i}$, $c_i = 0$ and $d_i = \frac{pV_i}{R_i T_i} (c_{p,i} - R_i)$ so the determinant of the \mathbb{A} -matrix simplifies to

$$\det(\mathbb{A}) = -p^{N-1} \sum_{i=1}^N \left(V_i \frac{c_{v,i}}{c_{p,i}} \right) \prod_{n=1}^N (m_n c_{p,n}) \quad (4.9)$$

which, though different at first glance, is the same as shown in [37].

Implications of the Determinant Structure

Because $\det(\mathbb{A}) \neq 0$ is required for existence and uniqueness of solution to the equation system it is interesting to see when, and if, $\det(\mathbb{A}) = 0$. If a_i , b_i , c_i , and d_i in (4.8) are replaced with their respective expressions the following equation is obtained

$$\det(\mathbb{A}) = -p^{N-1} \sum_{i=1}^N m_i \left(\frac{\partial u_i}{\partial p} + \frac{T}{p} \frac{c_{v,i}}{c_{p,i}} (c_{p,i} - c_{v,i}) \right) \prod_{n=1}^N m_n c_{p,n} \quad (4.10)$$

It is obvious that if one $m_n = 0$ then $\det(\mathbb{A}) = 0$ and thus the equation system does not have a unique solution. At a first glance the criterion for uniqueness of solution looks complex. However, as will be shown below, the introduction of multiple zones does not introduce any new criteria, other than those that apply for a single zone model. The statement is formalized in Theorem 4.1 below but the starting point is the definition of a *Well Behaved Gas*

Definition 4.1. A *Well Behaved Gas* is a gas for which the model

$$\left. \begin{aligned} a dp + b dT &= 0 \\ c dp + d dT &= dQ \end{aligned} \right\}$$

always has a unique solution for all p , T , and \mathbf{x} .

The model in Definition 4.1 is a reformulation of the general multizone model in (4.2) for one single zone with $dV = dm = 0$. The introduction of the *Well Behaved Gas* will be used to show that the multizone formulation does not introduce any new constraints on a gas for a unique solution to exist. The idea is to show that a *Well Behaved Gas* has properties that automatically fulfill the criteria from the multizone formulation and that therefore no new criteria are needed.

It is, however, unclear how hard this criteria is. Consider a control volume with a finite mass, if now $a d - b c = m^2 c_p \left(\frac{\partial u}{\partial p} + \frac{T}{p} \frac{c_v}{c_p} (c_p - c_v) \right)$ would tend to zero for a gas, then the differentials of temperature and pressure would tend to infinity when a small amount of heat is added to the system and such a gas would be a very peculiar one. It is therefore easy to accept that such gases are at least extremely rare and that most, if not all, gases are *Well Behaved*. All gases that have been tried out using CHEPP have turned out to be well behaved. In fact $\frac{T}{p} \frac{c_v}{c_p} (c_p - c_v)$ is typically at least about two times the size of $-\frac{\partial u}{\partial p}$.

For a *Well Behaved Gas* defined by Definition 4.1 it is possible to formulate the following lemma

Lemma 4.1

For a well behaved gas $\frac{a d - b c}{m^2} = c_p \left(\frac{\partial u}{\partial p} + \frac{T}{p} \frac{c_p}{c_v} (c_p - c_v) \right) > 0$ regardless of temperature, pressure or composition.

Proof: Firstly, to always have a unique solution to the model in Definition 4.1 we must have $a d - b c \neq 0$.

Secondly, because $ad - bc > 0$ for an inert gas and all gas-mixtures of inert and non inert mixtures are continuous in composition $ad - bc$ can never switch sign. From this it follows that $ad - bc > 0$ for all *Well Behaved Gases*. \square

Using Lemma 4.1 it is now possible to formulate the following theorem

Theorem 4.1

For all Well Behaved Gases $\det(\mathbb{A}) = 0 \iff \exists i \in \{1 \cdots N\}$ s.t. $m_i = 0$.

Or in other words, as long as the gas is *Well Behaved* then the only way to have $\det(\mathbb{A}) = 0$ is that at least one $m_i = 0$. On the other hand, if any $m_i = 0$ we always have $\det(\mathbb{A}) = 0$.

Proof: If we have $m_i = 0$ for some $i = i_0$, then $\prod_{n=1}^N m_n c_{p,n} = 0$ and therefore we have that $\det(\mathbb{A}) = 0$.

On the other hand, it follows from Lemma 4.1 that $ad - bc > 0$ and because $c_{p,n} > 0$ and $p > 0$, then if $\det(\mathbb{A}) = 0$ at least one $m_i = 0$. \square

4.4.2 Condition Number of the \mathbb{A} -matrix

Even if it is clear that there exists a unique solution as long as all of the zones have non-zero sizes, i.e. all $m_i \neq 0$, it is not clear whether the numerical properties of \mathbb{A} are sound and hence if it's really feasible to solve $\mathbb{A}\dot{x} = \mathbb{B}$ numerically with any accuracy. The condition number of a matrix gives an upper limit on how much the solution to an equation system varies with the parameters on the right hand side of the equation and is therefore a good measure of how errors in input parameters are translated into errors in the solution. In the case with $\mathbb{A}\dot{x} = \mathbb{B}$ the condition number of \mathbb{A} is an upper limit of how much errors in \mathbb{B} affects the estimated \dot{x} .

For a low condition number the problem is well-conditioned, while for a high condition number the problem is ill-conditioned. It turns out that the \mathbb{A} -matrix is ill-conditioned in general and for some cases, for example when some m_i :s are small the condition number tends to infinity. The condition numbers of the \mathbb{A} -matrix during a simulation is shown in Figure 4.1. For this particular case the condition number is as highest when either the burned volume is small or when the pressure is high.

A condition number as high as 10^{18} indicates that the solution of $\mathbb{A}\dot{x} = \mathbb{B}$ is extremely sensitive to errors in \mathbb{B} . It is however an upper limit and as will be shown later the large condition number in Figure 4.1 is a pessimistic estimate.

Adaptive Scaling of the \mathbb{A} -matrix

It is tempting to try to scale the problem to get a better conditioned problem as is done in [37]. It is important, however, to notice that even if the condition number of a scaled \mathbb{A} -matrix is lower than that of the original \mathbb{A} -matrix it's not going to affect the actual sensitivity of \dot{x} on \mathbb{B} . It will, however, give a more realistic upper limit and enable us to gain insight into where possible problems are.

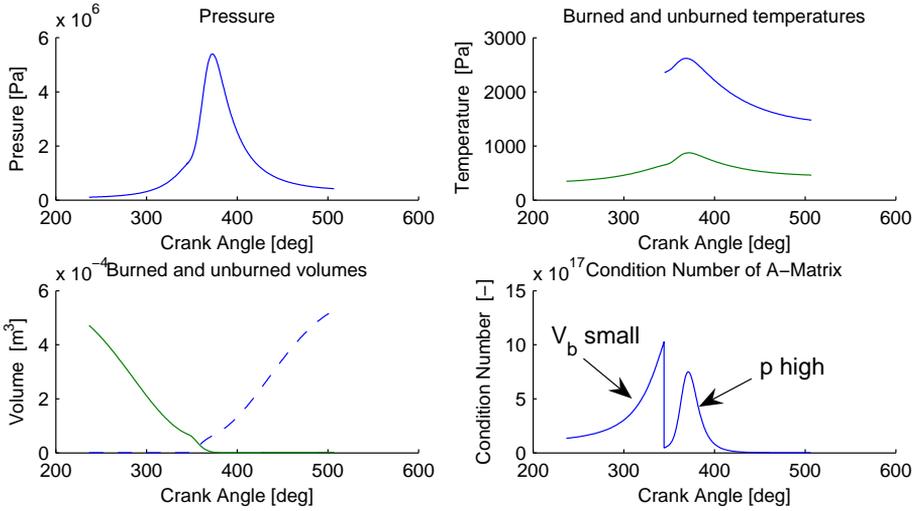


Figure 4.1: Condition number for the \mathbb{A} -matrix during a simulation of the two zone example setup in Appendix C. The burned zone is initialized to 10^{-11} of the total volume which is why the condition number is as high as 10^{18} during the compression stroke. Later in the simulation the high pressure is the dominating contributor. The combustion efficiency is set to 99% which is why the unburned zone never becomes small enough to affect the condition number.

In [37] an adaptive scaling scheme is developed which lowers the condition number to about 10^4 . The key is to scale the states and hence the state differentials with their respective sizes or nominal sizes so that all the states variables are of the same magnitude. In this presentation a similar approach is taken. Firstly, we can note that $\frac{c_i}{a_i}$ and $-\frac{b_i}{d_i} \in [0, 1]$. Secondly, we can note that the pressure is alone, except for a 1 on the first row, in every other column. It is possible to use this knowledge to factorize the \mathbb{A} -matrix in the following way

$$\mathbb{A} = \mathbb{D}_1 \mathbb{A}_D \mathbb{D}_2 = \mathbb{D}_1 \begin{pmatrix} 0 & \frac{a_1}{V_{\text{Tot}}} & 0 & \cdots & \frac{a_N}{V_{\text{Tot}}} & 0 \\ 1 & 1 & \frac{b_1}{d_1} & & & \\ \frac{c_1}{a_1} & 1 & 1 & & & \\ \vdots & & & \ddots & & \\ 1 & & & & 1 & \frac{b_N}{d_N} \\ \frac{c_N}{a_N} & & & & 1 & 1 \end{pmatrix} \mathbb{D}_2$$

where

$$\mathbb{D}_1 = \begin{pmatrix} \frac{V_{\text{Tot}}}{p} & & & & & \\ & a_1 & & & & \\ & & a_1 & & & \\ & & & \ddots & & \\ & & & & a_N & \\ & & & & & a_N \end{pmatrix} \quad \text{and} \quad \mathbb{D}_2 = \begin{pmatrix} 1 & & & & & \\ & \frac{p}{a_1} & & & & \\ & & \frac{d_1}{a_1} & & & \\ & & & \ddots & & \\ & & & & \frac{p}{a_N} & \\ & & & & & \frac{d_N}{a_N} \end{pmatrix}$$

The condition number of the \mathbb{A}_D -matrix is significantly smaller than that of the original matrix. In Figure 4.2 the condition number of a two zone model with one burned and one unburned zone is plotted against different state variables.

It is clear from Figure 4.2 that the alarming situation in Figure 4.1 might not be as bad as it looks at a first glance. The experience from working with the \mathbb{A}_D matrix is that the condition number has never exceeded 10 for a two zone model.

Interpretation of the Factorized \mathbb{A} -matrix

The reason for the \mathbb{A}_D -matrix to be better conditioned, than the original \mathbb{A} -matrix, is that the components that make the equation system numerically sensitive are put in the \mathbb{D} -matrices. Using $\mathbb{A} = \mathbb{D}_1 \mathbb{A}_D \mathbb{D}_2$ in $\mathbb{A} \dot{x} = \mathbb{B}$ we have that $\mathbb{D}_1 \mathbb{A}_D \mathbb{D}_2 \dot{x} = \mathbb{B}$ and therefore $\mathbb{A}_D (\mathbb{D}_2 \dot{x}) = (\mathbb{D}_1^{-1} \mathbb{B})$. One interpretation is that \mathbb{D}_1^{-1} magnifies the contributions from \mathbb{B} while \mathbb{D}_2 suppresses contributions to the final \dot{x} estimate.

Because both \mathbb{D}_1^{-1} and \mathbb{D}_2 contain $\frac{1}{m_i}$ factors they both have elements that tend to infinity when one $m_i \rightarrow 0$. It is therefore not obvious how the solution to the equation system is affected for this case. In Example 4.1 the effect that having a nearly empty zone has is investigated.

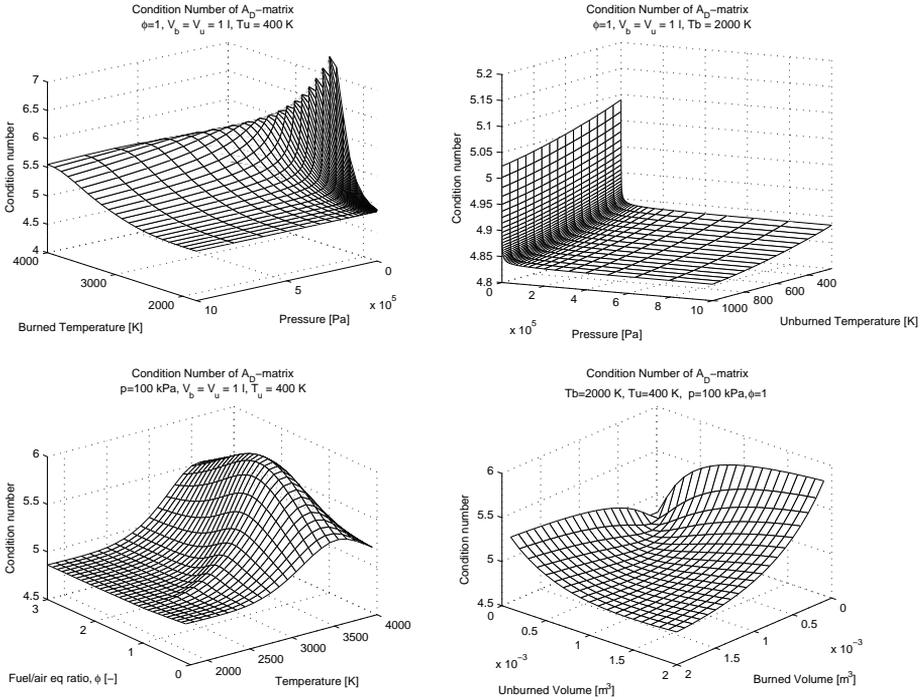


Figure 4.2: Condition number for the A_D -matrix for a two-zone model as a function of different state variables. There is only a slight dependence on unburned temperature and that the dependence on volume can be characterized as a dependence of volume differences. There is a peak at relatively high burned gas temperatures which is due to the fact that $c_{p,i}$ is high. There is also a strong dependence on pressure when it decreases towards zero. The lowest pressure in the top right plot is as low as 100 Pa which is well below the pressures that are present in the engine cylinder.

Example 4.1

The part of (4.2) that comes from the state equation and the energy equation for a zone is, using $\mathbb{A}_D (\mathbb{D}_2 \dot{x}) = (\mathbb{D}_1^{-1} \mathbb{B})$,

$$\begin{aligned} \frac{\bar{c}_i}{\bar{a}_i} \cdot (dp) + 1 \cdot \left(\frac{p}{m_i \bar{a}_i} dV_i \right) + 1 \cdot \left(\frac{\bar{d}_i}{\bar{a}_i} dT_i \right) &= \left(\frac{f_i}{m_i \bar{a}_i} \right) \\ 1 \cdot (dp) + 1 \cdot \left(\frac{p}{m_i \bar{a}_i} dV_i \right) + \left(\frac{\bar{b}_i}{\bar{d}_i} \right) \cdot \left(\frac{\bar{d}_i}{\bar{a}_i} dT_i \right) &= \left(\frac{e_i}{m_i \bar{a}_i} \right) \end{aligned}$$

where $\bar{a}_i = \frac{a_i}{m_i}$ has been used. Neither \bar{a}_i , \bar{b}_i , \bar{c}_i , \bar{d}_i , e_i , nor f_i depend on zone size. From the equation it is clear that as $m_i \rightarrow 0$ the significance of dp and dT_i in the equation becomes smaller and smaller. When $m_i = 0$ the equation does not make sense any because both \mathbb{D}_1^{-1} and \mathbb{D}_2 have elements that are infinite.

The use of \mathbb{D}_1^{-1} and \mathbb{D}_2 thus causes problems when trying to decide the state differentials for the nearly empty zone but on the other hand it enables us to see that having one nearly empty zone only has implications for the zone itself. The rest of the system is not affected by the zone. The value of dp can be determined from other equations in the total system while dT_i does not occur anywhere else and the conclusion is that as one $m_i \rightarrow 0$ then the corresponding dT_i becomes indeterminable.

Another way to interpret the division into zones is that a zero mass zone does not exist until the mass is non-zero. It is then natural that the temperature of the first element in the new zone is decided completely by an energy preservation equation for the element that enters the zone. In the same manner it is natural that the composition is decided by the composition of the first element that enters the zone, i.e $\mathbf{x}_i = g(\mathbf{x}_j)$.

In the following small example it is assumed that $dQ_i = 0$, i.e. no heat can be transferred to it. Using the formulation in (2.20) and inserting $\mathbf{x}_i = \hat{\mathbf{x}}_{d,j} = g(\mathbf{x}_j)$, $\hat{\mathbf{x}}_{s,j} = \mathbf{x}_j$, $\hat{T}_{s,j} = T_j$, and $dQ_i = 0$ then yields

$$\begin{aligned} \left. \begin{aligned} 0 dp + 0 dT_i + p dV_i &= (h(p, T_j, \mathbf{x}_j) - u(p, T_i, g(\mathbf{x}_j))) dm_j \\ 0 dp + 0 dT_i + p dV_i &= R_i T_i dm_j \end{aligned} \right\} \Rightarrow \\ \Rightarrow (h(p, T_j, \mathbf{x}_j) - h(p, T_i, g(\mathbf{x}_j))) dm_j &= 0 \end{aligned} \quad (4.11)$$

where $h = u + pV = u + RT$ has been used. Note that only one mass flow dm_j has been assumed. If (4.11) is extended with more than one mass flow it is clear that $h_i(p, T_i, \frac{\sum \mathbf{x}_j dm_j}{\sum dm_j})$ has to match $\frac{\sum_j h(p, T_j, \mathbf{x}_j) dm_j}{\sum dm_j}$ and that the only unknown variable is T_i . The interpretation is that for a zone that is being created T_i is decided by (4.11).

When implementing a multi-zone model it is sometimes practical to create all zones that are to be used during the simulation beforehand. In the initialization a temperature and volume is specified, later when the empty zone receives its first mass element the temperature of the zone can become non continuous which gives another insight into of why the equation system does not have a unique solution for $m_i = 0$.

4.4.3 Initialization and Depletion of Zones

In the previous section an interpretation to why the equation system does not have a unique solution for $m_i = 0$ was given. The cases of initialization and depletion of zones are directly related to this, as the masses either start from zero or go to zero and this issue will be treated below.

Simulation Aspects for Initialization of New Zones

A tempting strategy is to set $dT_i = dV_i = 0$ as long as there is no mass flow to the zone. At the first step with non-zero mass flow (4.11) can be used to calculate dV_i while dT_i is set to zero. This way, the first element of new mass gets the wrong temperature, T_i , but as more mass is flowing this error will be out weighted by the rest of the flow. However, even if there exists a unique solution for when $m_i \neq 0$ this strategy can cause a number of problems

1. When $\frac{m_i}{m_{\text{Tot}}}$ is close to machine precision, $\mathbb{D}_1^{-1}\mathbb{B}$ will be large and great care have to be taken to avoid numerical problems in the process of calculating $\dot{\mathbf{x}}$.
2. If T_i of the nearly empty zone differs from that of the flow the system will require a small step length for the ODE solver and can slow the simulation to a halt.
3. Other model components can make the system stiff for small volumes. For example, a badly designed heat transfer model can be a source of stiffness. This can occur because the volume/area ratio for a sphere approaches infinity as the sphere shrinks towards zero volume. Such problems are, however, easily avoided by adjusting the particular model for small zones.

The first problem can be avoided by simply setting a limit for when to consider V_i to be zero. Using $\frac{V_i}{V_{\text{Tot}}} > 10\epsilon$, where ϵ is the machine precision, as a limit has shown out to completely avoid the problem. The reason that V_i is used as limit instead of m_i is that it is a state variable and therefore readily available. Furthermore, since the connection between m_i and V_i is strong, similar results would be obtained when instead using a limit for $\frac{m_i}{m_{\text{Tot}}}$.

If the simulation tool has a possibility to change a state variable instantaneously the second problem can be avoided by setting T_i to a value that fulfills (4.11) as soon as $\frac{V_i}{V_{\text{Tot}}} > 10\epsilon$. If, however, that possibility is not available all is not lost. Another approach is to estimate the initial temperature that the empty zone will have when it is first created and initialize T_i with this temperature. This approach can, however, be risky because if the estimated T_i is off by the slightest amount the system is going to require a small step length anyway. Figure 4.3 shows a simulation where $\frac{V_i}{V_{\text{Tot}}} = 10\epsilon$ has been used as a limit. The initial temperature is off by 250 K which causes the step length to be shortened to 10^{-12} s which shows how the temperature error can slow down the simulation speed.

A third approach for solving the second problem is to raise the limit for when a zone is considered as zero from 10ϵ so that the zone may grow to a safe size before it is fully included in the model. This method has been used successfully in the application in Chapter 5 and the effect that this has on the simulation results is investigated in Section 4.5.5.

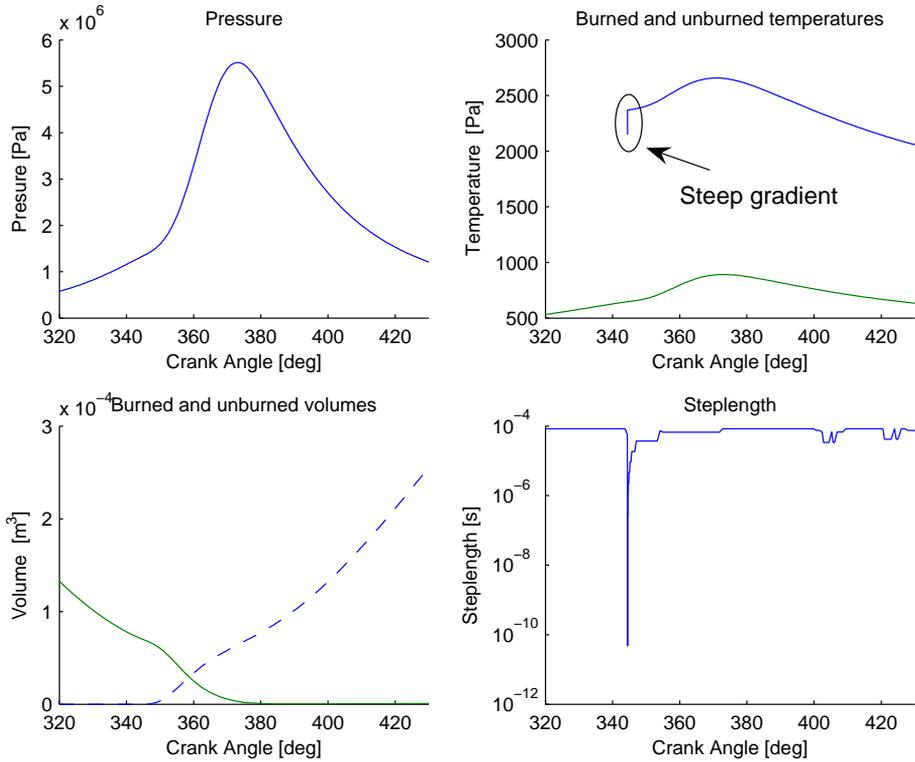


Figure 4.3: Results from simulation of the 2 zone model from Appendix C with zero volume in the burned zone at the beginning. Because the temperature of the empty zone is off by 250 K the burned gas temperature has a steep gradient as the combustion starts. Note that in this simulation the heat transfer coefficients have been set to zero to avoid other sources for transients.

Simulation Aspects for Depletion of Zones

Another important situation is when zones are depleted. There is risk that the ODE solver steps past the limit at which a zone is considered to be empty in one single step and then the equations are no longer valid since the volume is negative for some zone. A simple solution is to round negative volumes upwards and rely on the zero cross detection mechanisms in the ODE solver to find a step length that gives acceptable results according to the tolerances of the simulation.

If it is a possibility one could try to help the solver to find the zero crossing events by specifying predictable events such as start and end of combustion etc. Another possibility is to specify mass flows between zones as functions of the simulated states instead of as functions of an ideal state. An example of this is given in Example 4.2 below.

Example 4.2

The *Vibe* function, [51], is frequently used to model the fraction of burned mass.

$$x_B = \eta \left(1 - e^{-a \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^{m+1}} \right)$$

where η is a combustion efficiency parameter, m and a are shaping parameters while θ , θ_s , and θ_e are current crank angle, crank angle at start of combustion, and crank angle at end of combustion respectively. The burn rate is then the derivative of x_B with respect to either time or crank angle, i.e.

$$\frac{dx_B}{d\theta} = \eta \frac{a(m+1)}{\theta_e - \theta_s} \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^m e^{-a \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^{m+1}} \quad (4.12)$$

which can then be used to define the mass flow rate $\frac{dm_j}{dt}$ between the zones in (4.2) as

$$\frac{dm_j}{d\theta} = m_{\text{Tot}} \frac{dx_B}{d\theta}$$

where m_{Tot} is the total amount of mass available for combustion. The burn rate that is obtained by using (4.12) is specified as a beforehand given rate and does not depend on the state during the simulation. The equation can, however, be rewritten into a differential equation where the mass flow into the burned zone depends on the burned mass fraction, $x_B = \frac{m_B}{m_{\text{Tot}}}$, as

$$\frac{dm_j(\theta)}{d\theta} = m_{\text{Tot}} \frac{a(m+1)}{\theta_e - \theta_s} \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^m \left(\eta - \frac{m_B(\theta)}{m_{\text{Tot}}} \right) \quad (4.13)$$

Using the actual state variables of the model to model burn ratio as in (4.13) instead of the original model in (4.12) efficiently stops the ODE solver from trying to burn more than the total available amount of fuel even if the accuracy is low and the numerical solution is drifting.

4.5 Numerical Properties of the DAE Formulation

When using computer simulation to solve differential equations it has to be remembered that the numerical solutions do not always represent the true solution to the equations. As with Lorenz weather simulations, mentioned in the introduction, the equations can be so sensitive to the numerical accuracy that changing the accuracy of the solver yields completely different results. Another important issue is that for equations that are obtained by differentiation of other equations, as was done with the state and energy equations when (2.20) was derived, it is possible that the solution drifts so that the original equations are not fulfilled. The simulation tolerances can normally be tightened to minimize this drift but because it is assumed that these equations are valid, inconsistencies in the solution can cause problems for the ODE solver. Another issue that needs attention is differential equations that become stiff, which in some cases can be solved by using an ODE solver with stiff properties.

4.5.1 Investigation Setup

Because of the possible problems mentioned above it is interesting to study how the numerical solution to the DAE formulation behave for different ODE codes as well as different relative tolerances and gas models. It is also interesting to see how simplifications in the gas models affect the results. Unfortunately there is no analytic solution available for comparison and therefore another approach has to be taken instead. The closest to an analytical solution that we can get on average is a simulation with a really tight tolerance. Therefore a simulation with the full equilibrium gas model with a relative tolerance of 10^{-10} is used as a reference. Note that even if tighter tolerances perform better on average, random occurrence of events, such as start of combustion (SOC) or end of combustion (EOC) can affect the step length control for a simulation with a lower accuracy so that it yields better results for a particular simulation. Therefore the reference simulation is not necessarily the closest to the true solution and this needs to be considered when comparing results from simulations with different tolerances. It is also important to remember that differences between simulations are only accurate to a certain extent which is decided by the tolerance that is used, and this too has to be considered when calculating the relative difference between results from two simulations.

Properties that are easy to validate are the preservation of total mass, volume and energy. If the total heat transfer, $\int_{t_{\text{start}}}^t \dot{Q} d\tau$, and work, $\int_{t_{\text{start}}}^t \dot{W} d\tau$ are recorded the energy equation can be used to define a relative measure of how much energy that is gained or lost during the simulation

$$u_{\text{err}}(t) = \frac{\left| u_{\text{Tot}}(t) - \frac{1}{m_{\text{Tot}}(t_{\text{start}})} \int_{t_{\text{start}}}^t \left(\dot{Q}(\tau) + \dot{W}(\tau) \right) d\tau - u_{\text{Tot}}(t_{\text{start}}) \right|}{q_{LHV}} \quad (4.14)$$

where $u_{\text{Tot}} = \frac{1}{m_{\text{Tot}}} \sum_i \frac{p V_i}{R(p, T_i, \mathbf{x}_i) T_i} u(p, T_i, \mathbf{x}_i)$, q_{LHV} is lower heating value, and $m_{\text{Tot}} = \sum_i \frac{p V_i}{R(p, T_i, \mathbf{x}_i) T_i}$. In a similar manner the relative error in total mass and volume can be defined as

$$m_{\text{err}}(t) = \frac{|m_{\text{Tot}}(t) - m_{\text{Tot}}(t_{\text{start}})|}{m_{\text{Tot}}(t_{\text{start}})} \quad \text{and} \quad V_{\text{err}}(t) = \frac{|V_{\text{Tot}}(t) - V_{\text{Tot}}(t_{\text{start}})|}{V_{\text{Tot}}(t_{\text{start}})} \quad (4.15)$$

where $m_{\text{Tot}} = \sum_i \frac{p V_i}{R(p, T_i, \mathbf{x}_i) T_i}$ and $V_{\text{Tot}} = \sum_i V_i$.

In the analysis that follows the numerical properties of the DAE formulation is illustrated by studying the preservation and convergence properties given an example model setup. The investigation considers the following aspects:

ODE solver and tolerance where different solvers and tolerances are used to show how the simulations converge as tolerances are tightened and how the different ODE codes perform. This investigation shows that the formulation converges as tolerances are tightened and that the problem is not sensitive to selection of ODE code.

Gas model where the four gas models from Chapter 3 are used to study how the simplifications that are made affect the solver and the simulation results. This investigation shows that when using table interpolation the solver cannot preserve energy and mass completely because of the thermodynamic inconsistency. The investigation also shows that the simple equilibrium model is the one with least inconsistencies but with the largest errors.

Minimum volume where the effect that excluding a zone from the equations until the volume has reached a minimum volume fraction is analysed. This is used to show how small an initial zone can be if a certain accuracy is desired.

The model setup uses the `psClosedCylSIModel` model from Section 5.3.9 and the model setup is described in Appendix C. The following analysis is based on one single operating point and does by no means span over all the different possibilities. The analysis serves as an example of the type of studies that are needed to ensure that the numerical results from simulation of a new model setup are accurate. The DAE formulation can be simulated using any ODE code but in this presentation the solvers in MATLAB/SIMULINK are used for the simulations. Furthermore, the simulations and investigations also serve as a mean for validating the correctness of the implementation.

4.5.2 Short Introduction to ODE Solvers

To understand why different ODE solvers have different properties it is necessary to have a basic knowledge about computer simulation. Higher order methods can be classified either by their properties or by their construction. For example a distinction is made between *stiff* and *non-stiff* methods as well as *Runge Kutta* and *Linear Multi-step* methods [2].

Stiffness An informal definition of stiffness for a problem is that “An initial value problem is stiff in some interval if the step size needed to maintain stability of the forward Euler method is much smaller than the step size required to represent the solution accurately” [2]. The definition indicates a discrepancy between stability and accuracy of the solver but there are methods that have bigger stability regions which gives them the possibility to take longer steps without becoming unstable even for stiff problems. These methods are by construction implicit and requires that an equation system is solved in each simulation step. This implies that they are slower than explicit methods for a given step length but on the other hand they can take longer steps for stiff problems. At tight tolerances a short step length can, however, be required anyway and therefore an explicit method can be faster even if the problem is stiff according to the informal definition.

Single Step or Multi Step *Runge Kutta* based methods are single step methods. Therefore it is easy to change step length from one step to another. *Linear multi-step* methods can, however, be faster than *Runge Kutta* methods but with the drawback that they depend on earlier steps. When changing the step length the solver therefore has to restart completely or interpolate the solution in previous steps to obtain the solution at the required points. As a consequence *Runge Kutta* methods are usually faster for problems that need frequent change of step length while *Linear multi-step* methods can be faster for fixed step lengths or problems for which it is not necessary to change the step length frequently.

Order Selection The order of a method refers to how the accuracy of a method depends on the step length. A method is said to be of order p if the local truncation error, i.e. the maximum error that can be obtained at each step, is given by $\mathbf{d}_n = \mathcal{O}(h_n^p)$. Here \mathbf{d}_n is the local truncation error at step n while h_n is the step length and p the order of the method. The cost of a higher order method is, however, given by $\mathcal{O}(p \cdot h_n)$. One interpretation is that for crude tolerances a low order method can be faster than a high order method but when tolerances are tightened higher order methods become more attractive.

4.5.3 Drift and Convergence for Different Solvers and Tolerances

To make sure that the simulation results can be trusted and to check how the performance depends on the choice of solver a number of simulations were performed. In Figures 4.4 and 4.5 different tolerances are used for the `ode45` solver in SIMULINK. As can be seen in the figures the results converge towards a common point. The growth in mass and volume error clearly depends on the tolerances while for the energy error there is no difference between the three lower tolerances during the combustion. This can be related to the accuracy of the chemical equilibrium calculation. There is also a large step in energy and mass error as the burned mass is cooled below 1700 K and becomes frozen.

The sudden increase in error that is made due to the fact that the gas freezes and becomes a mixer indicates that it is not necessary to use tighter tolerances than about 10^{-4} . If the error poses a problem a solution is to restart the simulation as the temperature passes the frozen limit or change the limit so that the inconsistency becomes smaller. It is important to realize, however, that the error that is made is due to an assumption in the gas model which is only a simplified model. If the simulation is restarted when the temperature passes the frozen temperature there will instead be a step in pressure and temperature to compensate for the transition between reactor and mixer. The key to solving the problem therefore lies in making the gas model consistent. Before paying the gas model too much attention, in an effort to be able to increase the accuracy by tightening the tolerances, it is worthwhile to investigate what other sources of errors there are and what their contributions to the total error are. For example, steam at 773 K and atmospheric conditions deviates 0.1 percent from the ideal gas equation [7] and the mole fraction of steam in a stoichiometric mixture is about 15 percent. As another example, the linearity of typical research pressure sensors, e.g. [4], are about 0.1 percent.

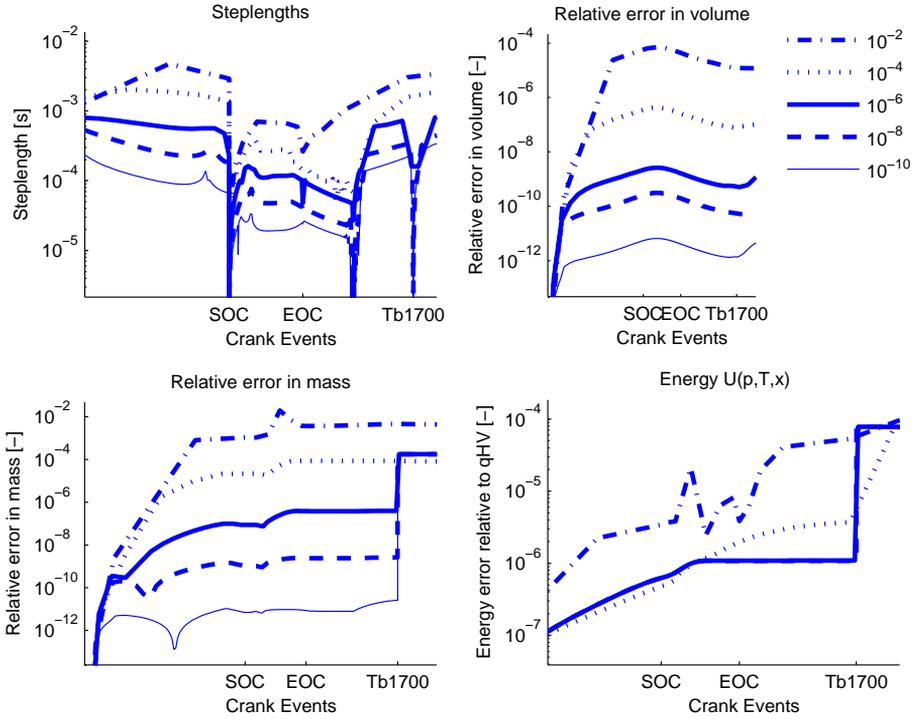


Figure 4.4: Mass, energy and volume preservation for different tolerances. At start of combustion (SOC) the step length is shortened considerably for all tolerances except 10^{-2} . During the combustion, between SOC and EOC the error correlates with the relative tolerance but at $T_b = 1700$ K the energy and mass preservation are affected by the fact that c_p is discontinuous when the burned gas passes the frozen limit. This affects the four simulations with lowest tolerances negatively which is also seen in the step length plot for three of them. The simulation setup is given in Appendix C and the solver that is used is `ode45` from SIMULINK.

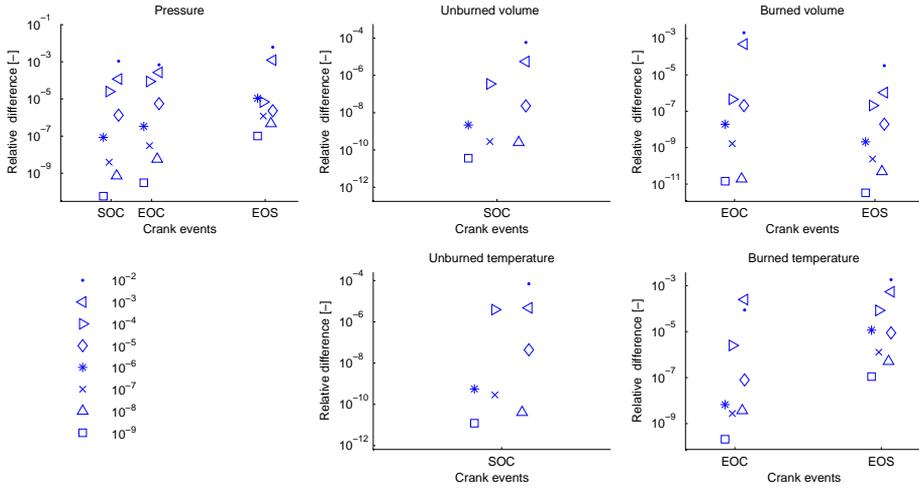


Figure 4.5: Relative difference in pressure, volumes, and temperatures at start of combustion (SOC), end of combustion (EOC) as well as end of simulation (EOS). The reference simulation is the one with the tightest tolerance, i.e. 10^{-10} , and as tolerances are tightened the simulations converge towards this simulation. The simulation setup is given in Appendix C and the solver that is used is *ode45* from SIMULINK. Note that the markers have been scattered in the x-direction to increase readability.

In Figures 4.6 and 4.7 different solvers and tolerances have been used in combination to see how well they all perform on the particular set up. All standard variable step solvers from SIMULINK have been used in the evaluation. The total simulation time for the different solvers is denoted in respective legend. The stiff solvers *ode15s*, *ode23s*, *ode23t*, and *ode23tb* are all outperformed by *ode45*, *ode113* and *ode23* when it comes to simulation time. For tolerances 10^{-3} and 10^{-4} the stiff solvers have about the same preservation properties but are about 10 times slower than their competitors. All solvers with a simulation time of more than 80 s are left out of the figures.

It is clear that for this particular simulation *ode113* is preferable if simulation speed is important. *ode113* does, however, give a larger growth in volume error than the others at SOC. There can be a number of reasons for this. According to the formula for *ode113* it is an *implicit* linear multi-step method. The method is, however, implemented in a way which does not take the implicit iteration to convergence which in turn makes the implementation explicit. Therefore the solver does not suffer from having to evaluate the function as many times as truly implicit methods. It is stated that such methods can be fast when the function evaluation is expensive and that they are suited only for non-stiff problems [2]. The large error growth at SOC can be related to the start up properties for *ode113* since it depends on earlier steps to calculate the solution or the fact that it is an explicit implementation of a implicit formulation. It is clear, however, that none of the other solvers suffer from this error growth at SOC.

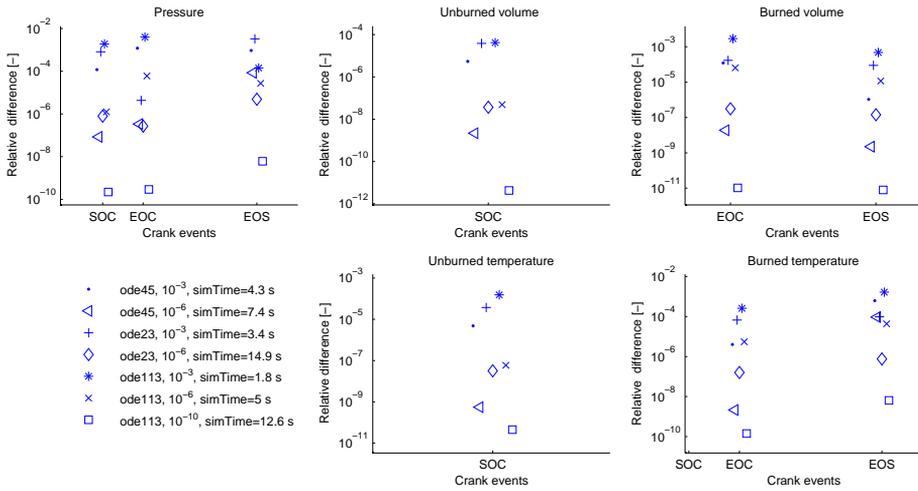


Figure 4.6: Relative difference in pressure, temperatures, and volume at different events. Simulations at different tolerances perform equally well regardless of solver. For all events and properties the *ode113* simulation with relative tolerance 10^{-10} is the closest to *ode45* with the same relative tolerance. For the simulations with a relative tolerance of 10^{-6} *ode113* yields the largest difference in burned properties compared to the reference simulation. This can be related to the volume error at SOC in Figure 4.7. The trend is however that the simulations are converging towards the reference simulation as the tolerances are tightened. The model setup is given in Appendix C.

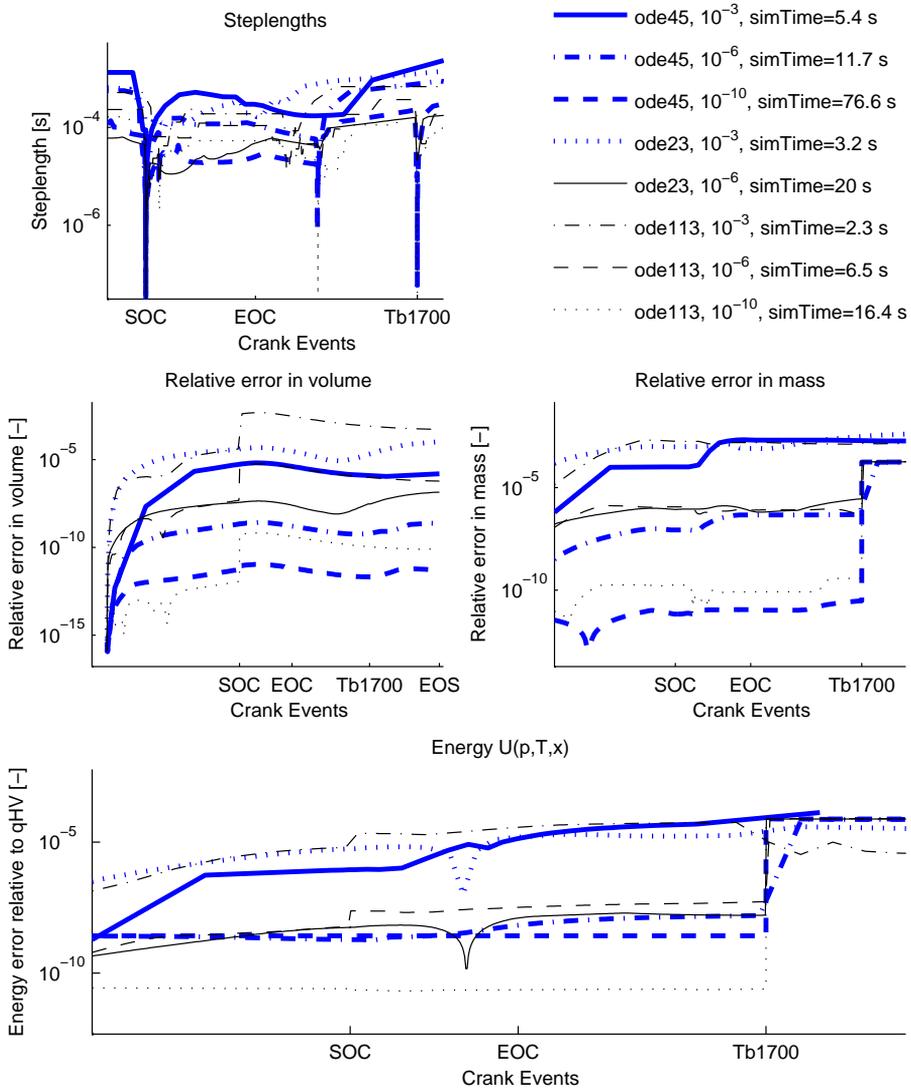


Figure 4.7: Mass energy and volume preservation for different solvers at different tolerances. *ode45*, *ode113* and *ode23* all behave quite similarly. *ode113* does, however, give a larger growth in volume error than the others at SOC. The model setup is given in Appendix C.

4.5.4 Drift and Convergence for Different Gas Models

It is interesting to see how the choice of gas model affects the simulation results and the performance. The different gas models are simplified and contain errors. Therefore it is, for example, not necessary to use a higher accuracy than the one that matches these errors. It is also interesting to see how inconsistencies, as for example with the interpolated properties of the tabular data model, affects the ODE solver.

The four models that are used are the:

cheppfull - 10 molecule full equilibrium model

chepp - the 4 composition parameters equilibrium model

simpleeq - the simplified equilibrium model with 4 composition parameters

tables - the table interpolation model with 4 composition parameters

Because **cheppfull** and **chepp** gives almost exactly the same results only **cheppfull** is used in the comparison. The tabular data is sampled with 50 K between the sample points in temperature and between 10 kPa and 1000 kPa for the pressure as shown in Figure C.3 in Appendix.

Preservation Properties for the Simpleeq and Tables Models In Figure 4.8 the mass, energy, and volume preservation properties of the **simpleeq** model is compared with **cheppfull**. The results for the two models are about the same except that the **simpleeq** model is not affected as the burned gas temperature passes the frozen limit. In Figure 4.9 the **tables** has instead been compared with the **cheppfull** model. As can be seen the **tables** model disturbs the solver and causes the step length to be shortened at every inconsistency in the thermochemical properties. If the results are examined closer it shows out that the lowest points on the curly energy error plot coincides with the simulation passing a sample point in temperature or pressure. The drift in volume depends almost only on the relative tolerance while the drift in mass and energy are affected by the gas model. The reason is that the volume preservation equation from the DAE formulation does not contain any gas properties while the energy equation suffers from the thermodynamic inconsistency. The conclusion is that for mass and energy preservation there is no need to use any lower tolerance than 10^{-6} .

Convergence Properties for the Tables and Simpleeq Models In Figure 4.10 the three models are compared to the **cheppfull** simulation with a relative tolerance of 10^{-10} . The figure shows the difference in pressure, temperature, and volumes for the three compared models at different relative tolerances. The simulated pressure, temperatures, and volumes does as expected not yield the same results for the different models. It is evident that for this particular choice of interpolation points there is no need for using any tighter tolerance than 10^{-3} for the **tables** or the **simpleeq** models.

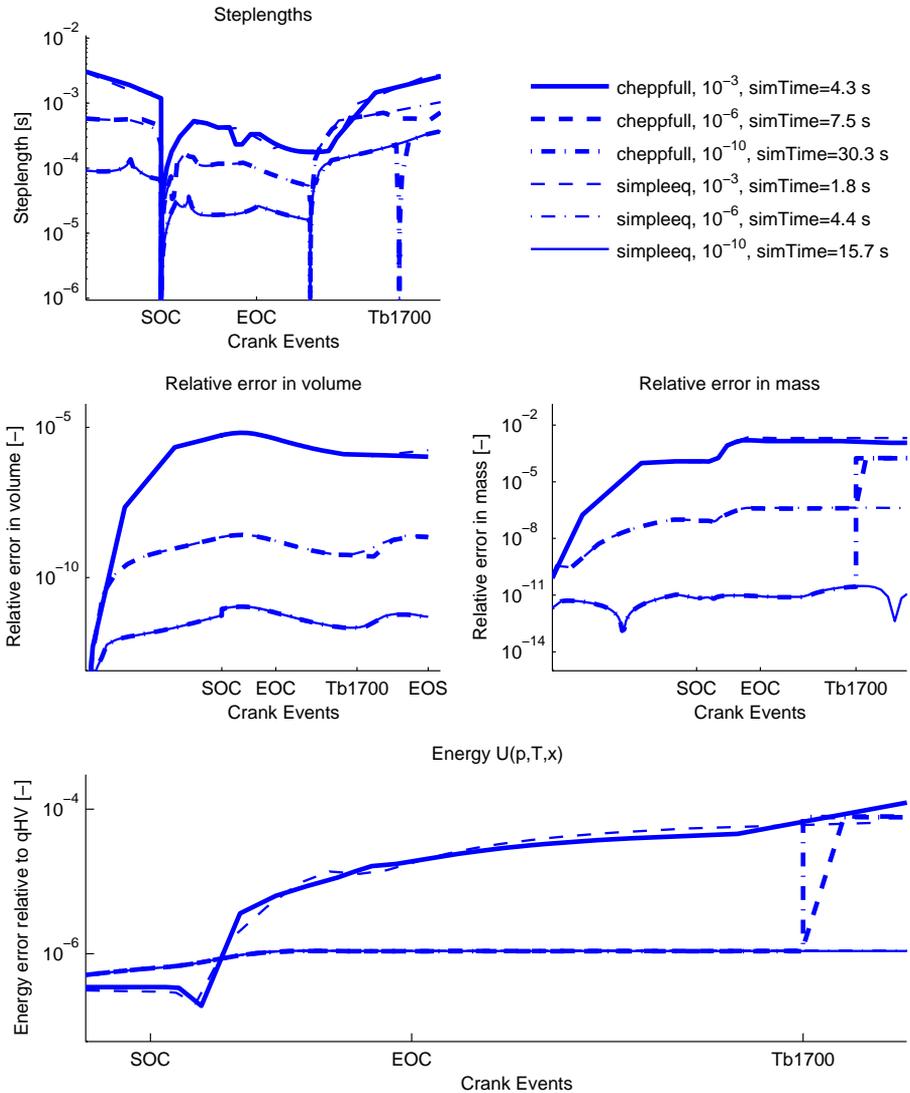


Figure 4.8: Preservation of total mass, energy, and volume for the *cheppfull* and the *simpleeq* models at different tolerances. The only difference between the preservation properties of the two models is that the *simpleeq* model is not affected as the burned gas temperature passes the frozen limit. This can be because the transition from frozen to equilibrium is smoother in the *simpleeq* model than in the *cheppfull* model. The model setup is given in Appendix C.

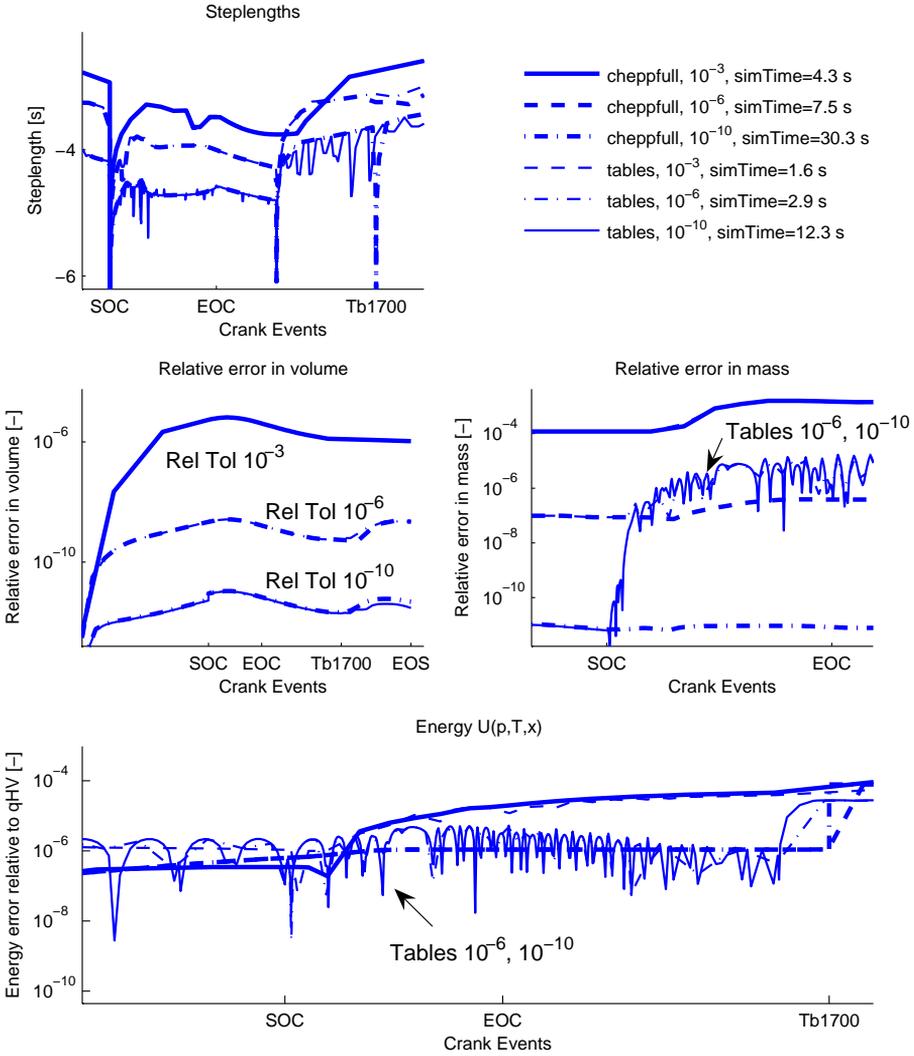


Figure 4.9: Preservation of total mass, energy, and volume for the *cheppfull* and the *tables* models at different tolerances. It is not efficient to simulate *tables* with high accuracy because the thermodynamic inconsistencies disturbs the solver. This, however, only occurs for relatively tight tolerances and since the *tables* model is only an approximation of the *cheppfull* model it has, as will be shown, quite large errors in itself and using tight tolerances is therefore not necessary. The model setup is given in Appendix C.

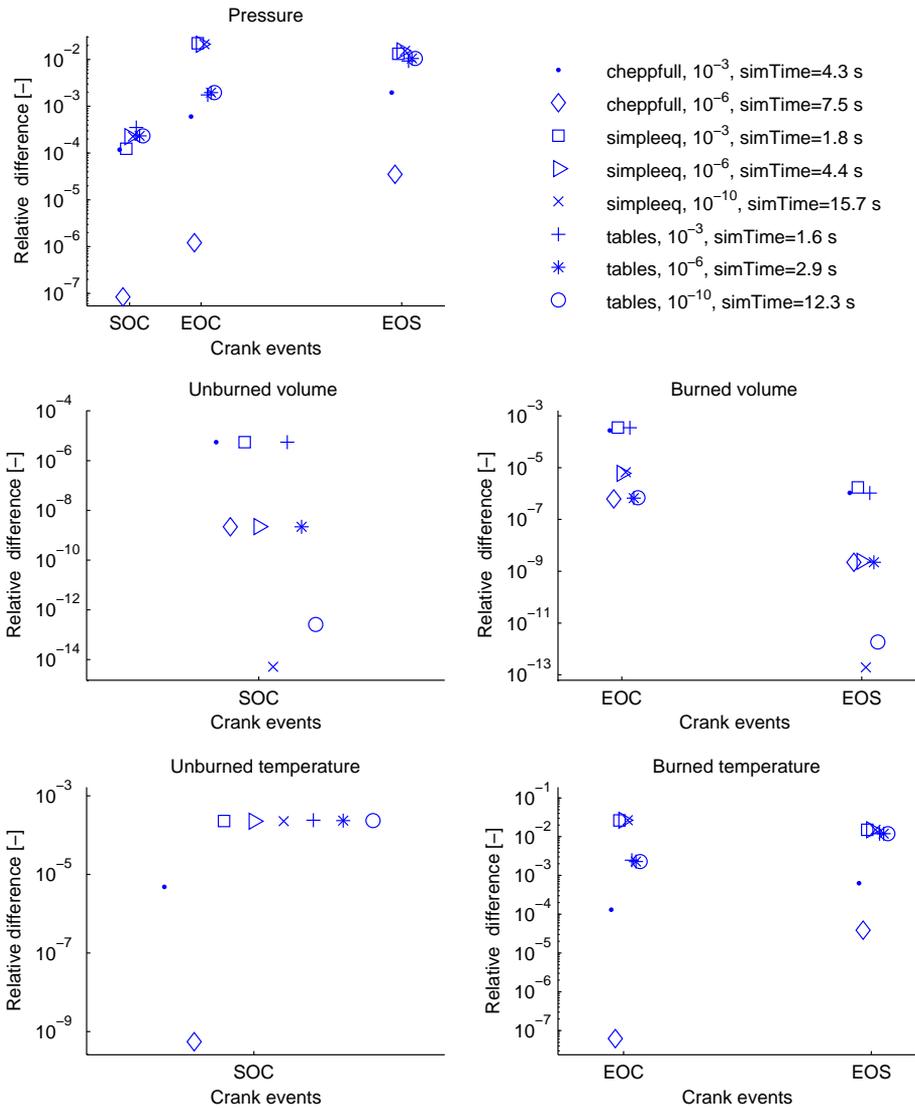


Figure 4.10: Relative difference in pressure, volume, and temperature at different crank events. At start of combustion (SOC) the difference pressure is about the same for tolerance 10^{-3} for the three models. At the end of simulation, however, none of the two simplified models come close to the equilibrium model even at tolerance 10^{-3} . This means that it is better to use the full equilibrium model at tolerance 10^{-3} than for example the table model with tolerance 10^{-6} for this particular simulation and that the cost is an increase in simulation time of about 40%. The temperatures behaves in a similar manner. The relative errors in volume does not, as expected from the model structure, depend on gas-model. The model setup is given in Appendix C.

Grid Sizes for the Tables Model In Figures 4.11 and 4.12 results from using the `tables` model with a temperature grid of 50K (as in Figure C.3 in Appendix) is compared to results from a `tables` model with a 5K temperature grid. The relative error in mass is affected in the same manner for both of the grid layouts. For the relative error in energy the 5K grid does, however, not oscillate as much as the 50K grid layout. At a relative tolerance of 10^{-6} the 5K grid gives significantly less oscillations in energy than the 50K grid and for a relative tolerance of 10^{-4} there are no oscillations at all. For the pressure, temperature, and mass the 5K grid table model with relative tolerances 10^{-6} and 10^{-8} yields errors with the same magnitude as for the `cheppfull` model using a relative tolerance of 10^{-4} . For the relative tolerance of 10^{-6} and 10^{-8} the 5K grid table model is, however, outperformed by the `cheppfull` model. A conclusion is that for this case a temperature grid with 5K between the samples instead of 50K lowers the relative errors by at least a factor ten, if the model is simulated using tolerances of at least 10^{-6} .

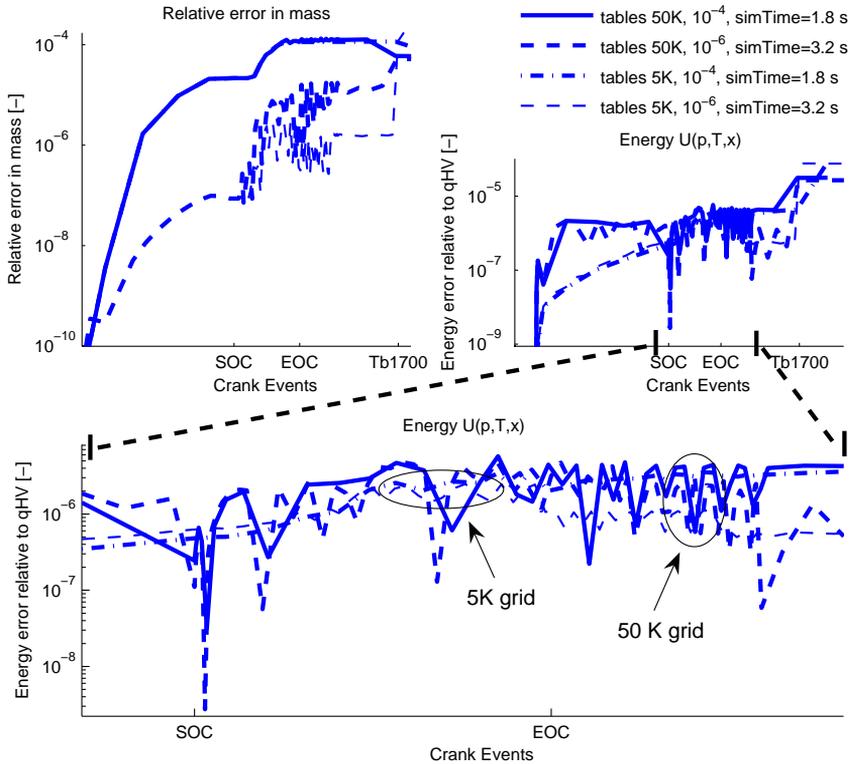


Figure 4.11: Preservation of total mass, energy, and volume for the *tables* model using different tolerances and table grids. When the temperature grid is 10 times as dense than for the default setup the relative error in mass is still affected by the inconsistency for the tightest tolerance (10^{-6}). For the relative error in energy the 5K grid model is, however, not as affected by the inconsistencies even if there is a slight oscillating tendency for the tightest tolerance. The model setup is given in Appendix C.

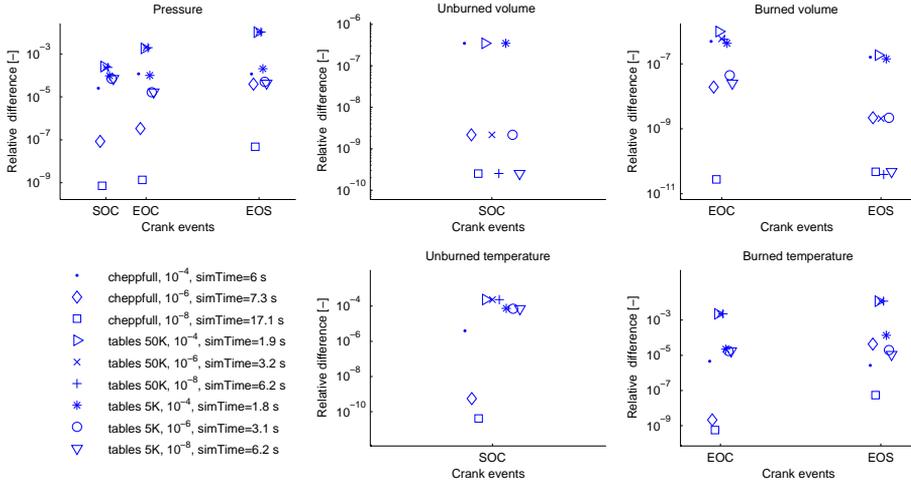


Figure 4.12: Relative difference in pressure, temperatures and volumes at the events SOC, EOC, EOS. The *cheppfull* model with a relative tolerance of 10^{-10} is used as reference. For the volumes the relative tolerance is the most important factor. For temperatures and pressures the situation is, however, different. For the 50K grid table model the pressure and temperatures have about the same errors regardless of tolerance but for the 5K grid table model with relative tolerances 10^{-6} and 10^{-8} the errors are of the same magnitude as for the *cheppfull* model using a relative tolerance of 10^{-4} . For the relative tolerance of 10^{-6} and 10^{-8} the 5K grid table model is, however, outperformed by the *cheppfull* model. Note that the markers have been scattered in the x-direction to increase readability.

In Table 4.2 the *cheppfull* model is compared to the *tables* model with temperature grids of 100K, 50K, 25K, 12.5K, and 5K at a relative tolerance of 10^{-10} . There is almost no variation in simulation time depending on the size of the tables and it is evident that there is a large decrease in error as the grid is lowered from 25 K to 12.5 K.

Table 4.2: Relative errors in pressure, temperatures, and volumes at SOC, EOC, and EOS where applicable. The simulation time for the *cheppfull* model at a relative tolerance of 10^{-10} is used as reference and the simulation time was 31.0 s. The same relative tolerance is used for all cases and the model setup that is used is described in Appendix C.

ΔT_{grid}	Rel. Error			Time
	SOC	EOC	EOS	
Pressure diff, $\Delta p /p_{\text{ref}}$				
100 K	$1.57 \cdot 10^{-04}$	$2.08 \cdot 10^{-03}$	$1.03 \cdot 10^{-02}$	11.1 [s]
50 K	$2.33 \cdot 10^{-04}$	$1.96 \cdot 10^{-03}$	$1.06 \cdot 10^{-02}$	12.0 [s]
25 K	$1.77 \cdot 10^{-04}$	$1.92 \cdot 10^{-03}$	$1.05 \cdot 10^{-02}$	12.5 [s]
12.5 K	$7.05 \cdot 10^{-05}$	$1.83 \cdot 10^{-05}$	$8.92 \cdot 10^{-05}$	12.6 [s]
5 K	$7.08 \cdot 10^{-05}$	$1.68 \cdot 10^{-05}$	$4.30 \cdot 10^{-05}$	12.1 [s]
Burned temp diff, $\Delta T_b /T_{b,\text{ref}}$				
100 K		$2.47 \cdot 10^{-03}$	$1.18 \cdot 10^{-02}$	11.1 [s]
50 K		$2.28 \cdot 10^{-03}$	$1.21 \cdot 10^{-02}$	12.0 [s]
25 K		$2.27 \cdot 10^{-03}$	$1.20 \cdot 10^{-02}$	12.5 [s]
12.5 K		$1.56 \cdot 10^{-05}$	$6.22 \cdot 10^{-05}$	12.6 [s]
5 K		$1.76 \cdot 10^{-05}$	$1.11 \cdot 10^{-05}$	12.1 [s]
Unburned temp diff, $\Delta T_u /T_{u,\text{ref}}$				
100 K	$1.57 \cdot 10^{-04}$			11.1 [s]
50 K	$2.33 \cdot 10^{-04}$			12.0 [s]
25 K	$1.77 \cdot 10^{-04}$			12.5 [s]
12.5 K	$7.05 \cdot 10^{-05}$			12.6 [s]
5 K	$7.07 \cdot 10^{-05}$			12.1 [s]
Burned volume diff, $\Delta V_b /V_{b,\text{ref}}$				
100 K		$5.82 \cdot 10^{-07}$	$1.11 \cdot 10^{-12}$	11.1 [s]
50 K		$5.68 \cdot 10^{-07}$	$2.26 \cdot 10^{-12}$	12.0 [s]
25 K		$5.41 \cdot 10^{-07}$	$1.32 \cdot 10^{-12}$	12.5 [s]
12.5 K		$2.57 \cdot 10^{-08}$	$1.05 \cdot 10^{-12}$	12.6 [s]
5 K		$2.53 \cdot 10^{-08}$	$1.02 \cdot 10^{-12}$	12.1 [s]
Unburned volume diff, $\Delta V_u /V_{u,\text{ref}}$				
100 K	$1.38 \cdot 10^{-12}$			11.1 [s]
50 K	$2.60 \cdot 10^{-13}$			12.0 [s]
25 K	$1.96 \cdot 10^{-12}$			12.5 [s]
12.5 K	$1.60 \cdot 10^{-12}$			12.6 [s]
5 K	$8.90 \cdot 10^{-13}$			12.1 [s]

4.5.5 Influence of Initialization

In Section 4.4.3 initialization issues for zones with zero volume were discussed. To avoid shortening of the step length it was suggested that the limit of $\frac{V_i}{V_{Tot}}$, for when a zone was fully included in the equations, was raised from 10 times the machine precision, or 10ϵ , to a level that does not suffer from the steep gradient problem. In Figure 4.13 the limit for the inclusion of the burned zone is varied from 10^{-6} to 10ϵ and the convergence to a simulation with the correct temperature from the beginning is studied. As the figures show the burned gas temperature converges to the simulation with the correct burned gas temperature as the limit is lowered. The correct burned gas temperature is calculated using (4.11) at the first time step that the zone is fully included in the equations. The difference

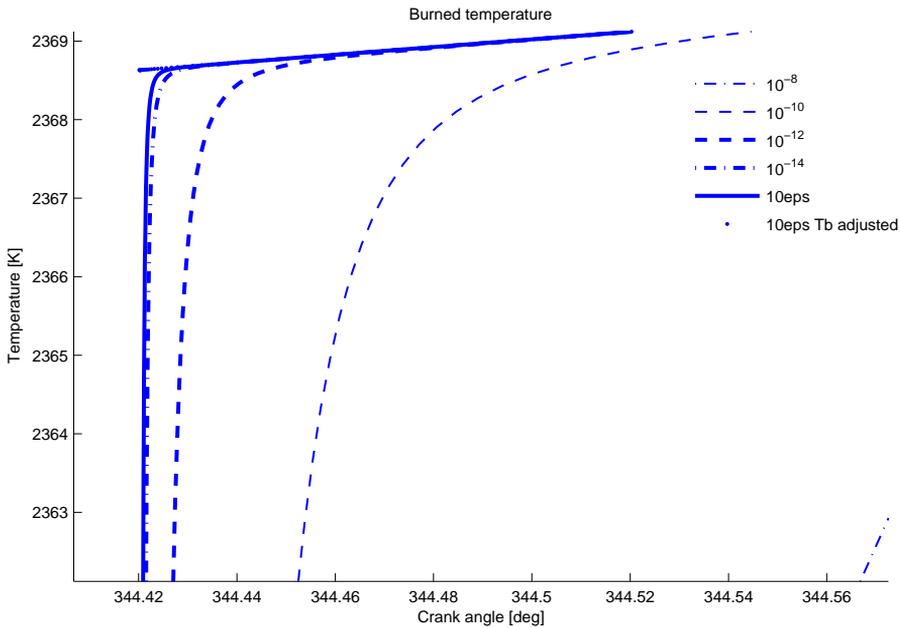


Figure 4.13: Temperatures at SOC for different minimum volumes, $\frac{V_i}{V_{Tot}}$. As the minimum volume, that is considered as non zero, decreases the burned temperature converges. The simulations that are not visible in the figure have a similar behaviour but with larger time constants. The heat transfer coefficient has been set to zero for this particular simulation to avoid sources of steep gradients as discussed in Section 4.4.3. The model setup is given in Appendix C. Note that the scales on the X- and Y-axes shows that all the gradients are steep. The initial difference in temperature is $\approx 219\text{K}$.

in pressure, temperature of burned zone, and volume of burned zone are shown in Table 4.3 for the different simulations. The difference in pressure, temperature, and volume clearly becomes smaller as the limit is lowered. The conclusion is that for temperature and pressure at EOC and EOS the initial zone fraction can be as large as 10^{-6} for a two

Table 4.3: Difference in pressure and burned zone temperature and volume for different volume limits and with an initial error in temperature of $\approx 219\text{K}$. The simulation time for the reference simulation with the adjusted temperature was 24.0s.

minVFrac	Rel. Error		Time
	EOC	EOS	
Pressure diff		$ \Delta p /p_{\text{ref}}$	
10^{-6}	$2.66 \cdot 10^{-06}$	$3.03 \cdot 10^{-06}$	22.3 [s]
10^{-8}	$2.67 \cdot 10^{-08}$	$2.97 \cdot 10^{-08}$	23.3 [s]
10^{-10}	$2.67 \cdot 10^{-10}$	$3.15 \cdot 10^{-10}$	23.7 [s]
10^{-12}	$2.14 \cdot 10^{-12}$	$1.62 \cdot 10^{-12}$	26.2 [s]
10^{-14}	$5.12 \cdot 10^{-13}$	$1.17 \cdot 10^{-12}$	26.9 [s]
10ϵ	$5.27 \cdot 10^{-13}$	$2.96 \cdot 10^{-14}$	28.3 [s]
Burned temp diff		$ \Delta T_b /T_{b,\text{ref}}$	
10^{-6}	$4.23 \cdot 10^{-08}$	$2.25 \cdot 10^{-07}$	22.3 [s]
10^{-8}	$4.19 \cdot 10^{-10}$	$2.57 \cdot 10^{-09}$	23.3 [s]
10^{-10}	$4.07 \cdot 10^{-12}$	$4.05 \cdot 10^{-11}$	23.7 [s]
10^{-12}	$3.65 \cdot 10^{-14}$	$5.17 \cdot 10^{-13}$	26.2 [s]
10^{-14}	$1.05 \cdot 10^{-13}$	$8.98 \cdot 10^{-14}$	26.9 [s]
10ϵ	$8.24 \cdot 10^{-14}$	$5.25 \cdot 10^{-16}$	28.3 [s]
Burned volume diff		$ \Delta V_b /V_{b,\text{ref}}$	
10^{-6}	$2.62 \cdot 10^{-06}$	$4.50 \cdot 10^{-07}$	22.3 [s]
10^{-8}	$2.63 \cdot 10^{-08}$	$3.74 \cdot 10^{-09}$	23.3 [s]
10^{-10}	$2.64 \cdot 10^{-10}$	$4.44 \cdot 10^{-11}$	23.7 [s]
10^{-12}	$2.82 \cdot 10^{-12}$	$5.02 \cdot 10^{-13}$	26.2 [s]
10^{-14}	$3.33 \cdot 10^{-13}$	$8.20 \cdot 10^{-14}$	26.9 [s]
10ϵ	$2.05 \cdot 10^{-13}$	$3.08 \cdot 10^{-15}$	28.3 [s]

zone model without affecting the results more than $3 \cdot 10^{-6}$. It has to be remembered, however, that if the initial temperature of the burned zone is important this estimate can be misleading. The effect that the volume fraction limit has on the result therefore has to be assessed from case to case.

4.6 Concluding Remarks about the DAE Formulation

The DAE formulation proposed in [37] has been extended and analyzed. The DAE formulation is founded upon the equation (2.20) and the equations for the thermodynamic properties in Chapter 3. The numerical properties of the \mathbb{A} matrix is further explored in this presentation compared to [37, 36].

It is shown that if at least one zone has zero mass, then the temperature differential of that zone becomes indeterminable. As a solution it is suggested that the temperature differential of the zero mass zone is left out of the equations and that the temperature is either set to an approximate value at start of simulation or reset to the temperature that it should have according to the energy equation as soon as there is mass in the zone.

It has also been shown that, for an example operating point, all simulations are consistent with the physical equations so that mass, energy and volume are preserved and that the drift in these properties is consistent with the relative tolerance of the simulation. The initialization issues have been discussed and it has been shown that the proposed solution does not influence the simulation results.

Finally, the influence upon choice of thermochemical property model has been discussed. It is therefore safe to make the following claims

- As long as all zones have positive mass the DAE formulation has a unique solution.
- Furthermore, as long as at least one zone has positive mass the DAE formulation can be altered to give a unique solution.
- The drift in mass, energy, and volume, that is introduced in the numerical solution is consistent with the relative tolerance of the simulation.
- Simplifications, such as using a 4 component gas model, interpolating thermochemical properties, and letting a new zone grow to a safe size before including in the calculations, have a known and controllable effect on the simulation result. The conclusions from these investigations are that
 - Using a temperature grid with 5K between sample points instead of 50K can yield at least 10 times smaller errors if simulation tolerances are as tight as 10^{-6} .
 - For the equilibrium model when the gas changes from being a reactor to a mixer the relative errors increases to a level that is consistent with using a relative tolerance of 10^{-3} .
 - Using $\frac{V_i}{V_{\text{Tot}}} > 10^{-6}$ as a limit when to consider a zone as empty gives a relative error in pressure, temperature, and volume of less than $3 \cdot 10^{-3}$ for an initial temperature error of ≈ 219 K.
- The simulation results for the SI engine model that uses the DAE formulation converges, for this particular operating point, to a common numerical solution of the equations (4.2) and (4.3).

It is therefore possible to draw the conclusion that the methodology gives a correct and consistent numerical solution to the original equations for the particular choice of engine model and operating point.

5

Implementation of the DAE Framework

As an application of the DAE framework a process simulation package, named PSPACK, has been developed. This chapter describes the structure of the package and its subdivision into modules together with the available choices for the different modules. It also documents the interfaces that have been developed, and highlights some important implementation choices.

5.1 Outline of the Chapter

In Section 5.2, **PSPACK, a Process Simulation Package, Basic Idea and Overview of the Simulation Package**, an overview of the structure of the simulation package is given and all the modules are introduced.

In Section 5.3, **Detailed Descriptions of the Modules**, the different modules and the available alternatives for respective module are documented.

5.2 PSPACK, a Process Simulation Package, Basic Idea and Overview of the Simulation Package

PSPACK is a tool that allows a user to combine different implementations of different models in a straight forward way. It gives structure to the simulation process, i.e. defining a simulation setup and performing single or batches of simulations. This is achieved with the aid of the developed DAE simulation framework for thermodynamic systems.

The package is designed to be modular, which means that all modules are replaceable with minimal effort. To support this, standard interfaces have been designed. Implementing

new models or modules is as easy as adding a directory with files that follow the given standard. The modular part of PSPACK is organized in a tree structure with sub-directories that contain the modules. The modules may include one or more implementations to choose between and these implementations are all stored in different directories of the module directory. In the root there is a small collection of necessary MATLAB m-scripts. These m-scripts make up the user interface and provides some tools for commonly used calculations, such as calculation of module variables that depend on other variables.

A graphical user interface (GUI) with scripting possibilities is implemented that supports the task of setting up simulations, both graphically and by MATLAB-scripts. There is a main user interface as well as interfaces for all modules. The initialization GUI's for the different modules all follow the same standard so that an interface for a module can easily be understood by anyone already familiar to other parts of the program package.

When setting up a simulation two things are needed:

- A selection of implementation for modules that are needed. This is done by adding the path of the desired module implementations.
- Parameters for the modules. Parameters can for example be entered manually or by scripting. There are also defaults for all parameters.

Both these tasks are either done using the graphical user interface or by manually setting up a simulation structure. One option is to set up and save a default simulation structure and then load altered versions of this simulation structure. In this manner it is easy to create scripts that can perform for example batch simulations, sensitivity studies and parameter optimization.

5.2.1 Overview of PSPACK modules

PSPACK is divided into the following modules

- `psModel` contains engine models which defines zone layouts and connects the different parts that are needed for a simulation
- `psKernel` solves the DAE equation
- `psThermProp` calculates thermochemical properties
- `psValve` models for cylinder valve lifts
- `psGeometry` calculates cylinder volume, area etc.
- `psHeat` calculates heat transfer
- `psVibe` contains mass fraction burned models

The connection between the different modules is illustrated in Figure 5.1. The engine model is responsible for setting up a zone-layout and defining an initial state vector with the aid of the other modules, i.e. geometry, valve lifts, thermochemical properties, and mass fraction burned. During the simulation the model first calls the modules to calculate necessary properties. The kernel is then called to calculate the state derivatives given the zone layout. For larger models, with more than one control volume the kernel may be called several times.

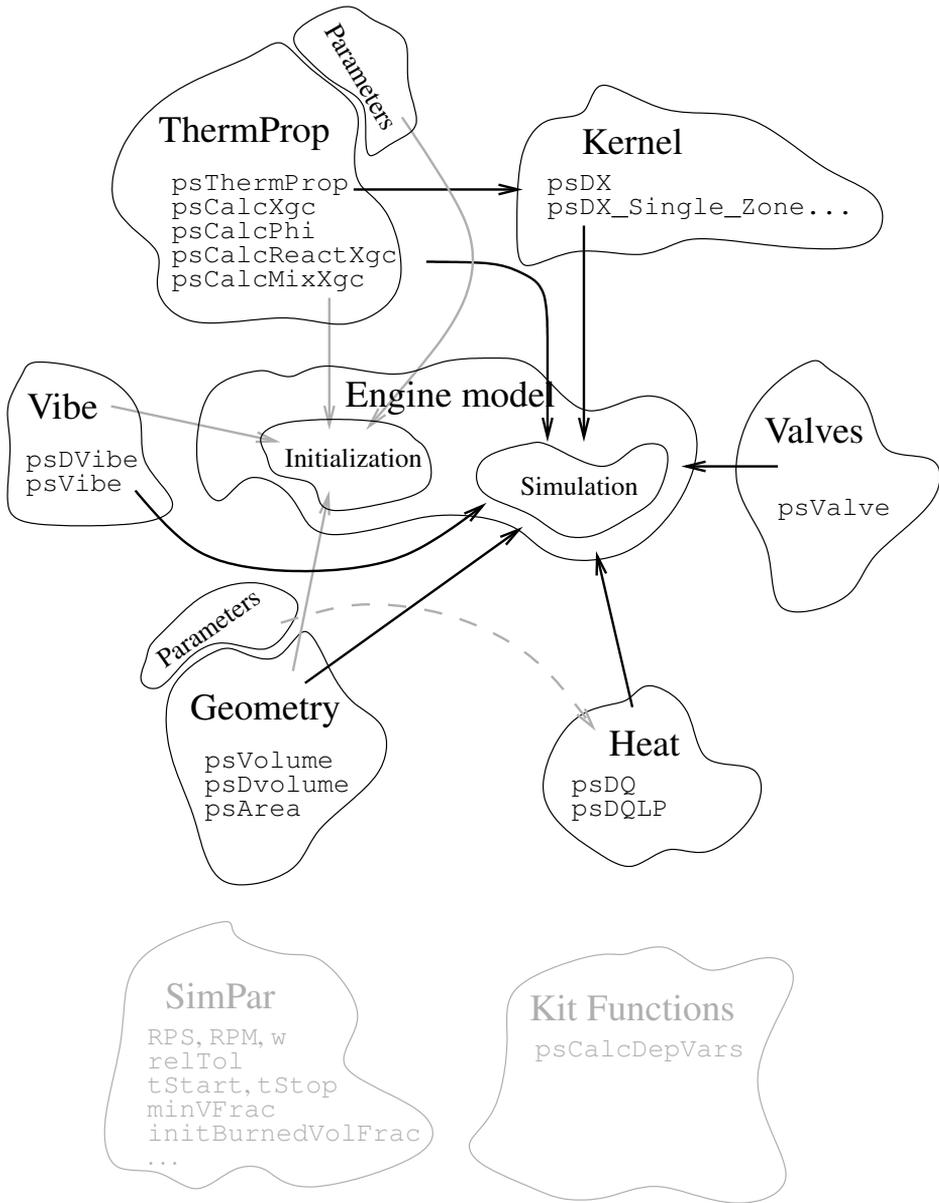


Figure 5.1: Overview of the sum-modules in PSPACK. The black arrows represents data functions that are called during the simulation while the gray arrows represent functions that are called during initialization. The dash-dotted arrow represents parameters that are needed during simulation by other modules. The gray clouds represent global parameters that do not belong to any of the modules, and functions that are common for all the modules.

5.2.2 Structure – `psSimStruct`

The first time PSPACK is started a global structure called `psSimStruct` is created. It is a collection of global variables that are used during the simulation but it also has information about these variables as well as the module they belong to. All the simulation choices are stored in `psSimStruct` which can therefore be saved at one point and reloaded later on if a simulation is to be repeated.

When PSPACK is first started a skeleton for `psSimStruct` is created. In this skeleton all modules are represented. The skeleton is shown in Figure B.1 in Appendix. When a module is initialized, the initialize flag is changed from zero to one and the module part of `psSimStruct` is filled with variables and variable information. When some module, for example `psSubModule`, is initialized the `psSimStruct` structure should be filled with variables as shown in Figure B.2 in Appendix.

The variables in PSPACK may be of the type `value`, `string`, or `table`. The variables that belongs to the `value`-class may either be free to change or can depend on other variables. Of the variables that are free to choose some makes sense to change by an optimization routine and these should therefore be listed in `optimVarNames`. The user can then use the fields `optimCtrl`, `optimLimits`, and `optimShonInfo` to turn on and of the optimization flag for the variables and to set limits for the optimization using either relative or absolute values. In this manner it is easy to create scripts that can perform parameter optimization.

5.3 Detailed Descriptions of the Modules

5.3.1 Module – psKernel – PsPACK Kernel

The `psKernel` module collects functions that have to do with the equation of state for an open or closed thermodynamic system.

The functions that are available in the kernel are

- `psDX`
- `psDX_SingleZone_ConstV`
- `psAdiabaticMix`

Function – psDX

`psDX` calculates the time derivative of the thermodynamic state variables for a closed/open system, e.g. for a cylinder with closed/open valves, with an arbitrary number of zones. An overview of the interface of `psDX` is shown in Table B.1 in Appendix. It is a straight forward implementation of (4.2) and (4.3). The equations are repeated here for reference.

$$\begin{pmatrix} 0 & 1 & 0 & \cdots & 1 & 0 \\ a_1 & p & b_1 & \cdots & 0 & 0 \\ c_1 & p & d_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_N & 0 & 0 & \cdots & p & b_N \\ c_N & 0 & 0 & \cdots & p & d_N \end{pmatrix} \begin{pmatrix} dp \\ dV_1 \\ dT_1 \\ \vdots \\ dV_N \\ dT_N \end{pmatrix} = \begin{pmatrix} dV \\ e_1 \\ f_1 \\ \vdots \\ e_N \\ f_N \end{pmatrix} \quad (4.2)$$

and

$$dx_i = \sum_j \frac{\hat{x}_{d,ij} - x_i}{m_i} dm_{ij} \quad (4.3)$$

Function – psDX_SingleZone_ConstV

`psDX_SingleZone_ConstV` is designed to be a faster implementation of `psDX` for the special case of a constant volume single zone model. This is useful when modeling for example the intake manifold but can also be used to model a series of single zone models as for example a pipe that will be discussed in Section 5.3.8. An overview of the interface to `psDX_SingleZone_ConstV` is shown in Table B.2 in Appendix.

Function – psAdiabaticMix

`psAdiabaticMix` calculates the result of the mixing of two zones. It is needed in transitions between different zone layouts, i.e. when a number of non empty burned zones are transformed to a single zone. An overview of the interface of `psAdiabaticMix` is

shown in Table B.3 in Appendix and a typical situation for an adiabatic mixing process is shown in Figure 5.2

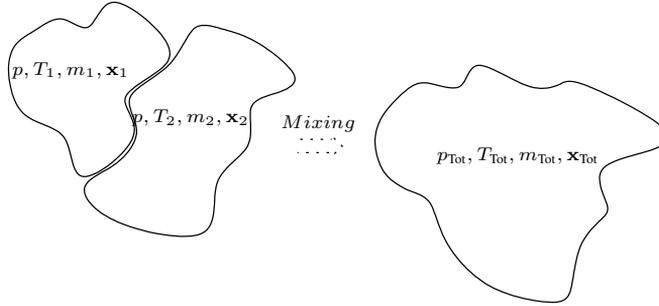


Figure 5.2: Before the mixing the two gases are thought to be separated by a barrier. When the barrier is removed it is assumed that the gases mix until both temperature and composition becomes homogenous.

During the mixing of a number of zones it is assumed that the mass and energy is conserved and that the resulting composition is homogeneous in composition and temperature. This is usually called an adiabatic mixing process because it is assumed that there is no heat transfer to the surrounding. Further, the state equation and Gibbs-Dalton's law of partial volumes is used. Gibbs-Dalton's law states that the volume of a mixture of substances is the sum of the volumes they occupied before they were mixed. Note that this ideal and instantaneous mixing process is a simplification of a real process. In particular the gases mix and may react with each other during the mixing process and therefore it is not guaranteed that the pressure will be equal before and after the mixing. The energy and mass before and after the mixing should be the same, that is

$$u_{\text{Tot}}(p_{\text{Tot}}, T_{\text{Tot}}, \mathbf{x}_{\text{Tot}}) = \sum_i u_i(p, T_i, \mathbf{x}_i) \quad (5.1)$$

$$p V_i = m_i R_i(p, T_i, \mathbf{x}_i) T_i \quad \text{and} \quad p_{\text{Tot}} V_{\text{Tot}} = m_{\text{Tot}} R_{\text{Tot}} T_{\text{Tot}}$$

where $\mathbf{x}_{\text{Tot}} = \frac{\sum_i \mathbf{x}_i m_i}{\sum_i m_i}$ and $m_{\text{Tot}} = \sum_i m_i$.

The implementation of `psAdiabaticMix` uses the Newton Rhapsod method together with a fixed point iteration for solving the non-linear equation in the following manner

1. Start with an initial guess of T_{Tot} and assume that $p_{\text{Tot}} = p$.
2. Calculate u_{Tot} , $c_{v,\text{Tot}}$ and R_{Tot}
3. Let T_{Tot} be $T_{\text{Tot}} + \frac{\sum_i u_i - u_{\text{Tot}}}{c_{v,\text{Tot}}}$, (*Newton Rhapsod part*)
4. Calculate a new p_{Tot} using $p_{\text{Tot}} = \frac{m_{\text{Tot}} R_{\text{Tot}} T_{\text{Tot}}}{V_{\text{Tot}}}$, (*fixed point part*)
5. Start over if update of T_{Tot} was large

Since $u \approx c_p T + c_{p,0}$ and the energy dependence on pressure is weak the algorithm typically converges in less than 5-6 iterations.

5.3.2 Module – psGeometry – Engine Geometry

The engine geometry information is needed for the calculation of the cylinder area and the cylinder volume as well as the derivative of the cylinder volume. Some of the geometry parameters are also used by psHeat to calculate the heat transfer.

The functions that are available in the geometry module are

- psVolume
- psDVolume
- psArea

The parameters that psGeometry need to create for other modules are listed in Table B.4 in Appendix.

Function – psVolume

This function calculates the volume of the cylinder given the crank angle and for a traditional engine depicted in Figure 5.3 we have

$$s = a \cos(\theta) + \sqrt{l^2 - (a \sin(\theta))^2}$$

$$V = V_c + \frac{\pi B^2}{4} (l + a - s)$$

An overview of the interface of psVolume is shown in Table B.5 in Appendix.

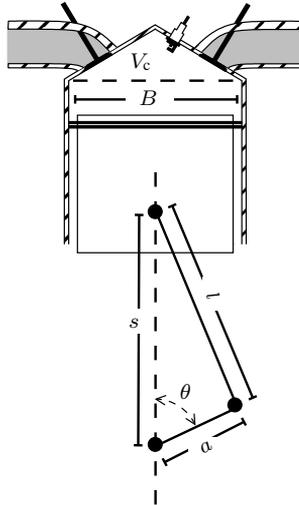


Figure 5.3: Geometry of typical standard engine. θ is crank angle, a is crank radius, B is cylinder bore, l is connecting rod length, and V_c is clearance volume

Function – psDVOLUME

This function is needed to calculate the derivative of the cylinder volume as a function of crank angle and for a traditional engine depicted in Figure 5.3 we have

$$\frac{ds}{d\theta} = -\sin(\theta) \left(a + \frac{a^2 \cos(\theta)}{\sqrt{l^2 - (a \sin(\theta))^2}} \right)$$

$$\frac{dV}{d\theta} = -\frac{\pi B^2}{4} \frac{ds}{d\theta}$$

An overview of the interface of psDVOLUME is shown in Table B.6 in Appendix.

Function – psArea

psArea is used to calculate the area of the cylinder/combustion compartment. As with psVolume and psDVOLUME it is a function of crank angle and for the engine depicted in Figure 5.3 we have

$$s = a \cos(\theta) + \sqrt{l^2 - (a \sin(\theta))^2}$$

$$A_{\text{Wall}} = \frac{\pi B^2}{2} + \pi B (l + a - s)$$

An overview of the interface of psArea is shown in Table B.7 in Appendix.

Available Implementations

The implementations that are available for the psGeometry module only differs by details in the use of variables. The available alternatives are

Generic This version uses variables that are stored in the external simulation structure psSimStruct and hence needs to look up all the global variables in every simulation step.

fixParGeometry This version stores the geometry parameters directly in locally generated files. In this way the computations can be sped up since no global variables need to be accessed during execution. The variables in Table B.4 still need to be created for other modules.

5.3.3 Module – psHeat – Heat Transfer

In the `psHeat` module, functions that calculates the heat transfer coefficient for heat transfer to and from the cylinder walls are collected. These functions are

- `psDQ`
- `psDQ1p`

Short review of heat transfer There are three types of heat transfer

- Convection
- Conduction
- Radiation

Of these, conduction and convection are the most important for a SI engine. Energy from the gases in the cylinder are mainly transferred to the wall by convection. The energy is then conducted through the cylinder walls and led away with the cooling system. In diesel engines, however, radiation may become more significant [25].

One approximation, that is used in the current implementation, is that the cylinder wall temperature is constant throughout the cycle. In that case only heat transfer through convection needs to be modeled on a cycle basis since it is assumed that all dissipated energy is conducted away to the cooling system in the same rate as it is conducted to/from the cylinder wall.

Heat transfer to and from the cylinder is formulated using Newton's law of external heat transfer, c.f. [25]

$$\dot{q} = h_c (T - T_{\text{wall}}) \quad (5.2)$$

where \dot{q} is the heat transfer per unit area, h_c is the heat transfer coefficient, T the temperature of the gas, and T_{wall} the cylinder wall temperature.

To obtain the total heat transfer it is necessary to multiply \dot{q} with the wall to gas contact area. For a single zone model the total cylinder area can be used. For a multi-zone model however, the total cylinder area has to be distributed between the zones. The choice of distribution depends on the geometry of the zones. During combustion, for example, the burned zone can be assumed to spread spherically from the spark plug and through the cylinder.

The choice of distribution can, however, have serious implications on simulation speed. In Figure 5.4 a spherical assumption is depicted. At the beginning of the combustion the flame front develops like a sphere from the spark plug. The consequence of this assumption is that the burned area $A_b = A_{\text{Tot}} \frac{1}{1 + \left(\frac{V_u}{V_b}\right)^{\frac{2}{3}}}$ which in turn implies that the area/volume ratio is $\frac{A_b}{V_b} = A_{\text{Tot}} \frac{1}{V_b + V_b^{\frac{1}{3}} V_u^{\frac{2}{3}}}$ which tends to infinity as the volume tends to zero. When the heat transfer per unit mass tends to infinity the temperature derivative will tend to infinity to and this shows that great care has to be taken to avoid making the system stiff for small zones.

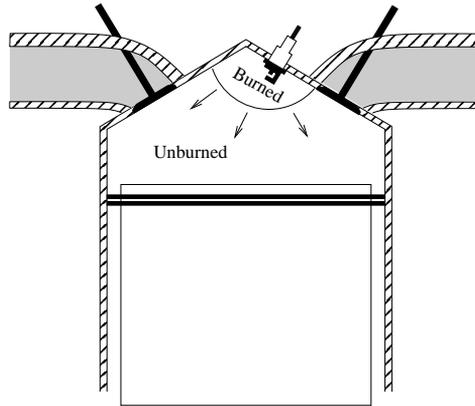


Figure 5.4: The volume distribution between the two zones of a SI engine can for example be modeled using a spherical assumption. The burned and unburned zones are then approximated as two perfect spheres.

Function – $psDQ$

Heat transfer during the high pressure part of the cycle (both intake and exhaust valves closed) is implemented in the function $psDQ$. The function implements (5.2) according to the specifications in Table B.8 in Appendix.

Function – $psDQ1p$

Heat transfer during the low pressure part of the cycle, i.e. when one or more of the inlet or outlet valves are partially open, is implemented in the function $psDQ1p$. The function implements (5.2) and an overview of the function interface is found in Table B.9 in Appendix.

Available Implementations

The heat transfer coefficient depends on a number of properties that needs to be translated to quantities like engine speed, geometry and cylinder pressure. Several publications have investigated heat transfer models, and the two most common are [53] and [26]. A thorough description of engine heat transfer, including the two different approaches, is given in [25]. The implementations that are available is discussed below.

Woschni's Heat-Transfer Correlation

In [53] the following correlation for the heat transfer coefficient is proposed

$$h = \frac{131 B^{-0.2} C_1 p_f^{0.8} w^{0.8}}{T^{0.53}}$$

with

$$w = 2.28 \left(\frac{0.0034 (p_f - p_m) T_{ivc} V_{disp} C_2}{p_{ivc} V_{ivc}} + U_p \right)$$

Where B is bore, p_f and p_m are fired and motored pressures in atmospheres, T is temperature, U_p is mean piston speed and C_1 and C_2 are constants. The subscript ivc as in V_{ivc} refers to conditions at intake valve closing.

With minor typographical modifications and conversion to SI units, as in [32], the correlation is

$$\begin{aligned} V_{char} &= C_1 \bar{V}_p + C_2 \frac{T_{ivc} V_d}{p_{ivc} V_{ivc}} (p - p_m) \\ h_c &= 0.013 B^{-0.2} (p V_{char})^{0.8} T^{-0.53} \end{aligned} \quad (5.3)$$

where V_{char} is a characteristic speed, T is the temperature of the gases, \bar{V}_p is mean piston speed, B is bore, p is pressure and p_m is motored pressure. The term motored pressure refers to the pressure that would have been obtained if the cycle would not have had any heat release. p_{ivc} , T_{ivc} and V_{ivc} are pressure temperature and volume at intake valve closure. Finally C_1 and C_2 are two tuning constants.

V_{char} in (5.3) has two contributions. The first part is the engine speed dependence and is thought to be proportional to mean piston speed while the second part represents the turbulence introduced during the combustion. Therefore $C_2 = 0$ should be used during the gas exchange and compression part, note that $p - p_m = 0$ during the compression.

Woschni found the following values to give good results

$$\left. \begin{aligned} C_1 &= 6.18 \\ C_2 &= 0 \\ C_1 &= 2.28 \\ C_2 &= 3.24 \times 10^{-3} \end{aligned} \right\} \begin{array}{l} \text{During the gas exchange period, i.e low pressure part.} \\ \text{During the compression and combustion, i.e high pressure part.} \end{array} \quad (5.4)$$

The `psHeat.C1` parameter in Table B.8 and B.9 in Appendix refers to $\frac{C_1}{2.28}$ or $\frac{C_1}{6.18}$ respectively so that a common parameter for both cases can be used.

Hohenberg's Heat-Transfer Correlation

In [26] Woschni's heat-transfer is reviewed and the following form is suggested

$$h = C_1 V^{-0.06} p^{0.8} T^{-0.4} (U_p + C_2)^{0.8} \quad (5.5)$$

where V is Volume, p_f is fired pressure in bar, T is gas temperature U_p is mean piston speed and C_1 , C_2 are constant parameters.

5.3.4 Module – psVibe – Burned Mass Fraction

As discussed earlier heat release can be modeled as a mass transport from a zone with unburned gases to a zone with burned gases. The burned mass fraction and its derivative are required during the simulation and the functions to calculate these quantities are collected in the `psVibe` module. The functions are

- `psVibe`
- `psDVibe`
- `psVibeEvents`

Function – psVibe

The burned mass fraction is implemented in the `psVibe` function. The interface of this function is specified in Table B.10 in Appendix.

Function – psDVibe

The derivative of the burned mass fraction is implemented in the `psDVibe` function. The interface of this function can be found in Table B.12 in Appendix.

Function – psVibeEvents

The purpose of the `psVibeEvents` function is to return a list of crank angles for start of combustion (SOC) as well as end of combustion (EOC). The function specification is shown in Table B.11 in Appendix.

Available Implementations

For the rate of heat release the Vibe [51] function is used. The equation describing the mass fraction of burned gases, x_B , is

$$x_B = \eta \left(1 - e^{-a \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^{m+1}} \right) \quad (5.6)$$

where θ is the crank angle, θ_s and θ_e are start of combustion and end of combustion, a and m are shaping parameters while η is the combustion efficiency.

Sometimes it is necessary to have the derivative of the heat release function. In its differentiated form (5.6) becomes

$$\frac{dx_B}{d\theta} = \eta \frac{a(m+1)}{\theta_e - \theta_s} \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^m e^{-a \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^{m+1}} = \frac{a(m+1)}{\theta_e - \theta_s} \left(\frac{\theta - \theta_s}{\theta_e - \theta_s} \right)^m (\eta - x_B)$$

where the last equality uses the definition of x_B itself to obtain a version that depends on the cylinder state as discussed in Example 4.2.

The implementations that are available for the `psvibe` module only differs by details in the use of variables. There are two implementations of the Vibe mass fraction burned function

Generic This version uses variables that are stored in the global simulation structure and hence needs to look up all the global variables in each simulation step. Either the parameters `thetaS` and `thetaE` are specified or the parameters `Phasing` and `Duration`.

fixParVibe This version stores the burned mass fraction parameters directly in locally generated files. In this way the computation can be sped up since no global variables need to be accessed during execution.

5.3.5 Module – `psSimPar` – Simulation Parameters

`psSimPar` is a module that collects simulation parameters that are necessary for the simulation but that are not model specific and that do not belong to other modules. An overview of the parameters that are available are listed in Table B.13 in Appendix.

5.3.6 Module – `psThermProp` – Thermochemical Properties

When simulating of an open or closed thermodynamic system thermochemical information about the medium, i.e. the gas, is needed. As shown in Section 2.7 the following properties are needed to be able to simulate the model in (2.20) and (4.2).

$$\left. \begin{array}{l} c_p \quad c_v \quad R \quad u/h \\ \frac{dR}{dp} \quad \frac{dR}{dT} \quad \frac{\partial u}{\partial p} \end{array} \right\} \quad \text{Scalar properties}$$

$$\left. \begin{array}{l} \nabla_{\mathbf{x}_r} u \quad \nabla_{\mathbf{x}_r} R \end{array} \right\} \quad \text{Vector properties}$$

The `psThermProp` module implements functions for calculating thermochemical properties that are needed during the simulation but also functions to support setting up a simulation and to convert from one representation to another. The following functions are available

```
-psThermProp
-psCalcMixXgc
-psCalcReactXgc
-psCalcPhi
-psCalcXgc
```

Function – `psThermProp`

The most important function in the module is the `psThermProp` function because it implements the calculation of thermochemical properties as discussed in Chapter 3. It is

a straight forward implementation of the equations for the different models. The requirements for the function are listed in Table B.15 in Appendix.

Function – `psCalcMixXgc`

The purpose of the `psCalcMixXgc` function is to calculate the composition that a given gas has when it is transported *into* a mixer. If for example a gas is transported from a reactor to a mixer the composition has to be changed to fit the description of the destination. The interface of the function is shown in Table B.16 in Appendix.

Function – `psCalcReactXgc`

The purpose of the `psCalcReactXgc` function is to calculate the composition that a given gas has when it is transported *in to* a reactor. If for example a gas is transported from a mixer to a reactor the composition has to be changed to fit the description of the destination. An overview of the specification for the function is shown in Table B.17 in Appendix.

Function – `psCalcPhi`

The function `psCalcPhi` is used to convert the internal representation of the gas to a fuel air equivalence ratio. The interface of the function is specified in Table B.18 in Appendix.

Function – `psCalcXgc`

The `psCalcXgc` function is used to calculate an internal composition parameter representation of a gas with a certain fuel/air equivalence ratio. In Table B.19 the interface of the function is specified.

Available Implementations

Chepp and CheppFull The `CheppFull` and `Chepp` models implement the models discussed in Sections 3.3 and 3.4.2 respectively.

Tables The `Tables` model implements the model discussed in Section 3.4.3. The model includes functionality for calculating tabular data from CHEPP. Parameters specific to the `Tables` model are listed in Table B.20 in Appendix.

SimpleEq The `SimpleEq` model is an implementation of the model discussed in 3.4.4 and the parameters specific to the mode is listed in Table B.21 in Appendix.

5.3.7 Module – psValve – Valve Areas

The `psValve` module contains functions for the calculation of the effective flow area of the intake and outlet valves for a given crank angle θ . The functions that are required during simulation are `psValve` and `psValveEvents` but the look-up table alternative also includes functions for off-line calculations of poppet valve geometries. The required functions are

- `psValve`
- `psValveEvents`

Function – psValve

The purpose of the `psValve` function is to calculate the relative effective flow areas of the inlet and outlet valves. The requirements for the function are specified in Table B.22 in Appendix. The relative effective flow area should be multiplied with the maximum effective flow area of the valve to obtain the actual effective flow area. The effective flow area as well as the discharge coefficients of the valves should be given by the `psGeometry` module.

Available Implementations

fixParValve This version uses functions built from cosine bumps and stores the parameters directly in the locally generated files. In this way the computation can be sped up since no global variables need to be accessed during execution. The cosine bumps are built according to the following scheme

$$u_{IV} = \begin{cases} 0 & \theta < \theta_{IVO} \\ 1 - \cos^2 \left(\frac{\pi}{2} \frac{\theta - \theta_{IVO}}{\theta_{DIV}} \right) & \theta \geq \theta_{IVO} & , \theta < \theta_{IVO} + \theta_{DIV} \\ 1 & \theta \geq \theta_{IVO} + \theta_{DIV} & , \theta < \theta_{IVC} - \theta_{DIV} \\ 1 - \cos^2 \left(\frac{\pi}{2} \frac{\theta - (\theta_{IVC} - \theta_{DIV})}{\theta_{DIV}} \right) & \theta \geq \theta_{IVC} - \theta_{DIV} & , \theta < \theta_{IVC} \\ 0 & \theta > \theta_{IVC} \end{cases}$$

Generic This version uses functions built upon cosine bumps but uses global parameters to calculate the opening areas.

LookupTable This version uses lookup tables to make linear interpolation of valve lift.

CVCPLookupTable This version implements a possibility to change the inlet and outlet valves maximum opening positions **during a cycle**, as would happen during a transient of an engine with a variable valve timing mechanism.

Geometry of Poppet Valves

Usually the valve lifts as a function of crank angle is known, either by measurements or by specification. In [25] it is shown that the minimal flow area, i.e the area that is governing

the flow, of a poppet valve can be calculated using the following equation

$$A_m = \begin{cases} \pi L_v \cos(\beta) (D_v - 2w + \frac{L_v}{2} \sin(2\beta)) & \text{if } 0 < L_v < \frac{w}{\sin(\beta) \cos \beta} \\ \pi D_m \sqrt{(L_v - w \tan(\beta))^2 + w^2} & \text{if } \frac{w}{\sin(\beta) \cos \beta} \leq L_v < w \tan \beta + \\ & + \sqrt{\left(\frac{D_p^2 - D_s^2}{4D_m}\right)^2 - w^2} \\ \frac{\pi}{4} (D_p^2 - D_s^2) & \text{if } w \tan \beta + \sqrt{\left(\frac{D_p^2 - D_s^2}{4D_m}\right)^2 - w^2} < L_v \end{cases}$$

where L_v is the valve lift, D_v is the head diameter, D_p is port diameter, D_s is stem diameter, D_m is the mean seat diameter w is the seat width and β is the seat angle. A thorough discussion as well as graphical illustrations can be found in [25].

Function – psValveEvents

The purpose of the `psValveEvents` function is to return a list of crank angles for exhaust and intake valve opening, EVO/IVO as well as exhaust and intake valve closure, EVC/IVC. The function specification is shown in Table B.23 in Appendix.

5.3.8 Frequently Used Model Components

The modules of PSPACK are focused on models which there are alternative choices for. Therefore some models, that are frequently used, have not yet made it into the framework even if it would make sense to reuse these components. Some of these frequently used components are described here.

Restrictions

From a thermodynamic viewpoint a restriction is something that governs the flow between two thermodynamic systems. Components where restrictions come in handy are

- Throttle
- Inlet and exhaust valves
- Turbine and waste gate
- Pipes

A frequently encountered approach for the modeling of a restriction is to assume that the fluid is compressible and has zero initial velocity before the restriction. These assumptions together with the assumption that the process is without heat transfer and reversible yields the following isentropic flow equation

$$\dot{m} = c_d A \frac{p_{in}}{\sqrt{RT_{in}}} \Psi(\Pi) \quad (5.7)$$

where \dot{m} is the mass flow through the valve, c_d is the discharge coefficient, A is the area of the flow-restriction, Ψ is the flow function, $\Pi = \frac{p_{out}}{p_{in}}$ is the pressure ratio over the

restriction and finally p_{in} and p_{out} are the pressures before and after the restriction in the flow direction. The flow function is

$$\Psi = \begin{cases} \sqrt{\gamma \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}} & \text{if } \Pi < \frac{2}{\gamma+1} \frac{\gamma}{\gamma-1} \\ \Pi^{\frac{1}{\gamma}} \sqrt{\frac{2\gamma}{\gamma-1} \left(1 - \Pi^{\frac{\gamma-1}{\gamma}}\right)} & \text{otherwise} \end{cases} \quad (5.8)$$

where γ is the ratio of specific heats. The first part of the equation represents choked flow, where the flow is limited by the speed of sound.

In some cases an approximate model where standard analytical integration rules can be applied is sought for. Factors of the type $\Pi^{\frac{2}{\gamma}}$ makes this analytical analysis cumbersome. Therefore simpler models may have to be considered. One such model, c.f. [21], is

$$\Psi(\Pi) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } \Pi < 0.5, \\ \sqrt{2\Pi(1-\Pi)} & \text{for } \Pi \geq 0.5. \end{cases} \quad (5.9)$$

As can be seen in Figure 5.5 setting $\gamma = 1.5385$ has about the same effect as using the simple model in (5.9).

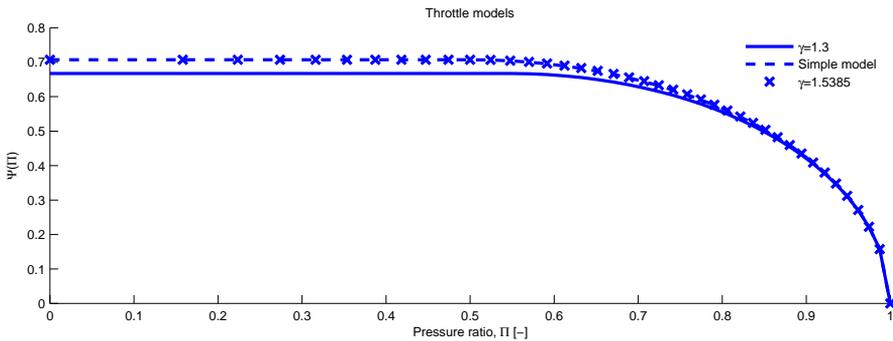


Figure 5.5: Comparison of Ψ function values for different models. The simple model gives about the same result as setting $\gamma = 1.5385$

Turbine restriction When modeling a turbine the standard equation for isentropic flow in (5.7) does not give accurate results. Different strategies to correct this problem have been proposed and one of the more appealing ones tries to explain the difference by studying a particle on its way through the turbine. This approach is used in [14] where it is stated that a particle that passes the turbine will be accelerated both in the stator and in the rotor of the turbine. It will then be the component with the largest pressure difference that governs the flow. If the pressure ratio over the stator and rotor are the same, i.e.

$$\Pi_t = \underbrace{\sqrt{\Pi_t}}_{\text{stator}} \cdot \underbrace{\sqrt{\Pi_t}}_{\text{rotor}} \quad (5.10)$$

Π should be replaced with $\sqrt{\Pi}$ in (5.7). This is shown to give good results for a turbine with closed waste gate [14].

Linearization A common problem when simulating pressure-driven flow systems is that the derivative of the flow function reaches infinity as the pressure ratio increases towards unity. This can cause a fixed step solver to become unstable at this point and a variable step solver to reduce the step size to a crawl thus crippling the simulation. A solution to this problem is to linearize the flow function for pressure ratios larger than a certain limit. A common choice is

$$\Psi_{\text{eff}}(\Pi) = \begin{cases} \Psi(\Pi) & \text{if } \Pi < \Pi_l \\ \frac{\Psi(\Pi_l) \cdot (\Pi - 1)}{(\Pi_l - 1)} & \text{otherwise} \end{cases} \quad (5.11)$$

where $\Pi_l < 1$ is the linearization limit. Π_l has a direct impact on the steady state pressure ratio over the restriction, for example during the intake stroke. Generally a higher Π_l gives a more accurate simulation but shorter step-lengths, Π_l should therefore be chosen carefully.

Pipe Dynamics

Most of the models used for the engine are zero dimensional models. However, to be able to capture the pipe dynamics one dimensional models are introduced. Since most of the dynamic effects in a pipe are in the flow direction, a one dimensional model can be sufficient depending on what accuracy is needed. There are commercially available software that can for example be used to model flow through intake and exhaust pipes using 1-dimensional models, e.g. [48] and [50], these are, however not used in this presentation.

The traditional way of modeling pipes is to use an energy preservation, momentum preservation and mass preservation equation and solve them using the method of characteristics or a finite difference method. These approaches are for example illustrated and discussed in [25].

In the DAE framework, mass and energy preservation is already taken care of. Therefore a slightly different approach is adopted here. The pipe is divided into a configurable number of sections that all have their own pressure, temperature and mixture. These sections are then treated as regular control volumes. The idea is shown in Figure 5.6 below. All boundaries are then associated with a plug-flow which defines the mass transfer between the sections. The velocity of the plug is modeled using Newtonian physics. In this way the ram effects in the ducts can be captured. The speed of the plugs are modeled using the following equation

$$\frac{dU_i}{dt} = \frac{2A(p_i - p_{i+1})}{(m_i + m_{i+1})} - U_i k_{Frict} \quad (5.12)$$

where p_i and m_i are the pressure and mass of section i and k_{Frict} is the friction coefficient while A is the area of the pipe. Because the residual mass fraction of each section is tracked by the simple bookkeeping in the control volumes diffusion of burned gases into the intake manifold is handled in a simple manner.

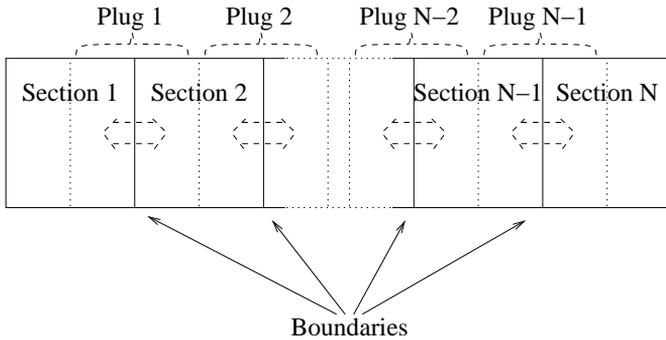


Figure 5.6: Pipe model with N sections and $N-1$ flowing plugs.

Note that it is assumed that all equations for ordinary control volumes are still valid for each section individually. The energy from the moving plugs are not linked to the internal energy of the gas in any way. The plugs simply act as flow governors. A similar approach is used in [30] where also artificial damping terms as well as the kinetic energy is considered.

If the pipe is connected to a traditional restriction, the properties of the last section of the pipe is used as input to this restriction. If, however, the pipe is connected to another control volume a part as large as the last section of the pipe of this control volume is assumed to be a part of the pipe. The flow between the pipe and the control volume is then modeled using the same strategy as for the rest of the sections.

In Figure 5.7 a simulation where the intake pipe parameters are matched to measured data from a modified SAAB/GM Ecotec-L850 engine is shown. For the simulation the model `ps4CylSIModel`, which will be described in Section 5.3.9, has been used. The amplitude, damping and wave length has a quite good agreement for the period after IVO.

The parameters for the pipe model are the number of sections, the pipe length, and the pipe area. In `ps4CylSIModel` the pipe model is used to model the intake and exhaust runners as well as the volumes of the intake and exhaust manifolds. For the runners 2 sections are used while for the manifolds only 1 section is used. In this way there are 3 plugs in the combined manifold/runner model. In Figure 5.8 the pipe lengths and areas of the intake and exhaust runners as well as the manifolds are doubled to illustrate the effect that these parameters have on the simulation result. The models are also simulated with and without friction and with 5 sections instead of 2 to illustrate the effect that these parameters have.

5.3.9 Module – `psModels` – Cylinder/Engine Models

A model is needed to put together an engine of the available components. A model in `psModel` may require different modules and can have different zones-layouts etc. For example two zones for gas exchange and one for compression may be used. The user can

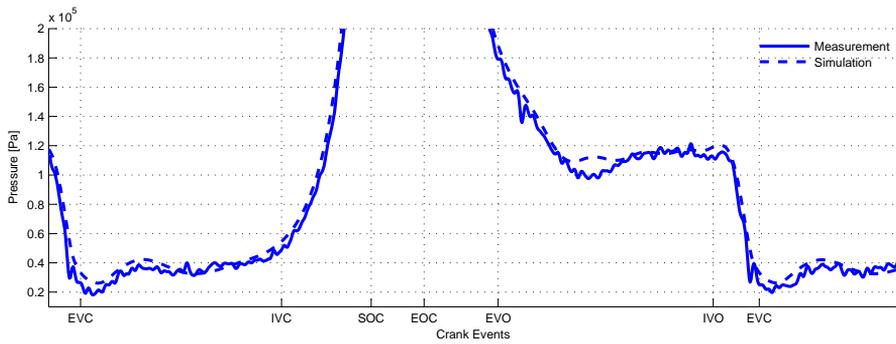


Figure 5.7: Simulation of `ps4CylSIModel`. The intake pipes parameters have been tuned to give a god agreement for the cylinder pressure during the intake stroke. The engine is operating at 4000 rpm and 120 mg of air per combustion. Intake valve maximum opening position is 475 deg and exhaust valve maximum opening position 250 deg. Note that the intake valve opening and closure events, IVO and IVC as well as the exhaust valve opening and closure events, EVO and EVC refer to the crank angles when the valves are completely closed or just about to open. This is why the pressure does not drop immediately after IVO.

choose how to implement a model and what solver to use for simulation as long as the following functions exist

- `psModelSim`
- `psConvertSimOutput`

Function – `psModelSim`

The purpose of the `psModelSim` function is to perform a simulation according to the selected set-up. The output can then be passed to `psConvertSimOutput` to obtain the simulation results on a standardized form. `psModelSim` does not require any arguments.

Function – `psConvertSimOutput`

When a simulation has been performed the `psConvertSimOutput` can be used to convert the simulation data to a predefined form. The interface of the function is specified in Table B.24 in Appendix.

Available Implementations

The model framework has been tested in a number of set-ups. It has been used to model full cycle simulations with 1D flows through intake and exhaust ports as well as closed cycle simulations where only the part between intake valve closure and exhaust valve

opening are simulated. The framework has been used to simulate both a spark ignited engine as well as a diesel engine.

The SI-engine model is used in the publications that make up Chapter 7 and Chapter 8 while the diesel engine model is discussed in Section 6.4.

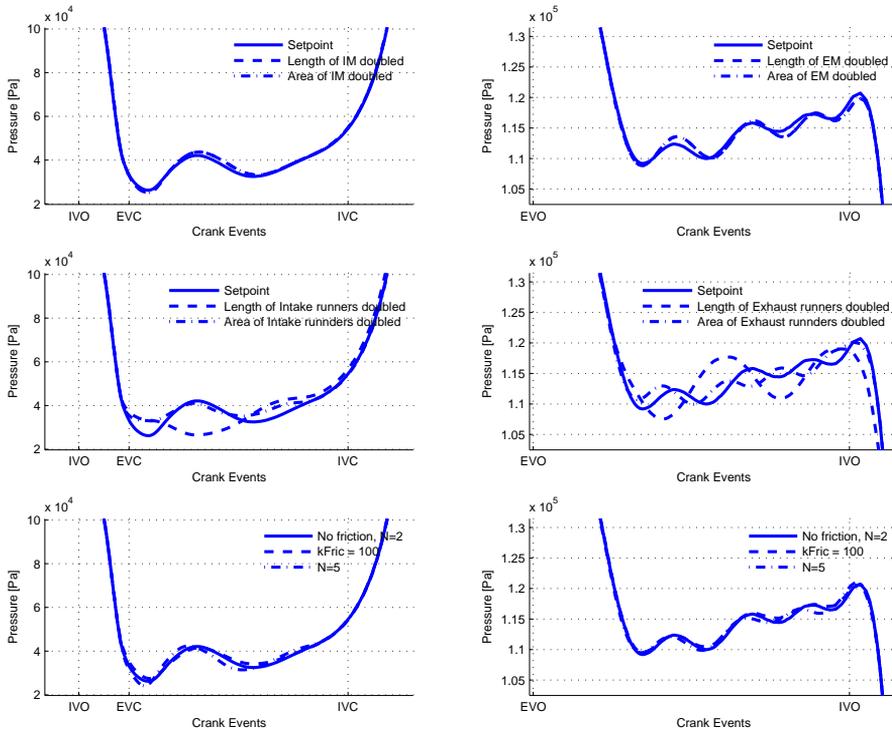


Figure 5.8: Pressures during the gas exchange process when different parameters are adjusted. The manifolds effect on the pressure is quite small both for the intake and exhaust manifolds. The length and the area of the runners does, however, have a large effect on the pressure trace. Elongating the pipe seems to decrease the wave-frequency while increasing the area seems to lower the amplitude. The friction and the number of sections have minor effects on the pressure trace. The friction, however, seems to dampen the oscillation slightly as expected.

Model – psClosedCylSIModel In `psClosedCylSIModel` a SIMULINK m-s-function, `ps2ZoneCylinder`, is used to implement a two-zone SI engine cylinder. The zone layout is depicted in Figure 5.9 and the parameters that are available for this model are listed in Table B.25 in the Appendix.

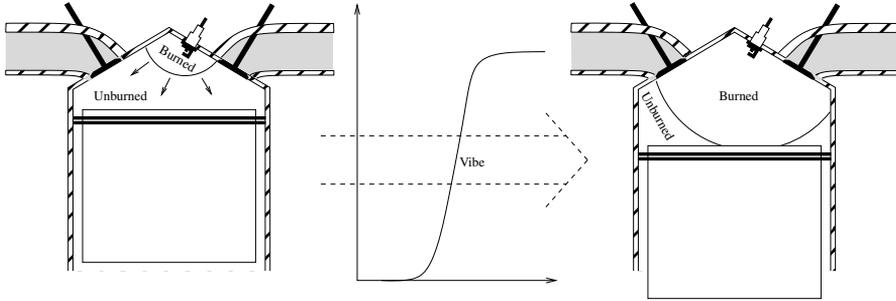


Figure 5.9: `psClosedCylSIModel` uses a 2-zone layout with a burned and an unburned fraction. A Vibe function is used to transport mass between the two zones.

The models that `psClosedCylSIModel` use are

- `ps2ZoneCylinder` which implements a cylinder with a 2-zone layout.
- `psToWorkspace` which is a model that collects the output from the simulations in a large structure.
- `psCycleSelect` which is a variable step system that makes SIMULINK take time steps at specific events, such as start of combustion (SOC).

Except for the dynamic models the following standalone functions are used by the model

- `psInitCombustion` which initializes the combustion part of the cycle using specified initial values.
- `psGetEvents` which collects events from `psVibeEvents` to get a list of events.

Except for the zone-layout the only implementation choice is how to distribute the heat transfer between the zones, i.e. their geometrical properties. A natural idea is to divide the cylinder wall area into two pieces as if the two zones were two perfect spheres. To avoid making the system stiff the heat transfer is divided between the two zones so that $dQ_b = A_{Tot} x_b h_{ht} (T_b - T_w)$ and $dQ_u = A_{Tot} (1 - x_b) h_{ht} (T_u - T_w)$ instead of the spherical assumption. The difference between the spherical approach and the linear approach is depicted in Figure 5.10.

As can be seen in the figure the heat transfer per mass unit tends to infinity as the zone become small for the spherical assumption while the linear model does not. The difference in total heat transfer is small and considering that the true wall contact area is unknown the model with the least numerical issues is the preferable choice.

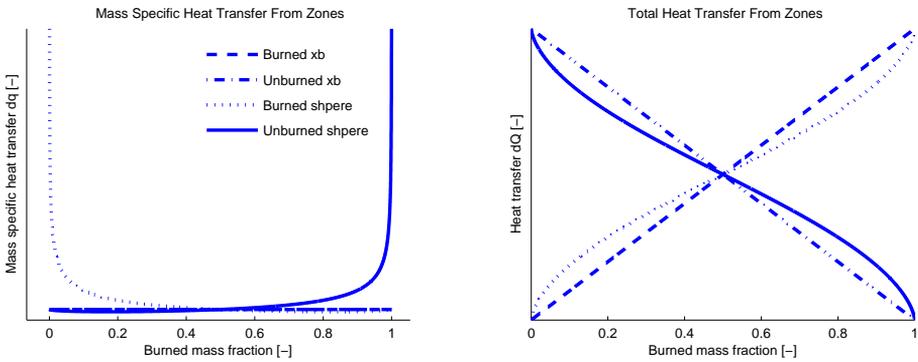


Figure 5.10: When the zones become small the mass specific heat transfer tends to infinity for the spherical assumption model. Scaling with x_b instead gives a better behaviour with a reasonable difference.

Model – ps4CylSIModel `ps4CylSIModel` models the temperatures and pressures of a complete engine, i.e. from throttle to exhaust manifold. The model uses the pipe model and restrictions from Section 5.3.8 together with an open 1-zone cylinder model for gas exchange, as well as the 2-zone model `psClosedCylSIModel` during the combustion. The model layout is shown in Figure 5.11 and a complete list of parameters for the model can be found in B.26 in Appendix.

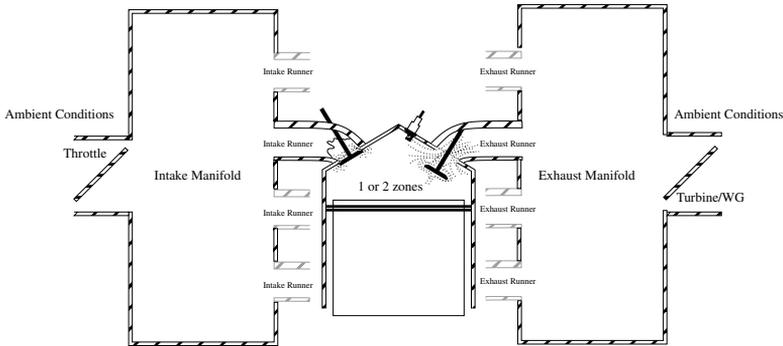


Figure 5.11: In `ps4CylSIModel` four cylinders are connected to an intake and exhaust manifold using pipes. The manifolds are pipes with one single section while the runners are multi-section pipes. The throttle and the turbine waste gate are modeled using standard isentropic flow equations in (5.8) while the turbine uses the model in (5.10).

The models that `ps4CylSIModel` use are

- `ps1ZoneOpenCylinder` which models a cylinder using a 1-zone layout. The control volume can be open which represents flow to and from the intake and exhaust runners.
- `ps2ZoneCylinder` which models a closed cylinder using a 2-zone layout. The 2-zone layout is the same model as in `psClosedCylSIModel`.
- `psMultiGasPipe` which is an implementation of the pipe model discussed in Section 5.3.8. As mentioned earlier this model is used both for the manifolds and the intake and exhaust runners.
- `psCycleSelect` which is a variable step system that makes SIMULINK take time steps at specific events, such as start of combustion (SOC) or intake valve closure (IVC).
- `psToWorkspace` which is a model that collects the output from the simulations in a large structure.

Except for the actual dynamic models the the following standalone functions are used by the model

- `psInitGasExchange` which initializes the gas exchange phase using either specified initial values or values from the combustion.

- `psInitCombustion` which initializes the combustion part of the cycle using either specified initial values or values from the gas exchange phase.
- `psStdRestr` which implements the equations from (5.9)
- `psTurbineRestr` which implements the equations from (5.10)
- `psGetEvents` which collects events from `psValveEvents` and `psVibeEvents` to get a complete list of events for a cycle

The transition from a 1-zone model to a 2-zone model is performed using *enabled* sub-systems in SIMULINK. An illustration of the model setup is shown in Figure 5.12. To make SIMULINK switch between the two zone layouts at a specified time the `psCycleSelect` is used to enable the correct sub-system.

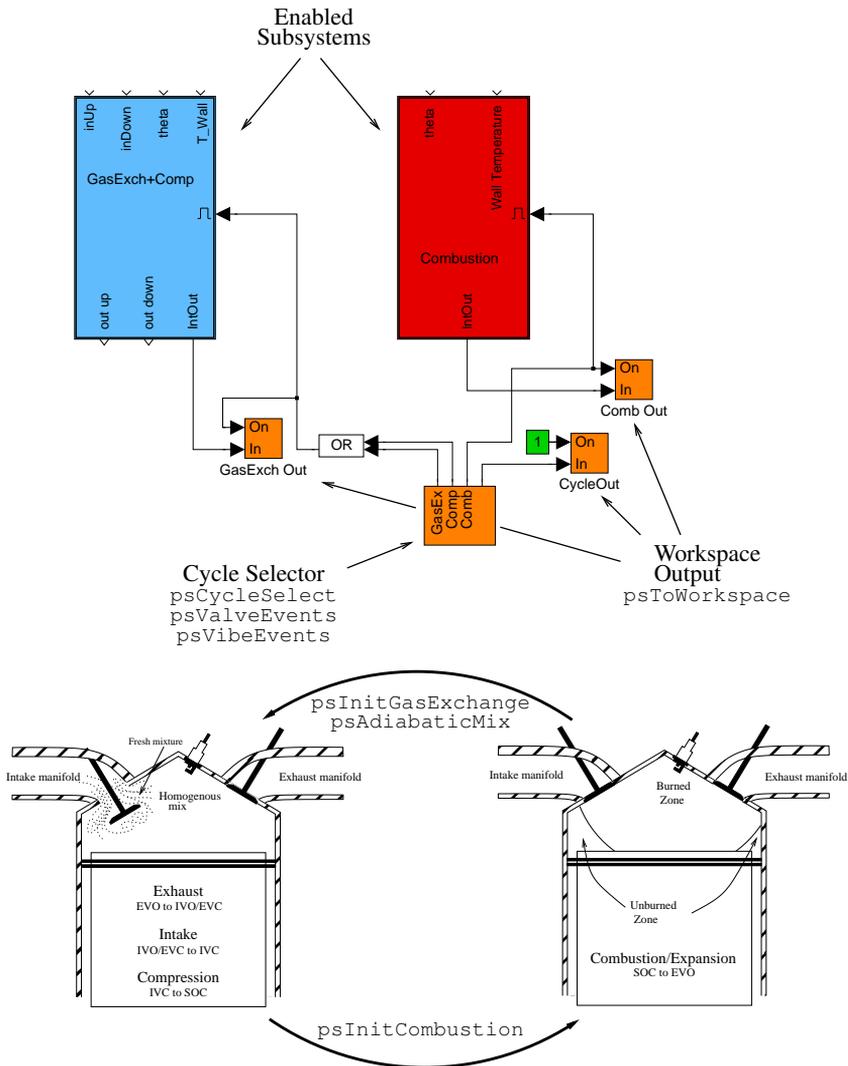


Figure 5.12: `ps4CylSIModel` switches between a 1-zone layout and a 2-zone layout during the four strokes of the engine. During exhaust, intake and compression a 1-zone model is used and during combustion a 2-zone model is used. The 2-zone model is the same as in `psClosedCylSIModel`. The 1 and 2-zone models are implemented in two SIMULINK *m-s*-functions, `ps1ZoneOpenCylinder` and `ps2ZoneCylinder`. The *s*-function blocks are stored inside two enabled subsystems and another *m-s*-function, `psCycleSelect`, with variable sample time is used to force SIMULINK to take time-steps at the exact time when the switch between the systems should take place. Because an adiabatic mixing of the two zones is used to initialize the gas-exchange the end pressure from the 2-zone model can be different from the start pressure of the 1-zone model at the same time-step. To be able to obtain both these pressures as output a replacement of the SIMULINK `ToWorkspace`-block, `psToWorkspace`, was constructed.

Part II

Applications of Theory

6

Modeling of SI and Diesel Engines

This chapter discusses the application of the DAE framework to the analysis of engine processes. The applications have also been used to drive the development of the framework and its implementation, that is now presented in Chapters 2–5. Different versions of the framework have been used in the two publications [38, 39] that are included as Chapter 7 and Chapter 8 respectively and also in two masters theses [28, 24].

For the simulation in Chapter 7 the fuel/air ratio or residual mass fraction for a zone were not allowed to change during its lifespan. The framework handled switching between two representations using adiabatic mixing, and almost empty zones were considered as empty in that no mass flow from the zone was allowed. This early version serves as a validation of the following features

- Multi-zone control volumes with any number of zones.
- Initialization and emptying of zones
- Adiabatic mixing of zones and switching between representations

As a consequence of not being able to change the fraction of residuals, a plug flow model for keeping track of the back flow into the intake manifold was needed. In the enhanced implementation that was used in Chapter 8 the residual mass fraction of the zones was allowed to change, which simplified the bookkeeping for the back flow. This enabled usage of one-dimensional pipes with a natural diffusion of residual gases into the intake manifold. The main feature that was added and validated is thus

- Control volume with variable mixture of fresh gases and residuals

To support the modeling of Diesel combustion in [28] the change of fuel/air ratio had to be added to the framework. This version of the framework is also used in [24] in which a sensitivity study of the `ps4Cyl1SIModel` from Chapter 5 is performed. In [28] the liquid fuel zone was treated separately but as will be shown in this chapter it can easily be

added to the DAE framework. The features that are added and validated are thus

- Control volume with variable fuel/air ratio
- Liquid zone

6.1 Outline of the Chapter

In Section 6.2, **Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines**, an overview to the previously published paper [38], that is included as Chapter 7, is given and the differences between the model in Chapter 5 and the model in the paper are discussed.

In Section 6.3, **Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines**, an overview to the previously published paper [39], that is included as Chapter 8, is given and the differences between the model in Chapter 5 and the model in the paper are discussed.

In Section 6.4, **Diesel Engine Modeling**, a diesel engine model that uses the framework is discussed. Simulation results from the model are used to show that the framework copes well with the changes in fuel/air ratio that are necessary when modeling diesel combustion.

6.2 Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines

In Chapter 7, [38], an early version of the implementation in Chapter 5 is used to evaluate three control oriented models that predict cylinder air charge and/or residual mass fraction, x_{RG} . The three models, presented in [15, 43, 35], are investigated and they all predict residual mass fraction. The model in [35] also predicts air charge.

The reference model used for the evaluation in Chapter 7 differs from the model in Chapter 5 in that it is assumed that the intake and exhaust manifolds are reservoirs with constant pressure and temperature and that the model does not allow the stoichiometric fuel/air ratio, ϕ , or the residual mass fraction, x_{RG} , of a zone to change during its lifespan. The back flow into the intake manifold during the valve overlap is therefore handled by integrating the mass flow and a switch of composition of the flowing gas instead of the simple diffusion in the pipes that were introduced in Chapter 5.

Another difference is the zone layout. The engine cycle is divided into three phases: gas exchange, compression, and combustion. During the gas exchange phase two zones are used instead of one single zone. The two zones in the gas exchange phase are used to separate the burned gas from the fresh mixture. The reason for this choice is the fact that the model does not allow the residual mass fraction, x_{RG} , of a zone to change during its lifespan. Another motive is that it makes the bookkeeping of the cylinder content straight forward. When entering the compression phase the simulation is stopped and the gases are mixed into one zone using an adiabatic mixing process before it is restarted again. In Figure 6.1 the bookkeeping of residual gas and fresh mixture is illustrated and explained.

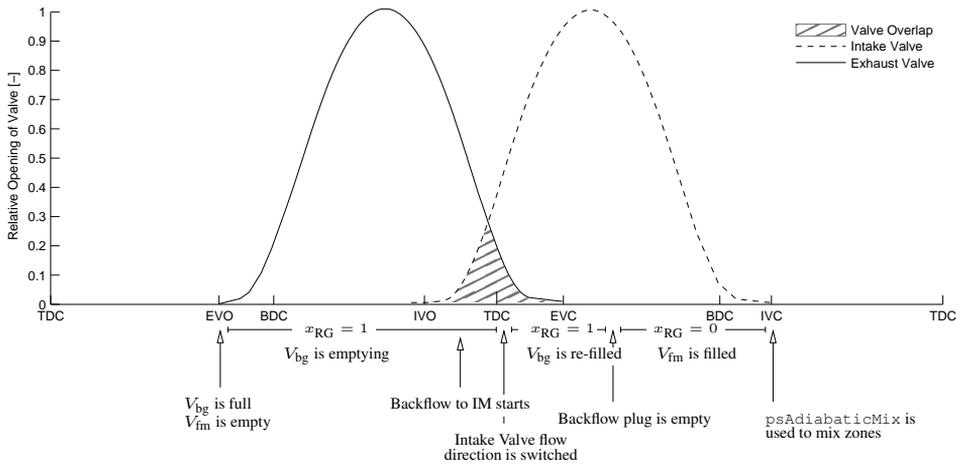


Figure 6.1: Zone filling and emptying during the gas exchange phase. At exhaust valve opening (EVO) the fresh mixture zone is initialized to having zero volume, $V_{fm} = 0$, while all gas that is left in the cylinder after the combustion, burned and unburned, are mixed and put in the burned gas zone, $V_{bg} = V_{Tot}$. During the exhaust stroke the gas from the burned gas zone, with $x_{GR} = 1$, flows out through the exhaust valve. At intake valve opening (IVO) back flow into the intake manifold starts. The temperature at this point is registered as T_{bf} and the total amount of back flow is integrated. When the flow direction is changed this temperature is used as the temperature of the flowing gas until the intake manifold is emptied of burned gas and the fresh mixture, with $x_{RG} = 0$, is ready to flow into the cylinder.

The only difference between the two-zone approach in [38] and using a single zone layout is that two separate temperatures are available instead of just one mean gas temperature. The difference in zone layout during the gas exchange phase is, however, not as significant as the assumption that the manifold pressures are constant. This is due to the fact that the gases are relatively cold during the gas exchange phase which implies that the difference in thermochemical properties is small when the two temperatures of the burned gas and the fresh mixture are used instead of one single mean gas temperature.

The parameters in the model are either kept constant for all operating points or taken from measurements in the specific operating point. Parameters that are related to the geometry of the engine are provided by the engine manufacturer while parameters that model flow efficiencies, heat transfer and cylinder wall temperature are tuned manually to values that are used for all operating points. The only parameters that are free to be optimized for each operating point are the parameters for the Vibe combustion function.

6.2.1 Gas Model

For the gas model that is used, the specific heat values, c_p and c_v , includes dissociation but the specific gas constant, R , depends only on ϕ . The model is

$$\begin{aligned}
 c_{p,u}(T) &= \alpha_3 T \phi + \alpha_2 T + \alpha_1 \phi + \alpha_0 \\
 h_u(T, \phi) &= \alpha_3 \frac{T^2}{2} \phi + \alpha_2 \frac{T^2}{2} + \alpha_1 \phi + \alpha_0 T - \\
 &\quad - 1.1082 \cdot 10^5 \cdot \phi - 3.045 \cdot 10^5 \\
 R_u &= 274 \\
 c_{p,b}(T) &= \beta_5 T^5 + \beta_4 T^4 + \beta_3 T^3 + \beta_2 T^2 + \beta_1 T + \beta_0 \quad (6.1) \\
 h_b(T, \phi) &= \beta_5 \frac{T^6}{6} + \beta_4 \frac{T^5}{5} + \beta_3 \frac{T^4}{4} + \beta_2 \frac{T^3}{3} + \beta_1 \frac{T^2}{2} + \beta_0 T + \\
 &\quad + \max(2.84 \cdot 10^6 (1 - \phi), 0.766 \cdot 10^6 (\phi - 1)) - 3.228 \cdot 10^6 \\
 R_b(\phi) &= 290 + \max(0, 56 (\phi - 1))
 \end{aligned}$$

$$\begin{aligned}
 c_p &= c_{p,u}(1 - x_{RG}) + c_{p,b} x_{RG} \\
 h &= h_u(1 - x_{RG}) + h_b x_{RG} \\
 R &= R_u(1 - x_{RG}) + R_b x_{RG}
 \end{aligned}$$

where the parameters of the model is fitted to data from CHEPP, c.f. [13]. The simplicity of the gas model affects mainly the pressure and temperature traces during the combustion.

6.3 Control Oriented Gas Exchange Models for CVCP Engines and their Transient Sensitivity

In Chapter 8, [39], another early version of the implementation in Chapter 5 is used to analyze the same set of control oriented models that were used in Chapter 7. There are two main additions made. The first addition is the investigation of how much transients in cam phasing affect the air charge as well as residual mass fraction and the models ability to capture these effects. Additions have also been made to the models of the intake- and exhaust ducts which here are one-dimensional models, that capture the ram effects in the manifolds, replacing the static pressure models used in the earlier publication.

The main difference between the reference model used for the evaluation in Chapter 8 and the model in Chapter 5 is the zone layout and the fact that the model does not allow the fuel/air ratio to be changed during the lifespan of a zone. The residual mass fraction can, however, change in the model in Chapter 8. The zone layout is the same as in Chapter 7 and the pipes that are used as intake and exhaust runners allow mixing of residuals and fresh air. The pipes allow diffusion of residuals into the intake manifold during periods of

back flow through the intake valve and eliminates the need to keep track of the back flow separately. The burned gas that is mixed with the fresh mixture in the intake manifold and that is flowing back into the cylinder is divided into its two components that are then placed in the fresh mixture zone and the burned gas zone respectively. In this way the two zones with burned gas and fresh mixture can be kept separate.

The argument for keeping the burned gas and the fresh mixture separate during the gas exchange phase even if it is not required by the framework is that it yields the possibility to have a geometrical interpretation. The interpretation is that the fresh mixture zone is close to the intake valve and the burned gas zone is close to the exhaust valve. This can be used to model the composition of the flow through the intake and exhaust valves when the mixture is not considered homogeneous. The flows through the cylinder ports are, however, complex and the cylinder ports are designed to yield a certain amount of turbulence to help the mixture to become homogeneous.

The gas that is used for the simulations in Chapter 8 is the same as in Chapter 7 and is thus also described by (6.1).

6.4 Diesel Engine Modeling

In [42] a model of diesel combustion is proposed and in [28] an implementation of the model is used to estimate engine torque. The diesel model in [28] used an early version of the implementation in Chapter 5 and the model has with minor modifications been adapted to the current framework and is now used here. The differences in the framework are, however, minor and does not influence the simulation results. The model is used to show the flexibility of the framework and that it manages to keep track of energy, mass and volumes throughout the simulation.

The zone layout that is suggested in [42], and which is based on [11], is an air zone, an incompressible liquid zone, and a mixing zone for each injection nozzle. In [28] the zone layout is modified slightly to have an air zone, an incompressible liquid fuel zone, a gaseous fuel zone, and a burnt zone. The two zone layouts are depicted in Figures 6.2 and 6.3. The differences between the two choices are thus the air entrainment and the choice to keep the residuals together with the burned gas instead of the air. This has implications on the rate equations for evaporation and combustion because the temperature of the zones differ between the layouts.

In the following analysis the implementation choices from [28] are left as is. However, it is possible to use the same layout as suggested in [42]. An example setup of the Diesel model implementation from [28] has been simulated and the parameters have been set to the default values of the implementation. While the choices in [28] have a profound effect on the rate equations for fuel evaporation, entrained air and combustion the effects of these choices are not analysed here. It has to be stressed that the model itself is not evaluated or ratified in this thesis, instead it is the thermodynamic framework that is put to the test.

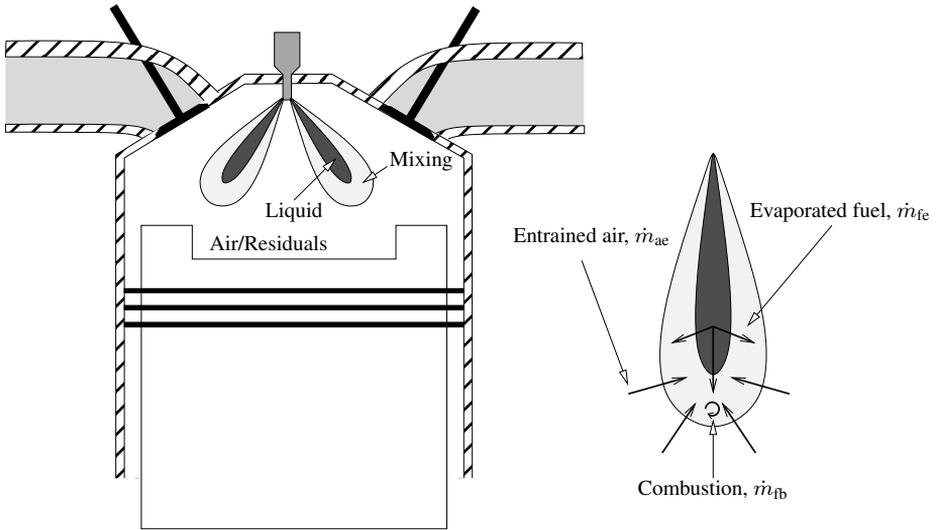


Figure 6.2: Zone layout as suggested in [42]. The air zone is used to keep track of the air and residuals while the mixing zone is used to keep track of the entrained air, evaporated fuel as well as the burnt fuel. The liquid zone is used to keep track of the injected fuel.

6.4.1 Adding a Liquid Incompressible Zone

The DAE equation (4.2) is formulated for gaseous substances. In the model the injected fuel is, however, treated as an incompressible liquid. This is easily handled by subtracting the liquid volume from the total cylinder volume so that $V_{\text{Tot}} = V_l + \sum_i V_i$. The volume of the liquid zone is determined by the mass of fuel in the liquid zone and the density of the fuel, $dV_l = \frac{\dot{m}_l}{\rho_l}$. Equation (4.2) can be extended to handle this liquid zone by adding one row and one column to the \mathbb{A} -matrix in the following manner

$$\begin{pmatrix} \mathbb{A} & 1 \\ & 0 \\ & \vdots \\ & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} dx \\ \\ \\ dV_l \end{pmatrix} = \begin{pmatrix} \mathbb{B} \\ \\ \\ \frac{dm_l}{\rho_l} \end{pmatrix}$$

It is easy to see that that the determinant of the \mathbb{A} -matrix is not affected by the extension and the liquid volume can be eliminated by subtracting the last row from the first row.

6.4.2 Evaluation of Mass, Energy, and Volume Conservation

Figure 6.4 shows the temperature of the zones while Figures 6.5, 6.6, and 6.7 show how the fuel and air flows between the zones in the example setup. The solver that is used is

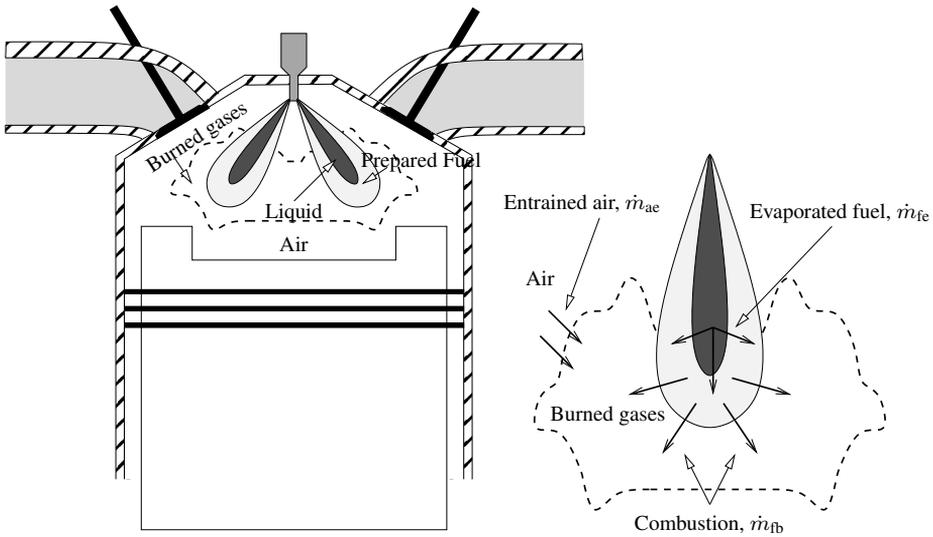


Figure 6.3: Zone layout from [28] (the model tested here). Note that the geometric interpretation is only for illustrative purposes. The air zone is used to keep track of the air and a prepared zone is used to keep track of the gaseous fuel. Air that entrains the fuel spray, \dot{m}_{ae} , is passed directly to the burned zone where also the residuals are kept. Combustion is modelled as a mass flow of fuel, \dot{m}_{fb} , from the prepared zone into the burned zone. The liquid zone is still used to keep track of the injected fuel which then evaporates into the prepared zone, \dot{m}_{fe} .

ode45 and the relative tolerance is 10^{-9} . Figure 6.8 shows the relative errors in total mass, volume, and energy for relative tolerances 10^{-8} , 10^{-9} , and 10^{-10} . The definition of the relative error in total mass, energy, and volume is the same as in equations (4.14) and (4.15).

When fuel is injected it starts to evaporate immediately and is stored in the prepared zone. Figure 6.6 shows the amount of fuel in the prepared and burned zones as well as the total amount of injected and prepared fuel. At the same time as fuel evaporates air starts to entrain the fuel spray. In the model it is assumed that the air is combusted and therefore transported to the burned zone. This choice was motivated by the fact that because the mixture in the burned zone is already lean, i.e. $\phi < 1$, the extra air does not have a large impact on the chemical equilibrium and hence no energy is released when the air is transported into the burned zone.

The fuel/air ratio of the burned zone decreases as air flows in. Figure 6.5 shows the fuel/air equivalence ratio and normalized fuel and air flows to the burned zone. When combustion starts the combustion rate increases rapidly during the premixed combustion until the fuel in the prepared zone is depleted. Figure 6.7 shows the mass flow rates to the prepared zone and the burned zone. The fuel/air ratio of the burned zone increases again as fuel enters the burned zone. When the prepared zone is depleted the combustion process becomes mixing-controlled and the mixing and combustion rates are the same.

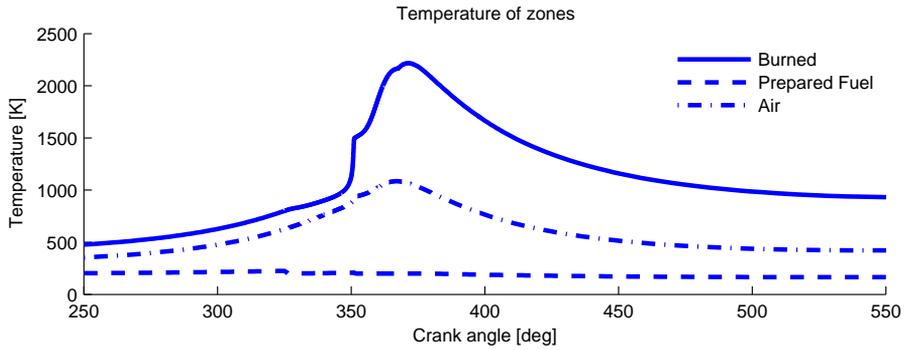


Figure 6.4: Temperatures for the burned, prepared, and air zones of the diesel model example. The prepared zone is cooled down by the injected fuel and therefore the temperature does not increase during the compression. The temperature of the air zone is only affected by compression and cooling. At about 345 crank angle degrees the fuel is ignited and the combustion starts. This is seen as a large temperature gradient of the burned zone.

During the simulation the fuel/air equivalence ratio of the burned zone is as low as 0.35 and as high as 0.75.

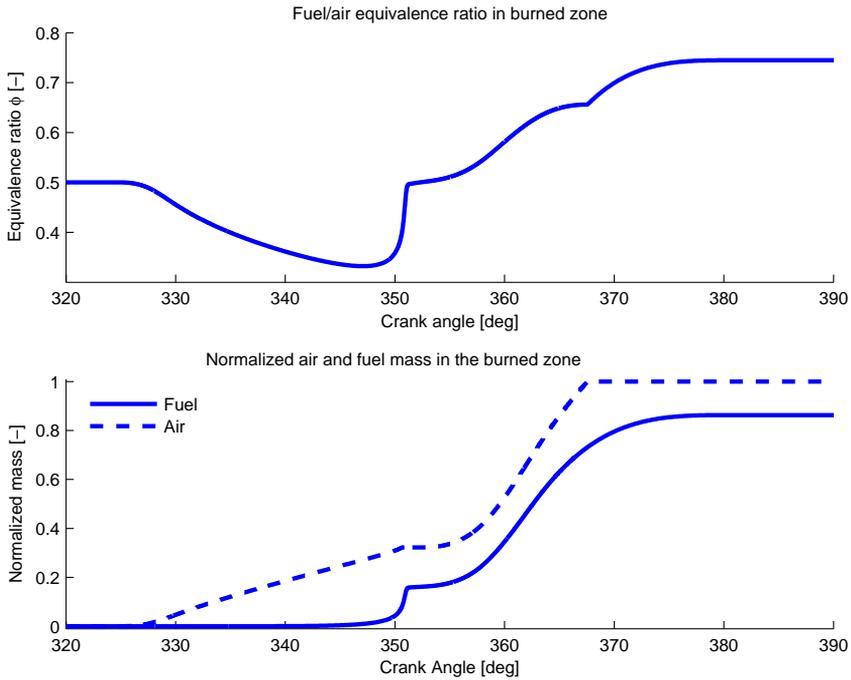


Figure 6.5: Normalized fuel and air masses that have entered the burned zone. The air is normalized with the final air mass, while the fuel is normalized with the final air mass divided by the stoichiometric air/fuel ratio, $\left(\frac{A}{F}\right)_s \approx 14.7$. At first, when air is entrained the fuel/air equivalence ratio, ϕ , of the burned zone decreases. During the premixed combustion the fuel/air ratio increases rapidly again until the prepared zone is depleted and the combustion becomes mixing-controlled.

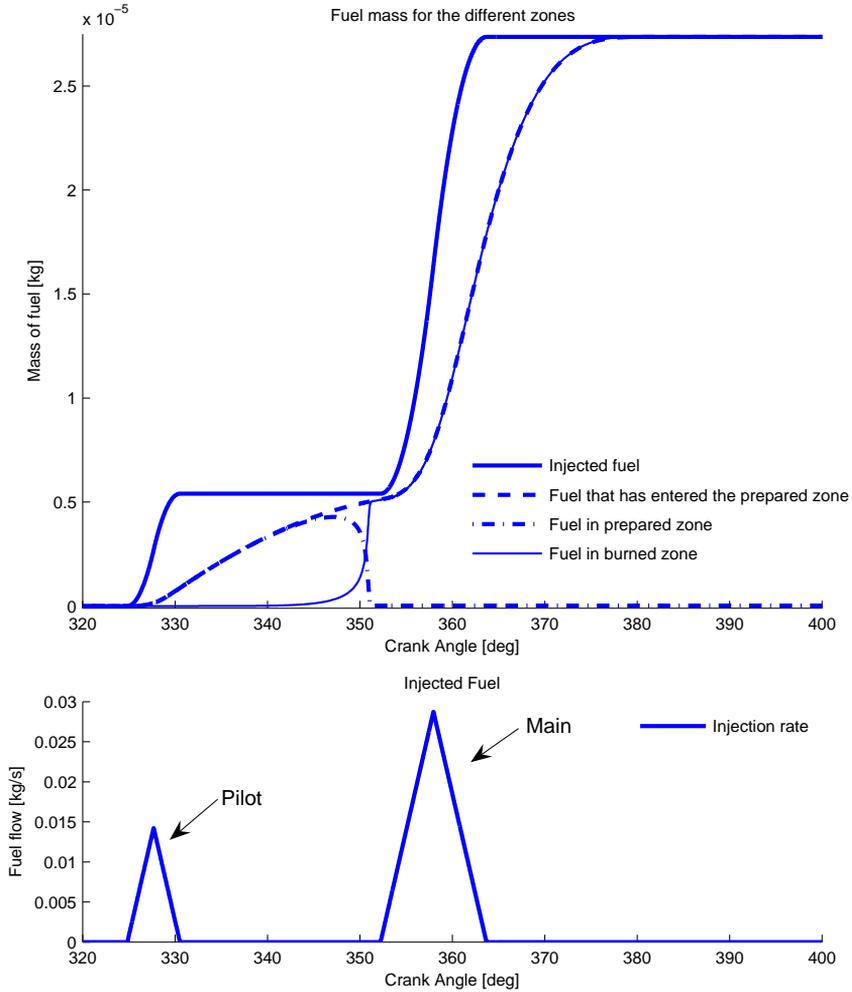


Figure 6.6: Fuel mass of the prepared and burned zone as well as total amount of injected and evaporated fuel for the diesel example setup. When the fuel is first injected it ends up in the liquid zone. As it evaporates it enters the prepared zone. When the combustion starts the fuel in the prepared zone is quickly transported into the burned zone and the prepared zone is emptied. After ignition, the fuel that enters the prepared zone is directly transported to the burned zone and the combustion rate is thus controlled by the evaporation rate.

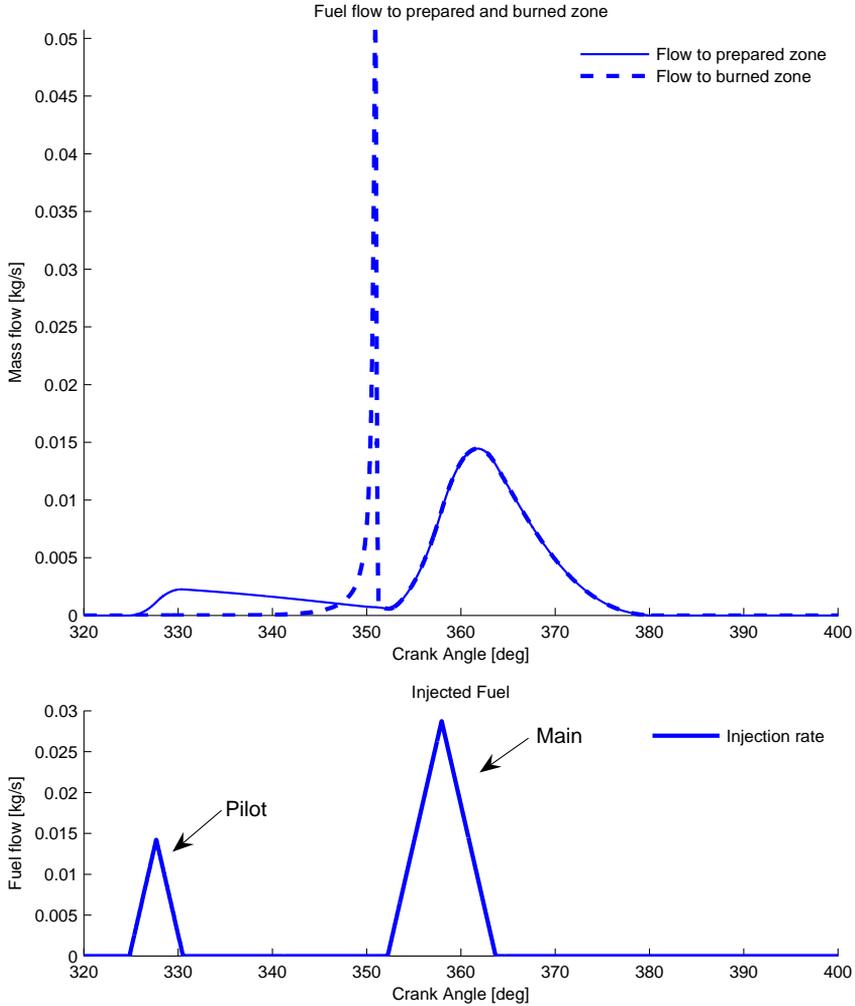


Figure 6.7: Fuel flows for the example setup of the diesel model. At first the fuel evaporates into the prepared zone and stays there. As the amount of fuel in the liquid zone becomes smaller the evaporation rate decreases. The burn rate is small until the mixture suddenly gains speed. The prepared zone is then quickly depleted and the combustion becomes controlled by the evaporation rate.

In Figure 6.8 the relative errors in total mass, volume, and energy are shown for three simulations with relative tolerances 10^{-8} , 10^{-9} , and 10^{-10} . As can be seen the relative errors decrease when the tolerance is tightened. However, at about 360 crank angle degrees the burned mixture passes the frozen temperature limit which makes the error in energy pop up to about 10^{-3} and the error in mass to about 10^{-5} . The conclusion is that the framework is able to preserve the total mass, energy and volume for this diesel model setup.

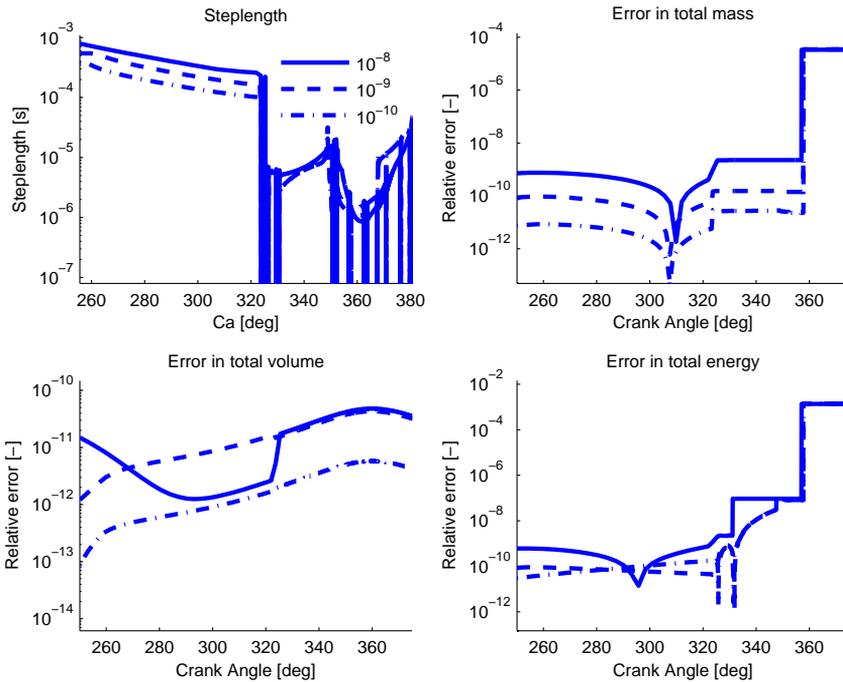


Figure 6.8: The solver that has been used is `ode45` but other solvers yield results that are consistent. The sudden increase in energy and mass error at about 360 crank angle degrees is due to the cooling of the burned gas. Unfortunately the model is stiff and using a relative tolerance higher than 10^{-8} does not shorten the step length. Other solvers has been used with similar results.

6.4.3 Investigation of Gas Model Dependence

As mentioned in Section 3.2.1 there are implications of choosing a state vector with four components. For an SI engine model there are no significant differences in pressure or temperature. In this diesel implementation where the fuel and air are first kept separate and then mixed as they enter the burned zone there are, however, differences. The implemented diesel model has been simulated using both the full equilibrium and the four component gas models. In Figure 6.9 the relative difference in pressure and burned temperature between the results for the two models are shown. As can be seen in the figure there is a large difference at the beginning of the combustion at about 350 crank angle degrees. In Figure 6.10 the combustion timing of the premixed combustion phase is shown. The figure shows that the combustion timing for the four component model is advanced by 0.1 crank angle degree which explains the large peak in pressure and temperature error.

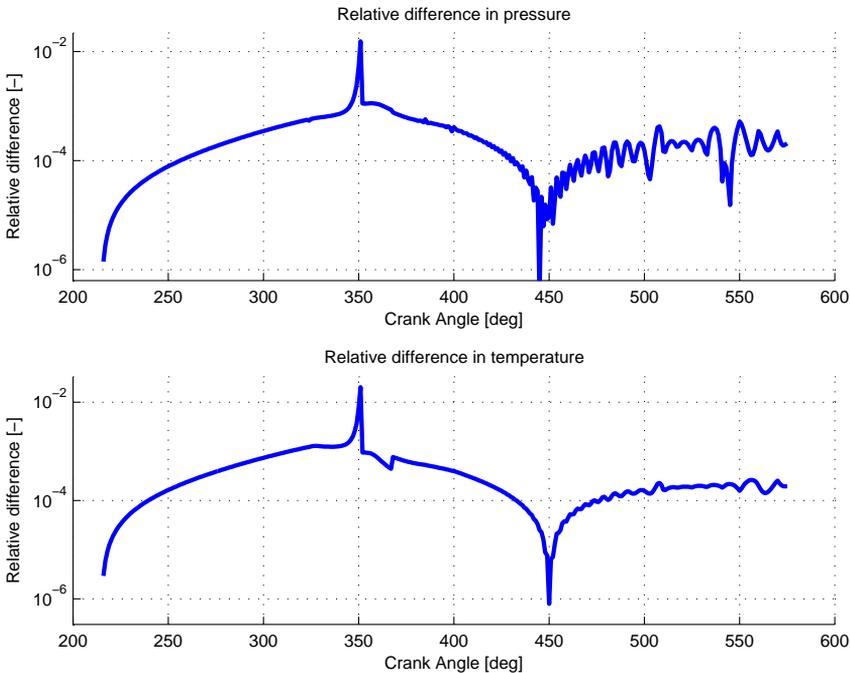


Figure 6.9: Comparison of pressure and burned temperature between the full equilibrium model and the four component model. The pressure and temperature difference are as large as a couple of percent. This difference is mostly due to the difference in combustion timing and the rapid pressure and temperature increases associated with it.

To understand why a relative difference in temperature of about 10^{-3} can have such large effects on the combustion timing it is beneficial to look at the rate equation for the burned

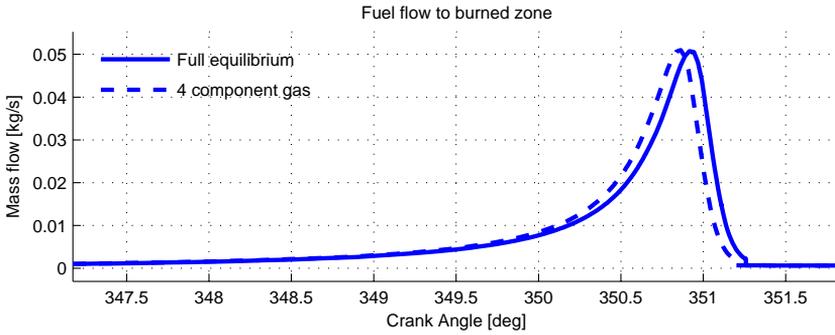


Figure 6.10: Comparison of combustion timing between the full equilibrium model and the four component model. Due to the differences between the two models the combustion is advanced 0.1 crank angle degrees. The maximum difference in magnitude in burn rate is about 20% which directly affects the pressure and temperature traces.

fuel. The premixed combustion rate in the diesel model is governed by an Arrhenius equation [42]

$$\dot{m}_{f,b} = m_{f,p} f(T_{\text{mean}}) \cdot e^{-\frac{T_A}{T_{\text{mean}}}}$$

where T_{mean} is the mean gas temperature of all zones, T_A is the activation temperature, and $m_{f,p}$ is the mass of fuel in the prepared zone. The exponential function in the rate equation makes the combustion timing sensitive to the mean gas temperature and small differences can therefore have a significant impact on the timing. The timing can, however, be delayed by adjusting the activation temperature.

A conclusion from the diesel simulations with different gas models is that modeling the combustion as a temperature controlled rate, as in the Diesel example, is more sensitive to effects that affect the temperature before the start of combustion, than using a fixed rate, as in the SI example. The main conclusion is that the framework is able to preserve the total mass, energy and volume as well as handle the incompressible zone for this diesel model setup.

Control Oriented Modeling of the Gas Exchange Process in Variable Cam Timing Engines

SAE Technical Paper 2006-01-0660 ¹

Per Öberg and Lars Eriksson
Vehicular Systems, Dept of EE
Linköpings Universitet
SE-58183 Linköping
SWEDEN

Abstract

Variable cam timing engines pose new questions for engine control system designers. The cam timing directly influences cylinder air charge and residual mass fraction. Three models that predict residual mass fraction are investigated for a turbocharged dual independent Variable Cam Timing (VCT) engine. The three models (Fox et. al. 1993, Ponti et. al. 2002, and Mladek et. al. 2000) that all have real time capabilities are evaluated and validated against data from a crank angle based reference model. None of these models have previously been validated to cover this engine type.

It is shown that all three models can be extended to dual independent VCT engines and that they also give a good description of the residual gas fraction. However, it is shown that the two most advanced models, based on a thermodynamic energy balance, are very sensitive to the model inputs and proper care must therefore be taken when these models are used.

¹Reprinted with permission from SAE paper 2006-01-0660 (c) 2006 SAE International. This chapter is a edited version of [38]. The changes are minor typographical changes, moving of the reference list to the end of the thesis, and adaption to single column.

7.1 Introduction

Air charge is important for engine fuel and torque control while residual mass fraction is a crucial factor that limits stable engine operation since it influences the combustion variability. Therefore it is essential for the control system to know the air charge and residual mass fraction, hence models that are accurate enough are required. When cam timing is used to its full extent existing control oriented models for residual mass fraction and cylinder air charge have the shortcoming of not covering the entire engine operating region.

The three models (Fox et al. 1993 [15], Ponti et al 2002 [43] and Mladek et al. 2000 [35]) that are investigated in this paper all predict residual mass fraction. The model in [35] also predicts air charge. When it comes to applicability the model in [15] is stated to be applicable for low to medium engine speeds. In [35] no operating points where the Variable Valve Timing (VVT) system has been active are used in the estimating procedure. In [43] both VVT and External EGR is used in the model but the model has not been validated. None of these three models have previously been validated to for a dual independent variable cam timing engine.

There is thus a need to investigate if the models can describe the residual gas fraction for dual independent VCT engines and to validate them over the engines' full operating range.

There are other publications that propose control and or estimation algorithms for different types of VVT systems, but the focus has been on the air charge and not on the residual gas fraction. In [27] and [34] air charge for dual equal and intake only VCT systems, that have moderate valve overlap, is studied. In [17] the focus is on fuel injection for the same type of engines as mentioned earlier.

7.1.1 The Models

Model A, Fox et al. 1993 [15], is a simple generalized flow restriction model created with physical insight. Two model parameters are tuned to measurements or simulations of residual mass fractions.

Model B, from Ponti et al 2002 [43], and Model C, from Mladek et al. 2000 [35], are both based on thermodynamic relations, e.g. an energy balance at IVC. They both require measurement of cylinder pressure.

The reference model that is used is a crank angle based multi-zone model. A reference model is necessitated by the fact that it is very difficult to directly measure the actual residual mass fraction in a production engine.

A straightforward validation of the residual mass fraction from the reference model is impossible. Therefore the reference model is tuned and validated against measured cylinder and manifold pressures, temperatures and mass flows through repeated simulations. Experimental data for the validation is obtained from a four cylinder, turbocharged engine with dual-independent continuously variable cam timing.

7.2 Experimental Setup

Measurements have been obtained for 2000 and 4000 rpm using six different air masses at 7×7 different cam positions. The span in air mass and cam position is chosen to cover as much of the engine operating region as possible. Altogether 427 of the 588 possible operating points are used because the engine does not produce work for all combinations.

Intake and exhaust manifold pressures have been measured using standard pressure sensors. Temperatures in exhaust and intake manifolds are measured using standard thermo elements and mass flow into the engine have been measured using the built in hot-film air mass flow meter. The cylinder pressures has been measured with a resolution of one crank angle degree. The number of cycles for each measurement is more than 300.

7.3 Reference Model

As reference model a crank angle based zero-dimensional multi-zone model has been used. The model is tailored to track the flows of fresh and burned gases during the gas exchange period. It is built upon the method presented in [37] that solves the in-cylinder differential equations.

In the reference model the engine cycle is divided into three phases, gas exchange, compression and combustion. During the gas exchange and the combustion phase two zones are used. During the compression phase however only one zone is used.

The two zones in the gas exchange phase are used too separate the residual gas and the fresh charge. When entering the compression phase the gases are instantaneously mixed into one zone. In the combustion phase one zone is used to track the unburned gases and the other is used to track the burned gas. Combustion is modeled as a mass transfer from the unburned to the burned zone.

Flow into and out of the cylinder is modeled by standard flow restrictions with constant discharge coefficients and the valves open areas are modeled by the geometries and valve lifts from the actual engine. There are no dynamics in the exhaust or intake manifold and heating/cooling by the exhaust/intake ports is not considered. The fuel that enters the model is regarded as well mixed and no consideration is taken to charge cooling by evaporation.

In cylinder heat transfer is modeled using Woschni [53] with constant wall temperature. For burn ratio the standard Vibe [51] function has been used.

Thermochemical data for the model is obtained by using polynomials fitted to calculations using the chemical equilibrium program package, CHEPP, presented in [13]. For c_p and c_v dissociation is considered but the specific gas constant, R , is constant. The fuel used is isoctane.

7.3.1 Model Parameters

Most of the parameters in the reference model is either kept constant for all operating points or taken from measurements in the specific point. The only parameters that are free are the parameters in the Vibe combustion function.

A complete list of parameters and constants used in the reference model can be found in Table 7.1 in Appendix 7.B.

Fitting of Vibe Parameters

The reference model is fitted to measurements using a least square optimization scheme. As residual the following measure is used

$$res = \sqrt{\left(\frac{m_{air,s} - m_{air,m}}{m_{air,m}}\right)^2 + \sum_i \left(\frac{p_{s,i} - p_{m,i} - p_o}{10^6 [\text{Pa}]}\right)^2}$$

where $m_{air,s}$ and $m_{air,m}$ represents measured and simulated cylinder air mass per combustion. $p_{s,i}$ represents samples of simulated cylinder pressure and $p_{m,i}$ represents measured. p_o is the cylinder pressure offset that minimizes the residual in each optimization step.

The parameters that are sought for are θ_s , θ_e and m in the Vibe function. An initial estimate of these parameters is calculated using the techniques presented in [45].

It has to be stressed that all other parameters in the reference model are left as constants or calculated from measurements during the least square fitting. For example, there is no fitting of heat transfer coefficients or temperatures to make the residuals smaller.

Signal Conditioning

The data from the measurements is resampled and the crank angle offset from TDC estimated before the parameter fit. The crank angle offset is obtained using the same procedure as when fitting the Vibe parameters above. Here thermochemical data for pure air is used instead of an air/fuel mixture and the Vibe parameters are kept constant while the crank angle offset is free.

7.3.2 Accuracy of Reference Model

The reference model is used to obtain data for the other methods since they are used exclusively on simulated data. Even so it is interesting to see how well the reference model describes a real engine. The results from comparisons with measured data are found in Appendix 7.B. The global agreement is good considered that most of the parameters are constants or measured values that are constant within the estimation procedure.

7.4 The Evaluated Models

7.4.1 Model A

Model A, Fox et al. [15], defines an overlap factor (OF) that measures the amount of overlap between intake and exhaust valves. The overlap factor is then used as effective area in a generalized flow restriction model. The model has two parameters that need tuning. These parameters are tuned to residual mass fractions from measurements or simulations. The following quantities are needed when using the model

- Intake manifold pressure, p_{im}
- Exhaust manifold pressure, p_{em}
- Engine speed, N
- Fuel air ratio, Φ
- Valve positions (which gives OF)

Computational Scheme

The key element in Model A is the overlap factor which is defined as

$$OF = \frac{D_i A_i + D_e A_e}{V_d} \quad (7.1)$$

where D_i and D_e are the inner seat diameters of the intake and exhaust valves and V_d is the displacement volume of the engine. A_i and A_e are the areas under the valve-lift/crank-angle curves and are defined as

$$A_i = \int_{IVO}^{L_i=L_e} L_i d\theta \quad \text{and} \quad A_e = \int_{L_i=L_e}^{EVC} L_e d\theta \quad (7.2)$$

where L_i and L_e are the intake and exhaust valve lifts respectively. In this implementation L_i and L_e have been calculated using the assumption that they are proportional to the valves effective flow areas, i.e. $L_i \propto \frac{A_{eff,i}}{D_i}$ and $L_e \propto \frac{A_{eff,e}}{D_e}$. Since the valve lifts in the calculation of A_i and A_e are small this has little effect on the calculation of OF .

Finally the residual mass fraction can be calculated using the following expression

$$x_{RG} = C_{1x_{RG}} \cdot \left(\frac{p_{em}}{p_{im}} \right)^{\left(\frac{\gamma+1}{2\gamma} \right)} \cdot \left(\frac{OF}{N} \right) \cdot \sqrt{p_{em} - p_{im}} \\ + C_{2x_{RG}} \cdot \frac{1}{\lambda \cdot r_c} \left(\frac{p_{im}}{p_{em}} \right)^{\frac{1}{\gamma}} \quad (7.3)$$

where $C_{1x_{RG}}$ and $C_{2x_{RG}}$ are constants that have to be fitted to data from measurements or simulations. These constants have been tuned to data from the reference model and the results are shown in Figure 7.13 in Appendix 7.C.

7.4.2 Model B

Model B is presented in [43] and is a simplification of Model C. The model was originally used for an engine with external EGR and VVT. External EGR is omitted in this implementation since the engine used lacks external EGR capabilities. The key element in the method is an energy balance at IVC. Using this energy balance the residual mass can be calculated from the following equation

$$\begin{aligned} (m_{FC} c_{v,FC} + m_{RG} c_{v,RG}) T_{IVC} &= \\ &= m_{FC} c_{v,FC} T_{FC} + m_{RG} c_{v,RG} T_{RG} \end{aligned} \quad (7.4)$$

as long as T_{IVC} , T_{FC} and T_{RG} are known. Estimates of T_{FC} and T_{IVC} are calculated using measurements of the following quantities

- In cylinder pressure, p_{cyl}
- Intake manifold temperature, T_{im}
- Air mass entering the cylinder, m_{FC}
- Engine speed, N

A good estimate of T_{RG} , however, is trickier and is therefore obtained using a correlation between T_{RG} and $m_{Tot} \cdot N$.

Algorithm

The first step of the algorithm is to adjust the cylinder pressure to the correct level. The reason for this is that the pressure measurements system used have a slowly varying offset. This adjustment is done by considering the cylinder pressure during the compression phase as a polytropic process for which $p \cdot V^\gamma$ is constant. A special variant of this technique is used in [43] but since the model is used on simulated data no compensation is needed and therefore the description of this technique is left out.

The temperature at IVC is calculated using cylinder pressure at IVC, measured air charge, m_{FC} , an estimate of the residual mass fraction, x_{RG} , and the following equations

$$m_{Tot} = \frac{m_{FC}}{1 - x_{RG}} \quad (7.5)$$

$$R_{IVC} = R_{RG} x_{RG} + R_{FC} (1 - x_{RG}) \quad (7.6)$$

$$T_{IVC} = \frac{p_{IVC} V_{IVC}}{R_{IVC} m_{Tot}} \quad (7.7)$$

In the next step the temperature of the fresh charge has to be estimated. In [35] the estimation of the fresh charge temperature for this model is discussed in detail. It is stated that the heat flux from the walls to the fresh charge is partly compensated for the fuel evaporating process. Therefore, the fresh charge temperature can be estimated using a polytropic compression from manifold conditions to in-cylinder conditions.

$$T_{FC} = T_{im} \cdot \left(\frac{p_{im}}{p_{IVC}} \right)^{\frac{1-\gamma_{FC}}{\gamma_{FC}}} \quad (7.8)$$

When it comes to the temperature for the residual gas other measures have to be taken. In [35] a correlation between mass flow and burned gas temperature at IVC is used. The model is

$$T_{RG} = -(C_{1T_{RG}} \cdot (m_{Tot} \cdot N))^{C_{2T_{RG}}} + C_{3T_{RG}} \quad (7.9)$$

where the three constants $C_{1T_{RG}}$, $C_{2T_{RG}}$ and $C_{3T_{RG}}$ are estimated using simulations. These parameters have been tuned to data from the reference model and results from the tuning are shown in Figure 7.14 in Appendix 7.C.

Finally the new residual mass fraction can be estimated using the energy balance at IVC in Equation (7.4)

$$x_{RG} = \frac{c_{v,FC}(T_{IVC} - T_{FC})}{c_{v,RG}(T_{RG} - T_{IVC}) + c_{v,FC}(T_{IVC} - T_{FC})} \quad (7.10)$$

where the specific heat values at constant volume $c_{v,FC}$, $c_{v,RG}$ and $c_{v,IVC}$ are the same as in the reference model.

Given an initial estimate of x_{RG} and using Equation (7.5)–(7.10) a new x_{RG} can thus be calculated. The problem is solved by fixpoint iteration and gives the final estimate of x_{RG} . The iterative process is summarized by the flowchart in Figure 7.1.

Unfortunately the original fixpoint iteration does not always converge. In [43] this has been addressed and a stabilizing scheme has been suggested. Instead of updating the residual estimate with the new value in each iteration the following weighting is performed

$$x_{RG,used} = x_{RG,old} + \frac{x_{RG,new} - x_{RG,old}}{2}$$

7.4.3 Model C

Model C is presented in [35]. The key elements are energy balance equations at two different points during the engine cycle. Using these equations both cylinder air charge and residual mass fraction can be calculated. In addition to well known thermodynamic relations two approximations are used to get estimates of the residual mass temperature at IVC and the relative heat loss to cylinder walls at $x_b = 50\%$.

The model requires on-line measurements of

- In cylinder pressure, p_{cyl}
- Intake manifold temperature, T_{im}
- Engine speed, N

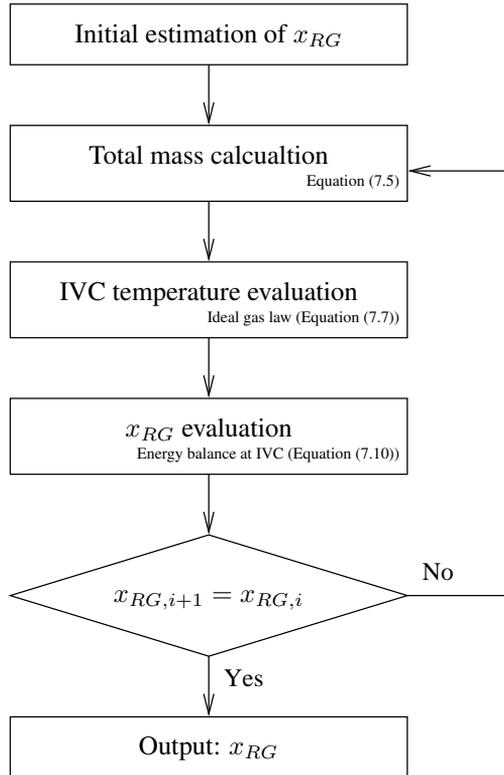


Figure 7.1: Flowchart of Model B. Using an initial estimate of x_{RG} a new value can be obtained using Equation (7.5),(7.7) and(7.10). Repeating this in an iterative manner a final value of x_{RG} is obtained.

Original Algorithm

Model C uses much of the same principles as Model B. As mentioned before Model B is in fact a simplification of Model C. The difference is that the fresh charge, m_{FC} , is known in Model B but is estimated in Model C. Equation (7.6)–(7.10) is thus the same for Model C as for Model B.

One extra equation is needed to be able to estimate the total mass in the cylinder, m_{Tot} , that was considered as known in Model B. Energy balance at $x_b = 50\%$ is therefore introduced. The position for $x_b = 50\%$ is estimated using a burn rate analysis as follows:

1. The end of combustion is found by using the following relation $ca_{BE} = argmax(p \cdot V^{1.15}) + 10$.
2. Weighting values are calculated so that the final x_b will be 0% at start of combustion and 100% at end of combustion. The weighting values are

$$\begin{aligned} \dot{p}_{CS} &= p_{CS} \left(\frac{V_{CS}}{V_{TDC}} \right)^{\gamma_{comp}} \\ \dot{p}_{CE} &= p_{CE} \left(\frac{V_{CE}}{V_{TDC}} \right)^{\gamma_{exp}} \end{aligned} \quad (7.11)$$

where γ_{comp} and γ_{exp} are the polytropic indices for compression and expansion phases respectively. Their values are set to 1.32 and 1.27, as suggested in [35].

3. x_b is then calculated as

$$x_b = \frac{p \left(\frac{V}{V_{TDC}} \right)^{\gamma(ca)} - \dot{p}_{CS}}{\dot{p}_{CE} - \dot{p}_{CS}} \quad (7.12)$$

where $\gamma(ca)$ is equal to γ_{comp} for $ca \leq ca_{TDC}$ and γ_{exp} for $ca > ca_{TDC}$.

4. Finally the crank angle for $x_b = 50\%$ is obtained from the estimated x_b .

The energy balance equation at $x_b = 50\%$ is formulated as follows

$$\begin{aligned} \bar{T}_{50} &= \frac{\int_{ca_{IVC}}^{ca_{50}} p dV}{m_{Tot} c_{v50}} + \frac{\bar{T}_{IVC} c_{vIVC}}{c_{v50}} \\ &+ \frac{(1 - q_{Cooling}) \cdot 0.5 \cdot x_c \cdot (1 - x_{RG}) \cdot \frac{1}{1+\lambda \cdot AF_s} \cdot q_{HV}}{c_{v50}} \end{aligned} \quad (7.13)$$

where \bar{T}_{50} is the mean gas temperature at $x_b = 50\%$, $q_{Cooling}$ is a factor describing the heat loss to the cylinder wall, x_c is a compensating factor for incomplete combustion, AF_s is the stoichiometric air fuel ratio, λ is the normalized air fuel ratio and q_{HV} is the higher heating value of the fuel.

Since $q_{Cooling}$ is unknown it has to be estimated and this is done using the following correlation

$$q_{Cooling} = C_{1q_{cool}} \cdot \bar{T}_{50} - C_{2q_{cool}} \quad (7.14)$$

where $C_{1q_{cool}}$ and $C_{2q_{cool}}$ are constants that have to be tuned to data from simulations. Values obtained for the reference model can be found in Figure 7.15 in Appendix 7.C.

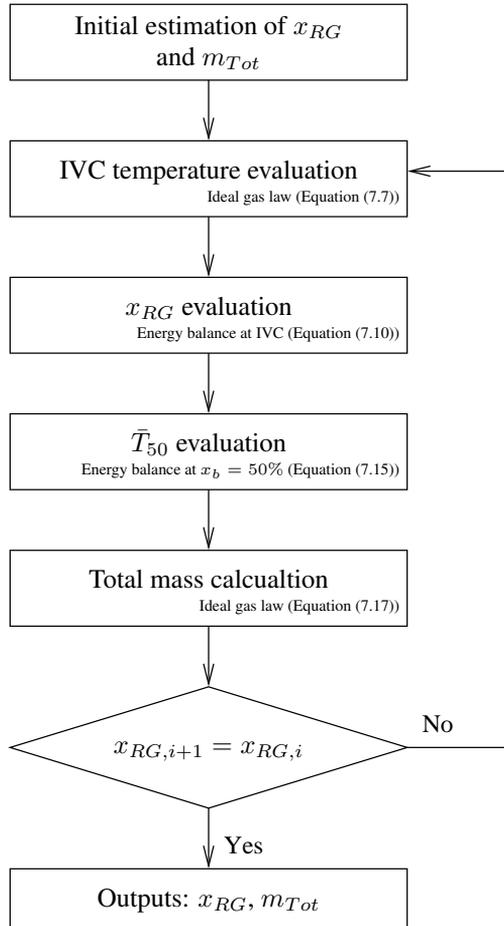


Figure 7.2: Flowchart of Model C. Using an initial estimate of x_{RG} and m_{Tot} , new values can be obtained using Equation (7.7), (7.10), (7.15) and (7.17). Repeating this in an iterative manner final values of x_{RG} and m_{Tot} is obtained.

Algorithm Modification

Using the $c_{v,50}$ calculated from the gas composition, i.e. calculated in the same way as for the reference model, will not give accurate results when using the energy balance in Equation (7.13) above. This is because

$$\Delta U = \int_{T_1}^{T_2} c_v(T) dT = mc_{v2}T_2 - mc_{v1}T_1$$

only holds if $c_v(T)$ is constant. Since $c_v(T) \approx c_{v1} + \frac{c_{v2}-c_{v1}}{T_2-T_1}(T-T_1)$ for the typical temperature interval Equation (7.13) has been modified in this implementation. Another issue is that $\int_{ca_{IVC}}^{ca_{50}} pdV$ is negative up till TDC since the volume is decreasing. Since work is added to the gas during this period a sign shift is necessary for the equation to be correct. The final expression is therefore

$$\begin{aligned} \bar{T}_{50} = T_{IVC} - \frac{\int_{ca_{IVC}}^{ca_{50}} pdV}{m_{Tot}(c_{v,IVC} + c_{v,50})/2} \\ + \frac{(1 - q_{Cooling}) \cdot 0.5 \cdot x_c \cdot (1 - x_{RG}) \cdot \frac{1}{1+\lambda \cdot AF_s} \cdot q_{HV}}{(c_{v,IVC} + c_{v,50})/2} \end{aligned} \quad (7.15)$$

Finally, when \bar{T}_{50} is known a new total mass estimate can be performed. Since no dissociation is considered in the reference model the gas constant for the composition is straight forward to calculate and is equal to R_{IVC} . Hence the total mass in the cylinder can be calculated using the relations

$$R_{50} = R_{IVC} \quad (7.16)$$

$$m_{Tot} = \frac{p_{50}V_{50}}{R_{50}T_{50}} \quad (7.17)$$

Complete Algorithm

The residual mass fraction, x_{RG} , can be calculated using an initial estimate of x_{RG} and the total mass in the cylinder, m_{Tot} , much in the same way as in Model B. A flowchart of the process is shown in Figure 7.2.

7.5 Results

7.5.1 Model A

Given the simplicity of Model A it performs rather well. Looking at Figures 7.3(a) and 7.3(b) it can be noted that the relative error is well under 70%. Studying the figures closer however reveals that there are effects that are not captured by the model since the errors have systematic content.

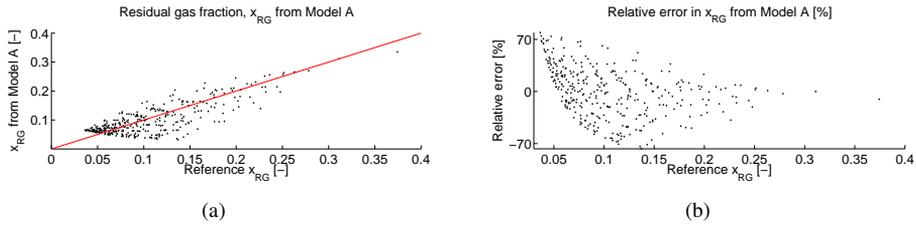


Figure 7.3: At first sight the modeled residual fraction gives a rather good fit to the reference. Studying the figure closer however reveals that there are effects that are not captured by the model (a). The relative error for Model A is well under 70% (b). Given the simplicity of the model this has to be considered as a good result.

The reasoning behind this conclusion is that if the model captured all physical effects the error should look more like noise and have no systematic content. Another issue with the model is that it uses the square root of the difference in manifold pressures in Equation (7.3). Since this value is not always positive for a turbocharged engine problems arise. It is not obvious how to include this into the model and therefore these points have been left out from the estimation.

7.5.2 Model B

Studying the residual mass fractions from Model B in Figure 7.4 it is evident that the model have some difficulties for small residual fractions. A comparison with the relative error for Model A in Figure 7.3(b) shows that the error is even larger than for Model A.

Since the greater part of the error is an offset and a scale error it is interesting to try and find the cause of the problems. Analyzing the method a number of observations can be made.

The method relies on Equation (7.10). The equation is repeated here for simplicity

$$x_{RG} = \frac{c_{v,FC}(T_{IVC} - T_{FC})}{c_{v,RG}(T_{RG} - T_{IVC}) + c_{v,FC}(T_{IVC} - T_{FC})}$$

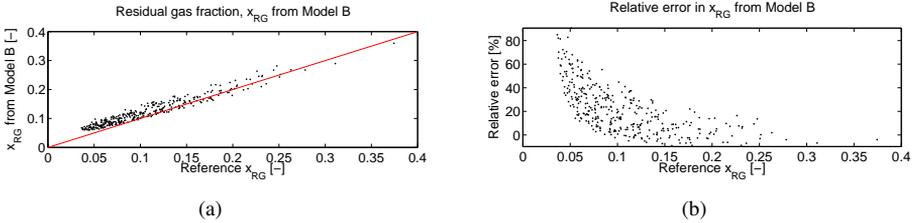


Figure 7.4: There is an offset and a scale error in the residual mass fraction from Model B (a). Studying the relative error for Model B shows that the relative error is as large as 80% for small residual fractions (b).

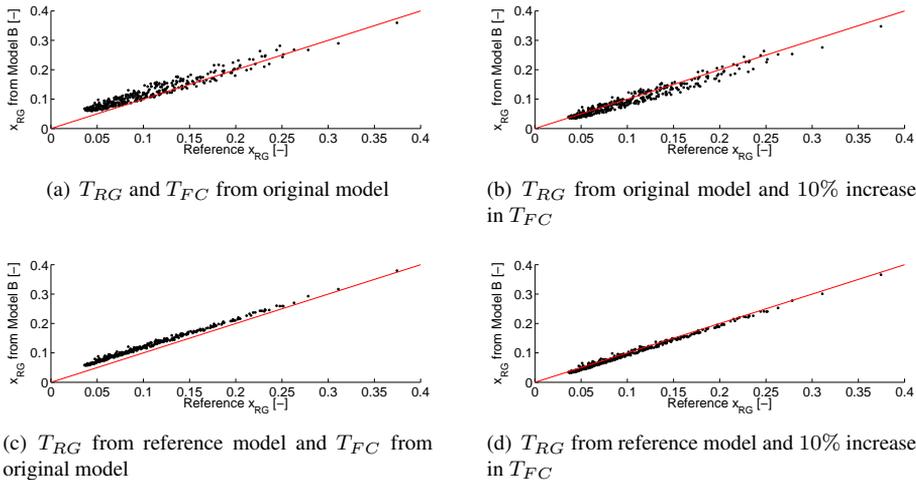


Figure 7.5: Using the original model for T_{RG} and T_{FC} gives a slight offset for small residual fractions (a). Adding a 10% heat effect on T_{FC} completely removes this offset (b). Using T_{RG} from the reference model and T_{FC} from the original model makes the spreadage smaller but does not affect the offset (c). Using the correct T_{RG} and a 10% adjustment to T_{FC} however makes the method perform very well. Note that the 10% heat correction is an ad hoc value and hence may significantly contribute to the remaining errors (d).

Given the correct temperatures obtained from the reference model, the mass fractions obtained also agrees well with the values from the reference model. It is therefore concluded that the error lies in the temperature estimates. Studying Equation (7.5)–(7.9) that leads up to Equation (7.10), it can be noted that

1. T_{IVC} is sensitive to errors in p_{cyl} and it is therefore important that the method of adjusting the measured p_{cyl} is chosen well. It is especially important that the intake valve is really closed for the offset estimation to work which poses interesting problems with a VCT engine.
2. T_{FC} is subject to charge heating from the cylinder walls which has shown to introduce an error up to about 10% in the estimation of T_{FC} . This is not included in Equation (7.5)–(7.9). Another issue is that charge cooling is not included in the reference model.
3. T_{RG} is obtained from a model that is not intended for use with a VCT engine. See for example the result from the parameter tuning in the T_{RG} model to the residual temperature of the reference model in Figure 7.14, Appendix 7.C.

However, the T_{IVC} estimate will be correct since the method is used on simulated data. This is because there is no offset in p_{cyl} and that all other parameters in Equation (7.7) are well known.

In Figure 7.5 corrected T_{RG} and T_{FC} estimates have been used to illustrate the methods sensitivity to errors in these estimates. The correction for T_{RG} consisted of replacing the modeled value with the correct value from the reference model. For T_{FC} a 10% increase to compensate for the charge heating were added.

Looking at Figure 7.5(b) it is possible to see that a 10% addition to T_{FC} to compensate for charge heating solves the immediate problems for the method. Figure 7.5(c) shows that using the correct T_{RG} has a much smaller effect on the result even though the relative error in T_{RG} is in the same range. The error in 7.5(d), where both compensations have been used, is even smaller. The important observation is however that the method is more sensitive to errors in the T_{FC} estimates than errors in T_{RG} estimates.

Note also that the remaining error in 7.5(d) is not only due to model imperfections but also from the fact that the 10% heat addition is not necessarily a good model of the heat exchange. The heat addition may very well vary between operating points.

In Figure 7.6 the relative error for the variant with 10% correction of T_{FC} but with the original T_{RG} from the model, is shown. This represents a relative error that is obtainable if the charge heating is included in the model. Note that the errors are well under 30% except for one outlier and that this is half the error of Model A.

7.5.3 Model C

Model C is the only model that gives both residual mass fraction and total mass in the cylinder. Model C suffers from the same problems with T_{FC} underestimating as Model B since they share Equation (7.8) and (7.10) with each other. A 10% increase in T_{FC} is therefore added from the start.

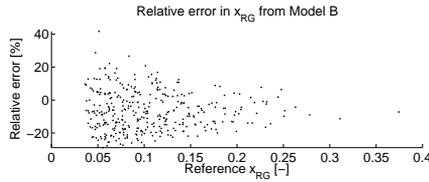


Figure 7.6: Looking at the relative error it can be concluded that using a 10% heat compensation in T_{FC} takes care of the larger errors and takes the relative error down to a maximum about 30%

As can be seen in Figure 7.7 Model C is quite accurate except for a small offset in the mid range air flows.

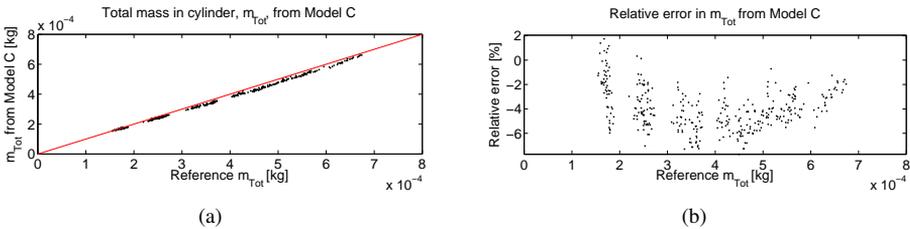


Figure 7.7: Total mass estimation using the T_{FC} increase works rather well for Model C. There is however a small error for medium mass flows in (a). The relative error in (b) also shows this dependence on airflow.

When studying the residual mass fraction instead, the model behaves well as can be seen in Figure 7.8. The spread in residual fraction is however larger than for Model B. The only difference between Model B and Model C is that Model C relies on its own mass estimate for the estimation of x_{RG} . A small error of about 6% in the mass charge is thus, for some operating points, propagated through the model to the estimate of x_{RG} where it causes an error of up till 50% extra relative error.

As with Model B the model has been used with the correct values for correlated quantities, i.e. correct values of T_{RG} and $q_{Cooling}$. The results are shown in Figure 7.9 and 7.10. The results are somewhat better than when using only the charge heating effect but the errors in the mass estimate still affect the residual estimate. Note that the 10% heat addition only is an ad hoc correction to the T_{FC} estimates and thus that the remaining errors in T_{FC} may be a major contributor to the remaining errors in the x_{RG} and m_{Tot} estimates.

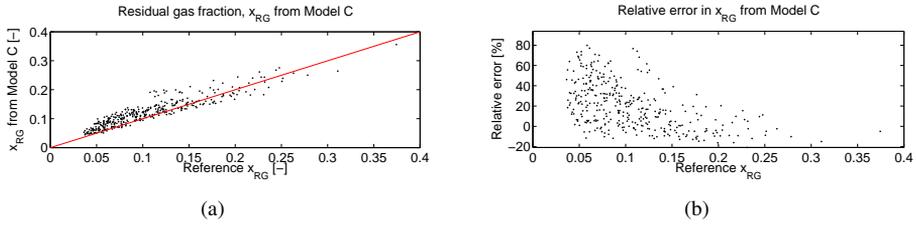


Figure 7.8: The spread in the residual mass estimation using the 10% increase of T_{FC} is larger than for Model B (a). The maximal relative error in (b) is about 75%.

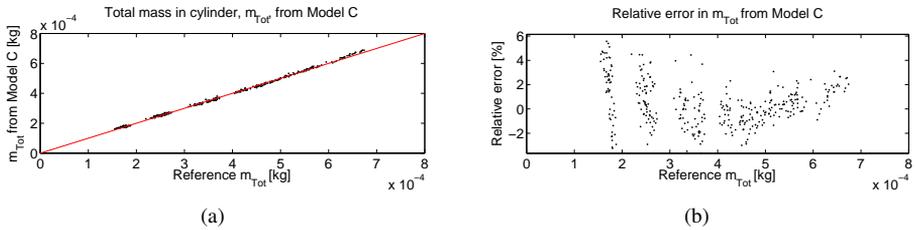


Figure 7.9: Using the correct values for $q_{Cooling}$ and T_{RG} as well as the 10% increase of T_{FC} eliminates much of the error in the total mass estimate in (a). This also shows up in the relative error in (b) which is now centered about zero.

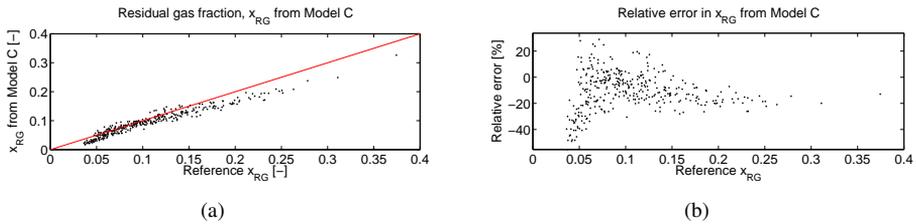


Figure 7.10: Using the correct values for $q_{Cooling}$ and T_{RG} together with the 10% increase of T_{FC} still leaves a quite large error in x_{RG} (a). However, looking at the relative error in (b) the maximal error is down to about 50%. Note that 10% is just an ad hoc value.

7.6 Conclusions

Three models, that predict residual mass fraction which of one also predicts air charge, have been investigated.

Model A gives good results right away considered that is very simple. There are however effects that are not taken care of and the relative error for small residual mass fractions is as large as 70%.

Model B performs well on simulations even though it is sensitive to the mean gas temperature at IVC, T_{IVC} , and the temperature of the fresh gas at IVC, T_{FC} . The error in temperature for the residual gas, T_{RG} , that is introduced by using the T_{RG} model in Equation (7.9), does not however significantly effect the x_{RG} estimate.

Because of the sensitivity to T_{IVC} and T_{FC} proper care have to be taken when applying the model to measured data since for measured data large errors may be introduced. Pressure measurements, air mass measurements, charge heating effects and fuel evaporation effects are especially important.

Model C estimates the air mass charge well. However the small errors in mass propagate to the estimates of the residual mass fraction that are affected in a negative manner. This effect is coupled to the sensitivity to T_{IVC} and T_{FC} in Model B.

The sensitivity to T_{IVC} and T_{FC} in Model B and Model C introduces larger errors in the residual mass estimate than the errors introduced by the $q_{Cooling}$ and T_{RG} models. The issues with T_{IVC} and T_{FC} is the same for both VCT and conventional engines even though the models for $q_{Cooling}$ and T_{RG} works better for a conventional engine.

Proper handling of Model B and Model C is crucial for good performance while Model A gives good results right away.

Appendix

7.A Nomenclature

Special quantities

$p_{s,i}$	Sample i of simulated cylinder pressure
$p_{m,i}$	Sample i of measured cylinder pressure
p_o	Cylinder pressure sensor offset
$m_{air,s}$	Simulated air mass per combustion
$m_{air,m}$	Measured air mass per combustion
\bar{T}	Average temperature
AF_s	Stoichiometric air fuel ratio
λ	Normalized air fuel ratio
Φ	Normalized fuel air ratio
$q_{Cooling}$	Heat loss due to cooling
q_{HV}	Higher heating value of fuel
x_c	Maximal combustion due to imperfections
p_{cyl}	Crank angle based cylinder pressure
c_v	Specific heating value for constant volume
c_p	Specific heating value for constant pressure
r_c	Compression ratio
γ	Ratio of specific heating values ($\frac{c_p}{c_v}$)
OF	Overlap factor
V_d	Displacement volume of engine
N	Engine speed in rps
D_i	Inner seat diameter of intake valve
D_e	Inner seat diameter of exhaust valve
$A_{eff,i}$	Effective flow area of intake valve
$A_{eff,e}$	Effective flow are of exhaust valve
L_i	Intake valve lift
L_e	Exhaust valve lift
x_{RG}	Residual mass fraction
x_b	Burn ratio

Abbreviations

EGR	Exhaust Gas Recirculation
VVT	Variable Valve Timing
VCT	Variable Cam Timing
IVC	Intake Valve Closure
TDC	Top Ded Center
EVC	Exhaust Valve Closure

Used indices

IVC	At Intake valve closure
CS	At combustion start
50	At $x_b = 50\%$
BE	At end of combustion
$comp$	During compression
exp	During expansion
RG	For Residual Gas part
FC	For Fresh Charge part
T_{ot}	Total
im	In Intake Manifold
em	In Exhaust Manifold

7.B Reference Model

A list of parameters used when simulating the reference model is shown in Table 7.1.

Table 7.1: Parameters and constants for the reference model.

Parameter	Value	Description
ϕ_{im}	1 [-]	Intake manifold fuel air ratio
$C_{d,ev}$	0.8 [-]	Intake valve discharge coefficient
$C_{d,iv}$	0.8 [-]	Exhaust valve discharge coefficient
T_{wall}	470 [K]	Cylinder wall temperature
p_{im}	(measured)	Mean value of intake manifold pressure
T_{im}	(measured)	Mean value of intake manifold temperature
p_{em}	(measured)	Mean value of exhaust manifold pressure
T_{em}	(measured)	Mean value of exhaust manifold temperature
$C_{1,heat}$	1	Woschni heat transfer coefficient
$C_{2,heat}$	0.4386	Woschni heat transfer coefficient
$\theta_{s,vibe}$	(estimated)	Vibe start of combustion
$\theta_{e,vibe}$	(estimated)	Vibe end of combustion
m_{vibe}	(estimated)	Vibe shaping parameter
a_{vibe}	6.9	Vibe shaping parameter

The reference model has been fitted to data from measurements in a subsection of $7 \times 7 \times 2$ operating points. Altogether 427 of the 588 possible operating points are used because the engine does not produce work for all combinations. The operating point grid consists of 7 exhaust cam phasings, 7 intake cam phasings, 6 airflows and 2 engine speeds. The engine speed is 2000rpm and 4000rpm. The span in air mass and cam position is chosen to cover as much of the engine operating region as possible.

Results from fitting the reference model to these measurements are shown in Figure 7.11. As can be seen the air charge for the reference model gives a good fit to measurements. The maximal relative error is about 20%. Since there are only two different engine speeds it is possible to see that the dependence on cam timing is large. It has to be stressed that no other parameters than the parameters for the Vibe function are used in the least squares optimization.

For completeness two simulated cylinder pressures are shown together with measurements in Figure 7.12(a) and 7.12(b). The simulation in Figure 7.12(a) represents the best possible fit and the simulation in Figure 7.12(b) the worst possible fit.

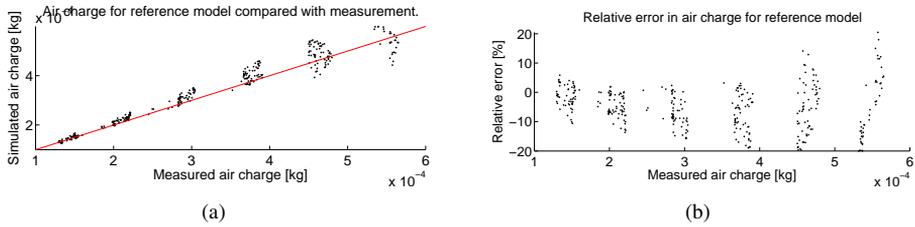


Figure 7.11: The air charge for the reference model gives a good fit although with some spreading (a). The maximum relative error is about 20% (b).

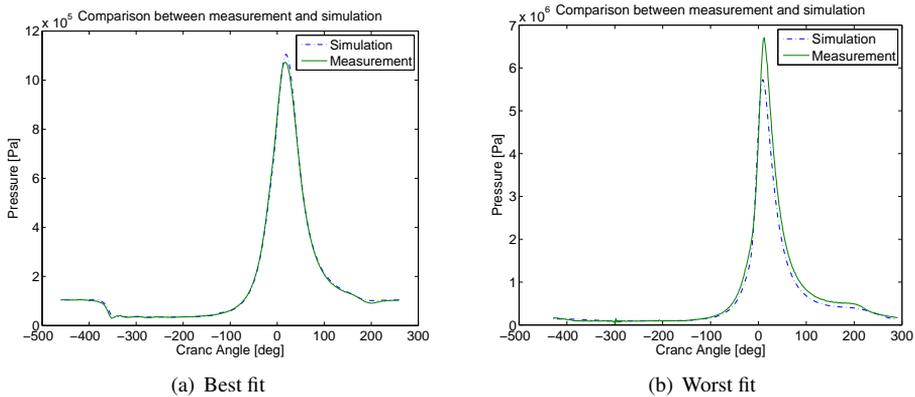


Figure 7.12: For the best case the accuracy of the reference model is good. The small difference in peak pressure is due to difference between the Vibe function and the actual burn rate. Note that this is the case that gives the smallest mean square error when comparing the measured and simulated cylinder pressure over the whole engine cycle. For the worst case the simulated cylinder pressure never reaches the same value as the measured signal. This is due to the fact that the error in air charge is large.

7.C Tuning of Model Parameters

Tuning of Parameters for Model A

The correlation for x_{RG} in Equation (7.3), Section 7.4.1, is directly used in Model A as the single most important component. The equation is repeated here for reference

$$x_{RG} = C_{1x_{RG}} \cdot \left(\frac{p_{em}}{p_{im}}\right)^{\left(\frac{\gamma+1}{2\gamma}\right)} \cdot \left(\frac{OF}{N}\right) \cdot \sqrt{p_{em} - p_{im}} + C_{2x_{RG}} \cdot \frac{1}{\lambda \cdot r_c} \left(\frac{p_{im}}{p_{em}}\right)^{\frac{1}{\gamma}}$$

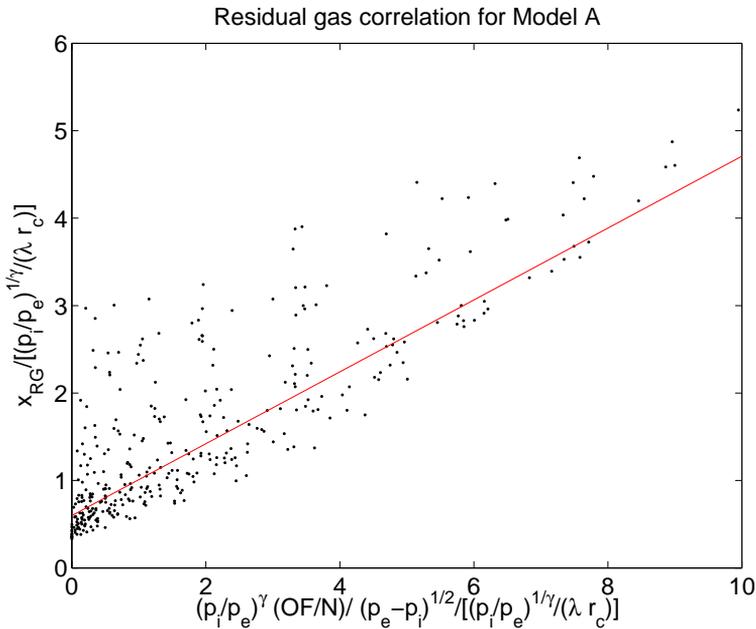


Figure 7.13: The constants $C_{1x_{RG}}$ and $C_{2x_{RG}}$ can be taken directly from the figure or fitted using a least square fit.

The parameters can be found by using a least square fit or by studying the data in Figure 7.13. The parameters have been fitted to:

$$C_{1x_{RG}} = 0.4205$$

$$C_{2x_{RG}} = 0.5869$$

Model for T_{RG}

The model for T_{RG} in Equation (7.9), Section 7.4.1, is used both in Model B and Model C. The equation is repeated here for reference

$$T_{RG} = -(C_{1T_{RG}} \cdot (m_{Tot} \cdot N))^{C_{2T_{RG}}} + C_{3T_{RG}}$$

Looking at Figure 7.14 there is little chance of getting a good fit between the data and the model used. This is not surprising since a VCT engine has been used instead of a conventional engine as in [35]. For a VCT engine the residual gas temperature, T_{RG} , dependence on cam timing is much stronger than the dependence on mass flow.

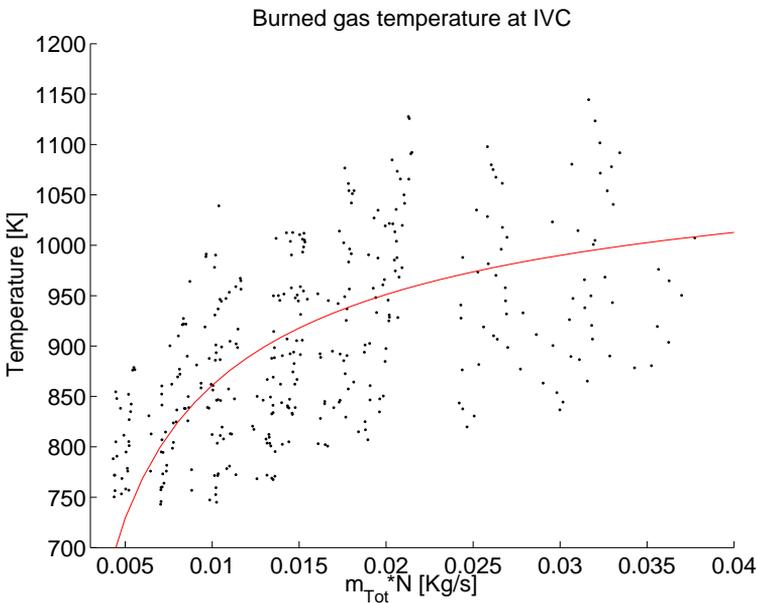


Figure 7.14: The correlation between mass flow and temperature of residual gas at IVC is weak. The reason for this is that the burned gas temperature varies more in a VCT engine than in an conventional engine.

The model has been tuned to the data in Figure 7.14 with the following values

$$\begin{aligned} C_{1T_{RG}} &= 0.0033 \\ C_{2T_{RG}} &= -0.5483 \\ C_{3T_{RG}} &= 1145.5 \text{ [K]} \end{aligned}$$

Model for $q_{Cooling}$

$q_{Cooling}$ is introduced in Equation (7.14), Section 7.4.1. The equation is repeated here for reference

$$q_{Cooling} = C_{1q_{cool}} \cdot \bar{T}_{50} - C_{2q_{cool}}$$

For $q_{Cooling}$ similar problems occur as with T_{RG} . This can be seen in Figure 7.15. The model has been tuned to the data with the parameters:

$$C_{1q_{cooling}} = -9.0622 \times 10^{-6}$$

$$C_{2q_{cooling}} = 0.0438$$

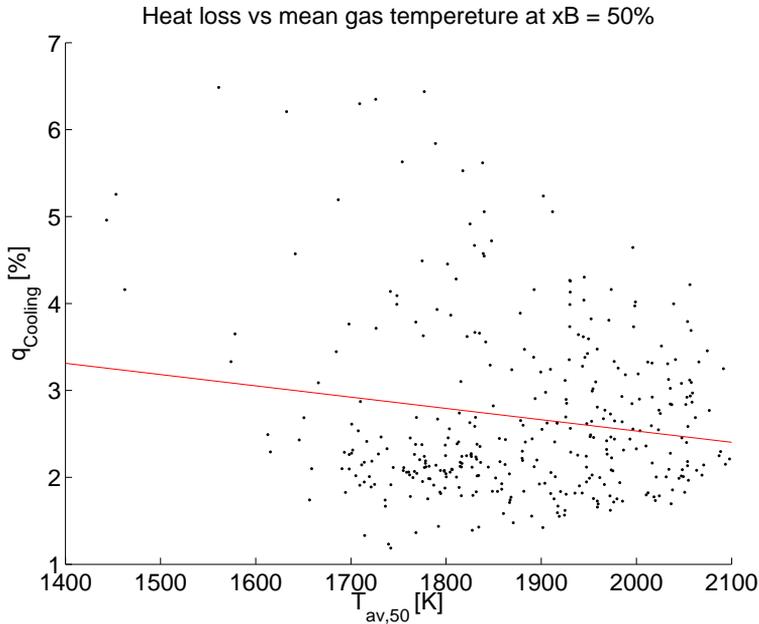


Figure 7.15: The strong dependence of heat loss on mean gas temperature does not hold for a VCT engine.

Control Oriented Gas Exchange Models for CVCP Engines and their Transient Sensitivity

P. Öberg and L. Eriksson¹

Division of Vehicular Systems, Department of Electrical Engineering, Linköping University 581 83 Linköping - Sweden

Abstract

Control Oriented Gas Exchange Models for CVCP Engines and their Sensitivity — The paper analyzes a set of control oriented models for the gas exchange phase in engines with continuously variable cam phasing (CVCP). These models describe the mass flow of fresh gases and the residual gases caught in the cylinder during the gas exchange phase. Simulations with CVCP transients are also performed to analyze the models performance during transients.

Résumé

Modèles d'échange de gaz contrôlé pour les moteurs CVCP et leur sensibilité — Le présent article analyse un ensemble de modèles de contrôle de la phase d'échange des gaz dans les moteurs à synchronisation des cames variable et continue (CVCP). Ces modèles décrivent le flux massif de gaz nouveaux et de gaz résiduels pris dans le cylindre au cours de la phase d'échange des gaz. Des simulations à l'aide de CVCP provisoires sont également effectuées afin d'analyser la performance des modèles au cours des essais provisoires.

¹This chapter is a edited version of the paper *Control Oriented Gas Exchange Models for CVCP Engines and their Transient Sensitivity*[39], Oil & Gas Science and Technology – Rev. IFP, Vol. 62 (2007), No. 4, pp. 573-584 Copyright ©2007, Institut francais du pétrole, DOI: 10.2516/ogst:2007041. The changes are minor typographical changes, moving of the reference list to the end of the thesis, and adaption to single column.

Introduction

There are high demands on the Engine control units (ECU) since they are central for achieving good performance such as stable combustion, good torque response, good fuel economy and low emissions. New technologies are introduced to meet the increasing demands from legislators and customers and these must also be handled and properly controlled by the ECU.

Continuously variable cam phasing (CVCP) provides an interesting possibility to reduce the fuel consumption but it also influences the amount of air and residual gases in the engine.

Air charge is important for engine fuel- and torque control while residual mass fraction is a crucial factor that limits stable engine operation since it influences the combustion variability. Therefore it is essential for the control system to know the air charge and residual mass fraction but these are not measured and hence models are required. When cam timing is used to its full extent the existing control oriented models for residual mass fraction and cylinder air charge have the shortcoming of not covering the entire engine operating region. Another issue is that transient effects introduced by the cam phasing mechanism have to be handled correctly.

Three models [15],[43] and [35]) that all predict residual mass fraction are investigated in this paper. The model in [35] also predicts air charge.

These models have previously been studied in [38] under static conditions. In particular their sensitivity to input data and model parameters were analyzed. In the paper it is concluded that while the more advanced models in [43] and [35] perform well on data from simulations they are sensitive to input data and model parameters, and that while the model in [15] is less sensitive it has other shortcomings. For example it is concluded that Model B and C are not very sensitive to errors in residual gas temperature but very sensitive to errors in mass flow and fresh gas temperature.

There are two main additions made in this paper. The first addition is the investigation of how much transients in cam phasing affect the air charge as well as residual mass fraction and the models ability to capture these effects. Additions have also been made to the models of the intake- and exhaust ducts which are here 1-D models, that captures the ram effects in the manifolds, instead of the static pressure models used in the earlier publication.

There are other publications, e.g. [27, 34, 17], that propose control and or estimation algorithms for different types of VVT systems, but the focus has been on the air charge and not on the residual gas fraction. In [27] and [34] air charge for dual equal and intake only CVCP systems, that have moderate valve overlap, is studied. In [17] the focus is on fuel injection for the same type of engines as mentioned earlier.

The Models

Model A, in [15], is a simple generalized flow restriction model created with physical insight. Two model parameters are tuned to measurements or simulations of residual mass fractions.

Model B, from Ponti et al 2002 [43], and Model C, from Mladek et al. 2000 [35], are both based on thermodynamic relations, e.g. an energy balance at IVC. They both require measurement of cylinder pressure.

The reference model that is used is a crank angle based multi-zone model. A reference model is necessitated by the fact that it is very difficult to directly measure the actual residual mass fraction in a production engine.

The Investigation

An important issue in engine control are transients and the focus of this investigation is on how sensitive these three models are to transients in CVCP position. Therefore simulations have been performed to obtain cylinder pressure and reference data for static operating points and for transients in cam phasing. Only effects due to the different effective valve-lifts that is the results of transients in CVCP position are studied and the intake and exhaust manifold pressures are assumed to be measured.

8.1 Reference Model

As reference cylinder model a crank angle based, single cylinder, zero-dimensional, c.f. [44], multi-zone model with 1-D flow models for intake and exhaust ducts. The in cylinder model is tailored to track the flows of fresh and burned gases during the gas exchange period. It builds upon the method presented in [37] that solves the in-cylinder differential equations.

In the reference model the engine cycle is divided into three phases, gas exchange, compression and combustion. During the gas exchange and the combustion phase two zones are used. During the compression phase only one zone is used.

The two zones in the gas exchange phase are used to separate the residual gas and the fresh charge. When entering the compression phase the gases are instantaneously mixed into one zone. In the combustion phase one zone is used to track the unburned gases and the other is used to track the burned gas. Combustion is modeled as a mass transfer from the unburned to the burned zone.

Flow into and out of the cylinder is modeled by standard flow restrictions with constant discharge coefficients and the valve opening areas are modeled using the geometries and valve lifts from the engine. Intake and exhaust manifold dynamics are modeled using a pipe connected to the valve in one end and to a constant pressure source in the other. The pipe is divided into a configurable number of sections that all have their own pressure, temperature and mixture. All boundaries are associated with a plug-flow which defines the mass transfer between the sections. The velocity of the plug is modeled using Newtonian physics. This way the ram effects in the ducts are captured but further improvement could be made by using a multi cylinder model and including wave reflection for example from the supercharger.

Heating/cooling by the exhaust/intake ports is not considered. The fuel that enters the model is regarded as well mixed and no consideration is taken to charge cooling by evaporation.

In-cylinder heat transfer is modeled using Woschni [53] with constant wall temperature. For burn ratio the standard Vibe [51] function is used.

Thermochemical data for the model is obtained by using polynomials fitted to calculations using the chemical equilibrium program package, CHEPP, [13]. For the specific heat values, c_p and c_v , dissociation is considered but the specific gas constant, R , is constant. The fuel used is isooctane.

Geometries and valve timings for the model are those of a 2.0 dm³ turbocharged SAAB 9³ Sport Sedan engine (SAAB B207R) equipped with a CVCP mechanism. A complete list of parameters and constants used in the reference model can be found in Table 8.5 in the Appendix 8.B.

8.2 The Evaluated Models

8.2.1 Model A

Model A, in [15], defines an overlap factor (OF) that measures the amount of overlap between intake and exhaust valves. The overlap factor is then used as effective area in a generalized flow restriction model. The model has two parameters that need tuning. These parameters are tuned to residual mass fractions from measurements or simulations. The following quantities are needed when using the model

- Intake manifold pressure, p_{im}
- Exhaust manifold pressure, p_{em}
- Engine speed, N
- Fuel air ratio, Φ
- Valve positions (which gives OF)

Computational Scheme

The key element in Model A is the overlap factor which is defined as:

$$OF = \frac{D_i A_i + D_e A_e}{V_d} \quad (8.1)$$

where D_i and D_e are the inner seat diameters of the intake and exhaust valves respectively and V_d is the displacement volume of the engine. A_i and A_e are the areas under the valve-lift/crank-angle curves and are defined as:

$$A_i = \int_{IVO}^{L_i=L_e} L_i d\theta \quad \text{and} \quad A_e = \int_{L_i=L_e}^{EVC} L_e d\theta \quad (8.2)$$

where L_i and L_e are the intake and exhaust valve lifts respectively. In this implementation L_i and L_e have been calculated using the assumption that they are proportional to the valves effective flow areas, i.e. $L_i \propto \frac{A_{eff,i}}{D_i}$ and $L_e \propto \frac{A_{eff,e}}{D_e}$. Since the valve lifts in the calculation of A_i and A_e are small this has little effect on the calculation of OF .

Finally the residual mass fraction is calculated using the following expression:

$$x_{RG} = C_{1x_{RG}} \cdot \left(\frac{p_{em}}{p_{im}} \right)^{\left(\frac{\gamma+1}{2\gamma} \right)} \cdot \left(\frac{OF}{N} \right) \cdot \sqrt{p_{em} - p_{im}} \\ + C_{2x_{RG}} \cdot \frac{1}{\lambda \cdot r_c} \left(\frac{p_{em}}{p_{im}} \right)^{\frac{1}{\gamma}} \quad (8.3)$$

where $C_{1x_{RG}}$ and $C_{2x_{RG}}$ are constants that have to be fitted to data from measurements or simulations. These constants have been tuned to data from the reference model and the results are shown in Figure 8.9 in the Appendix.

Extension for Turbocharging

When using Model A for a turbocharged engine the pressure difference in (8.3), $p_{em} - p_{im}$, becomes negative for some operating points. To avoid taking the square root of negative numbers the absolute value of the pressure difference is used. The sign is then moved outside of the square root to obtain:

$$x_{RG} = C_{1x_{RG}} \cdot \left(\frac{p_{em}}{p_{im}} \right)^{\left(\frac{\gamma+1}{2\gamma} \right)} \cdot \left(\frac{OF}{N} \right) \cdot \text{sgn}(p_{em} - p_{im}) \cdot \sqrt{|p_{em} - p_{im}|} + C_{2x_{RG}} \cdot \frac{1}{\lambda \cdot r_c} \left(\frac{p_{em}}{p_{im}} \right)^{\frac{1}{\gamma}} \quad (8.4)$$

where sgn is the sign operator.

Since the first term of the equation may now become negative it is possible to obtain negative residual mass fractions. This would indicate flow through of fresh gas to the exhaust manifold. The residual mass fraction in the cylinder is zero under these conditions. Therefore the residual mass fraction is set to zero when negative values are obtained.

8.2.2 Model B

Model B is presented in [43] and is a simplification of Model C. The model was originally used for an engine with external EGR and VVT. External EGR is omitted in this implementation since the engine used lacks external EGR capabilities. The key element in this method is an energy balance for the mixing of ideal gases at IVC:

$$\begin{aligned} & (m_{FC} c_{v,FC} (T_{IVC}) + m_{RG} c_{v,RG} (T_{IVC})) T_{IVC} = \\ & = m_{FC} c_{v,FC} (T_{FC}) T_{FC} + m_{RG} c_{v,RG} (T_{RG}) T_{RG} \end{aligned} \quad (8.5)$$

where

m_{FC}	is mass of fresh charge
m_{RG}	is mass of residual gas
$c_{v,FC}$	is specific heat at constant volume for fresh charge
$c_{v,RG}$	is specific heat at constant volume for residual gas
T_{FC}	is temperature for fresh charge
T_{RG}	is temperature for residual gases
T_{IVC}	is mean gas temperature at IVC

Using this equation the residual mass can be calculated as long as T_{IVC} , T_{FC} and T_{RG} are known. Estimates of T_{FC} and T_{IVC} are calculated using measurements of the following quantities

- Intake manifold pressure, p_{im}
- In cylinder pressure, p_{cyl}

- Intake manifold temperature, T_{im}
- Air mass entering the cylinder, m_{FC}
- Engine speed, N

A good estimate of T_{RG} , however, is trickier and is therefore obtained using a correlation between T_{RG} and $m_{Tot} \cdot N$.

Algorithm

The first step of the algorithm in [43] is to adjust the cylinder pressure to the correct level. The reason for this is that the pressure measurement system used have a slowly varying offset. This adjustment is done by considering the cylinder pressure during the compression phase as a polytropic process for which $p \cdot V^\gamma$ is constant. Here p is the pressure, V the volume and γ the ratio of specific heats. A special variant of this technique is used in [43] but since the model is validated on simulated data no compensation is needed and therefore the description of this technique is omitted.

The temperature at IVC is calculated using cylinder pressure at IVC, measured air charge, m_{FC} , an estimate of the residual mass fraction, x_{RG} , and the following equations:

$$m_{Tot} = \frac{m_{FC}}{1 - x_{RG}} \quad (8.6)$$

$$R_{IVC} = R_{RG} x_{RG} + R_{FC}(1 - x_{RG}) \quad (8.7)$$

$$T_{IVC} = \frac{p_{IVC} V_{IVC}}{R_{IVC} m_{Tot}} \quad (8.8)$$

In the next step the temperature of the fresh charge is estimated. In [35] the estimation of the fresh charge temperature for this model is discussed in detail. It is stated that the heat flux from the walls to the fresh charge is partly compensated for the fuel evaporating process. Therefore, the fresh charge temperature can be estimated using a polytropic compression from manifold conditions to in-cylinder conditions:

$$T_{FC} = T_{im} \cdot \left(\frac{p_{im}}{p_{IVC}} \right)^{\frac{1-\gamma_{FC}}{\gamma_{FC}}} \quad (8.9)$$

where γ_{FC} is the ratio of specific heat for the fresh charge.

When it comes to the temperature for the residual gas other measures have to be taken. In [35] a correlation between mass flow and burned gas temperature at IVC is used. The model is:

$$T_{RG} = -(C_{1T_{RG}} \cdot (m_{Tot} \cdot N))^{C_{2T_{RG}}} + C_{3T_{RG}} \quad (8.10)$$

where the three constants $C_{1T_{RG}}$, $C_{2T_{RG}}$ and $C_{3T_{RG}}$ are estimated using simulations. Here these parameters have been tuned to data from the reference model and results from the tuning are shown in Figure 8.10 in the Appendix.

Finally the new residual mass fraction can be estimated using the energy balance at IVC in (8.5):

$$x_{RG} = \frac{\left(c_{v,FC}(T_{IVC}) T_{IVC} - c_{v,FC}(T_{FC}) T_{FC} \right) / \left(c_{v,RG}(T_{RG}) T_{RG} - c_{v,RG}(T_{IVC}) T_{IVC} + c_{v,FC}(T_{IVC}) T_{IVC} - c_{v,FC}(T_{FC}) T_{FC} \right)}{\quad} \quad (8.11)$$

where the specific heat values at constant volume $c_{v,FC}$, $c_{v,RG}$ and $c_{v,IVC}$ are the same as in the reference model.

Given an initial estimate of x_{RG} and using (8.6)–(8.11) a new x_{RG} can thus be calculated. The problem is solved by fixed point iteration which gives the final estimate of x_{RG} . Unfortunately the original fixed point iteration does not always converge. In [43] this has been addressed and a stabilizing scheme has been suggested. Instead of updating the residual estimate with the new value in each iteration the following weighting is performed:

$$x_{RG,i+1} = x_{RG,i} + \frac{\hat{x}_{RG} - x_{RG,i}}{2} \quad (8.12)$$

The iterative process is summarized by the flowchart in Figure 8.1.

Extension for Charge Heating

In [35] charge cooling due to evaporation is said to be partly compensated by charge heating by heat transfer from the cylinder wall. In the reference model charge cooling due to evaporation is not included. Therefore charge heating has to be added to compensate for this. A crude model, proposed in [38], is that the fresh charge estimate is heated about 10% by the cylinder walls. Equation (8.9) is therefore replaced by:

$$T_{FC} = 1.10 \cdot T_{im} \cdot \left(\frac{p_{im}}{p_{IVC}} \right)^{\frac{1-\gamma_{FC}}{\gamma_{FC}}} \quad (8.13)$$

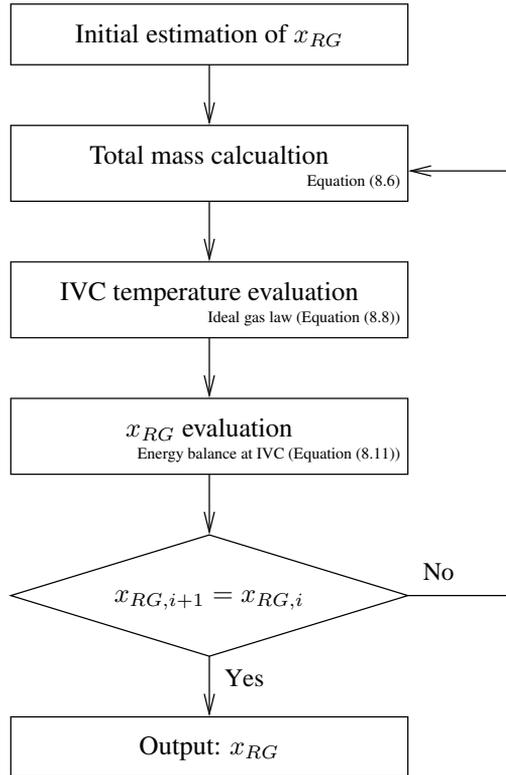


Figure 8.1: Flowchart of Model B. Using an initial estimate of x_{RG} a new value can be obtained using (8.6),(8.8) and(8.11). Repeating this in an iterative manner a final value of x_{RG} is obtained.

8.2.3 Model C

Model C is presented in [35] and its key elements are energy balance equations at two different points during the engine cycle. Using these equations both cylinder air charge and residual mass fraction can be calculated. In addition to well known thermodynamic relations two approximations are used to get estimates of the residual mass temperature at IVC and the relative heat loss to cylinder walls at $x_b = 50\%$.

The model requires on-line measurements of

- Intake manifold pressure, p_{im}
- In cylinder pressure, p_{cyl}
- Intake manifold temperature, T_{im}
- Engine speed, N

Original Algorithm

Model C uses much of the same principles as Model B. As mentioned before Model B is in fact a simplification of Model C. The difference is that the fresh charge, m_{FC} , is considered known in Model B but is estimated in Model C. Equations (8.7)–(8.13) are thus the same for Model C as for Model B.

One extra equation is needed to be able to estimate the total mass in the cylinder, m_{Tot} , energy balance at 50% mass fraction burned, $x_b = 50\%$, is therefore introduced. The position for $x_b = 50\%$ is estimated using a burn rate analysis as follows:

1. The end of combustion is found by using the following relation $ca_{EOC} = \text{argmax}(p \cdot V^{1.15}) + 10$, where ca_{EOC} is crank angle in degrees at end of combustion.
2. Weighting values are calculated so that the final x_b will be 0% at start of combustion and 100% at end of combustion. The weighting values are:

$$\begin{aligned} \dot{p}_{SOC} &= p_{SOC} \left(\frac{V_{SOC}}{V_{TDC}} \right)^{\gamma_{comp}} \\ \dot{p}_{EOC} &= p_{EOC} \left(\frac{V_{EOC}}{V_{TDC}} \right)^{\gamma_{exp}} \end{aligned} \quad (8.14)$$

where $_{SOC}$, $_{EOC}$, $_{TDC}$ is start of combustion, end of combustion and top dead center respectively and γ_{comp} and γ_{exp} are the polytropic indices for compression and expansion phases respectively. The values of the latter are set to 1.32 and 1.27, as suggested in [35].

3. x_b is then calculated as:

$$x_b = \frac{p \left(\frac{V}{V_{TDC}} \right)^{\gamma(ca)} - \dot{p}_{SOC}}{\dot{p}_{EOC} - \dot{p}_{SOC}} \quad (8.15)$$

where $\gamma(ca)$ is equal to γ_{comp} for $ca \leq ca_{TDC}$ and γ_{exp} for $ca > ca_{TDC}$.

4. Finally the crank angle for $x_b = 50\%$ is obtained from the estimated x_b .

The energy balance equation at $x_b = 50\%$ is stated and solved to give:

$$\bar{T}_{50} = \frac{\int_{ca_{IVC}}^{ca_{50}} p dV}{m_{Tot} c_{v50}} + \frac{T_{IVC} c_{vIVC}}{c_{v50}} + \frac{(1 - q_{cool}) \cdot 0.5 \cdot x_c \cdot (1 - x_{RG}) \cdot \frac{1}{1 + \lambda \cdot AF_s} \cdot q_{HV}}{c_{v50}} \quad (8.16)$$

where \bar{T}_{50} is the mean gas temperature at $x_b = 50\%$, q_{cool} is a factor describing the heat loss to the cylinder wall, x_c is a compensating factor for incomplete combustion, AF_s is the stoichiometric air fuel ratio, λ is the normalized air fuel ratio and q_{HV} is the higher heating value of the fuel.

Since q_{cool} is unknown it has to be estimated and this is done using the following correlation:

$$q_{cool} = C_{1q_{cool}} \cdot \bar{T}_{50} - C_{2q_{cool}} \quad (8.17)$$

where $C_{1q_{cool}}$ and $C_{2q_{cool}}$ are constants that have to be tuned to data from simulations. Values obtained for the reference model can be found in the Appendix where Figure 8.11 shows the result of the fit.

Algorithm Modification

Using the $c_{v,50}$ calculated from the gas composition, i.e. calculated in the same way as for the reference model, will not give accurate results when using the energy balance in (8.16) above. This is because:

$$\Delta U = \int_{T_1}^{T_2} c_v(T) dT = m c_{v2} T_2 - m c_{v1} T_1$$

is only guaranteed to hold if $c_v(T)$ is constant. Since $c_v(T) \approx c_{v1} + \frac{c_{v2} - c_{v1}}{T_2 - T_1} (T - T_1)$ for the typical temperature interval (8.16) has been modified in this implementation. Another issue is that $\int_{ca_{IVC}}^{ca_{50}} p dV$ is negative up till TDC since the volume is decreasing. Since work is added to the gas during this period a sign shift is necessary for the equation to be correct. The final expression is therefore:

$$\bar{T}_{50} = T_{IVC} - \frac{\int_{ca_{IVC}}^{ca_{50}} p dV}{m_{Tot} (c_{v,IVC} + c_{v,50})/2} + \frac{(1 - q_{cool}) \cdot 0.5 \cdot x_c \cdot (1 - x_{RG}) \cdot \frac{1}{1 + \lambda \cdot AF_s} \cdot q_{HV}}{(c_{v,IVC} + c_{v,50})/2} \quad (8.18)$$

note, once again, that $c_{v,50}$ and $c_{v,IVC}$ are functions of their respective temperature.

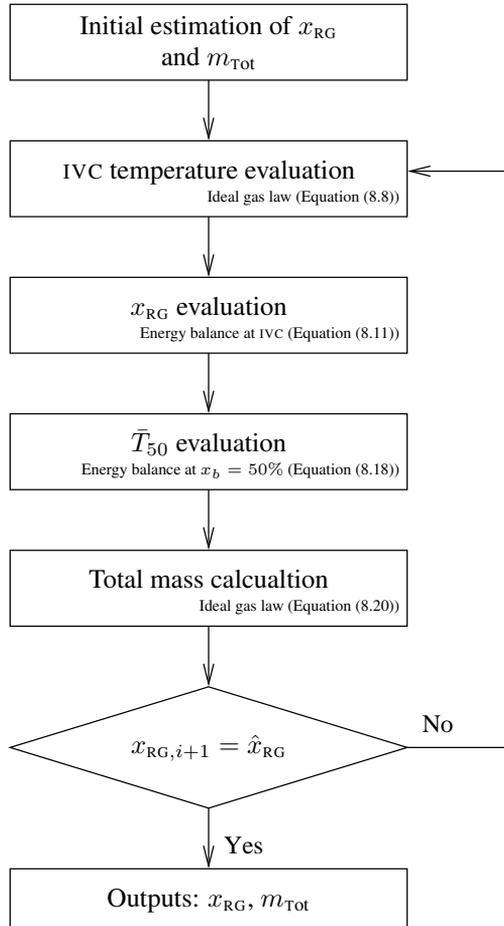


Figure 8.2: Flowchart of Model C. Using an initial estimate of x_{RG} and m_{Tot} , new values can be obtained using (8.8),(8.11),(8.18) and (8.20). Repeating this in an iterative manner final values of x_{RG} and m_{Tot} is obtained.

Finally, when \bar{T}_{50} is known a new total mass estimate can be performed. In the model no dissociation is considered. Therefore the gas constant for the composition is straight forward to calculate and is equal to R_{IVC} . Hence the total mass in the cylinder can be calculated using the relations:

$$R_{50} = R_{IVC} \quad (8.19)$$

$$m_{Tot} = \frac{p_{50} V_{50}}{R_{50} T_{50}} \quad (8.20)$$

Complete Algorithm

The residual mass fraction, x_{RG} , can be calculated using an initial estimate of x_{RG} and the total mass in the cylinder, m_{Tot} , much in the same way as in Model B. A flowchart of the process is shown in Figure 8.2.

8.3 Investigation Setup

The CVCP mechanism on the engine has a cam phase shifting speed of 100 deg/s measured in terms of crank angle revolution. This corresponds to a change of about 3 degrees in intake valve duration if a transient in cam position is triggered at intake valve opening when running at 800 rpm's. The purpose of this investigation is to capture the effect that this difference in duration has on the air charge and residual mass fraction and to evaluate the performance of Model A – C under these conditions.

Simulations have been performed to obtain cylinder pressure and reference data for static operating points and for transients in cam phasing. The simulations are designed to capture the difference in air charge and mass flow between the last static cycle before the transient and the first cycle during the transient. In Figure 8.3 the two concepts *static*– and *transient cycle* are defined. The last cycle before the transient will from hereon be referred to as the *static cycle* while the first cycle in the transient will be referred to as the *transient cycle*.

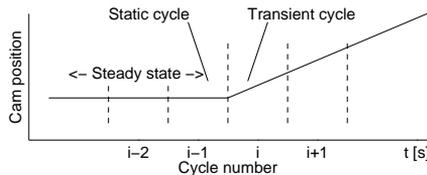


Figure 8.3: Definition of transient and static cycle. The last cycle before the transient is referred to as the static cycle while the first cycle of the transient is referred to as the transient cycle.

Both the static and the transient cycles are recorded and the data for the static cycles and the transient cycles are then used to evaluate Model A – C.

Table 8.1: Investigated operating points. Here *iMOP* and *eMOP* are intake and exhaust valve maximum opening position respectively.

Engine speed	1000,3000,5000	RPM
Exhaust manifold pressure	100,130	kPa
Intake manifold pressure	40,70,100,130,160	kPa
<i>iMOP</i>	435,464,475	deg
<i>eMOP</i>	240,252,280	deg

The transients are simulated by steps up and down in cam phasing for all static operating points. The steps have been large enough not to finish within the actual cycle and all combinations of positive and negative steps have been performed for both the intake and exhaust cam. Altogether this yields 8 different types of steps. For transients in intake cam position the transient is triggered at IVO and for transients in exhaust cam position the transient is triggered at EVO.

The static operating points are shown in Table 8.1. However, operating points that give residual mass fractions above 50% are removed since the engine does not run in these points. Altogether 36 of the 2430 simulations are removed because of this.

8.4 Deviations for Transient Cycles

In Figure 8.4 the differences in residual mass fraction and total mass in the cylinder between static and transient cycles are shown. The relative difference in mass flow can be as large as 80% and the difference in residual mass fraction can be as large as 20%. Because of the low number of operating points for each variable it is not advantageous to plot the differences in mass flow and residual mass fraction against any of them. It is however possible to see a strong dependence on engine speed which occurs because the relative speed of the CVCP mechanism is higher for lower engine speeds.

8.5 Evaluation of Model Performance

The transient and the static cycles are used to evaluate the performance of Model A – C. To be able to quantify the performance of the evaluated models the term *detrended error* is introduced. The detrended errors is defined as follows

$$e_{d,i} = x_i - \frac{(\hat{x}_i - m)}{k}$$

where k and m are the values that minimize $\sum_i |e_{d,i}|^2$ and \hat{x} is the estimate of x . The standard deviation of the detrended error may now be used to quantify the spreading of data around a thought straight-line fit of a plotted data set from an evaluated model.

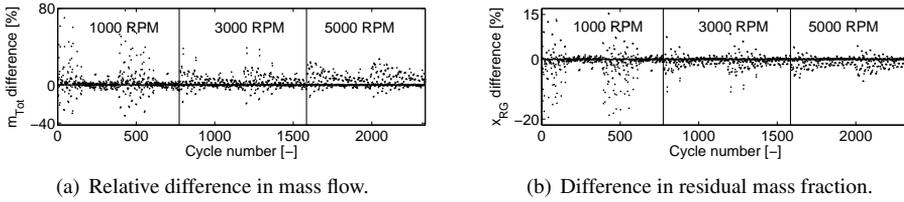


Figure 8.4: Comparison between static cycles and transient cycles. As can be seen both in (a) and (b) the differences are larger for low engine speeds. For 1000 rpm the maximum difference in mass flow is about 80% and for 3000 rpm the maximum difference is about 40%. The same effect is visible for the residual mass fraction where the maximum difference at 1000 rpm is about 20% while the maximum difference at 3000 rpm is about 10%. The reason for the higher impact at low speeds this is that the relative speed of the CVCP mechanism is higher for lower engine speeds.

Model A

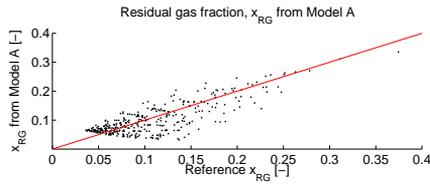
The main advantage with Model A is that it does not require measurement of in cylinder pressure. Another advantage is that it directly yields the residual mass fraction by the use of simple calculations. As long as it is possible to pre-calculate everything that is not directly measurable this is very efficient. This is the case for static operating points since the overlap factor, OF, only depends on the difference in intake and exhaust maximum opening position $iMOP$ and $eMOP$. For transients however the overlap factor has to be calculated for the actual effective valve lifts which may be different for every single cycle during the transient.

Another issue with Model A is that while it is feasible to obtain data for the calculation of the constants $C_{1x_{RG}}$ and $C_{2x_{RG}}$ in static operating points it is problematic to include transient cycles in the model tuning. This might be a problem if the residual mass fractions is not only dependent of overlap factor but also the shape of the valve lift curves.

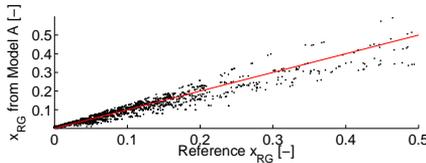
Altogether it is interesting to see how Model A performs if the overlap factors of the static cycles are used in contrast to the performance when the overlap factors are recalculated with the effective valve lifts of the transient cycles.

In Figure 8.5 the performance when used on static cycles (a), transient cycles with recalculated overlap factor (b) and transient cycles with overlap factor from the static cycle (c) are shown. Since there are 8 times as many different operating points for the transient cycles than for the static cycles it is hard to spot any difference between Figure 8.5(a) and (b). Therefore the standard deviation of the detrended errors have been used to quantize the differences. The standard deviations of the detrended errors are shown in Table 8.2.

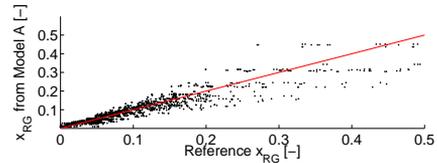
The increase in standard deviation is about 17% between the results in Figure 8.5(a) and (b). The difference between 8.5(b) and (c) is another 33% relative to 8.5(a). The increase in error in Figure 8.5(c) looks smaller at first sight but remember that the number of different operating points are the same as in Figure 8.5(b).



(a) Static cycles.



(b) Transient cycles with recalculated overlap factor.



(c) Transient cycles with same overlap factor as the static operating points.

Figure 8.5: Residual gas fraction estimates, x_{RG} , from Model A. The spread in x_{RG} for the transient cycles are larger than for the static cycles but remember that there are 8 times as many operating points in (b) compared to (a). Using the overlap factor from the static cycles and comparing with the residual mass fractions from the transient cycles decreases the performance even more (c).

Table 8.2: Standard deviations of detrended errors for Model A

Type of investigation	$std(e_{d,i})$
Static cycles	2.62×10^{-2}
Transient cycles with recalculated OF	3.07×10^{-2}
Transient cycles with static OF	3.94×10^{-2}

Model B

For Model B it might look like there is no reason why the performance should be different between the transient cycles and the static cycles since the estimation is based on measurable quantities that is not affected by the cam phasing. There are however some caveats. As shown in Figure 8.4(a) the difference in mass flow might be as large as 80% between two consecutive cycles. Model B relies on the measured air flow and since there are delays and dynamics in air flows it is likely that Model B is fed the wrong air mass flow during transients. For model B this gives two possible problems.

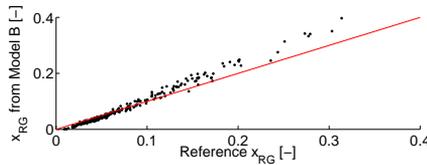
Firstly, the residual gas temperature of the first transient cycle is that of the last static cycle. When calculating the residual gas temperature it is necessary to use the mass flow of the last cycle. However, the model that predicts the residual gas temperature is a rather crude model and in [38] it is shown that Model B is not very sensitive to this type of error. Secondly, in [38] it is also concluded that the residual mass fraction estimate of Model B (and C) is sensitive to errors in its input quantities coupled to the fresh gas and that small errors in mass flow estimates may result in large errors in residual mass fraction estimates.

In Figure 8.6 the residual fraction estimates are shown for both transient and static cycles. In Figure 8.6(a) and (b) it can be seen that residual mass fraction estimates are almost as good for transient cycles as for static cycles. This is also supported by the standard deviation of the detrended errors in Table 8.3 where it can be seen that the increase in standard deviation is about 13%. Note that the standard deviations for Model B are one order of magnitude lower than the standard deviations for Model A. Using the wrong

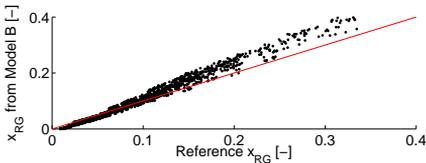
Table 8.3: Standard deviations of detrended errors for Model B

Type of investigation	$std(e_{d,i})$
Static cycles	7.60×10^{-3}
Transient cycles	8.58×10^{-3}
Transient cycles with mass estimate from previous cycle	2.94×10^{-2}

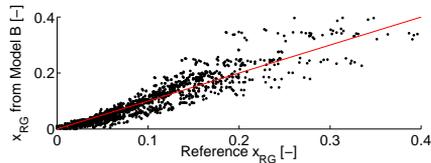
mass estimate does however have a detrimental effect on the residual mass fraction as can be seen in Figure 8.6(c). The increase in standard deviation is about 287% which takes it is close to the standard deviation of Model A.



(a) Static cycles.



(b) Transient cycles.



(c) Transient cycles using the previous cycles' mass flow in estimates.

Figure 8.6: Residual gas fraction estimates for Model B. The errors for the transient cycles in (a) are of the same size as for the static cycles in (b). Using the wrong air mass estimate however lowers the performance substantially. Note once again that there are 8 times as many different transient cycles than static cycles.

Model C

In Figure 8.7 it can, in the same way as with Model B, be seen that the performance of the residual mass estimation for Model C does not degrade for transient cycles. The increase in standard deviation is as small as 3% as can be seen in Table 8.4. An interesting point is that while the residual mass estimates are worse than for Model B with the correct mass flow they are better than for Model B using the wrong mass flow. This is because Model C has the benefit of using its own mass flow estimate in the estimation process.

Table 8.4: Standard deviations of detrended errors for Model C

Type of investigation	$std(e_{d,i})$	
Static cycles	m_{Tot}	1.13×10^{-5}
	x_{RG}	1.34×10^{-2}
Transient cycles	m_{Tot}	1.17×10^{-5}
	x_{RG}	1.38×10^{-2}

The performance of the mass flow estimation is shown in Figure 8.8. The difference in error between the mass flows estimates is hardly visible and the standard deviations differs with about 3.5%.

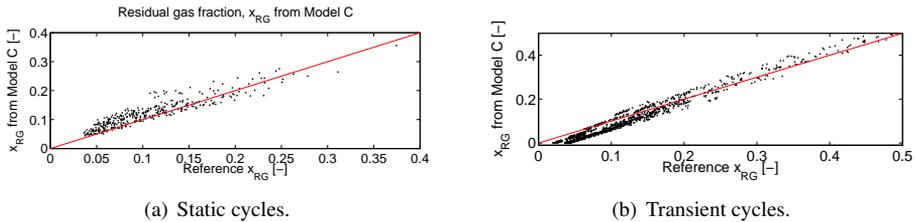


Figure 8.7: Residual gas fraction estimates for Model C. The errors for the transient cycles in (a) are of the same size as for the static cycles in (b). Note once again that there are 8 times as many different transient cycles than static cycles.

Conclusion

A CVCP engine has been simulated both in static operating points and with transients in CVCP position. It has been shown that transients in cam positions have a large impact on air charge and residual mass fraction.

Three models, that predict residual mass fraction of which one also predicts air charge, have been investigated and the models' sensitivity to transients have been studied.

Model A is very simple and it does not require measurement of the cylinder pressure but it is also the model with the largest errors and highest sensitivity to transients. The errors

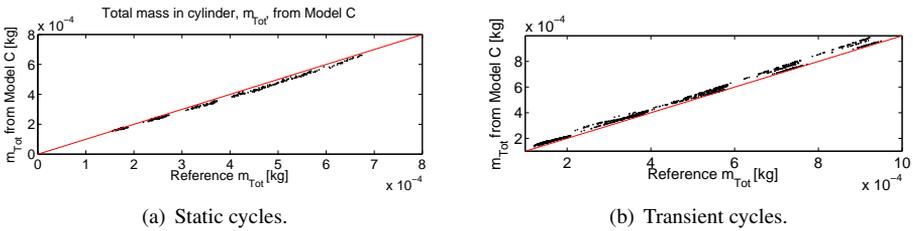


Figure 8.8: Total mass estimates for Model C. The errors for the transient cycles in (a) are of the same size as for the static cycles in (b). Even though there are 8 times as many transient cycles as static cycles the difference is hardly visible.

for the transient cycles are of the same magnitude as the errors for the static cycles even if the method is not adjusted to handle the transients. It is however, amongst the evaluated models, the model that gives the largest errors both for static cycles and for transient cycles.

Model B and C handles the transient cycles well when used correctly. The errors for the transient cycles are of the same magnitude as the errors for the static cycles for both models.

For static cycles it is advantageous to use an external measurement of air mass, as in Model B, since it makes the model less sensitive to input data. During transients however, if the external measurement is delayed, it is better to use Model C.

Acknowledgements

The Swedish Foundation for Strategic Research is gratefully acknowledged for their funding.

Appendix

8.A Nomenclature

Special quantities

p	Pressure
V	Volume
\bar{T}	Average temperature
AF_s	Stoichiometric air fuel ratio
λ	Normalized air fuel ratio
Φ	Normalized fuel air ratio
q_{cool}	Heat loss due to cooling
q_{HV}	Higher heating value of fuel
x_c	Maximal combustion due to imperfections
p_{cyl}	Crank angle based cylinder pressure
c_v	Specific heating value for constant volume
c_p	Specific heating value for constant pressure
R	Mass specific gas constant
r_c	Compression ratio
γ	Ratio of specific heating values ($\frac{c_p}{c_v}$)
OF	Overlap factor
V_d	Displacement volume of engine
N	Engine speed in rps
D_i	Inner seat diameter of intake valve
D_e	Inner seat diameter of exhaust valve
$A_{eff,i}$	Effective flow area of intake valve
$A_{eff,e}$	Effective flow are of exhaust valve
L_i	Intake valve lift
L_e	Exhaust valve lift
x_{RG}	Residual mass fraction
x_b	Burn ratio

Abbreviations

EGR	Exhaust Gas Recirculation
VVT	Variable Valve Timing
CVCP	Continuously Variable Cam Phasing
ECU	Engine Control Unit
IVC	Intake Valve Closure
IVO	Intake Valve Opening
TDC	Top Ded Center
EVC	Exhaust Valve Closure

Used indices

IVC	At Intake valve closure
SOC	At combustion start
50	At $x_b = 50\%$
EOC	At end of combustion
comp	During compression
exp	During expansion
RG	For Residual Gas part
FC	For Fresh Charge part
Tot	Total
im	In Intake Manifold
em	In Exhaust Manifold
cyl	In the cylinder

8.B Parameters for the Reference Model

A complete list of parameters and constants used in the reference model can be found in Table 8.5.

Table 8.5: Parameters and constants for the reference model.

Parameter	Value	Description
ϕ_{im}	1 [-]	Intake manifold fuel air ratio
$C_{d,iv}$	0.684 [-]	Intake valve discharge coefficient
$C_{d,ev}$	0.5 [-]	Exhaust valve discharge coefficient
T_{wall}	470 [K]	Cylinder wall temperature
T_{im}	310 [K]	Mean value of intake manifold temperature
T_{em}	800 [K]	Mean value of exhaust manifold temperature
L_{im}	20 [cm]	Intake manifold pipe length
A_{im}	25 [cm ²]	Intake manifold pipe area
N_{im}	3 [-]	Number of segments for intake manifold
L_{em}	10 [cm]	Exhaust manifold pipe length
A_{em}	12.5 [cm ²]	Exhaust manifold pipe area
N_{em}	3 [-]	Number of segments for exhaust manifold
S_{iv}	100 [deg/s]	Speed of CVCP mechanism for intake valve
S_{ev}	100 [deg/s]	Speed of CVCP mechanism for exhaust valve
$C_{1,heat}$	1	Woschni heat transfer coefficient
$C_{2,heat}$	0.4386	Woschni heat transfer coefficient
$\theta_{s,vibe}$	340 [deg]	Vibe start of combustion
$\theta_{e,vibe}$	400 [deg]	Vibe end of combustion
m_{vibe}	2 [-]	Vibe shaping parameter
a_{vibe}	6.9 [-]	Vibe shaping parameter

8.C Model Parameters

Tuning of Parameters for Model A

The correlation for x_{RG} in (8.4), Section 8.2.1, is directly used in Model A as the single most important component. The equation is repeated here for reference:

$$x_{RG} = C_{1x_{RG}} \cdot \left(\frac{p_{em}}{p_{im}} \right)^{\left(\frac{\gamma+1}{2\gamma} \right)} \cdot \left(\frac{OF}{N} \right) \cdot \text{sgn}(p_{em} - p_{im}) \cdot \sqrt{|p_{em} - p_{im}|} + C_{2x_{RG}} \cdot \frac{1}{\lambda \cdot r_c} \left(\frac{p_{em}}{p_{im}} \right)^{\frac{1}{\gamma}}$$

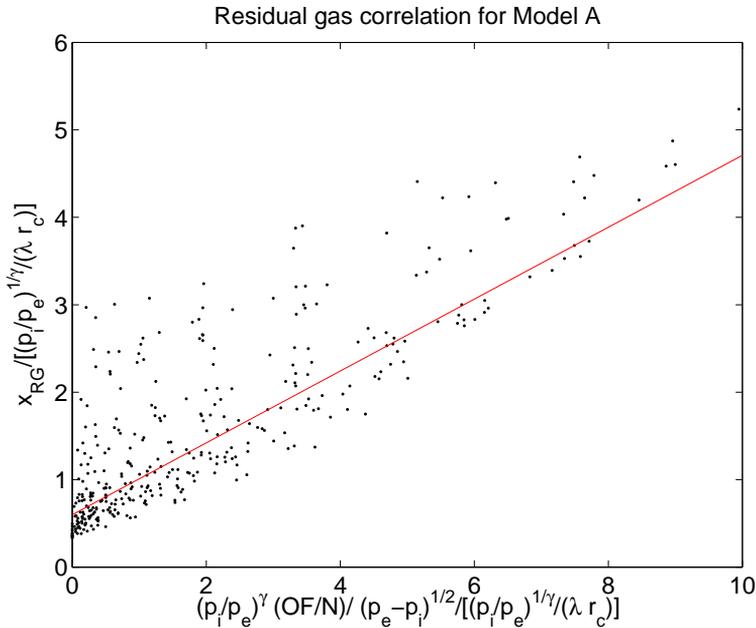


Figure 8.9: The constants $C_{1x_{RG}}$ and $C_{2x_{RG}}$ can be taken directly from the figure or fitted using a least square fit.

The parameters can be found by using a least square fit or by studying the data in Figure 8.9. The parameters have been fitted to:

$$C_{1x_{RG}} = 0.1578$$

$$C_{2x_{RG}} = 0.4712$$

Model for T_{RG}

The model for T_{RG} in (8.10), Section 8.2.1, is used both in Model B and Model C. The equation is repeated here for reference:

$$T_{RG} = -(C_{1T_{RG}} \cdot (m_{Tot} \cdot N))^{C_{2T_{RG}}} + C_{3T_{RG}}$$

Looking at Figure 8.10 there is little chance of getting a good fit between the data and the model used. This is not surprising since a CVCP engine has been used instead of a conventional engine as in [35]. For a CVCP engine the residual gas temperature, T_{RG} , dependence on cam timing is much stronger than the dependence on mass flow.

The model has been tuned to the data in Figure 8.10 with the following values:ifp-

$$C_{1T_{RG}} = 0.02456 \quad [-]$$

$$C_{2T_{RG}} = -0.6960 \quad [-]$$

$$C_{3T_{RG}} = 1101.2 \quad [K]$$

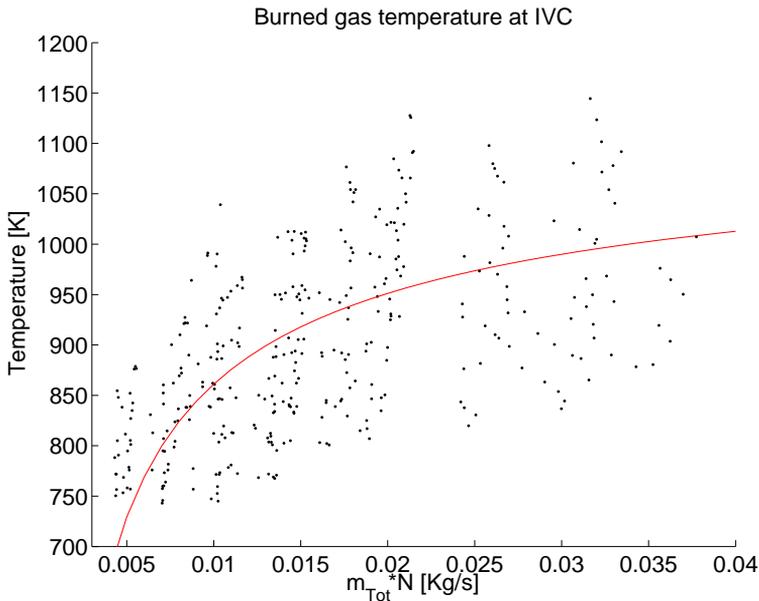


Figure 8.10: The correlation between mass flow and temperature of residual gas at IVC is weak for a CVCP engine.

Model for q_{cool}

q_{cool} is introduced in (8.17), Section 8.2.1. The equation is repeated here for reference:

$$q_{\text{cool}} = C_{1q_{\text{cool}}} \cdot \bar{T}_{50} - C_{2q_{\text{cool}}}$$

For q_{cool} the results are somewhat better than for the T_{RG} model. This can be seen in Figure 8.11. The model has been tuned to the data with the parameters:

$$C_{1q_{\text{cool}}} = -4.5226 \times 10^{-4} \quad [\text{K}^{-1}]$$

$$C_{2q_{\text{cool}}} = 0.8680 \quad [\text{K}]$$

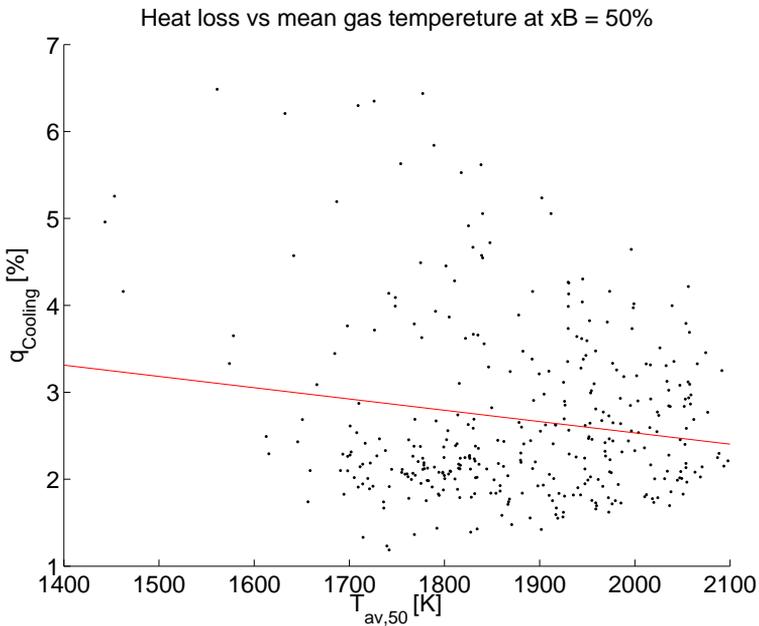


Figure 8.11: Heat loss vs mean gas temperature for a CVCP engine.

Bibliography

- [1] Ingemar Andersson and Lars Eriksson. A parametric model for ionization current in a four stroke si engine. *Journal of Dynamic Systems, Measurement, and Control*, 131(2), March 2009.
- [2] Uri M. Ascher and Linda R. Petzold. *Computer Metehods for Ordinary Differential Equations*. Siam, 1998.
- [3] P. W. Atkins. *Physical Chemistry*. Oxford University Press, fift edition, 1994.
- [4] AVL. Gu21d transducer uncooled, datasheet/specification. "<http://www.avl.com>", June 2009.
- [5] Gary L. Borman and Kenneth W. Ragland. *Combustion Engineering*. McGraw-Hill, international edition, 1998.
- [6] D. Brand. *Control-Oriented Modeling of NO Emissions of SI Engines*. PhD thesis, ETH, Swess Federal Institute of Technology, 2005.
- [7] Yunus A. Çengel and Robert H. Turner. *Fundamentals of Thermal-Fluid Sciences*. McGraw-Hill, international second edition, 2005.
- [8] Gunnar Cedersund. System identification of nonlinear thermochemical systems with dynamical instabilities. Technical report, Department of Electrical Engineering, Linköpings Universitet, 2004.
- [9] Daniel Claesson. Improved experimental agreement of ionization and pressure peak location by adding a dynamical no-model. Master's thesis, Linköpings Universitet, SE-581 83 Linköping, Sweden, 2004.
- [10] H. A. Cook. Digital computer analysis and interpretation of turbocharged diesel engine performance. *SAE Transactions*, 67:532–544, 1959.

- [11] John E. Dec. A conceptual model of di diesel combustion based on laser-sheet imaging. *SAE Technical Paper 970873*, 1997.
- [12] Lars Eriksson. Thermodynamics of unsteady flows and zero dimensional in-cylinder models. Internal course literature (Vehicular Systems).
- [13] Lars Eriksson. CHEPP – A chemical equilibrium program package for matlab. In *Modeling of Spark Ignition Engines*, number 2004-01-1460 in SAE Technical paper series SP-1830, 2004.
- [14] Lars Eriksson. Modeling and control of turbocharged si and di engines. *Oil & Gas Science and Technology - Rev. IFP*, 62(4):523–538, 2007.
- [15] Jonathan W. Fox, Wai K. Cheng, and John B. Heywood. A model for predicting residual gas fraction in spark-ignition engines. *SAE Technical Paper 931025*, 1993.
- [16] J. A. Gatowski, E. N. Balles, K. M. Chun, F. E. Nelson, J. A. Ekchian, and John B. Heywood. Heat release analysis of engine pressure data. *SAE Technical Paper 841359*, 1984.
- [17] Ali Umut Genc. *Linear Parameter-Varying Modelling and Robust Control of Variable Cam Timing Engines*. PhD thesis, Wolfson College Cambridge, 2002.
- [18] Pascal Giansetti. *Contrôle moteur à allumage commandé, Estimation / prédiction de la masse e de la composition du mélange enfermé dans le cylindre*. PhD thesis, Université D’Orleans, September 2005.
- [19] Johan Gill. A java package for simulation of combustion engines. Master’s thesis, Linköpings Universitet, SE-581 83 Linköping, Sweden, 2003.
- [20] Gordon and McBride. Computer program for calculation of complex chemical equilibrium compositions, rocket performance, incident and reflected shocks and chapman-jouget detonations. Technical Report SP-273, National Aeronautics and Space Administration, 1976. N78-17724.
- [21] Lino Guzzella and Christofer H. Onder. *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Springer, 2004.
- [22] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II*. Springer, second revised edition, 1996.
- [23] Shahrokh Hajireza, Bengt Sundén, and Fabian Mauss. A three-zone model for investigation of gas behaviour in the combustion chamber of si engines in relation to knock. *SAE Technical Paper 1999-01-0219*, 1999.
- [24] Pär Hammarlund. Simulation and analysis of a continuously variable cam phasing engine. Master’s thesis, Linköpings Universitet, SE-581 83 Linköping, 2008.
- [25] John B. Heywood. *Internal Combustion Engine Fundamentals*. McGraw-Hill series in mechanical engineering. McGraw-Hill, 1988.

- [26] Günter F. Hohenberg. Advanced approaches for heat transfer calculations. *SAE Technical Paper 790825*, 1979.
- [27] Mrdjan Jankovic and Stephen W. Magner. Variable cam timing: Consequences to automotive engine control design. *IFAC World Congress*, 2002.
- [28] Fredrik Jones and Christoffer Jezek. Diesel combustion modeling and simulation for torque estimation and parameter optimization. Master's thesis, Linköpings Universitet, SE-581 83 Linköping, 2008.
- [29] Komiyama K. and Heywood J. B. Predicting nox emissions and effects of exhaust gas recirculation in spark ignition engines. *SAE Technical Paper 730475*, *SAE Trans.*, 82, 1973.
- [30] Stig Kildegård Andersen, Henrik Carlsen, and Per Grove Thomsen. Control volume based modelling in one space dimension of oscillating, compressible flow in reciprocating machines. *Simulation Modelling Practice and Theory*, 14(8):1073–1086, November 2006.
- [31] Charles Kittel and Herbert Kroemer. *Thermal Physics*. W.H. Freeman and Company, second edition, 1980.
- [32] Marcus Klein. A specific heat ratio model and compression ratio estimation. Technical report, Linköpings Universitet, 2004. LiU-TEK-LIC-2004:33, Thesis No. 1104.
- [33] R. B. Krieger and G. L. Borman. The computation of apparent heat release for internal combustion engines. *ASME*, 1967.
- [34] Steve Magner, Mrdjan Jankovic, and Stephen Cooper. Methods to reduce air-charge characterization data for high degree of freedom engines. *SAE Technical Paper 2004-01-0903*, 2004.
- [35] Michael Mladek and Christopher H. Onder. A model for the estimation of inducted air mass and the residual gas fraction using cylinder pressure measurements. *SAE Technical Paper 2000-01-0958*, 2000.
- [36] Ylva Nilsson. *Modelling for Fuel Optimal Control of a Variable Compression Engine*. PhD thesis, Linköpings Universitet, 2007.
- [37] Ylva Nilsson and Lars Eriksson. A new formulation of multi-zone combustion engine models. *IFAC Workshop: Advances in Automotive Control*, pages 629–634, Karlsruhe, Germany, 2001.
- [38] Per Öberg and Lars Eriksson. Control oriented modeling of the gas exchange process in variable cam timing engines. In *SAE Technical Paper 2006-01-0660*, April 2006. SAE World Congress 2006.
- [39] Per Öberg and Lars Eriksson. Control oriented gas exchange models for CVCP engines and their transient sensitivity. *Oil & Gas Science and Technology - Rev. IFP*, 62(4):573–584, 2007.

- [40] Cherian Olikara and Gary L. Borman. A computer program for calculating properties of equilibrium combustion products with some applications to i.c. engines. *SAE Technical Paper 750468*, 1975.
- [41] D. J. Patterson and G. Van Wylen. A digital computer analysis of spark-ignited engine cycles. *SAE Technical Paper 633F*, 1963.
- [42] C. Pianese, G. Rizzo, A. Caraceni, P. Cioffi, G. Flauti, and I. Arsie. Thermodynamic modeling of jet formation and combustion in common rail multi-jet diesel engines. *SAE Thechnical Paper 2005-01-1121*, 2005.
- [43] F. Ponti, J. C. Piani, and R. Suglia. Residual gas model for on-line estimation for inlet and exhaust continous VVT engine configuration. *In proceedings of IFAC symposium on Advances in Automotive Control, Salerno, Italy*, 2004.
- [44] J. I. Ramos. *Internal combustion engine modeling*. Hemisphere Publishing Corporation, 1989.
- [45] Gerald M. Rassweiler and Lloyd Withrow. Motion pictures of engine flames correlated with pressure cards. *SAE Technical Paper 380139*, 1938.
- [46] W.C. Reynolds. The element potential method for chemical equilibrium analysis: Implementation in the interactive program STANJAN. Technical report, Department of Mechanical Engineering, Stanford University, 1986.
- [47] Gregory P. Smith, David M. Golden, Michael Frenklach, Nigel W. Moriarty, Boris Eiteneer, Mikhail Goldenberg, C. Thomas Bowman, Ronald K. Hanson, Soonho Song, William C. Gardiner Jr., Vitali V. Lissianski, and Zhiwei Q. Gri-mech 3.0. "http://www.me.berkeley.edu/gri_mech", June 2009.
- [48] Ricardo Software. Wave product information. "<http://www.ricardo.com/software>", June 2009.
- [49] F. Douglas Swesty. Thermodynamically consistent interpolation of equation of state tables. *Journal of Computational Physics*, 127:118–127, 1996.
- [50] Gamma Technologies. Gt-suite, virtual engine / powertrain / vehicle. "http://www.gtisoft.com/img/broch/broch_gtsuite.pdf", June 2009.
- [51] I. I. Vibe. *Brennverlauf und Kreisprozess von Verbrennungsmotoren*. VEB Verlag Technik Berlin, 1970. German translation of the russian original.
- [52] N. D. Whitehouse, A. Stotter, G. D. Goudie, and B. W. Prentice. Method of predicting some aspects of performance of a diesel engine using a digital computer. *Proceedings of Institution of Mechanical Engineers*, 176(9), 1962.
- [53] G. Woschni. A universally applicable equation for the instantaneous heat transfer coefficient in the internal combustion engine. *SAE Technical Paper 670931*, 1967.

A

Notational Conventions

Abbreviations

BDC	Bottom Dead Center
CAC	Cylinder Air Charge
CVCP	Continuously Variable Cam Phasing
DAE	Differential Algebraic Equation
ECU	Engine Control Unit
EGR	Exhaust Gas Recirculation
EOC	End Of Combustion
EOS	End Of Simulation
EVC	Exhaust Valve Closure
EVO	Exhaust Valve Opening
FC	Fresh Charge
GUI	Graphical User Interface
IVC	Intake Valve Closure
IVO	Intake Valve Opening
MOP	Maximum Open Position
ODE	Ordinary Differential Equation
RG	Residual Gas
SI	Spark Ignited
SOC	Start Of Combustion
TDC	Top Dead Center
VVT	Variable Valve Timing

Used Indices

50	At $x_b = 50\%$
comp	During compression
CS	At combustion start
cyl	In the cylinder
em	In Exhaust Manifold
EOC	At end of combustion
exp	During expansion
FC	For Fresh Charge part
im	In Intake Manifold
IVC	At Intake valve closure
RG	For Residual Gas part
SOC	At combustion start
Tot	Total

Special Quantities

AF_s	Stoichiometric air fuel ratio
$A_{eff,e}$	Effective flow area of exhaust valve
$A_{eff,i}$	Effective flow area of intake valve
D_e	Inner seat diameter of exhaust valve
D_i	Inner seat diameter of intake valve
L_e	Exhaust valve lift
L_i	Intake valve lift
$m_{air,s}$	Simulated air mass per combustion
$m_{air,m}$	Measured air mass per combustion
N	Engine speed in rps
OF	Overlap factor
p_{cyl}	Crank angle based cylinder pressure
$p_{m,i}$	Sample i of measured cylinder pressure
p_o	Cylinder pressure sensor offset
$p_{s,i}$	Sample i of simulated cylinder pressure
$q_{Cooling}$	Heat loss due to cooling
q_{HV}	Higher heating value of fuel
r_c	Compression ratio
\bar{T}	Average temperature
V_d	Displacement volume of engine
x_b	Burn ratio
x_{ba}	Fraction of burned air
x_{bf}	Fraction of burned fuel
x_c	Maximal combustion due to imperfections
x_{ua}	Fraction of unburned air
x_{uf}	Fraction of unburned fuel
x_{RG}	Residual mass fraction
λ	Normalized air fuel ratio
Φ	Normalized fuel air ratio
γ	Ratio of specific heating values ($\frac{c_p}{c_v}$)

Thermodynamic Properties

E	Total energy
H	Enthalpy
m	Mass
M	Molar weight
n	Number of moles of substance
p	Pressure
Q	Accumulated heat transfer
R	Mass specific gas constant
S	Entropy
T	Temperature
U	Internal energy
V	Volume
W	Accumulated work
\sim	Mole specific property
$-$	Average value
\wedge	Value where selection is necessary

Thermochemical Properties

c_p	Heating value at constant pressure
c_v	Heating value at constant volume
R	Gas constant
u	Internal energy
h	Enthalpy
$\frac{\partial R}{\partial p}$	Derivative of gas constant w.r.t pressure
$\frac{\partial R}{\partial T}$	Derivative of gas constant w.r.t time
$\frac{\partial u}{\partial p}$	Derivative of internal energy w.r.t pressure
$\nabla_{x,r} u$	Derivative of internal energy w.r.t composition
$\nabla_{x,r} R$	Derivative of gas constant energy w.r.t composition

Notation for Composition Properties

\mathbf{x}	Vector of mass fractions of molecule species
\mathbf{x}_r	Vector of mass fractions of reactant atoms
x_k	Mass fraction of molecule specie k
\mathbf{u}	Vector of mass specific internal energy of species
u_k	Mass specific internal energy of molecule specie k
$\tilde{\mathbf{x}}$	Vector of mole fractions of molecule species
$\tilde{\mathbf{x}}_r$	Vector of mole fractions of reactant atoms
\tilde{x}_i	Mole fraction of molecule specie i
$\tilde{\mathbf{u}}$	Vector of mole specific internal energy of species
\tilde{u}_k	Mole specific internal energy of specie k

Frequently Referenced Equations

The Maxwell Relations

Table 2.1: The four Maxwell Relations derived from the internal energy, the enthalpy, the Helmholtz energy, and Gibbs function.

$$\begin{aligned} \left(\frac{\partial T}{\partial V}\right)_V &= -\left(\frac{\partial p}{\partial S}\right)_V & \left(\frac{\partial T}{\partial p}\right)_S &= \left(\frac{\partial V}{\partial S}\right)_p \\ \left(\frac{\partial p}{\partial T}\right)_V &= \left(\frac{\partial S}{\partial V}\right)_T & \left(\frac{\partial V}{\partial T}\right)_p &= -\left(\frac{\partial S}{\partial p}\right)_T \end{aligned}$$

The Control Volume Model

$$\begin{aligned} m c_v dT + m \frac{\partial u}{\partial p} dp + p dV &= dQ + \sum_j \left((\nabla_{\mathbf{x}} u)^T (\mathbf{x} - \hat{\mathbf{x}}_{d,j}) + h(p, \hat{T}_{s,j}, \hat{\mathbf{x}}_{s,j}) - u \right) dm_j \\ p dV + \left(V - m T \frac{\partial R}{\partial p} \right) dp - m \left(R + T \frac{\partial R}{\partial T} \right) dT &= \sum_j \left(R + (\nabla_{\mathbf{x}} R)^T (\hat{\mathbf{x}}_{d,j} - \mathbf{x}) \right) T dm_j \\ d\mathbf{x} &= \sum_j \frac{\hat{\mathbf{x}}_{d,j} - \mathbf{x}}{m} dm_j \end{aligned} \quad (2.20)$$

where

$$\begin{aligned} \hat{\mathbf{x}}_{d,j} &= \begin{cases} g(\mathbf{x}_j) & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\ \hat{\mathbf{x}}_{s,j} &= \begin{cases} \mathbf{x}_j & \text{When flow is from outside } (dm_j > 0) \\ \mathbf{x} & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\ \hat{T}_{s,j} &= \begin{cases} T_j & \text{When flow is from outside } (dm_j > 0) \\ T & \text{When flow is from inside } (dm_j \leq 0) \end{cases} \\ u &= u(p, T, \mathbf{x}) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) \quad \text{or} \quad \mathbf{x}^T \mathbf{u}(T) \end{aligned}$$

The DAE Formulation

$$\begin{pmatrix} 0 & 1 & 0 & \cdots & 1 & 0 \\ a_1 & p & b_1 & \cdots & 0 & 0 \\ c_1 & p & d_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_N & 0 & 0 & \cdots & p & b_N \\ c_N & 0 & 0 & \cdots & p & d_N \end{pmatrix} \begin{pmatrix} dp \\ dV_1 \\ dT_1 \\ \vdots \\ dV_N \\ dT_N \end{pmatrix} = \begin{pmatrix} dV \\ e_1 \\ f_1 \\ \vdots \\ e_N \\ f_N \end{pmatrix} \quad (4.2)$$

and

$$d\mathbf{x}_i = \sum_j \frac{\hat{\mathbf{x}}_{d,ij} - \mathbf{x}_i}{m_i} dm_{ij} \quad (4.3)$$

where

$$a_i = V_i - m_i T_i \frac{\partial R_i}{\partial p} = / \text{ by Maxwell's relations (c.f. Table 2.1)} / = m_i \left(\frac{\partial u_i}{\partial p} + \frac{T}{p} (c_{p,i} - c_{v,i}) \right)$$

$$b_i = -m_i \left(R_i + T_i \frac{\partial R_i}{\partial T} \right) = / \text{ by Maxwell's relations} / = -m_i (c_{p,i} - c_{v,i})$$

$$c_i = m_i \frac{\partial u_i}{\partial p}$$

$$d_i = m_i c_{v,i}$$

$$e_i = \sum_j \left(R_i + (\nabla_{\mathbf{x}} R_i)^T (\hat{\mathbf{x}}_{d,ij} - \mathbf{x}_i) \right) T_i dm_{ij}$$

$$f_i = dQ_i + \sum_j \left((\nabla_{\mathbf{x}} u_i)^T (\mathbf{x}_i - \hat{\mathbf{x}}_{d,ij}) + h(p, \hat{T}_{s,ij}, \hat{\mathbf{x}}_{s,ij}) - u_i \right) dm_{ij}$$

and

$$\hat{\mathbf{x}}_{d,ij} = \begin{cases} g(\mathbf{x}_{ij}) & \text{When flow is from outside } (dm_{ij} > 0) \\ \mathbf{x}_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases}$$

$$\hat{\mathbf{x}}_{s,ij} = \begin{cases} \mathbf{x}_{ij} & \text{When flow is from outside } (dm_{ij} > 0) \\ \mathbf{x}_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases}$$

$$\hat{T}_{s,ij} = \begin{cases} T_{ij} & \text{When flow is from outside } (dm_{ij} > 0) \\ T_i & \text{When flow is from inside } (dm_{ij} \leq 0) \end{cases}$$

$$u = u(p, T, \mathbf{x}) = \mathbf{x}(p, T, \mathbf{x}_r)^T \mathbf{u}(T) \quad \text{or} \quad \mathbf{x}^T \mathbf{u}(T)$$

Shorthand DAE Notation

Using the notation

$$\mathbb{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 & 0 \\ a_1 & p & b_1 & & & \\ c_1 & p & d_1 & & & \\ \vdots & & & \ddots & & \\ a_N & & & & p & b_N \\ c_N & & & & p & d_N \end{pmatrix}, \quad \dot{x} = \begin{pmatrix} dp \\ dV_1 \\ dT_1 \\ \vdots \\ dV_N \\ dT_N \end{pmatrix}, \quad \mathbb{B} = \begin{pmatrix} dV \\ e_1 \\ f_1 \\ \vdots \\ e_N \\ f_N \end{pmatrix},$$

$$\dot{y} = \begin{pmatrix} dx_1 \\ \vdots \\ dx_N \end{pmatrix}, \quad \text{and} \quad \mathbb{B}_y = \begin{pmatrix} \sum_j \frac{\hat{x}_{d,1j} - x_1}{m_1} dm_{1j} \\ \vdots \\ \sum_j \frac{\hat{x}_{d,Nj} - x_N}{m_N} dm_{Nj} \end{pmatrix}$$

we can write the model, i.e. (4.2) and (4.3), as

$$\begin{pmatrix} \mathbb{A}(x, y) & \mathbf{0} \\ \mathbf{0} & I \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} \mathbb{B}(x, y, z) \\ \mathbb{B}_y(x, y, z) \end{pmatrix} \quad (4.4)$$

where $\mathbb{A}(x, y)$, $\mathbb{B}(x, y, z)$, and $\mathbb{B}_y(x, y, z)$ are functions of the volume, pressure, and temperature part, x , of the state vector as well as the composition part, y . $\mathbb{B}(x, y, z)$ and $\mathbb{B}_y(x, y, z)$ may also depend on the independent variable z , i.e. time or crank angle.

Factorisation of the \mathbb{A} -Matrix

$$\mathbb{A} = \mathbb{D}_1 \mathbb{A}_D \mathbb{D}_2 = \mathbb{D}_1 \begin{pmatrix} 0 & \frac{a_1}{V_{\text{Tot}}} & 0 & \cdots & \frac{a_N}{V_{\text{Tot}}} & 0 \\ 1 & 1 & \frac{b_1}{d_1} & & & \\ \frac{c_1}{a_1} & 1 & 1 & & & \\ \vdots & & & \ddots & & \\ 1 & & & & 1 & \frac{b_N}{d_N} \\ \frac{c_N}{a_N} & & & & 1 & 1 \end{pmatrix} \mathbb{D}_2$$

where

$$\mathbb{D}_1 = \begin{pmatrix} \frac{V_{\text{Tot}}}{p} & & & & & \\ & a_1 & & & & \\ & & a_1 & & & \\ & & & \ddots & & \\ & & & & a_N & \\ & & & & & a_N \end{pmatrix} \quad \text{and} \quad \mathbb{D}_2 = \begin{pmatrix} 1 & & & & & \\ & \frac{p}{a_1} & & & & \\ & & \frac{d_1}{a_1} & & & \\ & & & \ddots & & \\ & & & & \frac{p}{a_N} & \\ & & & & & \frac{d_N}{a_N} \end{pmatrix}$$

B

PSPACK Functions and Variables

This Appendix collects a number of implementation details of PSPACK and is referred to mainly from Chapter 5. A seed to PSPACK was developed by multiple authors at Vehicular Systems as a project in a thermodynamics course. The package was called PS, for process simulation, and implemented the DAE formulation in [37] as well as heat transfer, combustion, calculation of thermochemical properties, geometry and emissions. Many of the functions interfaces and names are inspired by this package.

The sections of this Appendix are the same as in Chapter 5 and the references to the figures and interfaces are thus mainly from the corresponding section of Chapter 5.

B.1 Basic Idea and Overview

B.1.1 Structure – psSimStruct

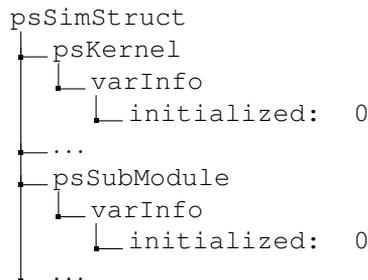


Figure B.1: Skeleton of *psSimStruct*.

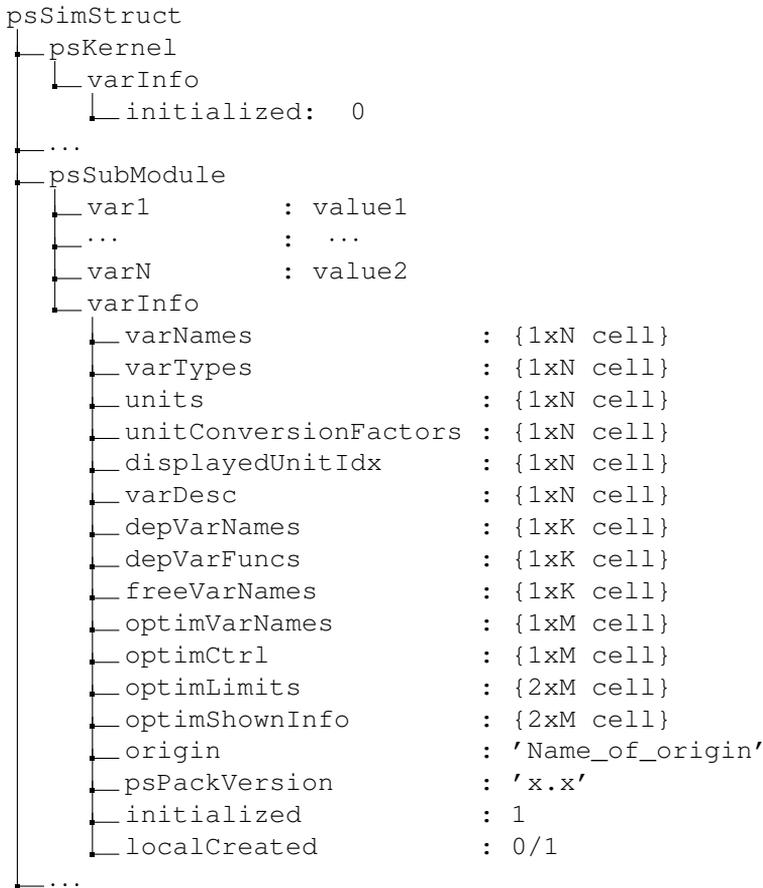


Figure B.2: Structure of `psSimStruct` when the example module `psSubModule` has been initialized. All variables in the structure, except `varInfo`, should have entries in the fields `varNames`, `varTypes`, `units`, `unitConversionFactors`, `displayedUnitIdx`, and `varDesc`. The variables may be of the type `value`, `string`, or `table`. The variables that belongs to the `value`-class may either be free to change or dependent on other variables and should therefore have entries in either `depVarNames` or `freeVarNames`. Of the variables that are free to choose some makes sense to change by an optimization routine and these should therefore be listed in `optimVarNames`. The user can then use `optimCtrl`, `optimLimits`, and `optimShonInfo` to turn on and of the optimization flag for the variables and to set limits for the optimization using either relative or absolute values.

B.2 Module – psKernel – PSPACK Kernel

B.2.1 Function – psDX

Table B.1: Overview of the function *psDX*.

Calling Syntax		
$[dXpVT \ dXgc] = psDX(XpVT, Xgc, Xgc_to, dV, dm, dQ, T_ext, Xgc_ext, Xgc_ext_to, dm_ext)$		
Name	Unit	Description
Inputs		
XpVT	[Pa, m ³ , K]	Pressure, volume and temperature state vector
Xgc	[-]	Composition state vector
Xgc_to	[-]	Composition state vector at destination of internal flows
dV	[m ³ /s]	Derivative of total cylinder volume
dm	[kg/s]	Matrix of mass exchange amongst the zones
dQ	[J/s]	Vector of heat transfer to the wall
T_ext	[K]	Temperature of external flow
Xgc_ext	[-]	Composition state vector for external flows
Xgc_ext_to	[-]	Composition state vector at destination of external flows
dm_ext	[-]	Vector of mass transfer to and from the system
Outputs		
dXpVT	[Pa/s; m ³ /s; K/s]	Change of thermodynamic state variables
dXgc	[-]	Change of composition state variables
Parameters		
psThermProp.K	[-]	Number of composition parameters in gas
psSimPar.minVfrac	[-]	Fraction of total volume for which to consider a zone to be empty
Dependencies (This function calls:)		
Functions: psThermProp		

B.2.2 Function – psDX_SingleZone_ConstV

Table B.2: Overview of the function *psDX_SingleZone_ConstV*. Note that the function may also be used for a series of single-zones, such as a pipe.

Calling Syntax		
[dXpT dXgc] = psDX_SingleZone_ConstV(XpT, V, Xgc, dm, dQ, TUp, XgcUp, dmUp, TDown, XgcDown, dmDown)		
Name	Unit	Description
Inputs		
XpT	[Pa, K]	Pressures, and temperatures of zones
V	[m ³]	Volumes of zones
Xgc	[-]	Compositions of the zones
dm	[kg/s]	Mass flow between zones
dQ	[J/s]	Heat transfer <i>from</i> the zones
External mass flows: There may only be flow to/from the first and last zone. (Note the sign/direction convention)		
TUp	[K]	Temperture outside the control volume
XgcUp	[-]	Gas composition upstreams
dmUp	[kg/s]	Mass flow TO the control volume
TDown	[K]	Temperture outside the control volume
XgcDown	[-]	Gas composition upstreams
dmDown	[kg/s]	Mass flow TO the control volume
Outputs		
dXpT	[Pa/s, K/s]	Change of thermodynamic state variables
dXgc	[-]	Change of composition state variables
Parameters		
psThermProp.K		
Dependencies (This function calls:)		
Functions: psThermProp		

B.2.3 Function – psAdiabaticMix

Table B.3: Overview of the function *psAdiabaticMix*.

Calling Syntax		
[p, T, Xgc] = psAdiabaticMix(p, V1, T1, V2, T2, Xgc1, Xgc2, stopFrac, Tinit)		
Name	Unit	Description
Inputs		
p	[Pa]	Pressure
T1	[K]	Temperature of zone 1
V1	[m ³]	Volume of zone 1
T2	[K]	Temperature of zone 2
V2	[m ³]	Volume of zone 2
Xgc1	[-]	Composition state vector of zone 1
Xgc2	[-]	Composition state vector of zone 2
stopFrac	[-]	Fraction of improvement to stop iteration
Tinit	[K]	Initial temperature guess
Outputs		
p	[Pa]	Pressure of resulting mix
T	[K]	Temperature of resulting mix
Xgc	[-]	Composition of resulting mix
Parameters		
- none -		
Dependencies (This function calls:)		
Functions: psThermProp		

B.3 Module – psGeometry – Engine Geometry

B.3.1 Global Parameter – psGeometry

Table B.4: Parameters that needs to be set up by the *psGeometry* module. Note that some of the parameters are dependent on each other.

Name	Unit	Description
psGeometry.B	[m]	Bore
psGeometry.a	[m]	Stroke
psGeometry.l	[m]	Connecting rod length
psGeometry.V_c	[m ³]	Clearance volume
psGeometry.n_cyl	[-]	Number of cylinders
psGeometry.r_c	[-]	Compression ratio
psGeometry.V_d	[m ³]	Displaced volume per cylinder
psGeometry.V_D	[m ³]	Total displaced volume of engine
psGeometry.S	[m]	Stroke ($2 * a$)
psGeometry.Ath_IV	[m ²]	Maximal flow area for intake valve
psGeometry.Ath_EV	[m ²]	Maximal flow area for exhaust valve
psGeometry.Cd_IV	[-]	Flow efficiency intake valve ([0-1])
psGeometry.Cd_EV	[-]	Flow efficiency exhaust valve ([0-1])

B.3.2 Function – psVolume

Table B.5: Overview of the function *psVolume*.

Calling Syntax		
V = psVolume (Ca)		
Name	Unit	Description
Input		
Ca	[rad]	Current crank angle
Output		
V	[m ³]	Volume of one cylinder
Parameters		
psGeometry.B	[m]	Bore
psGeometry.a	[m]	Stroke
psGeometry.l	[m]	Connecting rod length
psGeometry.V_c	[m ³]	Clearance volume
Dependencies (This function requires or calls:)		
- none -		

B.3.3 Function – psDVolume

Table B.6: Overview of the function psDVolume.

Calling Syntax		
[V] = psDVolume (Ca)		
Name	Unit	Description
Input		
Ca	[rad]	Current crank angle
Output		
V	[m ³]	Volume of the cylinder
Parameters		
psGeometry.B	[m]	Bore
psGeometry.a	[m]	Stroke
psGeometry.l	[m]	Connecting rod length
Dependencies (This function requires or calls:)		
- none -		

B.3.4 Function – psArea

Table B.7: Overview of the function psArea.

Calling Syntax		
[A] = psArea (Ca)		
Name	Unit	Description
Input		
Ca	[rad]	Current crank angle
Output		
A	[m ²]	Area of the combustion compartement
Parameters		
psGeometry.B	[m]	Bore
psGeometry.a	[m]	Stroke
psGeometry.l	[m]	Connecting rod length
Dependencies (This function requires or calls:)		
- none -		

B.4 Module – psHeat – Heat Transfer

B.4.1 Function – psDQ

Table B.8: Overview of the function *psDQ*.

Calling Syntax		
[DQ, h] = psDQ(pm, pf, T, V, A, mps, pivc, Tivc, Vivc, Ts)		
Name	Unit	Description
Input		
pm	[Pa]	Motored pressure approximation
pf	[Pa]	Actual cylinder pressure (fired pressure)
T	[K]	Temperature of zone
V	[m ³]	Volume zone
A	[m ²]	Contact area gas-wall
mps	[m/s]	Mean piston speed
pivc	[Pa]	Cylinder pressure at intake valve closure
Tivc	[K]	Temperature at intake valve closure
Vivc	[m ³]	Volume at intake valve closure
Ts	[K]	Wall temperature of cylinder
Output		
DQ	[W]	Heat transfer to the wall
h	[W/m ² K]	Heat transfer coefficient
Parameters (Same set of parameters for all available alternatives)		
psGeometry.B	[m]	Bore from geometry module
psGeometry.S	[m]	Stroke from geometry module
psHeat.C1	[-]	Heat transfer constant
psHeat.C2	[-]	Heat transfer constant
Dependencies (This function requires or calls:)		
SubModules: psGeometry		

B.4.2 Function – psDQ1p

Table B.9: Overview of the function *psDQ1p*.

Calling Syntax		
[DQ, h] = psDQ1p(p, T, A, mps, Ts)		
Name	Unit	Description
Input		
p	[Pa]	Cylinder pressure
T	[K]	Instantaneous temperature
A	[m ²]	Contact area gas-wall
mps	[m/s]	Mean piston
Ts	[K]	Wall temperature
Output		
DQ	[W]	Heat transfer to the wall
h	[W/m ² K]	Heat transfer coefficient
Parameters (Same set of parameters for all available alternatives)		
psGeometry.B	[m]	Bore from geometry module
Dependencies (This function requires or calls:)		
SubModules: psGeometry		

B.5 Module – psVibe – Burned Mass Fraction

B.5.1 Function – psVibe

Table B.10: Overview of the function *psVibe*. Note that some of the parameters are dependent on each other. The relations are $Duration = \theta_E - \theta_S$ and $Phasing = (\theta_E + \theta_S) / 2$.

Calling Syntax		
[xB] = psVibe(Ca)		
Name	Unit	Description
Input		
Ca	[rad] (0 = combustion TC)	Crank angle
Output		
xB	[-]	Burned mass fraction
Parameters		
psVibe.thetaS	[rad]	Start of combustion
psVibe.thetaE	[rad]	End of combustion
psVibe.m	[-]	Shaping parameter
psVibe.a	[-]	Shaping parameter
psVibe.eta	[-]	Combustion efficiency
psVibe.Phasing	[rad]	Phasing of combustion
psVibe.Duration	[rad]	Length of combustion
Dependencies (This function requires or calls:)		
- none -		

B.5.2 Function – psVibeEvents

Table B.11: Overview of the function *psVibeEvents*. The function should take the input structure and add it’s events to the already present list of events.

Calling Syntax		
[eventStruct] = psVibeEvents(eventstruct)		
Name	Unit	Description
Input		
eventStruct	[-]	Structure with events
.eventCa	[rad]	Ordered list of event angles (0 – 4π)
.SOC	[rad]	Angle for start of combustion event
.EOC	[rad]	Angle for end of combustion event
Output		
eventStruct	[-]	Structure with events
Parameters		
psVibe.*	[-]	See Table B.10 for a complete list of Vibe parameters
Dependencies (This function requires or calls:)		
- none -		

B.5.3 Function – psDVibe

Table B.12: Overview of the function *psDVibe*.

Calling Syntax		
[DxB] = psDVibe(Ca, xB)		
Name	Unit	Description
Input		
Ca	[rad] (0 = combustion TC)	Crank angle
xB	[-]	Actual mass fraction burned (optional parameter)
Output		
DxB	[1/rad]	Derivative of burned mass fraction with respect to crank angle
Parameters		
psVibe.*	[-]	See Table B.10 for a complete list of Vibe parameters
Dependencies (This function requires calls:)		
- none -		

B.6 Module – psSimPar – Simulation Parameters

Table B.13: Overview of common simulation parameters. Note that many of the parameters are dependent on each other.

Name	Unit	Description
Parameters		
psSimPar.RPM	[rev _{min}]	Engine speed
psSimPar.RPS	[rev _s]	Engine speed
psSimPar.w	[rad _s]	Engine speed
psSimPar.nrRev	[-]	Number of revolutions to simulate
psSimPar.thetaStart	[rad]	Simulation start angle
psSimPar.thetaStop	[rad]	Simulation stop angle
psSimPar.tStart	[s]	Simulation start time
psSimPar.tStop	[s]	Simulation stop time
psSimPar.minLapSteps	[-]	Minimum numbers of steps for ODE solver to take per lap
psSimPar.maxStep	[s]	Maxim steplength
psSimPar.resampleFac	[-]	Number of samples per revolution in interpolated output
psSimPar.relTol	[-]	Global relative tolerance
psSimPar.initBurnedVolFrac	[-]	Initial burned mass fraction at initialization of new zone
psSimPar.minVFrac	[-]	Minimal volume fraction where a zone is considered as empty
psSimPar.deltaT_ad	[K]	Approximation of adiabatic flame temperature
psSimPar.giveWarnings	[-]	Flag to control output of warnings during simulation

B.7 Module – psThermProp – Thermochemical Properties

B.7.1 Global Parameter – psThermProp

Table B.14: Parameters that needs to be set up by the psThermProp module.

Name	Unit	Description
Parameters		
psThermProp.T_frozen	[K]	Temperature at which to consider the composition as frozen
psThermProp.p_frozen	[Pa]	Pressure to use when frozen composition is assumed
psThermProp.K	[-]	Number of composition variables for the current gas
psThermProp.gammaApprox	[-]	Approximation of specific heat ratio
psThermProp.gammaAppro_T	[-]	Temperature at which specific heat ratio estimate is made
psThermProp.gammaAppro_p	[-]	Pressure at which specific heat ratio estimate is made
psThermProp.AFs	[-]	Stoichiometric air/fuel ratio
psThermProp.qLHVest	[J/kg]	Estimate of lower heating value of fuel
psThermProp.FuelName	[-]	Name of the fuel

B.7.2 Function – psThermProp

Table B.15: Overview of the function *psThermProp*.

Calling Syntax		
[h cp cv R dRdp dRdT dudp dudX dRdX] = psThermProp(p, T, Xgc, outputs)		
Name	Unit	Description
Input		
p	[Pa]	Common pressure
T	[K]	Temperature of zones
Xgc	[-]	Gas composition of zones
outputs	[-]	1x9 vector with flags to request calculation of the corresponding output
Output		
h	[J/kg]	Enthalpies
cp	[J/kg K]	Specific heats (constant pressure)
cv	[J/kg K]	Specific heats (constant volume)
R	[J/kg K]	Gas constants
dRdp	[J/kg K Pa]	Partial derivatives of the gas constant w.r.t. p
dRdT	[J/kg K ²]	Partial derivatives of the gas constant w.r.t. T
dudp	[J/kg Pa]	Partial derivatives of the specific internal energy w.r.t. p
dudX	[J/kg]	Partial derivatives of the specific internal energy w.r.t. X
dRdX	[J/kg K]	Partial derivatives of the gas constant wrt X
Parameters		
See respective model		
Dependencies (This function requires calls:)		
- none -		

B.7.3 Function – psCalcMixXgc

Table B.16: Overview of the function *psCalcMixXgc*.

Calling Syntax		
[Xgc] = psCalcMixXgc (Xgc, p, T)		
Name	Unit	Description
Input		
Xgc	[-]	Gas composition of zones at source
p	[Pa]	Common pressure
T	[K]	Temperature of zones
Output		
Xgc	[-]	Gas composition of zones at destination
Parameters		
See respective model		
Dependencies (This function requires calls:)		
- none -		

B.7.4 Function – psCalcReactXgc

Table B.17: Overview of the function *psCalcReactXgc*.

Calling Syntax		
[Xgc] = psCalcReactXgc (Xgc)		
Name	Unit	Description
Input		
Xgc	[-]	Gas composition of zones at source
Output		
Xgc	[-]	Gas composition of zones at destination
Parameters		
See respective model		
Dependencies (This function requires calls:)		
- none -		

B.7.5 Function – psCalcPhi

Table B.18: Overview of the function *psCalcPhi*.

Calling Syntax		
[Phi] = psCalcPhi(Xgc)		
Name	Unit	Description
Input		
Xgc	[-]	Gas composition of zone
Output		
Phi	[-]	Fuel/air equivalence ratio of mixer or reactor part of Xgc, whichever is relevant
Parameters		
See respective model		
Dependencies (This function requires calls:)		
- none -		

B.7.6 Function – psCalcXgc

Table B.19: Overview of the function *psCalcXgc*. Note that the fractions *x_react*, *x_mix_b*, and *x_mix_u* should sum up to one. Note also that *x_react* is either zero or one, i.e. a zone cannot be a mixer and reactor at the same time.

Calling Syntax		
[Xgc] = psCalcXgc(phi_react, x_react, phi_mix_b, x_mix_b, phi_mix_u, x_mix_u)		
Name	Unit	Description
Input		
phi_react	[-]	Fuel/air equivalence ratio of reactor part
x_react	[-]	Fraction that belongs to reactor part
phi_mix_b	[-]	Fuel/air equivalence ratio of mixers burned part
x_mix_b	[-]	Fraction that belongs to mixers burned part
phi_mix_u	[-]	Fuel/air equivalence ratio of mixers unburned part
x_mix_u	[-]	Fraction that belongs to mixers unburned part
Output		
Xgc	[-]	Gas composition of zone
Parameters		
See respective model		
Dependencies (This function requires calls:)		
- none -		

B.7.7 Available Alternatives

The Tables Model

Table B.20: Parameters for the *Tables* gas model.

Name	Unit	Description
Parameters (Tables)		
psThermProp.chemPropB	[-]	Table with thermochemical data for burned gas
.T	[K]	Temperature span vector
.p	[Pa]	Pressure span vector
.xf	[-]	Fuel fraction span vector
.phi	[-]	Fuel/air equivalence ratio vector
.hb	[J/kg]	Enthalpy matrix
.cpb	[J/kg K]	Specific heats
.cvb	[J/kg K]	Specific heats
.Mb	[kg/mol]	Average molecular weights
.dRbdp	[J/kg K Pa]	Partial derivatives or the gas constant w.r.t. p
.dRbdT	[J/kg K ²]	Partial derivatives or the gas constant w.r.t. T
.dubdp	[J/kg Pa]	Partial derivatives or the internal energy w.r.t. p
.dubdx f	[J/kg]	Partial derivatives or the internal energy w.r.t. the fraction of fuel
.dubdxa	[J/kg]	Partial derivatives or the internal energy w.r.t. the fraction of air
.dRbdxf	[J/kg]	Partial derivatives or the gas constant w.r.t. the fraction of fuel
.dRbdxa	[J/kg]	Partial derivatives or the gas constant w.r.t. the fraction of air
psThermProp.chemPropU	[-]	Table with thermochemical data for Unburned gas
.T	[K]	Temperature span vector
.ha	[J/kg]	Enthalpies of air
.cpa	[J/kg K]	Specific heat for air
.hf	[J/kg]	Enthalpies
.cpf	[J/kg K]	Specific heat of fuel
.Mf	[kg/mol]	Average molecular weight of fuel
.Ma	[kg/mol]	Average molecular weight of air

The SimpleEq Model

Table B.21: Parameters for the SimpleEq gas model.

Name	Unit	Description
Parameters (SimpleEq)		
psThermProp.psi	[-]	Molar N/O ratio (3.773 for air)
psThermProp.y	[-]	Molar H/C ratio of fuel
psThermProp.epsilon	[-]	Molar Fuel/Air equivalence ratio $\left(\frac{4}{y+4}\right)$

B.8 Module – psValve – Valve Areas

B.8.1 Function – psValve

Table B.22: Overview of the function *psValve*.

Calling Syntax		
[u] = psValve(Ca)		
Name	Unit	Description
Input		
Ca	[rad] (0 = combustion TC)	Crank angle
Output		
u	[-] (2x1-vector)	Valve flow area fractions [inlet, outlet]
Parameters (LookupTable and CVCPLookupTable)		
psValve.iMOP	[deg]	Inlet valve maximum open position
psValve.eMOP	[deg]	Outlet valve maximum open position
psValve.iv_table	[-]	Table with intake valve profiles
.ivo_rel	[deg]	Relative position of intake valve opening
.ivc_rel	[deg]	Relative position of intake valve closure
.x	[deg]	Position relative to iMOP
.y	[-]	Fraction of maximal opening area corresponding to x
psValve.ev_table	[-]	Table with exhaust valve profiles
.evo_rel	[deg]	Relative position of exhaust valve opening
.evc_rel	[deg]	Relative position of exhaust valve closure
.x	[deg]	Position relative to eMOP
.y	[-]	Fraction of maximal opening area corresponding to x
Parameters (Generic and fixParValve)		
psValve.DIV	[rad]	Duration for intake valve
psValve.DEV	[rad]	Duration for exhaust valve
psValve.IVO	[rad]	Intake valve opening
psValve.IVC	[rad]	Intake valve closure
psValve.EVO	[rad]	Exhaust valve opening
psValve.EVC	[rad]	Exhaust valve closure
Dependencies (This function calls:)		
- none -		

B.8.2 Function – psValveEvents

Table B.23: Overview of the function *psValveEvents*. The function should take the input structure and add it's events to the already present list of events.

Calling Syntax		
[eventStruct] = psValveEvents(eventStruct)		
Name	Unit	Description
Input		
eventStruct	[-]	Structure with events
.eventCa	[rad]	Ordered list of event angles ($0 - 4\pi$)
.IVO	[rad]	Angle for intake valve opening
.EVO	[rad]	Angle for exhaust valve opening
.IVC	[rad]	Angle for intake valve closure
.EVC	[rad]	Angle for exhaust valve closure
Output		
eventStruct	[-]	Structure with events
Parameters		
psValve.*	[-]	See Table B.22 for a complete list of valve parameters
Dependencies (This function requires or calls:)		
- none -		

B.9 Module – psModels – Cylinder/Engine Models

B.9.1 Function – psConvertSimOutput

Table B.24: Overview of the function *psConvertSimOutput*.

Calling Syntax		
[psSimResult] = psConvertSimOutput (psSimOutput)		
Name	Unit	Description
Input		
psSimOutput	[-]	A Structure with time generated by SIMULINK or some other simulation tool
Output		
psSimResult.rawData	[-]	Simulation data in raw form
psSimResult.rawData.t	[s]	The simulation time from simulink
psSimResult.rawData.xx	[-]	some simulation signal
...	[...]	...
psSimResult.resampData	[-]	Structure with re-sampled data
Parameter		
- none -		
Dependencies (This function requires or calls:)		
- none -		

B.9.2 Available Alternatives

psClosedCylSIModel

Table B.25: Parameters for the model *psClosedCylModel*. Note that X_{c_0} is dependent on the other composition parameters.

Name	Unit	Description
Parameters		
psModel.T_wall	[K]	Constant wall temperature
psModel.p0	[Pa]	Initial pressure
psModel.T0	[K]	Initial temperature
psModel.phiMix_b	[K]	Fuel/Air equivalence ratio of residual gas in unburned zone
psModel.xMix_b	[K]	Fraction residual gas in unburned zone
psModel.phiMix_u	[K]	Fuel/Air equivalence ratio of fresh mixture in the unburned zone
psModel.xMix_u	[K]	Fraction of fresh mixture in the unburned zone
psModel.Xc_0	[K]	Composition state of unburned zone at initialization

ps4CylSIModel

Table B.26: Parameters for the model *psClosedCylModel*. Note that X_{c_0} is dependent on the other composition parameters.

Name	Unit	Description
Parameters		
psModel.T_Wall	[K]	Wall temperature (Constant)
psModel.NumCyls	[-]	Number of cylinders
psModel.N_pipe	[-]	Number of sections in pipes
psModel.length_common_IM	[m]	Length of intake manifold
psModel.area_common_IM	[m ²]	Area of intake manifold
psModel.length_pipe_IM	[m]	Length of intake runners
psModel.area_pipe_IM	[m ²]	Area of intake runners
psModel.friction_IM	[-]	Friction in intake runners
psModel.volume_IM	[m ³]	Total volume of intake manifold
psModel.length_common_EM	[m]	Length of exhaust manifold
psModel.area_common_EM	[m ²]	Area of exhaust manifold
psModel.length_pipe_EM	[m]	Length of exhaust “runners”
psModel.area_pipe_EM	[m ²]	Area of exhaust “runners”
psModel.friction_EM	[-]	Friction in exhaust manifold pipe
psModel.volume_EM	[m ³]	Total volume of exhaust manifold
psModel.ThrottleArea	[m ²]	Effective flow area of throttle
psModel.TurbineEffArea	[m ²]	Effective flow are of turbine
psModel.TurbineWgArea	[m ²]	Effective flow are of waste-gate
psModel.pAmb	[Pa]	Ambient pressure
psModel.TAmb	[K]	Ambient temperature
As well as:		
Ambient gas composition		
Exhaust pipe gas definition (Downstream turbocharger)		
Intake manifold initial condition (composition, pressure and temperature)		
Exhaust manifold initial condition (composition, pressure and temperature)		

C

Example SI Engine Setup

In Chapter 3 and Chapter 4 different aspects of the different gas models as well as the DAE formulation are visualized using results from simulations of an example implementation of an SI engine. In this appendix the model that is used there is described in more detail.

The model setup is the `psClosedCylSIModel` from Chapter 5 which is a two-zone implementation that can simulate the closed part of an engine cycle. The model is developed for a SAAB/GM Ecotec-L850 engine and the parameters that are used, unless specified otherwise, are listed in Table C.1. The different parameters are documented in Appendix B. The modeled engine is spark ignited with a cylinder volume of 0.5 l and compression ratio 9.5, and the reference operating condition is 2000 RPM, with an initial cylinder pressure of 106 kPa, initial temperature of 347 K, and 11% residual gases at stoichiometric conditions. The fuel that is used is iso-octane and the default gas model is the `cheppfull` model.

The default solver is the MATLAB/SIMULINK `ode45`.

In Figures C.1– C.2 pressure, temperature, volumes, burned fraction, heat transfer, and work for the model setup are shown and in Figure C.3 the data grid for the `tables` model is shown together with pressure/temperature traces of the burned and unburned zone for the example setup.

Table C.1: Parameters for the example setup of *psClosedCylSIEngine*. The parameters use in the model are documented in Chapter 5 and Appendix B.

Name	Value	Unit
psHeat		
C1	1	[-]
C2	0.4386	[-]
psModel		
T_Wall	470	[K]
p0	1.0628	[bar]
T0	347.4269	[K]
phiMix_b	1	[-]
xMix_b	0.11	[-]
phiMix_u	1	[-]
xMix_u	0.89	[-]
psGeometry		
a	0.0430	[m]
l	0.1455	[m]
B	0.0860	[m ²]
V_c	$5.877 \cdot 10^{-5}$	[m ³]
n_cyl	4	[-]
r_c	9.5	[-]
V_d	$5 \cdot 10^{-4}$	[m ³]
V_D	0.0020	[m ³]
S	0.0860	[m]
psThermProp		
T_frozen	1700	[K]
p_frozen	1	[bar]
gammaApprox	1.3468	[-]
AFs	15.0673	[-]
qLHVest	44.61	[MJ/kg]
psVibe		
thetaS	6.0113	[rad]
thetaE	6.7556	[rad]
m	2	[-]
a	6.9	[-]
eta	1.0	[-]
psSimPar		
RPM	2000	[rpm]
thetaStart	4.1312	[rad]
thetaStop	8.8466	[rad]
relTol	10^{-10}	[-]
initBurnedVolFrac	0	[-]
minVfrac	10^{-12}	[-]
deltaT_ad	1900	[K]

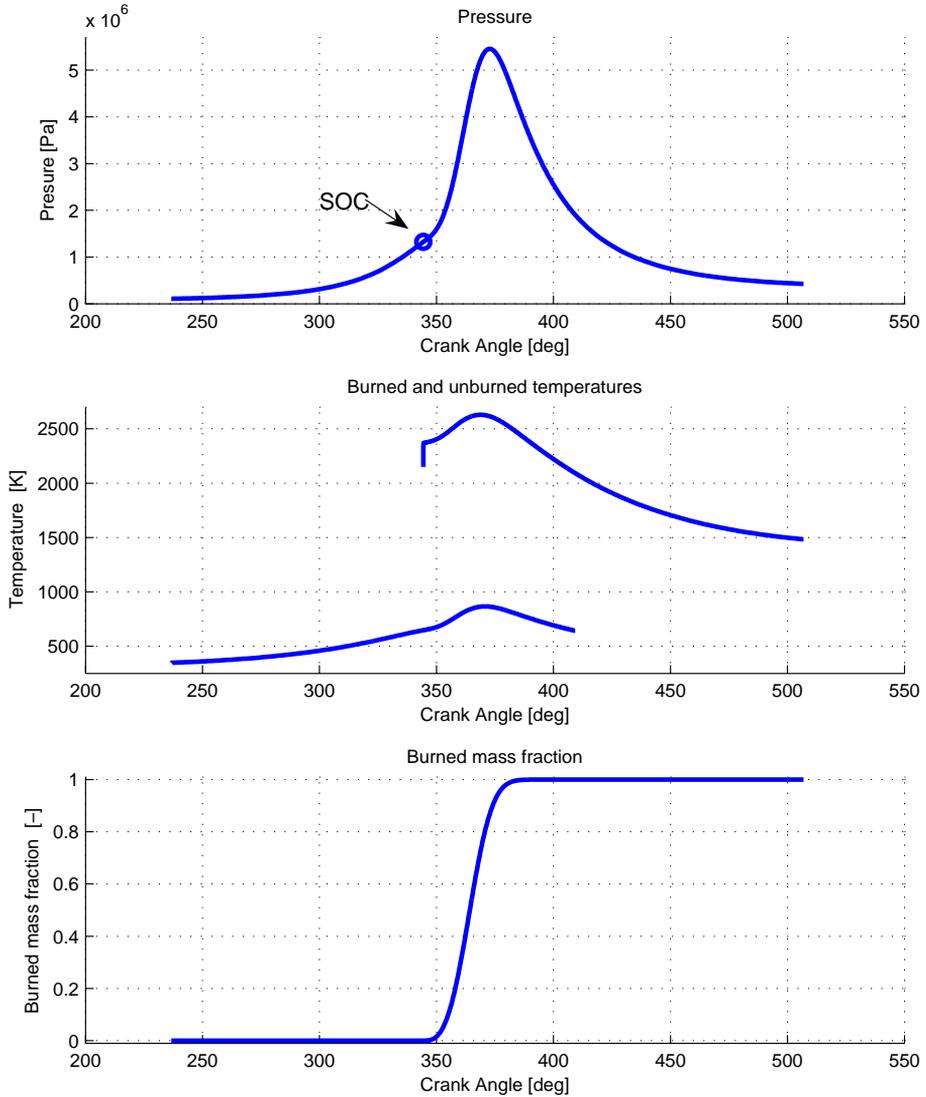


Figure C.1: Pressure and temperature traces for the default conditions of example setup. The initial steep gradient of the burned temperature is due to the fact that the initial temperature is estimated at start of simulation.

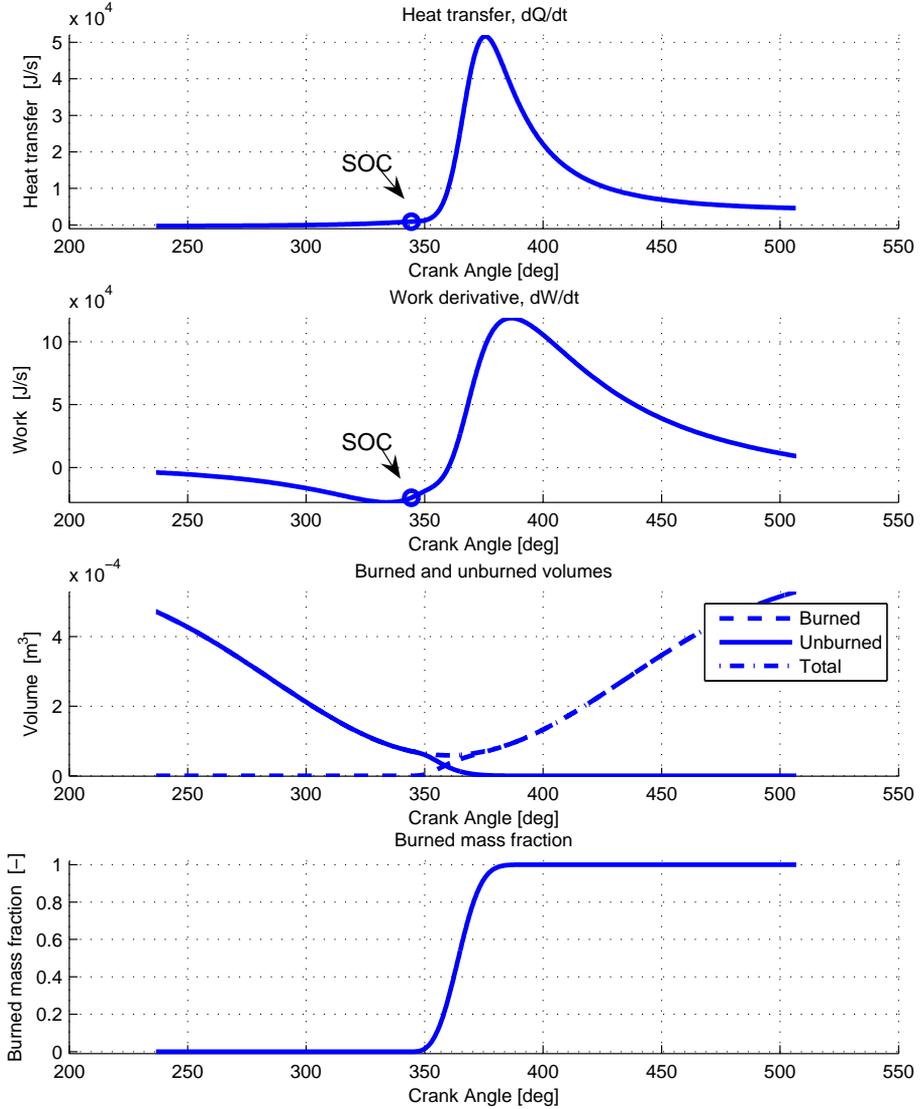


Figure C.2: Heat transfer, work derivative, volumes and burned mass fraction for the default conditions of example setup.

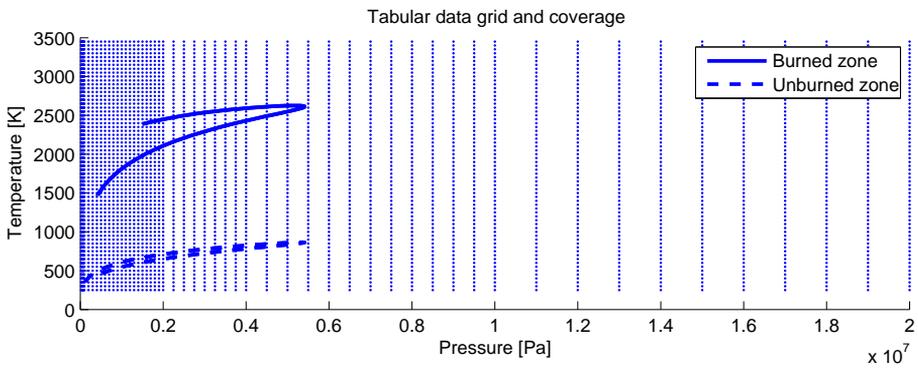


Figure C.3: The data for the model denoted *tables* is sampled with 50 K between the sample points in temperature and 10 kPa, 100 kPa, 250 kPa, 500 kPa, and 1000 kPa for the pressure.

Linköpings Studies in Science and Technology
Department of Electrical Engineering
Dissertations, Vehicular Systems

- No. 10 *Efficient Simulation and Optimal Control for Vehicle Propulsion*
Anders Fröberg, Dissertation No. 1180, 2008
- No. 9 *Single-Zone Cylinder Pressure Modeling and Estimation for Heat Release Analysis of SI Engines*
Markus Klein, Dissertation No. 1124, 2007
- No. 8 *Modelling for Fuel Optimal Control of a Variable Compression Engine*
Ylva Nilsson, Dissertation No. 1119, 2007
- No. 7 *Fault Isolation in Distributed Embedded Systems*
Jonas Biteus, Dissertation No. 1074, 2007
- No. 6 *Design and Analysis of Diagnosis Systems Using Structural Methods*
Mattias Krysander, Dissertation No. 1033, 2006
- No. 5 *Air charge Estimation in Turbocharged Spark Ignition Engines*
Per Andersson, Dissertation No. 989, 2005
- No. 4 *Residual Generation for Fault Diagnosis*
Erik Frisk, Dissertation No. 716, 2001
- No. 3 *Model Based Fault Diagnosis: Methods, Theory, and Automotive Engine Applications*, Mattias Nyberg, Dissertation No. 591, 1999
- No. 2 *Spark Advance Modeling and Control*
Lars Eriksson, Dissertation No 580, 1999
- No. 1 *Driveline Modeling and Control*
Magnus Pettersson, Dissertation No. 484, 1997



Linköping University
INSTITUTE OF TECHNOLOGY

**Department of Electrical Engineering
Linköping University
SE-581 83 Linköping**

