LUND UNIVERSITY

Development of an Automatic Reduction Tool for Chemical Mechanisms and an Optimized Sparse Matrix Solver for Systems of Differential and Algebraic Equations

Arvidsson, Andreas

2010

Link to publication

Citation for published version (APA):
Arvidsson, A. (2010). *Development of an Automatic Reduction Tool for Chemical Mechanisms and an Optimized Sparse Matrix Solver for Systems of Differential and Algebraic Equations*. [Doctoral Thesis (monograph), Combustion Physics].

Total number of authors:
1

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

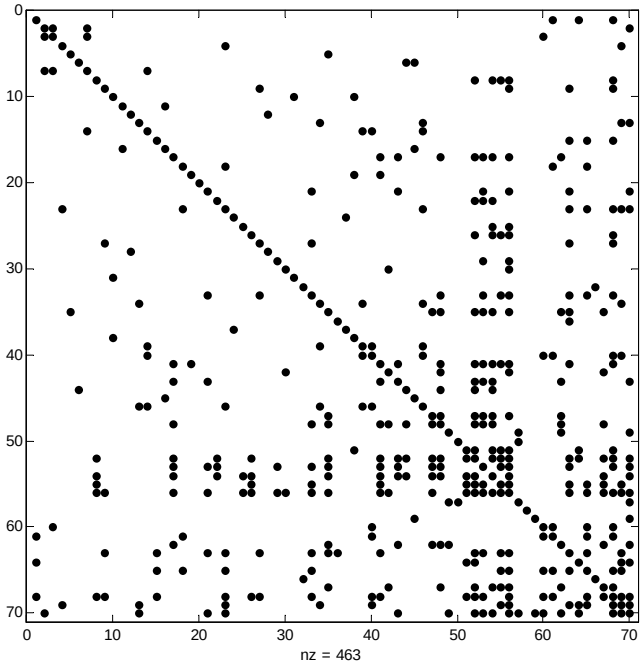
Download date: 23. Apr. 2024

Development of an automatic reduction tool for chemical mechanisms and an optimized sparse matrix solver for systems of differential and algebraic equations

Andreas Arvidsson

Division of Combustion Physics
Lund University

Doctoral Thesis



Development of an automatic reduction tool for chemical mechanisms and an optimized sparse matrix solver for systems of differential and algebraic equations

Andreas Arvidsson

Doctoral Thesis

Division of Combustion Physics
Lund University

Lund 2010

Printed at Mediatryck AB, Lund, Sweden
December 2010

Lund Reports on Physics, LRCP-141
ISBN 978-91-7473-074-6
ISSN 1102-8718

Andreas Arvidsson
Division of Combustion Physics
Lund University
P.O. Box 118
S-221 00 Lund
Sweden

Abstract

An N-Heptane mechanism and a Methane/Propane mechanism have been reduced by an Automatic Reduction Tool (ART) and simulated with two different solver combinations, which solve the set of ordinary *differential* equations governing the time evolution of the species simultaneously with solving *algebraic* equations for species that can be considered to be in quasi steady state. The most successful of the two solver combinations is an optimized combination of Newton solvers. The algebraic part of the solver is based on a Newton solver and is given a speed-up by using the fact that the sparseness pattern of the Jacobian is constant in time. This allows for automatically written source code and an optimization of the sparseness pattern in a preprocessing step. The optimization method is based on a simulated annealing procedure that minimizes the number of operations in the algebraic part of the solver. The speed-up of the Newton solver for the algebraic equations is one of the major developments presented in this thesis. The other one is the development of the ART and the reduction of the N-Heptane and the Methane/Propane mechanisms using the ART.

A reduction down to 37 out of 110 species and 23 out of 118 species is achieved for the N-Heptane and Methane/Propane mechanism respectively, while the accuracy of the solution is maintained and the CPU time is significantly lower than that of the detailed mechanism. Less, but still greatly reduced mechanisms are generated for larger ranges of physical conditions.

Also, the two solver combinations were implemented into a commercial Computational Fluid Dynamics (CFD) code. CFD simulations were then performed for a detailed and reduced mechanism. The implementation involving the optimized combination of Newton solvers resulted in a speed-up for the reduced mechanism compared to the detailed mechanism, while the accuracy of important species for the reduced mechanism was well within acceptable limits.

Contents

1. Introduction..... 21

1.1. Main objectives 21
1.2. Background..... 21
1.2.1. The positive aspects of combustion
1.2.2. The negative aspects of combustion
1.2.3. Reduction of the negative aspects of combustion
1.2.3.1. Computer simulations of combustion processes
1.2.3.2. Validity of a model
1.3. "Accuracy vs speed" problem..... 28
1.3.1. "Accuracy vs speed" problem solution
1.3.2. Is the reduction procedure worthwhile?
1.4. Previous work..... 34
1.5. Limitations..... 35
1.6. Chapter References..... 35

2. Modeling of the physics and chemistry..... 36

2.1. Chapter Introduction..... 36
2.2. The physical system..... 36
2.2.1. The dynamics of the physical system
2.3. Modeling of the physical system..... 41
2.3.1. Choosing an appropriate model
2.3.2. Systems of Differential and Algebraic Equations (DAE)
2.3.2.1. Input/output model of a system of DAE
2.3.3. 3-D models of chemically reactive flow
2.3.4. 0-D models for combustion processes
2.3.4.1. System of ODE for the Constant Volume Reactor (CVR)
2.3.4.1.1. System of ODE for the Adiabatic Constant Volume Reactor (ACVR)
2.3.4.2. System of ODE for the Constant Pressure Reactor (CPR)
2.3.4.2.1. System of ODE for the Adiabatic Constant Pressure Reactor (ACPR)
2.3.4.3. Coupling between 0-D and 3-D models
2.3.4.3.1. Operator splitting method
2.3.5. Chemical Kinetics
2.3.5.1. Reaction rates
2.3.5.2. The Arrhenius equation
2.3.5.3. A system of chemical reactions
2.3.5.4. Quasi Steady State Approximation (QSSA)
2.3.5.4.1. QSSA applied to a system of ODE
2.3.5.4.2. Errors due to the QSSA
2.3.5.5. The Ignition Process
2.3.5.5.1. Chain reactions

2.3.5.5.2. Important molecules for the mechanisms in this thesis
2.4. Chapter references…… 81
2.5. Chapter Appendix……..…….... 82

3. Numerical methods……....87

3.1. Chapter introduction…… 87
3.2. Integration of Ordinary Differential equations (ODE)……..... 88
3.2.1. Explicit and implicit methods
3.2.2. Stiffness
3.2.3. Integration of stiff systems of ODE
3.2.3.1. Predictor Corrector (PC) methods
3.3. The numerical method for the system of ODE…….... 94
3.3.1. The basic Newton method
3.3.1.1. Convergence of the Newton method
3.3.2. The Corrector
3.3.2.1. Damping of the Newton method
3.3.3. The Predictor
3.3.3.1. Gear's Backward Differentiation Formulas (BDFs).
3.3.3.2. Adaptive time step size
3.3.4. The Newton method in detail
3.3.4.1. The Jacobian
3.3.4.1.1. The condition number of the Jacobian
3.3.4.2. Ordinary Gaussian Elimination (GE)
3.3.4.3. Back Substitution (BS) of a triangular system
3.3.4.4. CPU cost of the Newton method
3.4. Two numerical methods for systems of NAE…… 117
3.4.1. The Newton method
3.4.1.1. The Jacobian for the system of NAE
3.4.2. The FP iteration method
3.4.2.1. The FP method applied to a system of NAE
3.5. Two combinations of numerical methods for stiff systems of DAE…… 125
3.5.1. Newton-FP method
3.5.2. Newton-Newton method
3.5.2.1. Newton-Newton method with operator splitting
3.5.3. CPU cost for the Newton-Newton method and the Newton-FP method
3.6. Chapter References…… 137
3.7. Chapter Appendix……... 138

4. Solver Combinations…… 142

4.1. **An overview of the solver combinations…… 142**
4.1.1. Input
4.1.2. Output
4.1.3. Solver structure
4.1.4. Simplified flow chart for the solver combinations for the system of DAE

4.2. **The Newton Solver for the system of ODE…….. 142**
4.2.1. Convergence
4.2.2. Flow chart for the Newton solver for the system of ODE
4.3. **The solver for the system of NAE…….. 150**
4.3.1. The Newton solver for the system of NAE
4.3.1.1. Convergence
4.3.1.2. Flow chart for the Newton solver for the system of NAE
4.3.2. Arguments for using a modified Newton solver for the system of NAE
4.3.3. Modifications of the Newton solver
4.3.3.1. Fixed sparseness pattern
4.3.3.1.1. Pre-processing
4.3.3.2. Using the sparseness pattern
4.3.3.2.1. Linked lists
4.3.3.2.2. Sparse GE
4.3.3.2.3. Sparse BS
4.3.3.3. Sparseness optimization
4.3.3.3.1. Simulated annealing
4.3.3.3.2. Metropolis Based Simulated Annealing (MBSA) algorithm
4.3.3.3.3. Simulated annealing applied to sparseness optimization
4.3.4. The FP solver for the system of NAE
4.3.4.1. Flow chart for the FP solver for the system of NAE
4.4. **The combined solvers for the system of DAE…….. 177**
4.4.1. Newton-Newton solver
4.4.1.1. Flow chart for the Newton-Newton solver
4.4.2. Newton-FP solver
4.4.2.1. Flow chart for the Newton-FP solver
4.5. **Chapter References……..... 178**

5. Mechanism Reduction……..179

5.1. Chapter Introduction…….. 179
5.2. The goal of the reduction…… 179
5.3. Mechanism generation ……... 180
5.3.1. Global and semi-global mechanisms
5.3.2. Detailed mechanisms
5.4. Mechanism reduction …… 182
5.4.1. The total reduction procedure
5.4.1.1. Chemical lumping and lumped mechanism
5.4.1.2. Species removal skeletal mechanism
5.4.2. The reduced mechanism
5.4.2.1. QSS species selection
5.4.2.1.1. Time scale separation methods
5.4.2.1.1.1. Life time (LT) analysis
5.4.2.1.2. Sensitivity analysis
5.4.2.1.3. Level Of Importance (LOI)
5.4.2.1.4. Concentration

5.4.2.1.5. LTC
5.4.2.2. The Automatic Reduction Tool (ART)
5.4.2.2.1. The ART algorithm
5.4.2.2.2. Mechanism validation by the ART
5.5. Chapter References……. 199
5.6. Chapter Appendix……... 200

6. Results and Discussion…… 202

6.1. Chapter Introduction…….. 202
6.2. N-Heptane Mechanism……... 204
6.2.1. Variation of inner solver
6.2.1.1. Time step size variation
6.2.1.1.1. Reduction level of the reduced mechanisms
6.2.1.1.2. CPU time of the reduced mechanisms
6.2.1.1.3. Solver information
6.2.1.1.4. Accuracy of the reduced mechanisms
6.2.1.1.5. Species profiles
6.2.1.2. Variation of solver settings
6.2.1.2.1. CPU time of the reduced mechanisms
6.2.1.2.2. Solver information
6.2.1.3. Minimization and Maximization of CPU time
6.2.1.3.1. Sparseness pattern of the NAE Jacobian
6.2.2. Variation of ART ET
6.2.2.1. Variation of IDT HF limit
6.2.2.1.1. Reduction level of the reduced mechanisms
6.2.2.1.2. CPU time of the reduced mechanisms
6.2.2.1.3. Accuracy of the reduced mechanisms
6.2.2.2. Variation of IDT CF limit
6.2.2.2.1. Reduction level of the reduced mechanisms
6.2.2.2.2. CPU time of the reduced mechanisms
6.2.2.2.3. Accuracy of the reduced mechanisms
6.2.2.3. Variation of Max HO2 CF limit
6.2.2.3.1. Reduction level of the reduced mechanisms
6.2.2.3.2. CPU time of the reduced mechanisms
6.2.2.3.3. Accuracy of the reduced mechanisms
6.2.2.4. Variation of number of ET for chemical species
6.2.2.5. Variation of QSS species ranking lists
6.2.2.6. Additional effects in groups of QSS species
6.2.3. Variation of physical ranges
6.2.3.1. Case 2: Temperature range
6.2.3.1.1. Reduction level of the reduced mechanisms
6.2.3.1.2. CPU time of the reduced mechanisms
6.2.3.1.3. Accuracy of the reduced mechanisms
6.2.3.2. Case 3: Temperature and Fuel/air ratio range
6.2.3.2.1. Reduction level of the reduced mechanisms
6.2.3.2.2. CPU time of the reduced mechanisms

6.2.3.2.3. Accuracy of the reduced mechanisms
6.2.3.3. Case 4: Temperature, Fuel/air ratio and pressure range
6.2.3.3.1. Reduction level of the reduced mechanisms
6.2.3.3.2. CPU time of the reduced mechanisms
6.2.3.3.3. Accuracy of the reduced mechanisms
6.2.4. CFD Application
6.2.4.1. CPU time of the simulation
6.2.4.2. Accuracy of the simulation
6.3. Methane/Propane Mechanism……... 363
6.3.1. Case 1: One physical point
6.3.1.1. Reduction level of the reduced mechanisms
6.3.1.2. CPU time of the reduced mechanisms
6.3.1.3. Accuracy of the reduced mechanisms
6.3.1.4. Solver information
6.3.1.4.1. Sparseness pattern of the NAE Jacobian
6.3.2. Case 2: Temperature range
6.3.2.1. Reduction level of the reduced mechanisms
6.3.2.2. CPU time of the reduced mechanisms
6.3.2.3. Accuracy of the reduced mechanisms
6.3.3. Case 3: Temperature and pressure range
6.3.3.1. Reduction level of the reduced mechanisms
6.3.3.2. CPU time of the reduced mechanisms
6.3.3.3. Accuracy of the reduced mechanisms
6.4. Chapter References……..... 388
6.5. Chapter Appendix……... 389

7. Conclusions and Outlook……411
7.1. Chapter References…… 413
Acknowledgements…….. 414

Nomenclature

Latin Alphabet

- A is the pre-exponential factor in the Arrhenius equation []
- A is a constant related to the decomposition of the Jacobian []
- B is a constant related to the back substitution []
- $\bar{c}_{v,i}$ is the heat capacity at constant volume per mole of species i [J/Kmole]
- $c_{v,i}$ is the heat capacity at constant volume per kg of species i [J/Kkg]
- c_v total heat capacity at constant volume per kg [J/Kkg]
- $c_{p,i}$ is the heat capacity at constant pressure per kg of species i [J/Kkg]
- $\bar{c}_{p,i}$ is the heat capacity at constant pressure per mole of species i [J/Kmole]
- C_{CFD} is a constant that relates the CPU_{CFD} to the number of QSS species []
- C_{FP} is a proportionality constant, which depends on the number of iterations needed in order to achieve convergence for the FP method. []
- CPU_{CFD} is the CPU time for the CFD part of a chemically reactive flow calculation [s]
- CPU_{CHEM} is the CPU time for the chemical kinetics part of a chemically reactive flow calculation [s]
- CPU_{TOTAL} is the total CPU time for a chemically reactive flow calculation [s]
- $CPU_{total}(N_{QSS})$ is a function of the number of QSS species and represents the total CPU time for given reduction level for the entire simulation [s]
- N_{TS} is the number of time steps for the entire simulation []

- $CTS_i(N_{QSS})$ is a function of the number of QSS species and represents the CPU time related to the solver combination for time step i [s]
- $CPU_{NORM}^{Average}(N_{QSS})$ is the normalized CPU time, which is a function of the number of QSS species, and represents the total CPU time for a given reduction level divided by the CPU time for the detailed mechanism []
- $CPU_{total}^{Average}(N_{QSS})$ is a function of the number of QSS species and represents the total CPU time for the solver combination, based on average values, for a given reduction level for the entire simulation [s]
- $CPU_{total}^{Average}(0)$ is the same as $CPU_{total}^{Average}(N_{QSS})$ but for the detailed mechanism, that is, zero QSS species. [s]
- $E(x)$ is the "Energy"-function for the simulated annealing process []
- E_a is the energy barrier for a chemical reaction [J/mole] (if R is used in the Arrhenius expression)
- **g** is the gravitational acceleration. [m/s^2]
- $\boldsymbol{g}$ is the function vector in the system of ODE that shall be minimized by the Newton method. [$mole/m^3s$]
- H is the total enthalpy of the system [J]
- H_i is the enthalpy per mole of species i [J/mole]
- h_i is the enthalpy per kg of species i [J/kg]
- h is the step size in the Newton method []
- IDT CF is the Ignition Delay Time for the Cool Flame, also known as 1:st ignition [s]

- IDT HF is the Ignition Delay Time for the Hot Flame, also known as 2:nd ignition [s]
- J_D is the Jacobian matrix in the Newton method for the system of ODE
- J_A is the Jacobian matrix in the Newton method for the system of NAE
- $\mathbf{j}_i$ is the diffusion flux of species i [mole/m^{2}s]
- $\mathbf{j_q}$ is the heat flux of species i [J/m^{2}s]
- k_{AB} is the rate constant for a reaction between molecules A and B [m^3/mole s]
- K_k is the rate constant for reaction k given by the Arrhenius [m^3/mole s] (for a bimolecular reaction) (the unit depends on the number of reacting species).
- k_B Boltzmann factor [J/K]
- m total mass [kg]
- m_i mass of species i [kg]
- n is a fit parameter in the Arrhenius expression
- M is a third body in a chemical reaction [mole/m^3]
- Max HO2 CF is the maximum value of the HO_2 concentration at the first peak corresponding to the first ignition. []
- Max OH HF is the maximum value of the OH concentration corresponding to the second ignition. []
- N_A is the number of species in the system of NAE []
- N_{AV} is Avogadro's number [1/mole]
- N_D is the number of species in the system of ODE []
- N_R is the number of reactions in the mechanism including both forward and backward since they are treated separately. []

- N_S is the number of species in the mechanism []
- N_{QSS} is the number of QSS species []
- N_{NQSS} is the number of non QSS species []
- N_{TOT} is the total number species, i.e. $N_{TOT} = \left(N_{NQSS} + N_{QSS}\right)$ []
- $N_{II}^{i,j}$ is the number of iteration steps of the inner solver for the particular time step *i* and particular iteration step *j* of the outer solver []
- N_{OI}^{tot} is the total number of iterations of the outer solver for an entire simulation []
- N_{II}^{tot} is the total number of iterations of the inner solver for an entire simulation []
- N_{OI}^{i} is the number of iteration steps of the outer solver for the particular time step *i* []
- *P* is the pressure [N/m^2]
- $P_{Dj}\left(N_{NQSS}\right)$ is a step function between 0 and 1, which depends on the number of non QSS species, and can be interpreted as the probability for building and decomposing a new Jacobian for the system of ODE at the particular iteration step *j* []
- $P_{Ak}\left(N_{QSS}\right)$ is a step function between 0 and 1, which depends on the number of QSS species, and can be interpreted as the probability for building and decomposing a new Jacobian for the system of NAE at the particular iteration step *k* []

- $\overline{P}_{Dj}\left(N_{NQSS}\right)$ is function of the number of non QSS species which assumes values between 0 and 1 and can be interpreted as the average probability for building and decomposing a new Jacobian for the system of ODE []
- $\overline{P}_{Ak}\left(N_{QSS}\right)$ is function of the number of QSS species which assumes values between 0 and 1and can be interpreted as the average probability for building and decomposing a new Jacobian for the system of NAE []
- $\overline{\overline{p}}$ is the pressure tensor [N/m^2]
- P is a steric factor related to orientation of the colliding molecules []
- P is a probability []
- Q heat [J]
- $\dot{Q}$ [J/s]
- q_r is the heat radiation source term. [J/s]
- R is the universal gas constant [J/mole K]
- **r** is the spatial location [m]
- r_{AB} is the reaction rate for a reaction between molecules A and B [mole/m^{3}s]
- r_k is the reaction rate for reaction k [mole/m^{3}s]
- SOP: Sum of OPerations in the Gaussian elimination and back substitution of the solver for the algebraic equations []
- ΔSOP is the difference in SOP corresponding to two consecutive sparseness patterns of the Jacobian for the system of NAE. []
- T is the temperature [K]
- t is time [s]

- Δt_i is the difference between the future time point and i-1 time point previous to the present time point. [s]
- U_i is the internal energy per mole of species i [J/mole]
- V is the volume of the system [m^3]
- v is the specific volume of the system [m^3/kg]
- **v** is the velocity [m/s]
- W work [J]
- $\dot{W}$ work per time [J/s]
- W_i molecular weight of species i [kg/mole]
- $\mathbf{x}_D$ is the vector of species concentrations in the system of ODE [mole/m^3]
- $\mathbf{x}_A$ is the vector of species concentrations in the system of NAE [mole/m^3]
- $\mathbf{x}_A^{NAE}$ the concentration vector of the QSS species [mole/m^3]
- $\mathbf{x}_A^{ODE}$ the concentration vector of the species of the ODE system that are later set in QSS [mole/m^3]
- $\mathbf{x}_D^c$ is the solution to the corrector equations including truncation errors [mole/m^3s]
- $\mathbf{x}_D^p$ is the solution to the predictor equations including truncation errors [mole/m^3s]
- $\dot{\mathbf{x}}(t)$ is the time derivative of **x(t)** [mole/m^3s]
- y is a function value []
- Y_i mass fraction of species i. []

Greek Symbols

- α is a constant []
- $\alpha(N_{QSS})$ is a function of the number of QSS species which assumes values between 0 and 1 and accounts for the computational speed-up related to the optimisation of inner solver []
- α_{const} is a constant which assumes values between 0 and 1 and accounts for the computational speed-up related to the optimisation of inner solver []
- $\beta_i(N_{QSS})$ is a function of the number of QSS species and represents the CPU time related to additional operations in the program code for time step *i*. [s]
- $\beta_{tot}(N_{QSS})$ is a function of the number of QSS species and accounts for the CPU time related to additional operations in the program code for the entire simulation [s]
- Δ (x) difference of the quantity x [] the unit depends on the unit of x
- $\boldsymbol{\varepsilon_D}$ is the error is $\boldsymbol{x}_D$ [mole/m^3]
- $\boldsymbol{\varepsilon_A}$ is the error is $\boldsymbol{x}_A$ [mole/m^3]
- $\boldsymbol{\varepsilon_{dc}}$ is the discretisation and truncation error between the corrector equations and the exact equations for the ODE system
- $\boldsymbol{\varepsilon_{dp}}$, which is the discretisation and truncation errors between the predictor equation and the exact equations for the ODE system.
- κ(A) is the condition number of the matrix A []
- λ is a damping factor in the Newton method for the ODE system []

- $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ are the maximum and minimum eigenvalues of the matrix A []
- μ is the reduced mass [kg]
- υ_{ij} is the net stoichiometric coefficient for species *i* in reaction *j* []
- υ'_{ij} is the stoichiometric coefficient for the species *i* in reaction *j* on the reactant side []
- υ''_{ij} is the stoichiometric coefficient for the species *i* in reaction *j* on the product side []
- ρ density [kg/m^3]
- σ_{AB}, is the cross section for the collision between molecule *A* and molecule *B* []
- τ_i life time of species i [s]
- ω_i source term for species *i* [mole/(m^{3}s)]
- $\boldsymbol{\omega}_D$ is the source term vector in the system of ODE [mole/(m^{3}s)]
- $\boldsymbol{\omega}_A$ is the source term vector in the system of NAE [mole/(m^{3}s)]

Abbreviations

- ACVR: Adiabatic Constant Volume Reactor
- ACPR: Adiabatic Constant Pressure Reactor
- ART: Automatic Reduction Tool
- ART ET: ART Evaluation Target
- BS: Back Substitution
- CFD: Computational Fluid Dynamics
- CF: Cool Flame
- CGR: Chemistry Guided Reduction
- CI: Column Index
- CN: Condition Number
- CPR: Constant Pressure Reactor
- CPU: Central Processing Unit
- CVR: Constant Volume Reactor
- DAE: Differential and Algebraic Equations
- DNS: Direct Numerical Simulation
- ET: Evaluation Targets
- FP: Fixed Point
- GE: Gaussian Elimination
- HF: Hot Flame
- IDT: Ignition Delay Time
- JAC: Jacobian
- LES: Large Eddy Simulation
- LOI: Level Of Importance
- LT: Life Time
- LTC: Life Time Concentration
- NAE: Nonlinear Algebraic Equations
- NNZ: Number of Non Zeros
- NO: Number of Operations
- NTC: Negative Temperature Coefficient
- NZE: Non Zero Element
- ODE: Ordinary Differential Equations
- PC: Predictor-Corrector
- PCA: Principal Component Analysis
- PDE: partial Differential Equations
- PDF: Probability Density Function
- QSS: Quasi Steady State
- QSSA: Quasi Steady State Approximation
- RANS: Reynolds Average Navier Stoke
- RK: Runge-Kutta
- SOP: Sum of Operations

The Contributions of the Author

The author of this thesis has contributed to the following;

- The development of the Automated Reduction Tool (ART), which further develops the software REDKIN for mechanism reduction, and optimizes the usage of the software REDKIN and the software IGNITION for reactor simulation. The ART, REDKIN and IGNITION are discussed in section 5.4.2.2.
- The demonstration of the ART by reducing the mechanisms for n-Heptane and Methane/Propane efficiently. The results are shown in section 6.
- The development and optimization of the Newton solver for the NAE system, which is a part of the Newton-Newton solver combination. The optimization of the Newton solver for the NAE system is discussed in section 4.3.3, while the Newton-Newton solver is discussed in section 3.5.2.

Previous work at the department of Combustion Physics is;

- The program REDKIN
- The program IGNITION
- The Newton solver for the ODE system in the Newton-Newton and Newton-Fixed Point (FP) solver, which is discussed in section section 3.5.2.

Chapter 1.

Introduction

1.1. Main objectives

The main objectives of this thesis are;

- The development and optimization of a fast and accurate solver combination for a system of Differential and Algebraic Equations (DAE) in such fashion that CPU time decreases as the number of Nonlinear Algebraic Equations (NAE) increases, while the accuracy of the solution is maintained.
- The development of an Automatic Reduction Tool (ART), which automatically reduces detailed chemical mechanisms by converting as many of the Ordinary Differential Equations (ODE) as possible into NAE.

1.2. Background

Combustion processes have been used by mankind on smaller scales through the ages, but ever since the industrial revolution in the late 18^{th} and early 19^{th} century combustion processes have become an important source of energy for industry and society as a whole. Combustion processes are therefore used on a much larger scale today than in earlier ages. The future energy demand will likely be even larger than it is today, since the world has an increasing demand for energy as the world population grows and societies develop [3]. Today, about 80% of the world energy production comes from combustion of fossil fuels. It is believed that combustion of fossil fuels will be the dominant energy source for at least two decades to come [1]. The energy from combustion of fossil fuels is primarily used in power plants and in IC engines and gas turbines for transportation of people and goods.
However, combustion of fossil fuels produces different kinds of emission products, like CO_x, SO_x and NO_x and soot particles, which affect the environment and health both

globally and locally (see section 1.2.2). The CO_2 emissions from combustion of fossil fuels are the most infamous, since the molecule strongly contributes to the greenhouse effect that causes global warming [2]. However, the extent of the contribution is debated. Much political work has been and is done to decrease the CO_2 emissions by the use of improved combustion and emission reduction techniques and different kinds of legislation and taxes. However, the greenhouse effect caused by combustion processes of fossil fuels will continue to increase in the future for the reasons stated above.
Still, the fossil fuels reserves will not last forever, which means that renewable energy sources, there among combustion of bio-fuels, must satisfy the future energy demands. Combustion of bio-fuels also produces hazardous emissions, but the carbon atoms from the combustion process of bio-fuels are recycled from the CO_2 in the atmosphere when new bio-fuels are grown, which eliminates the global warming contribution from this bio-fuel combustion process. However, this is only true under the condition that no fossil fuels are used during the growth, harvesting, manufacturing and distribution of the bio-fuels.
There are issues with bio-fuels as well, like the "food versus fuel" debate, which is based on the fact that large areas must be used for growing the bio-fuel that otherwise could be used for growing food instead, and problems with deforestation and soil erosion. This means that there are positive and negative aspects of combustion of fossil fuels and bio-fuels, which are further discussed in section 1.2.1 and section 1.2.2. Reduction strategies for the negative aspects of combustion are discussed in detail in section 1.2.3.

In principle, combustion can simply be described as an exothermic process where fuel and oxidizer react and form products. The energy released can be utilized as heat and / or mechanical work. Despite this overall simplicity, the processes involved are quite complex. A complete description of a combustion process involves different disciplines such as chemistry, fluid dynamics and thermodynamics. A true description of the chemical reactions involved even for the simplest hydrocarbons, e.g. methane, involves hundreds of reaction steps and hundreds of intermediate species. Fluid dynamics and thermodynamics are needed to describe the transport processes for momentum, heat and chemical species. Further on, almost every combustion process in an application of practical importance is turbulent, i.e. the flow field features random fluctuations in space and time. The small scale convective motion enhances mass and heat transfer. The fluid dynamic system is highly non-linear as well as the chemical reaction processes. The processes also feature a wide range of time and space scales.
To properly describe a combustion process, it is of greatest importance to account for the interaction between the fluid dynamics and the chemical kinetics, particularly this is the case for non-steady combustion of relevance for internal combustion engines. Fluid dynamics control the transport of fuel and oxidizer to the reaction zone, thus affecting the flame, which in turn affects the temperature and pressure fields which then changes the conditions for the fluid dynamic system.
Even though combustion processes have been used successfully over a long time period, the knowledge of the details of the processes involved, and particularly the interaction between these processes, are not fully known. Many issues of combustion remain to be solved.

1.2.1. The positive aspects of combustion

The combustion of fossil fuel and bio-fuel produces energy which can be used in [1];

- Power plants to produce heat and electricity. The power plants can be based on combustion of oil, coal, gas and bio-fuel.
- Transportation of people and goods by the means of cars and trucks (combustion engine), airplanes (combustion engine and jet engine), boats (combustion engine) and trains (indirectly through power plants).
- Industrial processes like metal-refining industry (furnaces), cement manufacturing industry (rotary kilns) and soot manufacturing for process engineering.

The positive aspects of combustion can be enhanced by more efficient combustion processes. Higher efficiency of combustion processes can be achieved by both fundamental and applied research, which can be performed by experiments or computer simulations.
Computer simulations are discussed in detail in this thesis, while experiments are not.

1.2.2. The negative aspects of combustion

In a simple description of the combustion process the fuel reacts with oxygen and produces many intermediate species, which react with each other and oxygen to produce the final products like H_2O and CO_2. However, the combustion process also produces unwanted emissions of various kinds, which leads to environmental pollution, like smog, acid rain, ozone depletion in the atmosphere, ground ozone formation and global warming [2]. The combustion process also produces specific health hazards like respiratory problems and cancer. The major pollutants produced by combustion of various kinds are [1];

- Unburned or partially unburned hydrocarbons, which causes specific health hazards like cancer
- CO, is produced especially in combustion at fuel rich conditions and is toxic
- CO_2 , is a green house gas.
- Nitrogen oxides (NO_x), is produced especially at high temperature combustion and contributes to the greenhouse effect and causes smog, which in turn causes specific health hazards. When NO_x and Volatile Organic Compounds (VOC) react in the presence of sunlight they form photochemical smog and ozone, which both cause damage to lung tissue and reduction in lung function. NO_x also causes acid rain.
- Sulfur oxides (SO_2 and SO_3), contributes to the greenhouse effect and causes acid rain and increased respiratory problems.
- Soot Particles, which causes reduction in lung function. The environmental effect of soot particles are uncertain and under debate [2].

1.2.3. Reduction of the negative aspects of combustion

Large investments are made by governments and industries to facilitate basic and applied research at universities and companies with the aim to increase the general knowledge of combustion and to decrease the emissions from combustion processes, make engines more efficient in order to decrease the fuel consumption and develop alternative combustion technologies and facilitate the usage of alternative, environmentally friendlier fuels. New fuels are tested and new engines and combustion processes are developed, but the problems are not easily solved since the combustion process is very complex and involves many parameters to vary. Due to the complexity of the combustion process, experiments cannot always be easily performed nor to a low cost for large parameter ranges. Also, all parameters are not accessible in experiments.
With the ever increasing computing power, the alternative of simulation of combustion processes, as a complement to experiments, grows stronger year by year. The reason for this is that computer simulations can capture the physics under conditions and in situations that are hard to analyze experimentally and are also less expensive than some experimental apparatus. Computer simulations can nowadays function as a scientific tool, especially if Direct Numerical Simulation (DNS) is used, and not just an engineering tool. Computer simulation may indicate phenomena that experiments so far have not, which in turn can lead to the creation of new experiments that would not have been considered before.
However, it should be noted that the coefficients and parameters involved in computer simulation often are determined from experiments, which means that the experiments and computer simulations are closely tied together and that the evolution of the computer simulations depends on the accuracy of experiments. The computer models must also be validated, if possible, against experiments. Research in the combustion field is done by both experiments and computer simulations but this thesis will focus only on computer simulations.

Computer simulations have complemented and sometimes reached a dominant position compared to experiments in some research fields in the past decade. One example of this is the field of aerodynamics, where Computational Fluid Dynamics (CFD) simulations became a great complement to and sometimes replaced wind tunnel experiments when optimizing cars, trucks and airplanes. Another example is the Finite Element Method (FEM), which became a great tool for engineers who studied crash test and developed stability for cars, trucks and airplanes.

A possible future use of computer simulation lies in the field of engine development. Today, the development of a new engine requires engine experiment, in which numerous parameters are tested. The engine experiments make up a large part of the total cost of the engine development, which is in order of one billion euro. Consequently, large amounts of money and time can be saved using computer simulations of engines, if possible, instead of or as a complement to experiments.
Detailed chemical information is becoming an important part of combustion simulation codes for engines and other combustion devices. These simulations aim at optimizing the aspects of geometry and time on the combustion process and the resulting emissions for

the particular combustion device that is investigated. The simulation codes for combustion processes involve the cooperation between CFD codes, which transports chemical species in space and time, and chemical kinetic solvers, which are able to handle a stiff chemical mechanism on a grid cell level. The cooperation between CFD codes and chemical kinetic codes is promoted by the increased awareness of the importance of detailed chemistry for the predictions of control parameters and emission levels, in particular for the development of low-emission technologies. The numerical study of such devices requires particular accuracy of chemical species concentrations throughout the computations. The chemical kinetic codes and the cooperation with CFD codes are discussed in detail in section 2.3.3 and 2.3.4.3.
However, the computer simulations must give an accurate solution and they must be performed in a relatively short time, which sometimes are hard conditions to fulfill simultaneously. The speed of the computer simulations can be shortened by the use of reduced instead of detailed chemical mechanisms. However, the accuracy of the reduced mechanism is generally less than the detailed. Hence, mechanism reduction, which is discussed in detail in section 5.4, generally results in a trade off problem between speed and accuracy, which is discussed through out the thesis.

1.2.3.1 Computer simulations of combustion processes

Computer simulations of combustion processes are based on CFD programs, which are coupled with chemical kinetics programs.
The CFD programs are based on the balance equations of mass, energy, momentum and chemical species plus the equation of state [5]. The balance equations involve the chemical source term, commonly calculated in cooperation with chemical kinetics programs. The balance equations involve derivatives in space and time. In order to solve the conservation equations numerically, time and space are discretized into grid points. At each grid point the chemical source terms are (usually) calculated from the chemical kinetics programs. One alternative is to use a Constant Volume Reactor (CVR) or Constant Pressure Reactor (CPR) at each grid point. A more detailed discussion of this topic can be found in section 2.3.4.3.

CFD simulations of combustion processes can be done in up to 3D and are more expensive in terms of CPU time and memory requirements especially when higher dimensions are involved, since the calculations often involves millions of grid points. Unfortunately, the computers today are not strong enough to simulate detailed 3D combustion scenarios in reasonable time. Even in 2D the simulations can be quite costly. Engine simulations using DNS and detailed chemistry can sometimes reach the order of years. The CPU time can be reduced significantly by the use of CFD models and less detailed chemistry [4]. However, less detailed models will compromise with the accuracy of the solution. Hence, a trade off between speed and accuracy exists.

The motivation for CPU time reduction is mainly operative usage in industry. Optimization often means that many parameters must be varied, which means that the

simulations must be repeated numerous times and faster computer programs therefore save both time and money. Two things that can be done in order to decrease the CPU time of the computer simulations of combustion processes are;

- Simplifications of the system
- Development of faster numerical methods and/or solvers

This thesis will only focus on simplification of the chemistry and development of faster numerical methods and solvers for calculation of the chemical source term. Hence, development of faster numerical methods and/or solvers for the CFD codes is outside the scope of this thesis**.**
Much work is done to decrease the calculation time of the chemical source term, since it uses a large part of the CPU time in combustion simulations. The work is focused on developing faster algorithms in the computer codes that calculates the chemical source term and also to decrease the calculation time by simplifying the chemical system, without losing accuracy of the solution.

A simplification of the system often leads to shorter calculation time but also to a less accurate solution, which means that there is always a trade off between speed and accuracy of the computer simulations when simplifications are made.
Detailed chemical mechanisms have the advantage of providing accurate solutions to combustion problems. However, calculations with detailed chemical mechanisms require a great deal of CPU time, which is a major disadvantage. The detailed chemical mechanism can be simplified by reduction of the number of species and reactions. Mechanism reduction can be achieved through a sequence of reduction methods. These methods, which are described in section 5.4, are chemical lumping, species removal and application of the Quasi Steady State Approximation (QSSA).
The accuracy of the reduced mechanism must be controlled during the reduction procedure. Hence, the solution based on the reduced mechanism must not deviate too much from the solution based on the detailed mechanism. Consequently, the reduced mechanism is only valid for a limited set of physical conditions.
In this thesis, the chemical system is simplified by the use of the QSSA on some species. The species that are set as QSS species are not transported in CFD programs, but the species are still contributing to the solution even though the contribution is less accurate than before. This will be explained more in detail in section 2.3.5.4. The selection of the QSS species is difficult and various methods for doing so exist. This difficulty motivates the ART, which is discussed in section 5.4.2.2.

1.2.3.2 Validity of a model

The model can capture physics outside the range of the experimental data. An example of this is illustrated in Figure 1.1, which shows Ignition Delay Time (IDT) vs 1000/T for experimental data [6] and the simulations of the CVR model. The CVR is discussed in detail in section 2.3.4.1. The experimental data exist for a shorter temperature range than the model simulation. Hence, the model is only validated against experiments for the overlapping temperature region.
The question arises if the model can be trusted for the parameter ranges that have not been validated against experimental data. Since the model is validated against experimental data, it is reasonable to assume that the validity of the model decreases proportional to the distance from the region that contains experimental data. However, this decrease is problem dependent and does not have to be linear. This is illustrated in Figure 1.2.

It should be noted that the experiments also have limitations. Experiments are usually hard to conduct for extreme physical conditions. Hence, experiments are more trustworthy in some regions than others. This in turn means that the validation of the model against experiments is more trustworthy in some regions than others. This is further discussed in section 2.3.

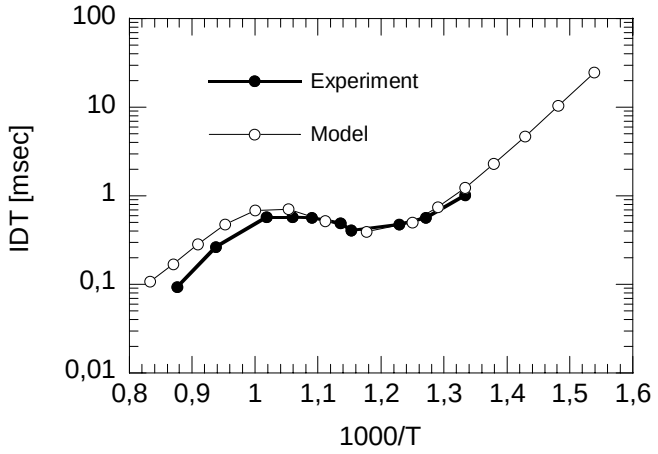


Figure 1.1. IDT vs 1000/T for experiments [6] and model. The model is only validated against experiment for the temperature region the experiments exist. Beyond that temperature region the model can predict the behavior of the experiments.

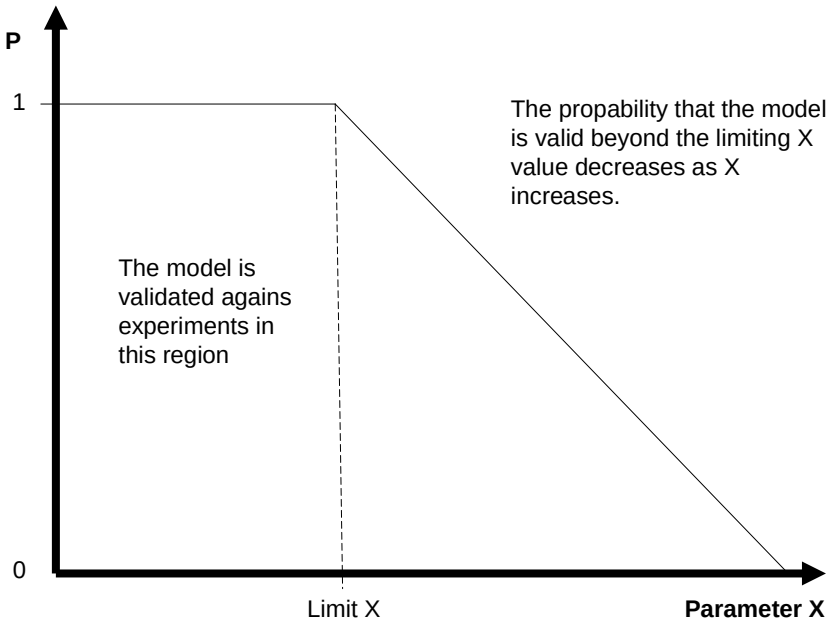


Figure 1.2. The model is validated against experiments for the parameter, X, up to the limit value of X. Beyond the limiting X value, the probability, P, that the model is valid decreases as X increases. The decreasing function is problem dependent and does not have to be linear.

1.3. "Accuracy vs speed" problem

In order to have fast computer simulations of combustion processes the CFD code needs to involve as few species as possible since the transportation of each species and the calculation of the chemical source term for each species use a lot of CPU time. At the same time there has to be enough species involved to describe the system accurately, otherwise the computer simulation gives an unrealistic solution to the problem. The computer simulation must be;

- Fast
- Accurate

Hence, there is an obvious trade off between the speed of the computer simulation and the accuracy of the solution. The way to attack this problem is to reduce the number of species that are transported without losing much accuracy of the solution.
The reduction of the number of transported species can be done by setting some of the species in a QSSA. This is also the reduction procedure in this thesis. The species that are set in QSSA will not be transported since their concentrations do not have to be calculated from a differential equation, but the species are still in the system and are

contributing to the solution of the system since they are calculated from an algebraic equation instead. This gives rise to a system of Differential and Algebraic equations (DAE) section 2.3.2. The numerical method and solver for a system of DAE is presented in section 3.5 and 4.4.

The CPU time for a combustion simulation, CPU_{TOTAL}, can be divided into two parts;

- CPU_{CFD}
- CPU_{CHEM}

Hence, CPU_{TOTAL} is the sum of CPU_{CFD} and CPU_{CHEM}. The CPU_{CFD} is the CPU time for solving the transport equations, while the CPU_{CHEM} is the CPU time for solving the chemical source term. The CPU_{CFD} decreases linearly (see Figure 1.3.) as the number transported species decreases and the number of species in QSSA increases. The calculation time and the number of transported species needed to solve the problem with desired accuracy depend on the physical problem, the accuracy demands of the solution and CFD model used.

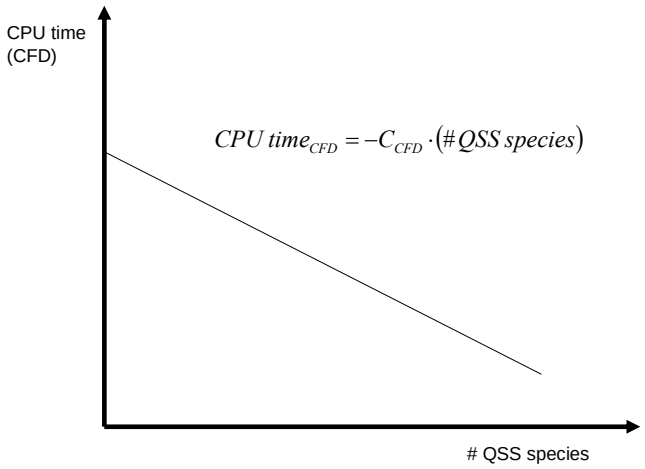


Figure 1.3. A schematic illustration of the CPU_{CFD} vs the number of QSS species. The CPU time decreases linearly with the number of QSS species. C_{CFD} is a positive constant.

One method for chemically reactive flow calculations is to use a CVR or a CPR at each grid point. This method is explained in section 2.3.4.3.1 and is also the method used in this thesis. For this method the CPU_{CHEM} has another behavior than CPU_{CFD}. The reason for this is that the system of algebraic equations must be solved more often than the system of differential equations in order to keep the accuracy of the solution, which leads to the behavior of the CPU_{CHEM} as can be seen in Figure1.4. This is discussed further in section 3.5.3.
CPU_{CHEM} initially decreases to a certain limit as the number of species in QSSA increases and then increases again as the number of species in QSSA increases. The exact behavior

depends on the numerical method and solver used, as well as the accuracy demands on the solution.
The accuracy of the solution from the reduced mechanism normally decreases in steps as the number of QSS species increases. A schematic illustration of this is shown in Figure 1.5. The reason for this is that some individual species, and also some QSS species combinations, can affect the solution of the reduced mechanism more than others.

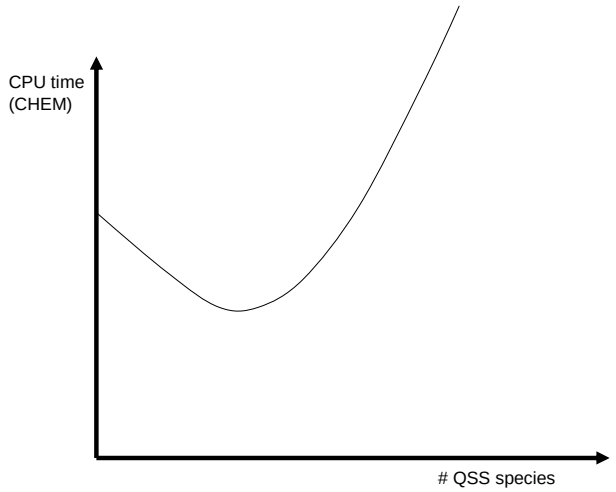


Figure 1.4. A schematic illustration of CPU_{CHEM} vs the number of QSS species. The CPU time first decreases and then increases with the number of QSS species.

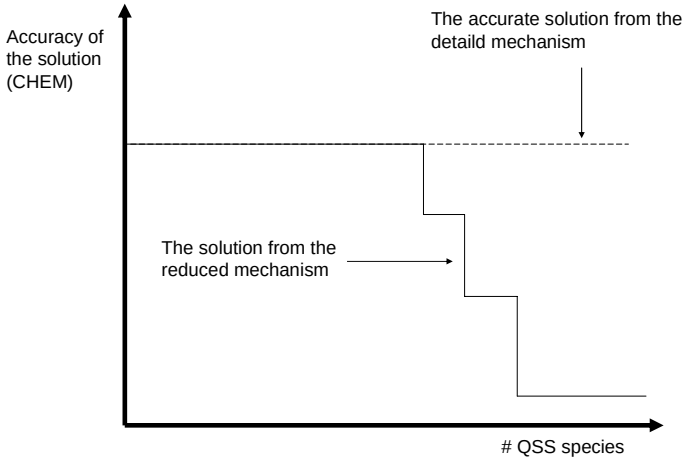


Figure 1.5. A schematic illustration of the accuracy of the solution vs number of QSS species. The figure shows the accurate solution from the detailed mechanism and the solution from the reduced mechanism. The deviation normally increases in steps, since some individual species, and also some QSS species combinations, can affect the solution of the reduced mechanism more than others.

The $CPU_{TOTAL} = CPU_{CFD} + CPU_{CHEM}$ will behave similar to the CPU_{CHEM}, which is shown in Figure 1.4. The reason for this is that $CPU_{CHEM} > CPU_{CFD}$ for most applications involving detailed chemical mechanisms. The accuracy of the solution is a combination of the approximations made in the CFD part and the approximations made in the chemical part. The accuracy of the solution decreases with the number of QSS species in a similar fashion to the accuracy of the chemical part, which is shown in Figure 1.5.

Since CPU_{CHEM} is a large part of the CPU_{TOTAL}, much CPU time can be saved on speeding up the calculation of the chemical source term. In order to speed up the calculation of the chemical source term two problems must be solved. The first problem is how to choose the species that are to be set in QSSA without losing too much accuracy of the solution. The second problem is how to solve the system of DAE in fast and accurate way. Hence, the “Accuracy versus speed” problem can be expressed as;

- Find a method to select the right species to be set in QSSA
- Find a fast and accurate solver combination for the system of DAE representing the chemical source term

1.3.1 “Accuracy vs speed” problem solution

The goal is to reduce the original system of ODE into a system of DAE by QSSA in order to transport as few species as possible to gain CPU time in CFD calculations. This reduction has to be done in such a way that the accuracy of the solution of the reduced system does not deviate too much from the solution of the original system (see Figure 1.6) and that the CPU time of the system of DAE decreases, in the fashion shown in Figure 1.7, with the number of species set in a QSSA.

The Newton-Newton solver combination, which this thesis is based on, moves the minimum of the CPU time curve downwards and toward higher number of QSS compared to the CPU time curve for the Newton-Fixed Point (Newton-FP) solver combination, which is based on previous work within the group. A schematic illustration of this is shown in Figure 1.7. The solver combinations are discussed in detail in section 4.4.

If high accuracy demands are put on the solution, few species can be set in QSS and vise versa. The accuracy of the solution also depends on which species combinations that are set in QSSA.
The Automatic Reduction Tool (ART), which is presented in section 5.4.2.2, automatically selects the QSS species so that the user defined accuracy demands are fulfilled for the reduced mechanism. Hence, the solutions to the “Accuracy versus speed tradeoff” problem according to this thesis are;

- To use solver combination that is presented in section 4.4. The performance of the solver combination will be presented in chapter 6.
- To use the Automatic Reduction Tool (ART) that automatically selects the QSS species. The performance of the ART will also be presented chapter 6.

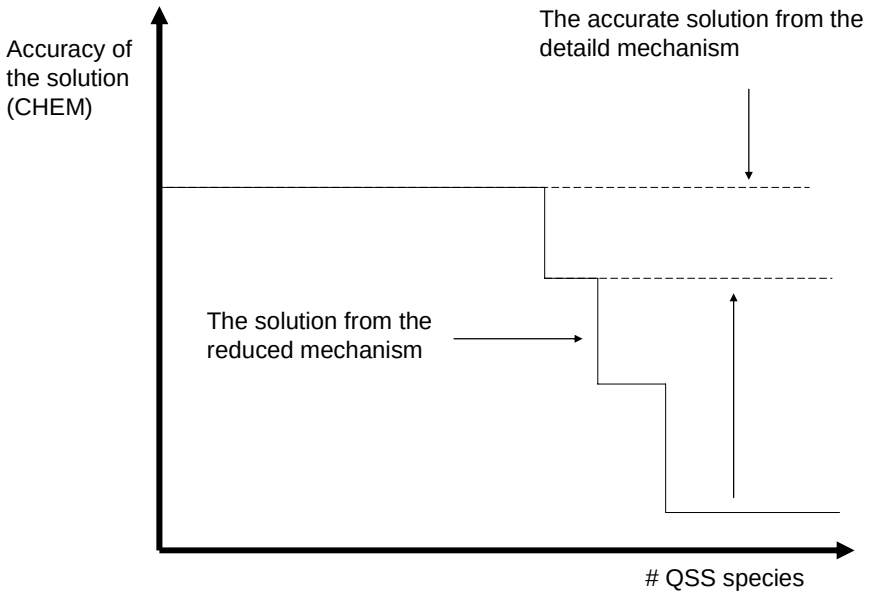


Figure 1.6. A schematic illustration of the accuracy of the solution vs number of QSS species. The aim is to decrease the difference between the accurate solution from the detailed mechanism and the solution from the reduced mechanism, by choosing the right species to set in QSSA.

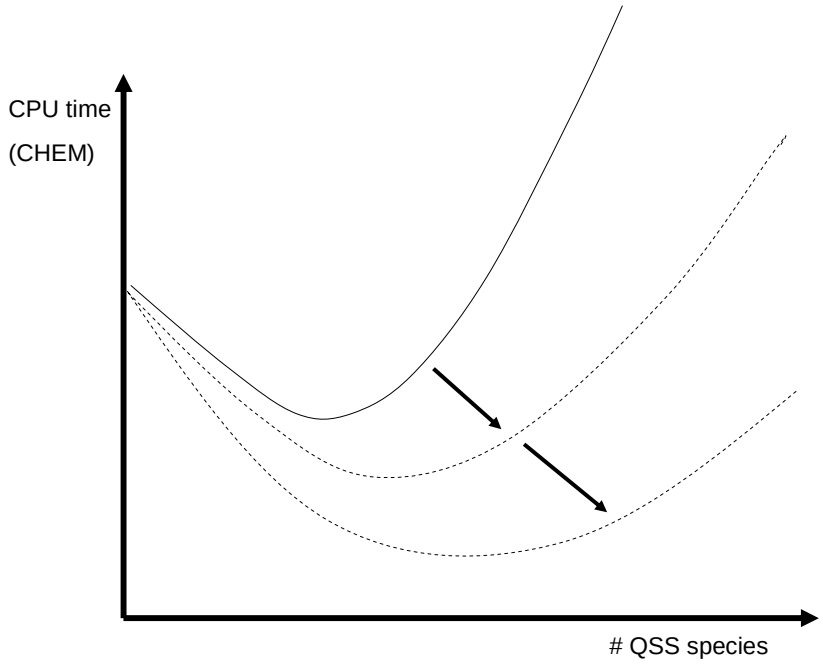


Figure 1.7. A schematic illustration of the CHEM part of the CPU time vs number of QSS species. The goal is to construct a solver combination that moves the minimum of the curve downwards and toward higher number of QSS.

1.3.2. Is the reduction procedure worthwhile?

The reduction procedure of the detailed mechanism costs CPU time. The question then arises if the CPU time gain from the use of the reduced mechanism will compensate for the CPU time cost from the reduction procedure. The answer to the question is that it all depends on how many times the reduced mechanism is used. The total time from the use of the detailed mechanism will surpass the total time from the use of the reduced mechanism plus the time from the reduction procedure when the number of simulations is large enough. This is illustrated in Figure 1.8. In practice, a reduced mechanism is often used many times in order to investigate various parameter combinations. Hence, the reduction procedure is worthwhile.

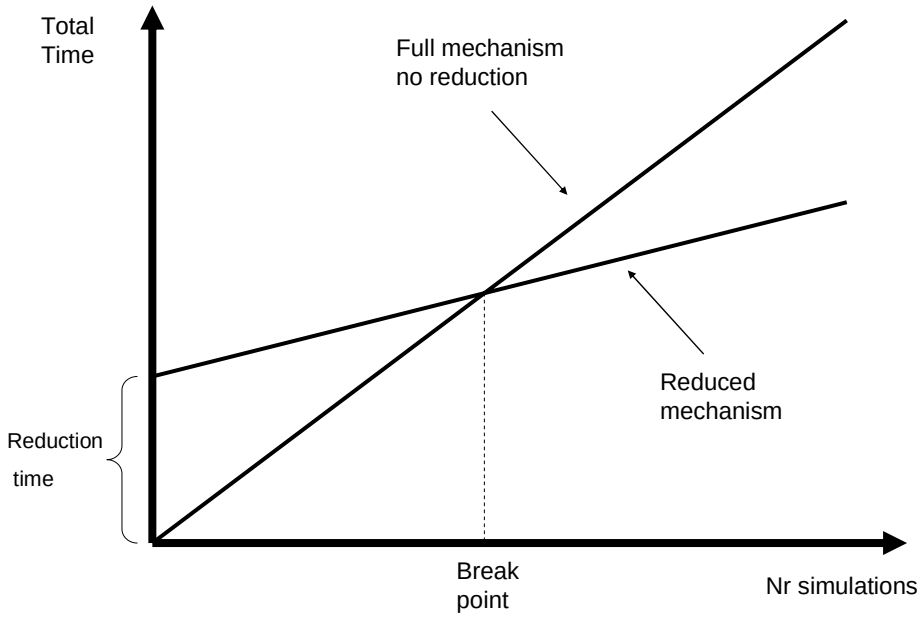


Figure 1.8. Total time vs Number of simulations for the reduced mechanism and the full mechanism without any reduction. The total time involves the accumulated time for all simulations plus the reduction time. The total time for the reduced mechanism becomes smaller than the total time for the full mechanism at the break point. The location of the breakpoint is problem dependent.

1.4. Previous work

Previous work at the Division of Combustion Physics at Lund University was based on a Newton solver for the differential equations and a fixed point solver for the system of algebraic equations, which leads to a behavior similar to that in Figure1.4, that is, a strong increase in CPU time compared to the detailed case when the number of QSS becomes large. This behavior motivated the work of this thesis.

Other solutions to the problem have been tried in other groups. Decoupling of the QSS species equation has been tested [7,10]. The speed up of the method increases with the amount of decoupling between the algebraic equations. However, the accuracy of the solution method decreases with the amount of decoupling between the equations.

The Principal Component Analysis (PCA) method has been tested in order to decrease the cost for the Jacobian calculations [8]. The speed up of the method increases with the amount of neglected modes. However, the accuracy of the solution decreases with the amount of neglected modes.

Whenever a simplifications of the system of equations are made in order to decrease the CPU time, the accuracy of the solution decreases. Hence, there is always a trade off between speed and accuracy.

It is hard to compare the amount of speed up and the accuracy reduction from article [7,10] and [8] to the work in this thesis, since different measures for speed up and error are used. Also, the numerical methods are tested on different problems with different chemical mechanisms.

1.5. Limitations

This thesis focuses mainly on the development and performance of the automatic reduction tool for mechanism reduction and the development and optimization of the solver combination for the system of DAE.
Consequently, a few things are mentioned which are outside the main objectives in this thesis. Those things are CFD, mechanism generation, alternative reduction methods for chemical mechanisms and alternative numerical methods for solving differential and algebraic equations.

1.6. Chapter References

[1] S. R. Turns, *An Introduction to Combustion Concepts and Applications* second edition McGRAW-HILL INTERNATIONAL EDITIONS 2000
[2] S. Solomon et al, *Climate change 2007 The Physical science basis*, Cambridge (2007)
[3] J. Fenger, J.C Tjell (Eds), *Air pollution*, Polyteknisk Forlag (2009)
[4] T Lu, C K Law, *Toward accommodating realistic fuel chemistry in large-scale computations,* Progress in Energy and Combustion Science 35 (2009) 192-215
[5] K.K. Kuo, *Principles of combustion*, John Wiley and sons (1986)
[6] K. Fieweger, R. Blumenthal, G, Adomeit, *Combust. Flame* 109 (1997) 599-619
[7] J.Y. Chen, Y.F. Tham, *Speedy solution of quasi-steady-state species by combination of fixed-point iteration and matrix inversion*, Elsevier, Combustion and Flame (2008)
[8] M.A. Singer, W.H. Green, *Using adaptive proper orthogonal decomposition to solve the reaction-diffusion equation*, Elsevier, Applied Numerical Mathematics 59 (2009) 272-279.
[9] T. Turanyi, A.S. Tomlin, M.J. Pilling, *On the error of the Quasi-Steady-state Approximation J. Phys. Chem*. 1993, 97, 163-172
[10] A.C. Zambon, H.K. Chelliah, *Explicit reduced reaction models for ignition, flame propagation, and extinction of $C_2H_4/CH_4/H_2$ and air systems,* Combustion and Flame 150 (2007) 71-91

Chapter 2.

Modeling of the physics and chemistry

2.1 Chapter Introduction

This chapter focuses on the mathematical models used to describe a real physical system. The focus also lies on the assumptions and simplifications made in the models and how and why the models differ from the physical system.
Almost all simulations in this thesis are based on the Adiabatic Constant Volume Reactor (ACVR) model, which is described in detail. The chapter also contains a brief theory about the applications of the ACVR model in CFD/Chemistry interaction models. Finally the chapter contains theory about chemical kinetics and the simplification that the Quasi Steady State Approximation (QSSA) means.

2.2 The Physical System

The work presented in this thesis is based on a model for chemical kinetics occurring in an Adiabatic Constant Volume Reactor (ACVR), which is a special case of the Constant Volume Reactor (CVR). The ACVR model is based on assumptions and simplifications of the real physical system which lead to a difference between the numerical solution of the model and the physical reality.
The features of a real physical system, which the ACVR is a model of, are listed below and are illustrated in Figure 2.1. The points where the ACVR model differs from the real physical system are written with bold letters. A detailed mathematical description and illustration of the ACVR is found in section 2.3.4.1.1.

- The volume is constant, which means that there is not any work done on or by the surroundings.
- The system is closed from the surroundings, which means that there is not any in or out flow of mass, energy and momentum due to mass flow through the walls of the system.
- Energy in form of heat can be exchanged with the surroundings by heat conduction through the walls of the system and thermal radiation, since adiabatic conditions are very hard to achieve in reality.
    - **This contribution is assumed to be zero in the ACVR model.**
- The total mass is constant
- The mass is not homogeneously distributed within the volume.
    - **The mass distribution is assumed to be homogeneous in the ACVR.**
- The mixture composition is not homogeneous within the volume.
    - **The mixture composition is assumed to be homogeneous in the ACVR.**
- The mass is distributed on different chemical species and the distribution changes with time due to chemical reactions and movement of molecules.
- The molecules move within the volume due to concentration, pressure and temperature gradients.
    - **The spatial movement of molecules is not considered in the ACVR, since the two previous points assumes that concentration and temperature gradients do not exist.**
- The internal energy is constant and is the total energy of the molecules composing the system.
- The total energy of a molecule is the sum of the thermal energy (the energy associated with translation, rotation and vibration energies) and the chemical energy (the energy associated with the atomic bonds in a molecule).
- All the molecules are in gas phase
- The initial fuel mixture consists of fuel and air
    - **The initial fuel mixture consists of fuel, oxygen and nitrogen only**.

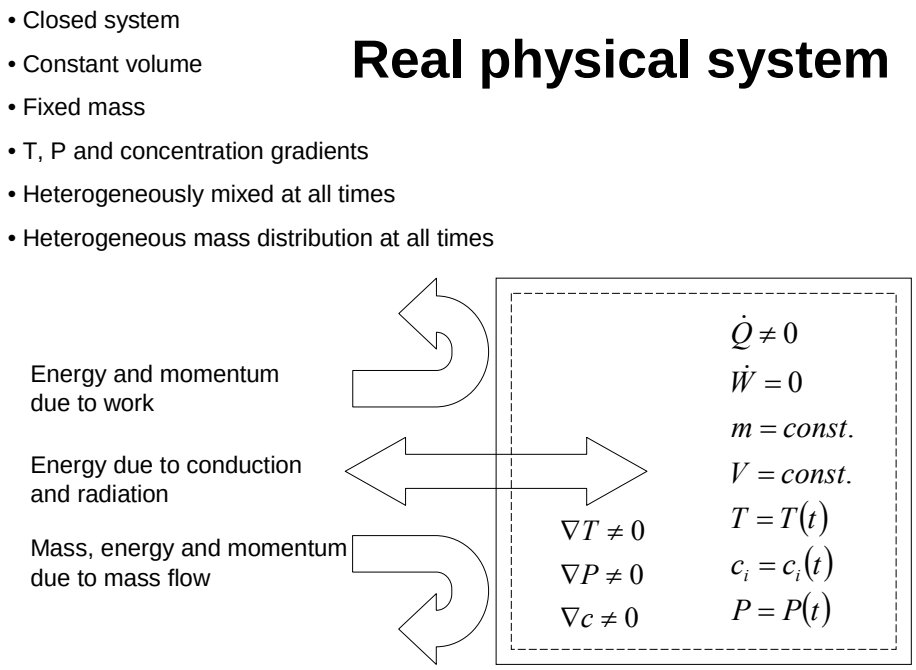


Figure 2.1. The features of the real physical system that the ACVR is a model of. The volume and total mass of the system is constant. The system contains temperature, pressure and concentration gradients, which gives a heterogeneous mixing and mass distribution. Energy due to conduction and radiation can enter or leave the system. The temperature, pressure and concentrations change with time.

2.2.1. The dynamics of the physical system

The dynamics of the physical system will be described without equations in this section, since an appropriate mathematical model will be chosen later in the text.
The physical system consists of atoms and molecules with different masses, velocities and energies. The molecules collide with each other, causing chemical reactions, and the walls of the system and exchange energy and momentum. This is illustrated in Figure 2.2. The collision frequency depends on the local density of the colliding molecules, the local density of other molecules and the velocities (which is a function of the thermal energy of the system) of the colliding molecules. When the molecules collide there is a probability for chemical reactions to occur, which depends on if the thermal energy of the molecules is larger than the activation energy for the reaction. A chemical reaction breaks bonds in the reacting molecules and forms new bonds in the product molecules (unless it is a pure dissociation reaction). The chemical energy associated with bond breaking and formation will, depending on if the chemical reaction is exothermic or endothermic, increase or decrease the amount of thermal energy in the system, which in turn is related to the temperature and pressure of the system.

The rate of the chemical reactions, and thereby the rate of change in chemical composition, depends on the temperature and the chemical composition of the system. The rate of change of temperature depends on the chemical composition, since it determines the chemical reactions that occur and therefore the amount of thermal energy that is released or absorbed in the reactions.
In a system involving many exothermic reactions, like a combustion system, the coupling between the heat release in the reactions and the thermal energies influence on the rate of the reactions can cause a very rapid change in the chemical composition and a very rapid increase in the temperature and pressure in the system.
Or in other words, energy release from the exothermic chemical reactions results in higher velocities of the chemical species in the system, which in turn gives higher temperature and pressure in the system. The higher velocities of the chemical species result in more frequent collisions between the chemical species, which in turn gives more frequent energy release into the system. This results in even higher velocities of the chemical species and so on. This process continues until there is a very rapid change in the chemical composition and a very rapid increase in the temperature and pressure in the system. Such a process is often called an ignition of the system.
For a more detailed description about the coupling between the chemical species concentrations and the temperature, a mathematical formulation is needed and can be found in section 2.3.5.
An example of the ignition process can be seen in Figure 2.3, which shows Temperature, Pressure, OH and HO_2 profiles for combustion of N-Heptane. The rapid increase in the profiles is typical for an ignition process. A more detailed illustration of the ignition process can be found in section 6.2.1.1.5 and section 6.3.1.3, showing a combustion simulation of N-Heptane and Methane/Propane respectively.

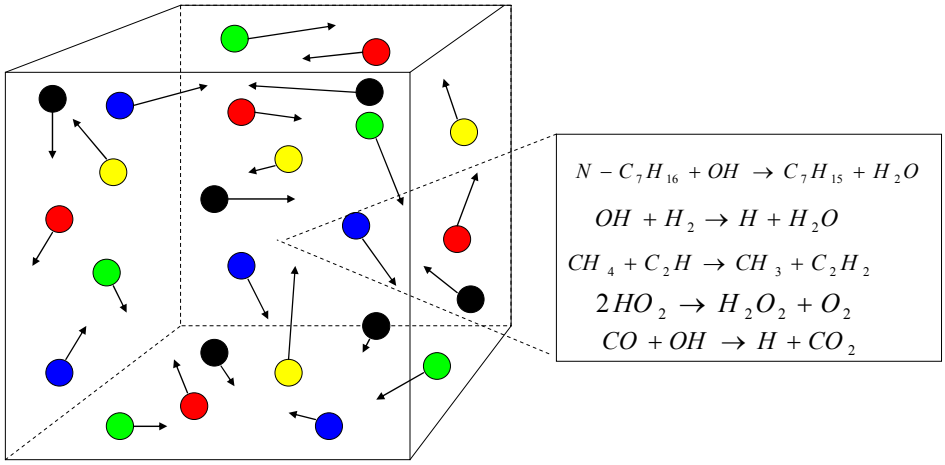


Figure 2.2. The physical system under investigation is a closed system with a constant volume and a fixed mass, which is divided up on different molecules. The molecules react with each other causing energy to be released and absorbed by the breaking and forming of chemical bonds. The temperature, pressure and concentrations of the species vary when time evolves. Different types of molecules are symbolized with different colors and different velocities are symbolized with different arrow lengths. The chemical formulas symbolize that chemical reactions occur within the system.

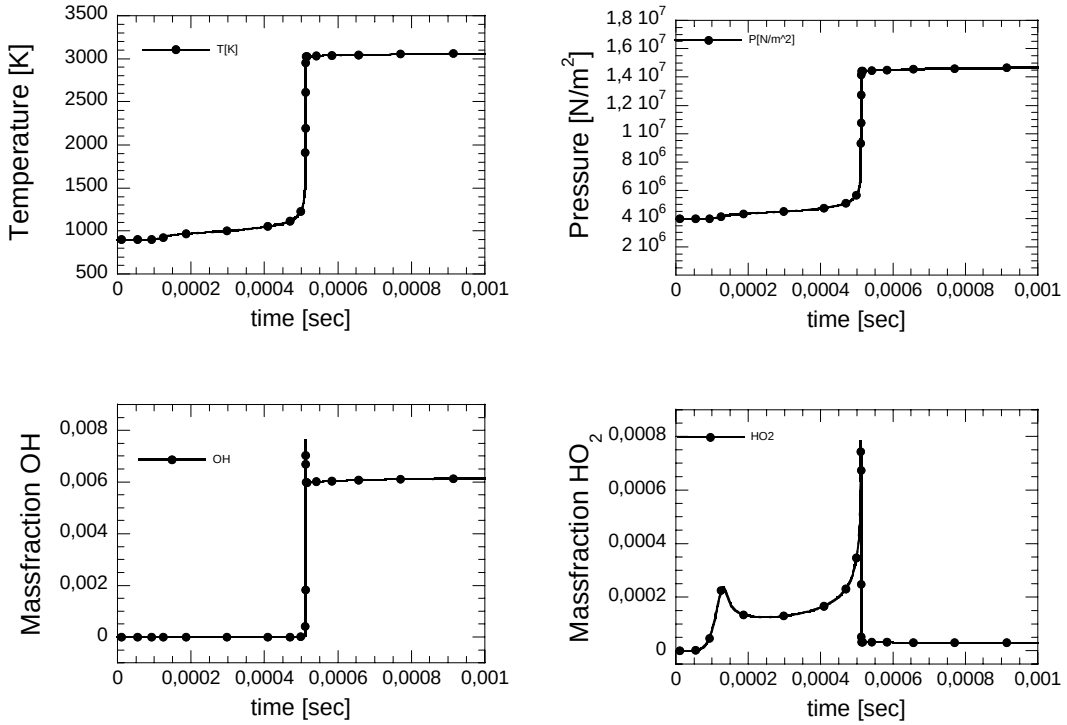


Figure 2.3. Temperature, Pressure, OH and HO_2 profiles for auto-ignition of N-Heptane. The rapid increase is typical for an ignition process.

2.3. Modeling of the physical system

The physicist Hans-Uno Bengtsson once said something like “The only full and accurate description of the universe is the universe itself. All other descriptions are simplifications”. Hence, when a physical phenomenon is described, it is always a simplified description in one way or another.
It is therefore important to state the simplifications and assumptions that are made and why they are made when the phenomenon is described. The needed accuracy of the description, which can be made in words or in mathematical terms, is case dependent.

It is important to distinguish between physical reality, the experiments, the mathematical model, the discretized mathematical model and the numerical solution of the discretized mathematical model. This is illustrated in Figure 2.4.

The physical reality is investigated through experiments and predicted by the numerical solution of more or less detailed mathematical models. However, the outcome of the experimental results and the numerical solution to the models are all projections of the physical reality. The projections of the physical reality contain uncertainties, which can be either systematic or random. However, the projections of the most detailed

experiments and the most detailed models are the best representation of the physical reality at the present time.
Often the numerical solution is compared to the experimental results in order to validate the model and numerical solution. This is a good approach if the projection of the physical reality due to experiments contains small uncertainties. However, if the projection of the physical reality due to experiments contains large errors, the comparison of the numerical solution to experiments can lead to incorrect optimization of the model. If the experimental data contains large errors, the comparison of a (simplified) model can be made to the most detailed model, which within the field of combustion corresponds to DNS simulations.

First of all, there is a difference between the physical reality and the mathematical model. This difference can either be due to (black dots correspond to general reasoning, while white dots correspond to the assumptions made in this thesis);

- The fact that not all physical effects have been considered in the mathematical model. Hence, the model contains simplifications and assumptions.
    - The mathematical model in this thesis is based on the system of ODE for the ACVR, which is described in section 2.3.4.1.1. The ACVR assumes that temperature, pressure and concentration gradients do not exist and neglects heat exchange with the surroundings in terms of conduction and radiation
    - Not all chemical species and reactions are considered in the chemical mechanisms
- Errors and uncertainties in the physical and chemical parameters.
    - The chemical mechanisms contain experimentally determined constants, which in turn contain errors, for the Arrhenius expression. Examples of constants are the activation energy and the pre exponential factor.
- The continuum approximation
    - The mathematical model in this thesis is based on the continuum approximation even though the real physical system contains individual molecules.

There is also a difference between the discretized mathematical model and the mathematical model. This difference can either be due to (black dots correspond to general reasoning, while white dots correspond to the assumptions made in this thesis);

- Discretization errors, when the mathematical model is discretized in order to be solved numerically
    - The finite time step size
    - The Jacobian in the outer Newton solvers is calculated with finite differences
- Truncation errors in, for instance, the Taylor expansions of the mathematical expressions
    - Both the inner and outer Newton methods use truncation of the Taylor expansion after the linear order for the iteration procedure

  - The predictor for the outer Newton solver use truncation of the Taylor expansion after the *n:th* order (*n* is user defined)
  - The time derivative for the outer Newton solver uses truncation of the Taylor expansion after the *n:th* order (*n* is user defined)

There is also a difference between the exact solution to the discretized mathematical model and the numerical solution. This difference can either be due to (black dots correspond to general reasoning, while white dots correspond to the assumptions made in this thesis);

- Numerical errors, like round off errors and errors due to floating point precision
  - Both the inner and outer Newton solvers use DOUBLE PRECISION
- Tolerance levels in the numerical method, that is, the relative change in species concentrations between two iteration steps that is acceptable for convergence
  - The tolerance levels for both the inner and outer Newton solvers are discussed in section 4.2.1.

Finally, there is a difference between the numerical solution of the discretized mathematical model and the physical reality. This error depends on the error between the physical reality and the mathematical model, the mathematical model and the discretized mathematical model and the error between the discretized mathematical model and the numerical solution of the discretized mathematical model.
Hence, the best way to get a good description between the physical reality and the numerical solution of the discretized mathematical model is to;

- Use a mathematical method that takes as many physical effects into consideration as possible
- Minimize the discretization and truncation errors in the discretized mathematical model
- Use a numerical method that has low tolerance levels and uses the highest floating point precision
- Use as accurate experiments as possible in order to get accurate constants in the model and accurate validation against the experiments

However, complicated mathematical models and numerical methods cost CPU time and require computer memory, which means that the aim is to minimize the difference between the physical reality and numerical solution to the lowest possible cost. Hence, there is always a trade off between the accuracy of the numerical solution on one hand and the CPU time and the memory requirements on the other.

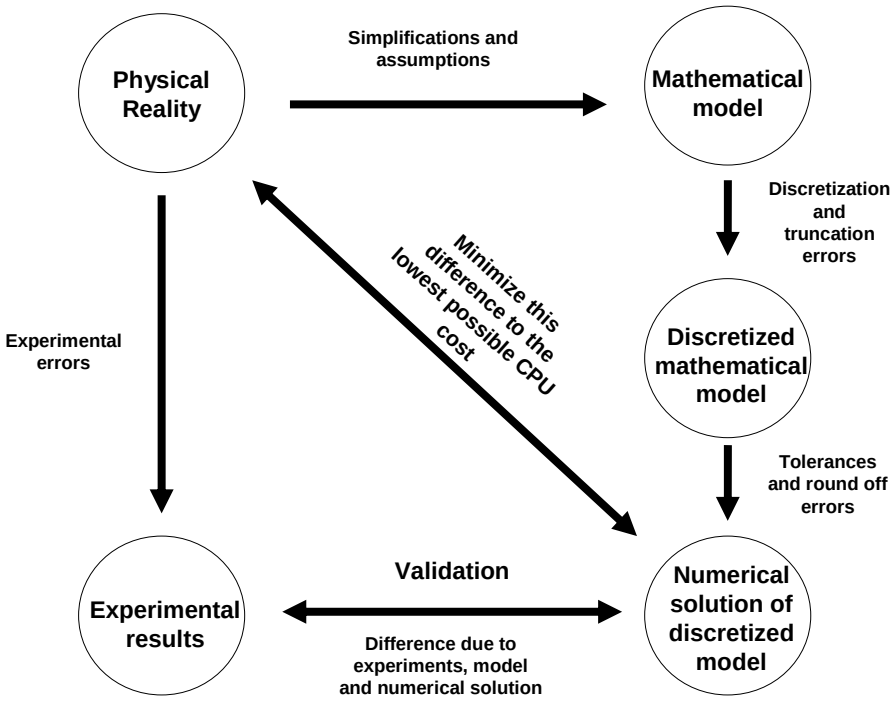


Figure 2.4. A mathematical model contains errors compared to the physical reality due to simplifications and assumptions. A discretized version of the mathematical model contains discretization and truncation errors. A numerical solution of the discretized mathematical model contains errors due to tolerances and round off errors. The numerical solution of the mathematical model is validated against the experimental results, which also contain errors. The aim is to minimize the error between the physical reality and the numerical solution of the discretized mathematical model to the lowest possible CPU cost. Hence, a trade off between accuracy and CPU time.

This thesis is based on reduction of detailed chemical mechanisms. The system of equations for the ACVR based on a detailed chemical mechanism is a mathematical model of the physical reality. The system of equations for the ACVR based on a reduced chemical mechanism is a simplified mathematical model, which is extracted from the detailed model by the use of assumptions and simplifications. The reduction, which in this thesis is based on the QSSA, must be done in a way that the accuracy of the solution based on the reduced mechanism is within acceptable limits of the solution based on the detailed mechanism.

In this thesis the detailed model contains errors compared to the physical reality due to simplifications and assumptions. The simplified model is extracted from the detailed model via further simplifications and assumptions. The discretized version of the detailed model and simplified model contains discretization and truncation errors. The numerical solution of the discretized detailed model and the discretized simplified model contains errors due to tolerances and round off errors.

Consequently, there is also a difference between the numerical solution to the discretized detailed model and the numerical solution to the discretized simplified model. This difference can be enhanced or reduced by the use of different numerical methods, discretization errors, truncation errors, tolerance levels and round off errors.
The numerical solution of the detailed model is validated against the experimental results, which also contain errors. The numerical solution of the simplified model is validated against the numerical solution of the detailed model. This is illustrated in Figure 2.5.

The acceptable difference between the numerical solution of the simplified model and the numerical solution of the detailed model is often subjective and user defined. However, when the acceptable difference is defined, the CPU cost for obtaining the solution to the simplified model must be minimized. This leads to a trade off between speed and accuracy. Hence, the simplifications and assumptions, the discretization and truncation errors and the tolerance levels and round off errors must be chosen wisely in order to obtain the best trade off between speed and accuracy.

The importance of making the correct assumptions in the mathematical model and choosing the appropriate numerical method becomes very clear if the system is complicated and the computational time is in the order of days, weeks or months. However, the difficulty of making the correct assumptions and simplifications in the mathematical model increases when the system becomes more complicated, which means that there is a strong driving force to develop new methods/tools for choosing these assumptions. Hence, in order to reduce the CPU time and the memory requirements, while maintaining the accuracy of the numerical solution within acceptable limits in a complex system, the following can be done;

- Use a method/tool that chooses the simplifications to be made in the mathematical model
- Speed up the numerical method without losing accuracy of the solution

The tool used for simplifying the mathematical model in this thesis is an Automatic Reduction Tool (ART), which is discussed in detail in section 5.4.2.2. The speed up of the numerical method is an optimization for the Newton solver for the system of NAE. This optimization is discussed in detail in section 4.3.3.
A mathematical model of a complex system often involves an input/output model of a system of differential and algebraic equations (DAE), which is presented in section 2.3.2.1.

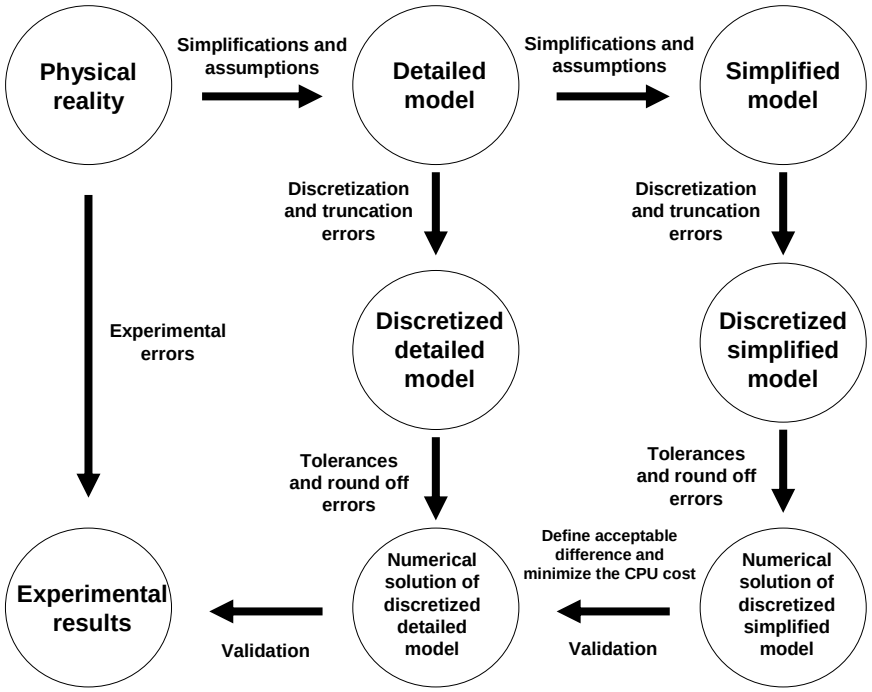


Figure 2.5. In this thesis the detailed model contains errors compared to the physical reality due to simplifications and assumptions. The simplified model is extracted from the detailed model via further simplifications and assumptions. The discretized version of the detailed model and simplified model contain discretization and truncation errors. The numerical solution of the discretized detailed model and the discretized simplified model contain errors due to tolerances and round off errors. The numerical solution of the detailed model is validated against the experimental results, which also contain errors. The numerical solution of the simplified model is validated against the numerical solution of the detailed model. The acceptable difference between the numerical solution of the simplified model and the numerical solution of the detailed model is defined by the user. Thereafter the numerical solution to the simplified model is then obtained to the lowest possible CPU cost. This leads to a trade off between speed and accuracy.

2.3.1. Choosing an appropriate model

In order to choose an appropriate model for this real physical system, the size of the system must be considered. Continuum mechanics can be used for a large system, but as the size of the system becomes smaller the continuum approximation becomes less valid. When the size of the system is less than the continuum limit, statistical physics must be used to describe the system.
The continuum approximation has the advantage that individual molecules do not have to be considered. Instead variables like density and species mass fraction can be used in the system of PDE, involving spatial derivatives of the variables, describing the system. The ACVR is based on the continuum approximation but also on other assumptions, which become more valid as the size of the system decreases. Hence, there is a size region for which the ACVR reactor is a good approximation for the real physical system. This is illustrated in Figure 2.6.
When the system is larger than the upper limit of this region, the equations for chemically reactive flow must be solved and when the system is smaller than the lower limit the continuum approximation is not valid and another model, based on statistical physics, must be used.

The models used in thesis are based on the continuum approximation, which means that the system can be modeled with differential equations for density, species mass fractions, momentum and energy. The mathematical formulation of the chemical kinetics is found in section 2.3.5.
The dynamics of individual atoms and molecules are not modeled, since the CPU time and memory requirements would be too large.

In order to choose an appropriate model for this real physical system, the physical size of the system must be considered. If the system is large the equations for chemically reactive flow (see section 2.3.4.3.1) must be solved, but if the system is small enough the assumptions of the ACVR model become more correct. However, the system cannot be smaller than for the continuum approximation to be valid.

There are many systems falling under the description presented above, which means that the derived equations and numerical methods presented in this thesis will be applicable to any (gas phased) chemical system that can be modeled by an ACVR. However, the applications of the numerical methods in the work behind this thesis will focus only on combustion systems.

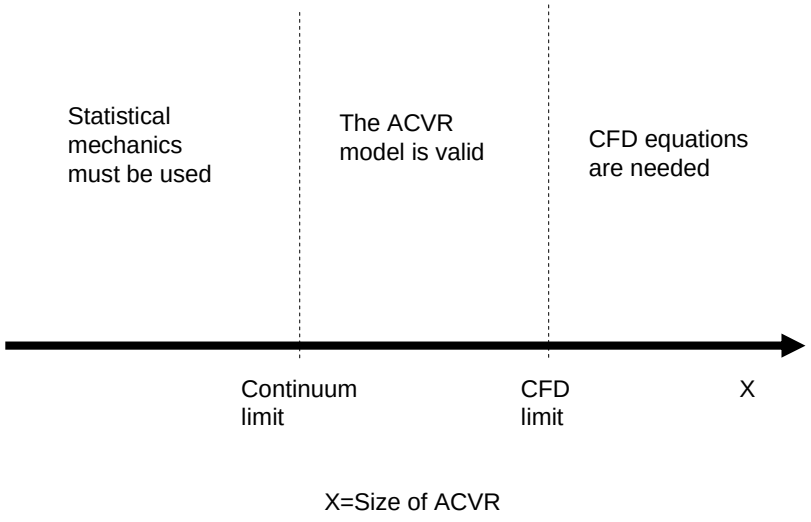


Figure 2.6. The assumptions in the ACVR are only valid for a certain size of the reactor. If the size is smaller than the continuum limit, the continuum approximation and ideal gas law are not valid anymore. If the size is larger than the CFD limit the assumptions 0-D are not valid anymore.

2.3.2. Systems of Differential and Algebraic Equations (DAE)

A system of DAE is a system of nonlinear differential equations and nonlinear algebraic equation plus boundary conditions. The system of differential equations in a system of DAE can either be Ordinary Differential Equations (ODE) or Partial Differential Equations (PDE).

This general system of DAE can describe (almost) anything like physical, chemical, biological, physiological, economical systems etc. The solution of such a general system must be obtained numerically. Numerical methods for solving such a system are presented in section 3.5. An input/output model of a system of DAE is presented in section 2.3.2.1. A general system of DAE, where the differential equations are of the ODE form, can mathematically be described as;

$$\begin{aligned} \frac{d(\mathbf{x}(t))}{dt} &= \mathbf{f}(\mathbf{x}(t),t) \\ 0 &= \mathbf{g}(\mathbf{x}(t),t) \end{aligned} \tag{2.1}$$

plus the initial conditions;

$$\mathbf{x}(t=0) = \mathbf{x}_0 \tag{2.2}$$

The vector of unknowns is;

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_N(t) \end{bmatrix} \tag{2.3}$$

where N is the number of equations. The time derivative of the vector of unknowns is;

$$\frac{d(\mathbf{x}(t))}{dt} = \begin{bmatrix} \frac{d(x_1(t))}{dt} \\ \vdots \\ \frac{d(x_N(t))}{dt} \end{bmatrix} \tag{2.4}$$

The function-vectors are;

$$\mathbf{f}(\mathbf{x}(t),t) = \begin{bmatrix} f_1(\mathbf{x}(t),t) \\ \vdots \\ f_N(\mathbf{x}(t),t) \end{bmatrix} \tag{2.5}$$

and

$$\mathbf{g}(\mathbf{x}(t),t) = \begin{bmatrix} g_1(\mathbf{x}(t),t) \\ \vdots \\ g_N(\mathbf{x}(t),t) \end{bmatrix} \tag{2.6}$$

where the function vectors have explicit and implicit dependence on time.

However, the system of DAE, which the simulations this thesis is based on, only involves ODE from the ACVR (see section 2.3.4.1.1.) and the algebraic equations come from applying the QSSA (see section 2.3.5.4.) to some of the ODE for the chemical species. The system of DAE for the chemical kinetics is a special case of the system described above, since they only have an implicit dependence on time and do not have an explicit dependence on time.

An example of a system of PDE is the balance equations for mass, chemical species, momentum and energy used in CFD calculation for chemically reactive flow, which are presented in section 2.3.4.3.1. If the differential equations are of the PDE form, **f** and **g** are operators and involve derivatives of **x**.

2.3.2.1. Input/output model of a system of DAE

Modeling of systems in natural science (and other sciences as well) often involves an input/output model for a system of some sort. This is illustrated in Figure 2.7. The input/output model consists of the input-vector, the system, described in terms of PDE, ODE or DAE plus boundary conditions and initial conditions, and an output vector from the system.
Modeling of physical/chemical systems in general means that appropriate physical/chemical theories are chosen to describe the systems in terms of ODE, PDE or DAE. The theories this thesis is based on are Thermodynamics and Chemical kinetics. The input vector provides the initial conditions for the equations and (if needed) parameter settings for the solver. The system then evolves in time with the help of a solver and thereafter produces an output vector.
The input vector in work behind this thesis consists of initial species concentration, initial temperature, initial pressure and also solver settings, while the output vector consists of the final species concentrations, temperature, pressure and also diagnostic parameters. The choice of numerical method and implementation of the solver is critical if low CPU time and high accuracy of the solution is of importance. The choice of numerical method for this thesis is discussed in detail in chapter 3.

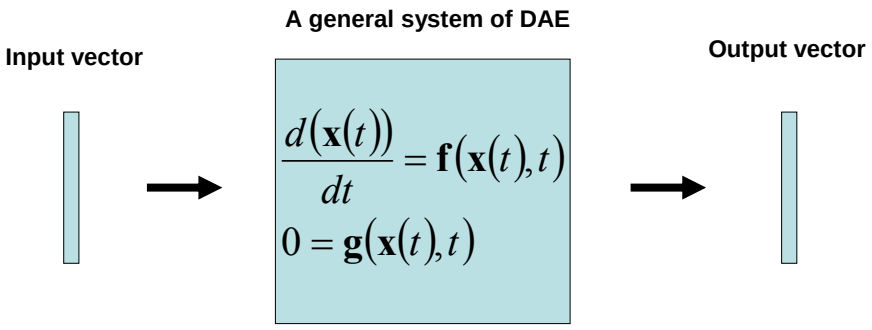


Figure 2.7. An input/output model of a general system of DAE, where the differential equations are of ODE form. ***x***, is the vector of unknowns, ***f***, and , ***g***, are function vectors. In the general case the differential equations in the system of DAE can either be PDE or ODE.

2.3.3. 3-D models of chemically reactive flow

The general mathematical formulation of systems involving chemically reactive flow consists of the balance equations for mass, momentum and energy [1]. These balance equations are coupled with PDEs, involving derivatives with respect to both time and space, for each chemical species, the equation of state (ideal gas law) plus additional equations for the boundary conditions to make up the final system of equations describing chemically reactive flow. The balance equations can be found in section 2.3.4.3.1.

When spatial transport effects can be neglected the coupled set of PDE's become a coupled set of ODE's for the species concentration and the energy, which evolves only in time. Systems that only evolve in time are often called 0-D systems and are described in section 2.3.4.
The computational cost and memory requirements for solving the system of PDE in 3-D depend on the chemical mechanism and the spatial resolution needed to perform the simulations with the desired accuracy. The computational cost and memory requirements also depend on the CFD model and flow-chemistry interaction model.

Direct Numerical Simulation (DNS) solves the 3D equation system directly with high accuracy of the solution, but often to a high computational cost, while other CFD models use different approximations in order to solve the equations to a lower computational cost, which often results in a decreased accuracy of the solution. Some CFD models used in combustion are listed in table 2.1.
Simulations of 3D systems with a detailed reaction mechanism are possible to perform with the computers available today if an appropriate CFD model and model for the fluid/chemistry interaction is chosen, but DNS is not possible if the flow Reynolds number is high, which it is for highly turbulent flow, and solution shall be obtained within a reasonable time (unless the region in space is extremely small).
Approximate CFD models and reduced chemical mechanisms are beneficial in order to reduce the CPU time and memory requirements for 3-D simulations, which is one of the motivators and future applications for the work behind this thesis. One way to couple CFD programs and chemical kinetics programs is described in section 2.3.4.3.
Practical combustion processes almost always occur in turbulent environment, which means that models accounting for turbulence-chemistry interaction must be used, since DNS is too time consuming. A problem in turbulent combustion modeling is the closure of the chemical source term. This is elaborated on further in textbooks [1,9] covering this subject and not treated in this thesis.

Table 2.1 shows columns with different CFD (combustion) models, different chemical models and different CFD/chemistry interaction models. The complexity of the models increases downwards in all columns. When modeling chemical reactive flow one must choose one model in each column.
Flamelets is independent of the number of species, since chemical libraries are used, which means that the transportation does not benefit much from a reduced mechanism.
All CFD models in table 2.1 can use a CVR or CPR in each grid point and each CVR or CPR benefits from a proper reduced mechanism in terms of CPU time (see section 6.2.4).
The DNS does not need a Flow-chemistry interaction model, since all equations are solved directly.
The preferred complexity level of each column in table 2.1 is problem dependent. However, similar complexity of all models is preferable, since the least complex model will be the bottle neck of the simulation in terms of accuracy.
The CFD calculation in Chapter 6 used RANS, reduced and detailed chemistry and laminar source term closure and operator splitting.

Table 2.1. The table shows different CFD models, chemistry models and Flow-chemistry interaction models. The complexity of all models increases downwards.

	CFD	**Chemistry**	**Flow-chemistry interaction**
Increasing complexity	RANS	1-step mechanism 4-step mechanism	Eddy dissipation Laminar source term closure
↓	LES	Reduced mechanism	Time scale Flamelet
	DNS	Detailed mechanism	Conditional moment closure Transported PDF

2.3.4. 0-D models

0-D models do not involve any spatial resolution at all and the system of equations evolves in time only, which basically means that the 0-D models are obtained if the time derivatives remain and the spatial derivatives are deleted from the 3-D models of chemically reactive flow. An example of 0-D models are the CVR and CPR, which are describe in section 2.3.4.1 and 2.3.4.2.
CPU time and memory requirements are not of great importance for the 0-D model when it is used on its own. However, when many 0-D models are coupled in a 3-D model and the number of grid points are the order of 10^6, the memory and CPU time requirements is of great importance. Hence, it is essential to minimize the CPU time and memory requirements of each CVR and CPR.

2.3.4.1. System of ODE for the Constant Volume Reactor (CVR)

An example of a 0-D model is the CVR. A CVR is a closed system with a constant volume, fixed mass and without mechanical work. There is not any in or out flow of mass or energy due to mass flow. However, energy can be exchanged with the surrounding due to heat conduction through the walls and thermal radiation. The species mixture is assumed to follow the ideal gas law and to be perfectly mixed and homogeneous throughout the volume. Hence, temperature, pressure and species concentration gradients do not exist in the CVR. The variables of the system are temperature, pressure and the mass fractions of the species. This is illustrated in Figure 2.8.

The system of ODE is described by one ODE for the temperature and one ODE for each species concentration. All ODEs in the system depend on the temperature and the species concentrations [5]. Hence;

$$\frac{dT}{dt} = f_T(\mathbf{x}, T) \tag{2.7}$$

$$\frac{dx_i}{dt} = \omega_i(\mathbf{x}, T) \quad i = 1,..,N_S \tag{2.8}$$

where N_S is the number of species. The ODE for the temperature is derived from the energy conservation equation. The derivation is found in the chapter appendix. Hence, the system of ODE for the CVR is;

$$\rho c_v \frac{dT}{dt} = \frac{\dot{Q}}{m} - \sum_{i=1}^{N_S} \omega_i(\mathbf{x}, T) \cdot U_i \tag{2.9}$$

$$\frac{dx_i}{dt} = \omega_i(\mathbf{x}, T), \; i = 1, N_S \tag{2.10}$$

The source term is described in detail in section 2.3.5.3.

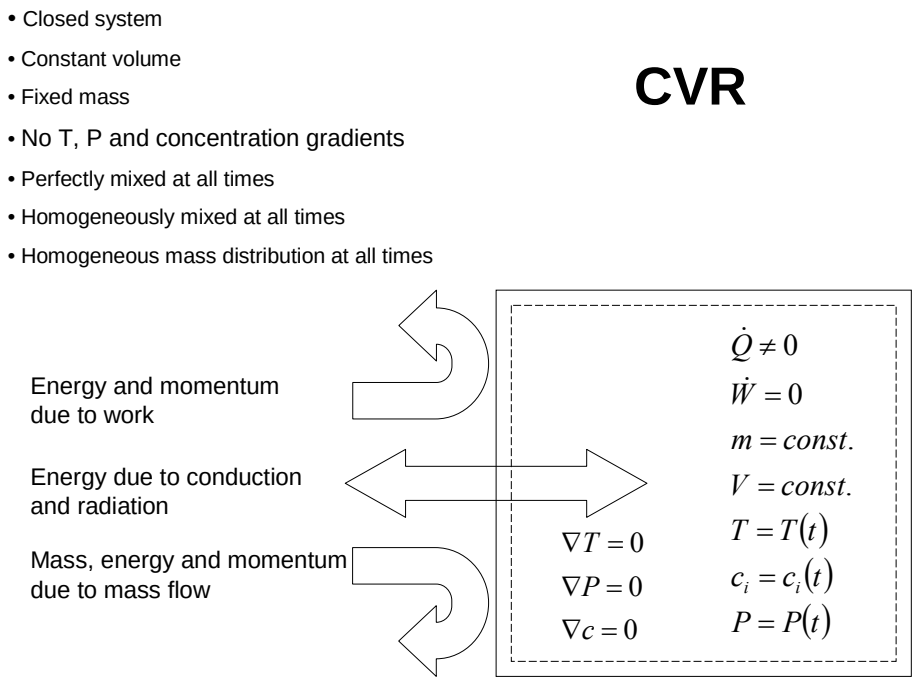


Figure 2.8. The CVR and its characteristics. The system is closed, which means that heat is allowed to be transported through the walls of the system. The volume and mass are constant, the work done on/by the system is zero. The temperature, pressure and species mass fractions change with time. The mixture is perfectly stirred and the mixture is homogeneous throughout the volume at all times.

2.3.4.1.1. System of ODE for the Adiabatic Constant Volume Reactor (ACVR)

The ACVR is a CVR under adiabatic conditions, which means that energy cannot be exchanged with the surroundings. In other word, if the following additional assumptions are made to the assumptions already made for the CVR;

- There is not any heat loss or gain due to thermal radiation
- There is not any heat loss or gain due to thermal conductivity through the reactor walls

the CVR becomes an ACVR. This is illustrated in Figure 2.9.
Hence, the system of ODE for the ACVR is;

$$\rho c_v \frac{dT}{dt} = -\sum_{i=1}^{N_S} \omega_i(\mathbf{x}, T) \cdot U_i \tag{2.11}$$

$$\frac{dx_i}{dt} = \omega_i(\mathbf{x}, T), \; i = 1, ..., N_S \tag{2.12}$$

The source term is described in detail in section 2.3.5.3.
All simulations in this thesis are based on the ACVR. The only exception is the CFD simulations in section 6.2.4, which are based on an ACPR.

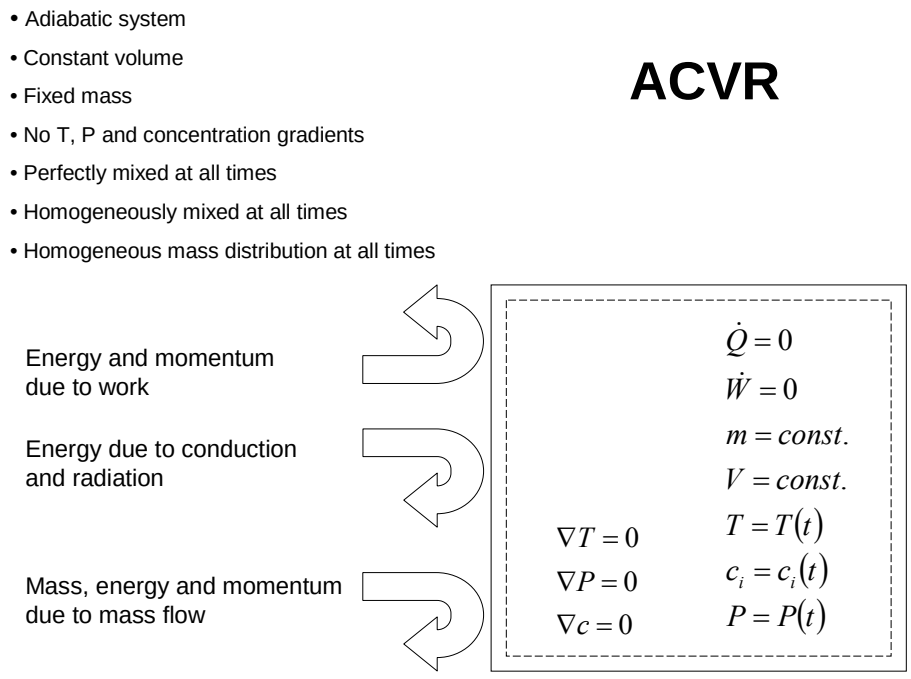


Figure 2.9. An ACVR and its characteristics. The system is isolated, which means that heat is not allowed to be transported through the walls of the system. The volume and mass are constant and the work done on/by the system is zero. The temperature, pressure and species mass fractions change with time. The mixture is perfectly stirred and the mixture is homogeneous throughout the volume at all times.

2.3.4.2. System of ODE for the Constant Pressure Reactor (CPR)

Another example of a 0-D model is the CPR. A CPR is a closed system with a constant pressure, fixed mass and mechanical work. There is not any in or out flow of mass or energy in due to mass flow. However, energy can be exchanged with the surrounding due to heat conduction through the walls and thermal radiation. The species mixture is assumed to follow the ideal gas law and to be perfectly mixed and homogeneous throughout the volume. Hence, temperature, pressure and species concentration gradients do not exist in the CPR. The variables of the system are temperature, pressure and the mass fractions of the species. This is illustrated in Figure 2.10.

The system of ODE is described by one ODE for the temperature and one ODE for each species concentration. All ODEs in the system depend on the temperature and the species concentrations [5]. Hence;

$$\frac{dT}{dt} = f_T\left(\mathbf{x}, T\right) \tag{2.13}$$

$$\frac{dx_i}{dt} = \omega_i\left(\mathbf{x}, T\right) \quad i = 1,..., N_S \tag{2.14}$$

where N_S is the number of species.
However, if the system of ODE is described by species mass fractions instead of concentrations the system of ODE becomes simpler. The system of ODE then becomes [6];

$$\rho c_p \frac{dT}{dt} = \frac{\dot{Q}}{V} - \sum_{i=1}^{N_S} \omega_i\left(\mathbf{x}, T\right) \cdot H_i \tag{2.15}$$

$$\frac{dY_i}{dt} = \frac{\omega_i\left(\mathbf{x}, T\right) W_i}{\rho}, \; i = 1,..., N_S \tag{2.16}$$

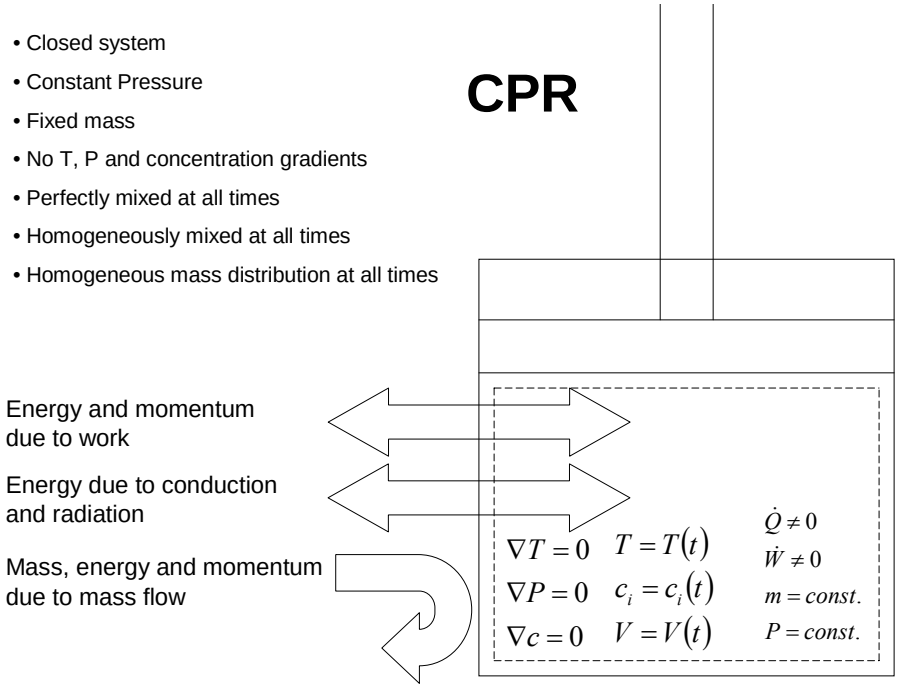


Figure 2.10. The CPR and its characteristics. Work is done on the system and heat is allowed to be transported through the walls of the system. The pressure and mass are constant. The temperature, volume and species mass fractions change with time. The mixture is perfectly stirred and the mixture is homogeneous throughout the volume at all times.

2.3.4.2.1. System of ODE for the Adiabatic Constant Pressure Reactor (ACPR)

The ACPR is a CPR under adiabatic conditions, which means that energy cannot be exchanged with the surroundings. In other words, if the following additional assumptions are made to the assumptions already made for the CPR;

- There is not any heat loss or gain due to thermal radiation
- There is not any heat loss or gain due to thermal conductivity through the reactor walls

The CPR becomes an ACPR. This is illustrated in Figure 2.11.
Hence, the system of ODE for the ACPR is;

$$\rho c_p \frac{dT}{dt} = -\sum_{i=1}^{N_S} \omega_i(\mathbf{x}, T) \cdot H_i \tag{2.17}$$

$$\frac{dY_i}{dt} = \frac{\omega_i(\mathbf{x}, T) W_i}{\rho}, \; i = 1, ..., N_S \tag{2.18}$$

The source term is described in detail in section 2.3.5.3.
The CFD simulations in section 6.2.4 are based on an ACPR.

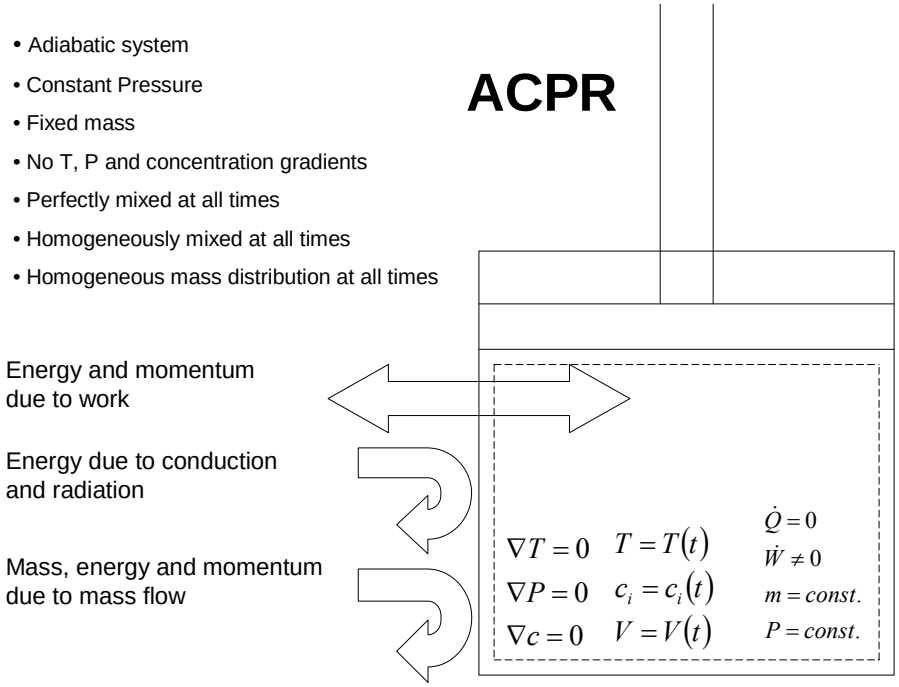


Figure 2.11. An ACPR and its characteristics. Work is done on the system but heat is not allowed to be transported through the walls of the system. The pressure and mass are constant. The temperature, volume and species mass fractions change with time. The mixture is perfectly stirred and the mixture is homogeneous throughout the volume at all times.

2.3.4.3. Coupling between 0-D and 3-D models

3-D models can be built up on 0-D models. For example, in some CFD codes for laminar flow, each grid point in a 3-D space can be made up of 0-D CVR or CPR. This is illustrated in Figure 2.12. Such a system can be solved with an operator splitting method, which is presented below.
It is however usual to employ CPR and CVR even in RANS turbulent combustion simulations. It should be noted that in this approach the chemical source terms are computed without considering turbulent fluctuations. Such a simulation can be found for combustion of N-Heptane in section 6.2.4.

The number of grid points can be of the order of 10^6 in a CFD simulation and each grid point must save the concentration of all species. Hence, a total memory requirement in the order of 10^8 DOUBLE PRECISION is needed when the number of species is in the order of 10^2.
The memory requirement for the Jacobian scales as the number of transported species squared. If the Jacobian is saved for each grid point approximately 10^{10} DOUBLE PRECISION are needed when the number of species is in the order of 10^2. For this reason the Jacobian is not saved at the end of each time step. Instead it is calculated at the beginning of each time step in this thesis.

The CPU time for the CFD part of total CFD/chemistry interaction program is linearly dependent of the number of transported species;

$$CPU_{CFD} = C_{CFD} \cdot N_D \tag{2.19}$$

where, C_{CFD}, is a constant and, N_D, is the number of transported species. The CPU time for the chemistry part of total CFD/chemistry interaction program is dependent of the number of transported species and QSS species in a way that is described in section 3.5.3. The dependence of the memory requirements and CPU time on the number of transported species suggests that reduced chemistry models should be used to save memory and CPU time. Using reduced chemistry models saves;

- memory requirements
- CPU time for the CFD part of total CFD/chemistry interaction program
- CPU time for the chemistry part of total CFD/chemistry interaction program (if the right solver combination is used).

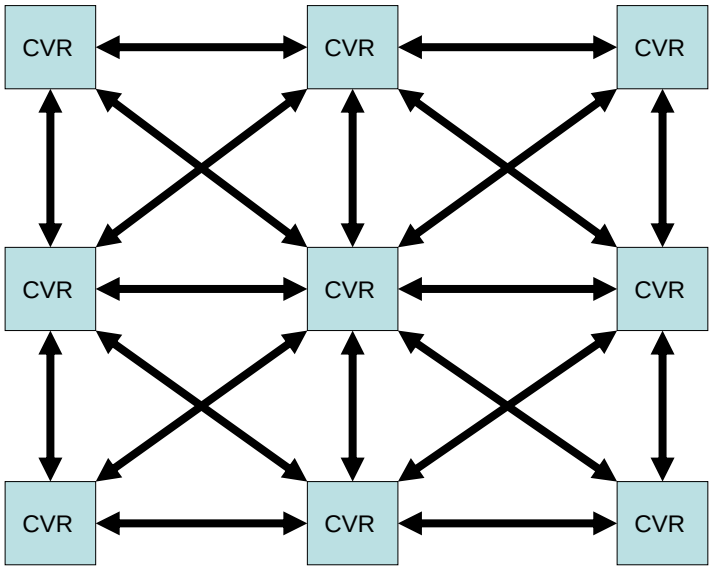


Figure 2.12. 3-D models built up of 0-D models. Each grid point in a 3-D space consists of a CVR or CPR. Species are transported between the grid points by the CFD equations. The CVR or CPR is solved for a certain time period at each grid point.

2.3.4.3.1. Operator splitting method

An operator splitting method separates the calculation of the chemical source term from the calculation of the flow.
The CFD code simulates the flow by solving the balance equations for total mass, species mass, momentum and energy, while the chemical source term is solved by a chemical kinetics program in each CVR or CPR.
The operator splitting method introduces a splitting error, which is proportional to the time step size. In other words, the time step size must be kept small to have a small splitting error.

The balance equations for total mass, species mass, momentum and energy are stated below [1]. It should be noted that the equations below are expressed in a quite compact form and that these equations can be expressed in various ways and in a more detailed manner. The balance equation for the total mass is;

$$\frac{\partial \rho(\mathbf{r},t)}{\partial t} + div\left(\rho(\mathbf{r},t)\mathbf{v}(\mathbf{r},t)\right) = 0 \tag{2.20}$$

where t is the time, ρ is the density, $\mathbf{v}$ is the velocity and $\mathbf{r}$ is the spatial location.

The balance equation for the species mass is;

$$\frac{\partial \rho_i(\mathbf{r},t)}{\partial t} + div(\rho_i(\mathbf{r},t)\mathbf{v}(\mathbf{r},t)) + div(\mathbf{j}_i) = W_i \omega_i(\mathbf{r},t) \tag{2.21}$$

where ρ_i is the species density, $\mathbf{j}_i = \rho_i \cdot v_i$ is the diffusion flux of species i, ω_i is the chemical source term and W_i is the molar mass of species i.
The balance equation for the momentum is;

$$\frac{\partial(\rho(\mathbf{r},t)\mathbf{v}(\mathbf{r},t))}{\partial t} + div(\rho(\mathbf{r},t)\mathbf{v}(\mathbf{r},t) \otimes \mathbf{v}(\mathbf{r},t)) + div(\bar{\bar{p}}) = \rho(\mathbf{r},t)\mathbf{g}(\mathbf{r},t) \tag{2.22}$$

where $\bar{\bar{p}}$ is the pressure tensor and $\mathbf{g}$ is the gravitational acceleration.
The balance equation for the energy in terms of internal energy is;

$$\frac{\partial(\rho(\mathbf{r},t)u(\mathbf{r},t))}{\partial t} + div(\rho(\mathbf{r},t)u(\mathbf{r},t)\mathbf{v}(\mathbf{r},t) + \mathbf{j}_q(\mathbf{r},t)) + \bar{\bar{p}} : grad(\mathbf{v}(\mathbf{r},t)) = q_r(\mathbf{r},t) \tag{2.23}$$

where u is the internal energy, $\mathbf{j_q}$ is the heat flux and q_r is the heat radiation source term.
The balance equation for the energy in terms of enthalpy is;

$$\frac{\partial(\rho(\mathbf{r},t)h(\mathbf{r},t))}{\partial t} - \frac{\partial p}{\partial t} + div(\rho(\mathbf{r},t)h(\mathbf{r},t)\mathbf{v}(\mathbf{r},t) + \mathbf{j}_q(\mathbf{r},t)) + \bar{\bar{p}} : grad(\mathbf{v}(\mathbf{r},t)) - div(p\mathbf{v}(\mathbf{r},t)) = q_r(\mathbf{r},t) \tag{2.24}$$

where h is the enthalpy and p is the pressure.
The energy equation chosen depends on the problem to be solved. A CVR is expressed in terms of internal energy and a CPR is expressed in terms of enthalpy. If the balance equations are expressed in this way, the chemical source term is only involved in the balance equation for the species mass. This simplifies the solution method when a CVR or CPR is used for calculation of the chemical source term.

The equations above are suited for DNS and do not involve any turbulence modeling. The operator splitting method can be used for DNS and CFD models like LES and RANS. The equations for the LES and RANS differ from the equations above. The difference lies in the turbulence modeling.
A RANS simulation based on operator splitting and CPR can be found for combustion of N-Heptane in section 6.2.4.

Figure 2.13 shows a simplified flow chart for DNS, LES and RANS simulations based on an operator splitting method. The flow chart shows the cooperation between the CFD program and the chemical kinetics program. A CVR or CPR is used in each grid point for calculation of the chemical source term.
At a given time step in the flow chart the following happens;

- The CFD program solves the balance equations for mass, energy and momentum based on the old species concentrations.
- The CFD program solves the species transport equations based on the new values from the balance equations for mass (ρ), energy (u or h) and momentum (v).
- The chemical kinetics program calculates the chemical source term from the CVR or CPR at each grid point. This provides a new species vector that can be used in the balance equations for mass, energy and momentum at the next time step.

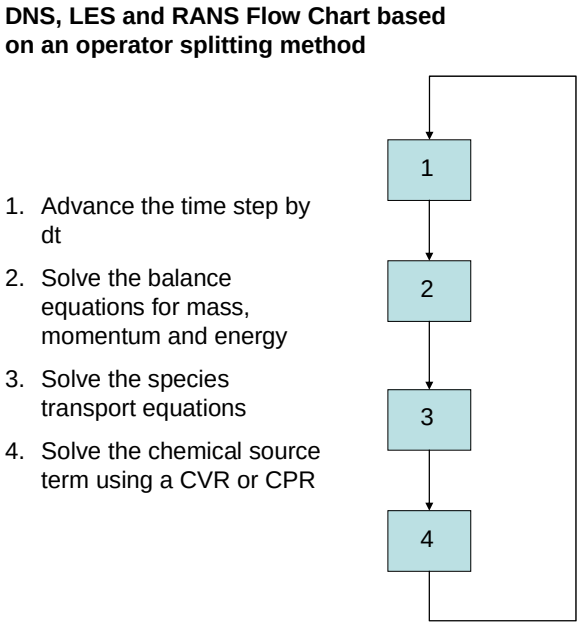


Figure 2.13. Simplified flow chart for DNS, LES and RANS simulations based on an operator splitting method. A CVR or CPR is used in each grid point for calculation of the chemical source term.

2.3.5. Chemical kinetics

If two (or more) molecules/atoms collide energy and momentum is transferred between the molecules. This can result in a chemical reaction causing rearrangement of the chemical bonds. Also, the energy transfer can leave the molecules in an excited state but without a chemical reaction occurring. A third possibility is that kinetic energy and momentum is transferred without excitation of the molecules or chemical reactions occurring. If the discussion is limited to collisions between molecules in a gas phase, the following must be fulfilled for a reaction occur [7].

- The molecules must collide in the right orientation to one another
- The molecules must have enough kinetic energy to overcome the energy barrier.
- The colliding molecules must be able to cause a rearrangement of the bonds in order to form one or more product molecules, which is something that not all molecules can do when they collide

In combustion systems many reactions are exothermic, which means that the energy in terms of enthalpy, ΔH, is released into the system when the products are formed from the reactants.
However, the reactants must have enough energy to overcome the energy barrier E_a in order to produce the products. The energy barrier, which is mostly determined experimentally, exists because of the repulsive forces of the electron clouds. Figure 2.14 shows an illustration of an exothermic reaction and the energy barrier.
For a more detailed description of the bond breaking and bond formation, quantum chemistry must be used, which is outside the scope of this thesis. The rates at which chemical reactions occur can be described by the Arrhenius equation (see section 2.3.5.2.) and the involved species concentrations.

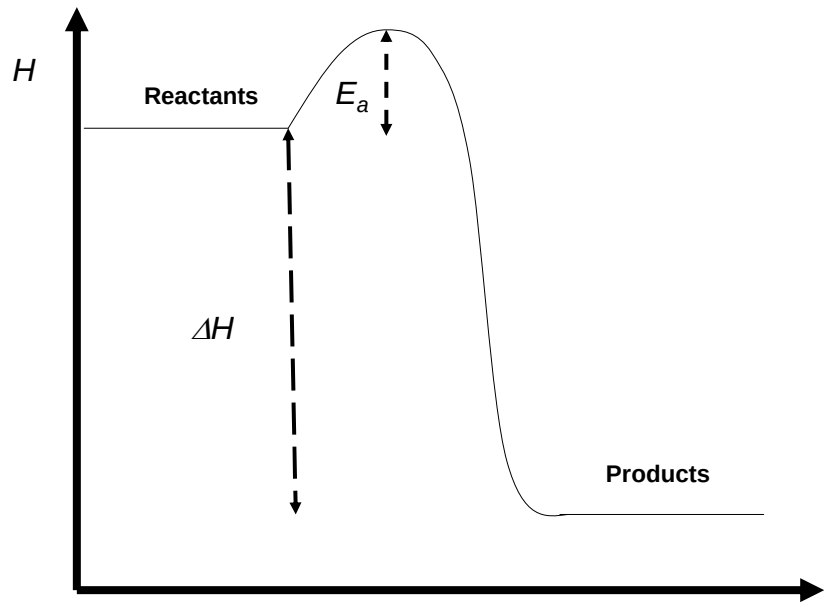


Figure 2.14. Exothermic reaction. The reactants must have enough energy to overcome the energy barrier E_a in order to produce the products. The enthalpy ΔH is released into the system.

2.3.5.1. Reaction rates

If two chemical species collide and have enough energy to overcome a bond breaking energy barrier, the transition into product species is possible from a thermodynamic point of view. The rate at which the transition into product species takes place is an important factor in chemical kinetics models and need to be known for all reactions.
The rate of a reaction depends on the rate constant and the concentration of the reacting species. The rate constant contains information on temperature dependence, possible pressure dependence and other parameters. For further discussion see section 2.3.5.2.

In order to formulate the rate of a reaction the following system of chemical reactions involving molecules A, B, C and D can be considered. If only forward reactions are considered, the system is described by the following reactions;

$$\upsilon'_A A + \upsilon'_B B \xrightarrow{k_{AB}} \upsilon''_C C + \upsilon''_D D \tag{2.25a}$$

$$\upsilon'_C C + \upsilon'_D D \xrightarrow{k_{CD}} \upsilon''_A A + \upsilon''_B B \tag{2.25b}$$

where ν_A, ν_B, ν_C and ν_D are the stoichiometric coefficient for molecule A, B, C and D respectively. The prime and double prime of the stoichiometric coefficients correspond to the reactant and products side of the reaction respectively. The rate constant is, k_{AB}, for reaction (2.25a) and k_{CD}, is the rate constant for reaction (2.25b). The reaction rate, r_{AB}, when ν_A molecules of type, A, and ν_B molecules of type molecule, B, collide is described by;

$$r_{AB} = k_{AB} \cdot [A]^{\upsilon'_A} \cdot [B]^{\upsilon'_B} \tag{2.26}$$

The reaction rate, r_{CD}, when ν_C molecules of type, C, and ν_D molecules of type molecule, D, collide is described by;

$$r_{CD} = k_{CD} \cdot [A]^{\upsilon'_C} \cdot [B]^{\upsilon'_D} \tag{2.27}$$

where, $[A]$, $[B]$, $[C]$ and, $[D]$, are the concentrations of molecule A, B, C and D respectively. Since the species concentrations are treated separately, the rest of the factors that affect a chemical reaction are involved in the rate constants, k_{AB} and k_{CD}, which are described by the Arrhenius equation (see section 2.3.5.2). The time derivatives of species, A, B, C and D are the sum of the reactions (weighted with the stoichiometric coefficient of the species) in which the species is involved in;

$$\frac{d[A]}{dt} = -(\upsilon'_A \cdot r_{AB}) + \upsilon''_A r_{CD} = -\upsilon'_A \cdot k_{AB} \cdot [A]^{\upsilon'_A} \cdot [B]^{\upsilon'_B} + \upsilon''_A k_{CD} \cdot [C]^{\upsilon'_C} \cdot [D]^{\upsilon'_D} \tag{2.28}$$

$$\frac{d[B]}{dt} = -(\upsilon'_B \cdot r_{AB}) + \upsilon''_B r_{CD} = -\upsilon'_B \cdot k_{AB} \cdot [A]^{\upsilon'_A} \cdot [B]^{\upsilon'_B} + \upsilon''_B k_{CD} \cdot [C]^{\upsilon'_C} \cdot [D]^{\upsilon'_D} \tag{2.29}$$

$$\frac{d[C]}{dt} = -(\upsilon'_C \cdot r_{CD}) + \upsilon''_C r_{AB} = -\upsilon'_C \cdot k_{CD} \cdot [C]^{\upsilon'_C} \cdot [D]^{\upsilon'_D} + \upsilon''_C k_{AB} \cdot [A]^{\upsilon'_A} \cdot [B]^{\upsilon'_B} \tag{2.30}$$

$$\frac{d[D]}{dt} = -(\upsilon'_D \cdot r_{CD}) + \upsilon''_D r_{AB} = -\upsilon'_D \cdot k_{CD} \cdot [C]^{\upsilon'_C} \cdot [D]^{\upsilon'_D} + \upsilon''_D k_{AB} \cdot [A]^{\upsilon'_A} \cdot [B]^{\upsilon'_B} \tag{2.31}$$

If a physical system involves many species, which can participate in many reactions that either produce or consume the species, the time derivative of the species is a sum of reaction rates of the reactions the species are involved in. In other words, if a chemical species, *i*, is consumed by the reaction with some species and formed by the reaction of other species, the time derivative of that species can be written as a sum of reaction rates [3];

$$\dot{x}_i(t) = \sum_{k=1}^{N_R} (\upsilon''_{ik} - \upsilon'_{ik}) \cdot r_k(t) = \sum_{k=1}^{N_R} \upsilon_{ik} r_k(t) \tag{2.32}$$

Where υ''_{ik} is the stoichiometric coefficient for the *i:th* species on the product side, υ'_{ik} is the stoichiometric coefficient for the *i*:th species on the reactant side and υ_{ik} is the net stoichiometric coefficient for the *i:th* species in the *k:th* reaction. N_R is the number of reactions. The term, r_k, is written as;

$$r_k(t) = \prod_{l=1}^{N_S} x_l^{\upsilon'_{lk}}(t) \cdot K_k(T(t)) \tag{2.33}$$

υ'_{lk}, is the stoichiometric coefficient for the *l*:th species on the reactant side in the k:th reaction. N_S is the number of species. $K_k(T(t))$ is the reaction constant for the k:th reaction and is the Arrhenius equation described in section 2.3.5.2.

2.3.5.2. The Arrhenius equation

An empirical observation is that many reactions have rate constants that follow the Arrhenius equation [7];

$$\ln k = \ln A - (\frac{E_a}{RT}) \tag{2.34}$$

where k is the rate constant in the reaction rate, E_a is the activation energy, A, is the pre-exponential factor, R is the universal gas constant. T is the temperature of the system. The activation energy is mainly determined from experimental results by plotting *ln k vs 1/T*. However, for those cases that experiments cannot be performed or trusted for some reason, quantum chemical calculations are performed.

The rate constant used in this thesis is described by the Arrhenius equation written in another modified form [7];

$$k = AT^n \exp(-\frac{E_a}{RT}) \tag{2.35}$$

The modification consists of the T^n term where n is a fit parameter. The modified Arrhenius equation is used for the reactions for which the normal Arrhenius equation does not apply.
The pre-exponential factor is found from collision theory [5] and is expressed as;

$$A = PN_{AV}\sigma_{AB}^2\left[\frac{8\pi k_B T}{\mu}\right]^{1/2} \tag{2.36}$$

Where, σ_{AB}, is the cross section for the collision between molecule A and molecule B. N_{AV} is Avogadro's number, P is a steric factor related to orientation of the colliding molecules, k_B is the Boltzmann factor and μ is the reduced mass;

$$\mu = \frac{m_A \cdot m_B}{m_A + m_B} \tag{2.37}$$

Where m_A and m_B are the mass of molecule A and molecule B respectively.

The parts of the Arrhenius equations can be understood as;

- The pre-exponential factor, A, represents the collision efficiency, which is a measure of the rate at which collisions occur irrespective of the kinetic energy of the molecules involved

- The exponential term is the Boltzmann factor, specifying the fraction of collisions that have an energy greater than the activation energy E_a.

The probability for a collision to occur depends on molecular size and average molecular velocities. These factors are considered in the collision frequency, A. The probability that the species have enough energy to overcome the energy barrier, in the form of activation energy, depends on the species velocities, which in turn depends on the temperature of the system. According to the Boltzmann distribution only a part of the molecules have a velocity large enough to break the activation-energy barrier, which motivates the exponential expression in the rate constant. The higher the temperature (velocity) is, the higher the probability is to overcome the energy barrier. This means that the exponential expression can be seen as the fraction of collisions that leads to chemical reactions.

2.3.5.3. A system of chemical equations

This section focuses on the system of chemical equations that describe a physical system which involves many species, which participates in many reactions that either produces or consumes the species. If many species are involved in a system and if we consider homogeneity in space, which means that we only consider time variation in the chemical system, the equations describing the system are;

$$\begin{aligned}\frac{d\mathbf{x}(t)}{dt} &= \boldsymbol{\omega}(\mathbf{x}(t), T(t)) \\ \frac{d(T(t))}{dt} &= f_T(\mathbf{x}(t), T(t))\end{aligned} \tag{2.38}$$

where T is the temperature and f_T is the source term for the temperature, which depends on the type of reactor used to model the system. The f_T for a ACVR and ACPR is found from eq(2.11) and eq(2.17).
The species concentration vector is;

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_{N_S}(t) \end{bmatrix} \tag{2.39}$$

where N_S is the number of species. The source-vector is;

$$\boldsymbol{\omega}(\mathbf{x}(t)) = \begin{bmatrix} \omega_1(\mathbf{x}(t)) \\ \vdots \\ \omega_{N_S}(\mathbf{x}(t)) \end{bmatrix} \tag{2.40}$$

Using the full expression of the source vector [3];

$$\frac{dx_i(t)}{dt} = \omega_i(\mathbf{x}(t), T(t)) = \sum_{k=1}^{N_R} \upsilon_{ik} r_k(\mathbf{x}(t), T(t)) = \sum_{k=1}^{N_R} \upsilon_{ik} \prod_{l=1}^{N_S} x_l^{\upsilon'_{lk}}(t) \cdot K_k(T(t)), \quad i = 1,.., N_S \tag{2.41}$$

If the Arrhenius equation is written out instead of the rate constant;

$$\frac{dx_i(t)}{dt} = \sum_{k=1}^{N_R} \upsilon_{ik} \prod_{l=1}^{N_S} x_l^{\upsilon'_{lk}}(t) \cdot \left[A \cdot T(t)^n \cdot \exp(-\frac{E_a}{RT(t)}) \right]_k, \quad i = 1,.., N_S \tag{2.42}$$

All the variables and constants are described above in section 2.3.5.1and 2.3.5.2. This clearly shows the implicit dependence on time via other species concentrations and temperature.

2.3.5.4. Quasi Steady State Approximation (QSSA)

If the QSSA is used for a species it means that the time-derivative of the species is set to zero and the species concentration can be found as a function of other species [2]. In symbols this becomes;

$$0 = \frac{dx_i}{dt} = \omega_i(\mathbf{x}_D, \mathbf{x}_A) \tag{2.43}$$

Where $\boldsymbol{x_D}$ and $\boldsymbol{x_A}$ is the species concentrations for the system of ODE and NAE respectively. If one or many ODEs are turned into NAE via QSSA, the entire chemical system is described by a system of DAE.
The QSSA is a good approximation for some species but not for others. If the QSSA is a good approximation for a species, it means that the approximation does not affect the behavior of the system of DAE much. The features the species must have in order not to affect the behavior of the system of DAE much is answered below.

To complicate things further, a particular species can be considered as a good QSS species for some points in time and as a bad QSS species for other points in time. However, this thesis focuses only on finding QSS species that do not affect the system of DAE much for the entire simulation time.

If a species does not change its concentration during a certain time period it can be considered to be in a steady state during that time period. Another way to express this is that the time derivative of the species concentration does not change during the time period. Mathematically this is expressed as;

$$\frac{dx_i}{dt} = 0 \tag{2.44}$$

This means that the ODE of the species can be replaced by an NAE.
If the species concentration changes significantly during the time period, that is;

$$\frac{dx_i}{dt} \neq 0 \tag{2.45}$$

the species cannot be considered to be in a QSS. If the ODE of the species is replaced by a NAE, an inaccurate approximation is made. This inaccurate approximation affects the entire system of DAE much and results in an inaccurate solution.

If the species concentration only changes slightly during the time period, that is;

$$\frac{dx_i}{dt} \approx 0 \tag{2.46}$$

the species can be considered to be in a QSS. If the ODE of the species is replaced by a NAE, that is $\frac{dx_i}{dt} = 0$, an approximation which affects the entire system of DAE is made. How much the approximation affects the system of DAE is very species dependent.

The three cases are illustrated in Figure 2.15., which shows three different concentration profiles during the time period t_1 to t_2. The highest concentration profile has $\frac{dx_i}{dt} \neq 0$ during the time period and cannot be considered to be in steady state or QSS. The middle concentration profile has an early fast decay and then has $\frac{dx_i}{dt} \approx 0$ during the major part of the short time period. This species can therefore be considered to be in QSS during the time period. The species in QSS are interesting from a reduction point of view. The lowest concentration profile has $\frac{dx_i}{dt} = 0$ during the time period and can be considered to be in steady state.

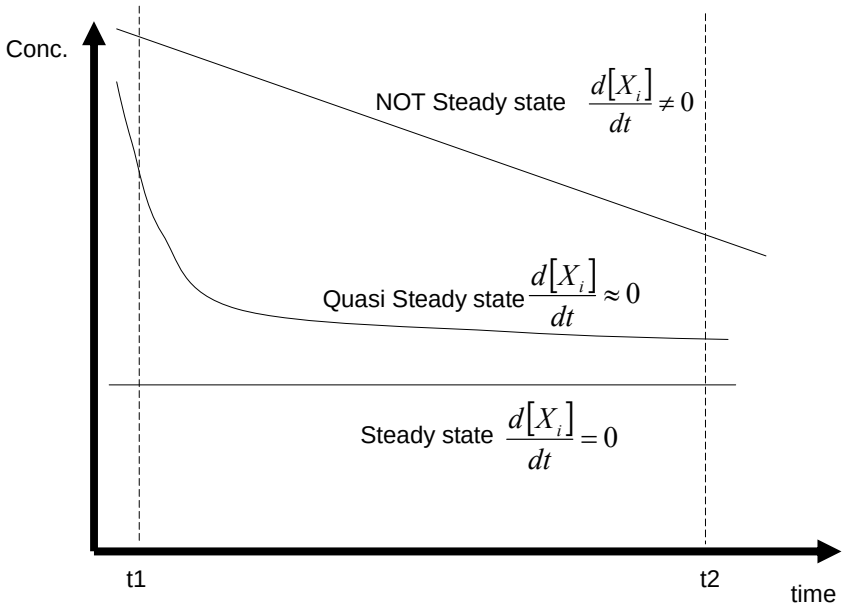


Figure 2.15. Different concentration profiles vs time. The lowest profile shows a truly steady state situation. The middle shows a profile that can be considered to be in QSS in the short time period (t_2-t_1) and the highest shows a profile that cannot be considered in QSS.

It should be noted that QSS species concentrations vary over time, even though the time derivative is zero. The reason for this is that the QSS species concentrations are functions of $\boldsymbol{x_D}$ and $\boldsymbol{x_A}$. When $\boldsymbol{x_D}$ and $\boldsymbol{x_A}$ in the source term of a QSS species change, the QSS species concentration changes as well. This is illustrated in Figure 2.16.

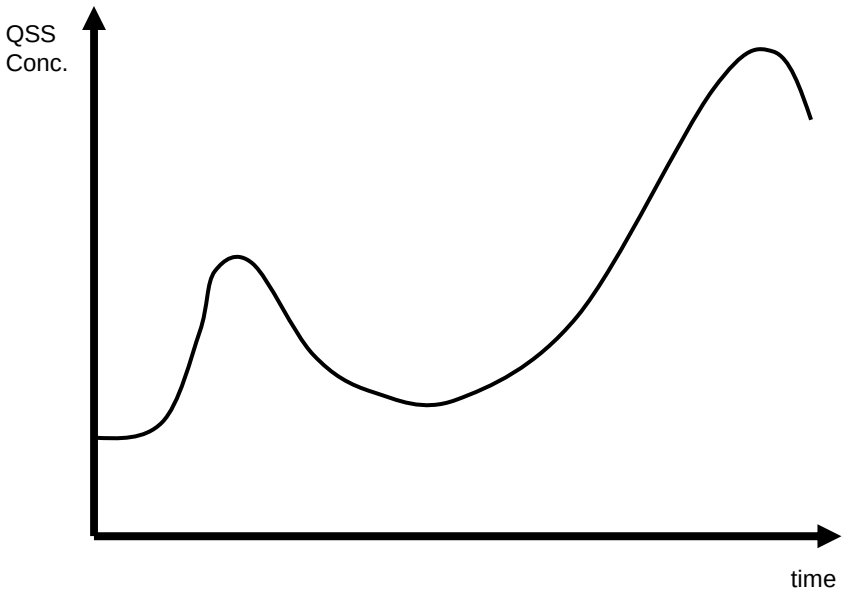


Figure 2.16. A schematic illustration of the concentration of a QSS species vs time. The concentration of a QSS species is a function of $\boldsymbol{x_D}$ and $\boldsymbol{x_A}$. When concentrations of species in the ODE system move in time the QSS concentration move in time as well.

Species with short Life Times (LT) are usually thought of as good QSS species [2,3]. The reason for this is that they quickly reach a time derivative that is close to zero for the time interval of interest. However, some important species, like radicals, that do have short LT are not good QSS species. The reason for this is that the system of DAE is very sensitive to the concentration of these species. Application of the QSSA to them approximates the system too much. For this reason a sensitivity measure [1,3] is used to classify species. The sensitivity is described in section 5.4.2.1.2.
Species with short LT and low sensitivity to the important variable, like temperature or OH concentration, are therefore "appropriate" QSS species, while species with long LT and high sensitivity are "non appropriate" QSS species. Most species are in the grey area between appropriate and non appropriate and can be considered as intermediate QSS species, that is, species with short LT and high sensitivity or vise versa. These intermediate QSS species are harder to classify. This is summarized in table 2.2.
Hence, indicators like LT and sensitivity analysis are used to classify the QSS species. These indicators and other indicator for QSSA are discussed further in section 5.4.2.1.

Table 2.2. The table shows the combinations of LT and sensitivity that gives appropriate, non appropriate or intermediated QSS species.

	Short LT	**Long LT**
Low sensitivity	Appropriate QSS species	Intermediate QSS species
High sensitivity	Intermediate QSS species	Non appropriate QSS species

2.3.5.4.1 QSSA applied to a system of ODE

A system of ODE for a chemical system without any QSSA is described by eq(2.38). Another form of this equation is;

$$\begin{aligned}
\frac{d\left(x_{D,1}(t)\right)}{dt} &= \omega_{D,1}\left(\mathbf{x}_D(t), T(t)\right) \\
&\vdots \\
\frac{d\left(x_{D,N_S}(t)\right)}{dt} &= \omega_{D,N_S}\left(\mathbf{x}_D(t), T(t)\right) \\
\frac{d\left(T(t)\right)}{dt} &= f_T\left(\mathbf{x}_D(t), T(t)\right)
\end{aligned} \tag{2.47}$$

where the species vector is;

$$\mathbf{x}_D(t) = \begin{bmatrix} x_{D,1}(t) \\ \vdots \\ x_{D,N_S}(t) \end{bmatrix} \tag{2.48}$$

The subscript *D* denoted that the species belong to a system of ODE.
Every time a species is put into QSS, it means an approximation to the original system and thereby results in a less accurate description of the physical reality than before. The approximation means that the original ODE for a species is turned into an algebraic equation by setting the time-derivative equal to zero, while the concentration of the species still can be found from a function of the other species.
If the QSSA is used for many species, a system of NAE is formed and the original system of ODE is now a coupled system of DAE. The non QSS species are found from ODE's and are functions of both non QSS species and QSS species, while the QSS species are found from algebraic equations and are also functions of both non QSS species and QSS species. This is the reason for the coupling between the system of ODE's and the system of NAE. The chemical system will be described by, after some species are put into QSSA, the following set of DAE;

$$\begin{aligned}
\frac{d\left(x_{D,1}(t)\right)}{dt} &= \omega_{D,1}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) \\
&\vdots \\
\frac{d\left(x_{D,b}(t)\right)}{dt} &= \omega_{D,b}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) \\
0 &= \omega_{A,b+1}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) \\
&\vdots \\
0 &= \omega_{A,N_S}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) \\
\frac{d\left(T(t)\right)}{dt} &= f_T\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)
\end{aligned} \tag{2.49}$$

Where the non QSS species vector is;

$$\mathbf{x}_D(t) = \begin{bmatrix} x_{D,1}(t) \\ \vdots \\ x_{D,b}(t) \end{bmatrix} \tag{2.50}$$

And the QSS species vector is;

$$\mathbf{x}_A(t) = \begin{bmatrix} x_{A,b+1}(t) \\ \vdots \\ x_{A,N_S}(t) \end{bmatrix} \tag{2.51}$$

If the QSSA is applied to the system of ODE's from the ACVR in section 2.3.4.1.1 the resulting set of DAE's is;

$$\begin{aligned}
&\frac{d\left(T(t)\right)}{dt} = f_T\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) = \\
&= \frac{\left(\sum_{i=1}^{N_D} U_i \omega_i\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) + \sum_{i=1}^{N_A} U_i\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right) \cdot \omega_i\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)\right)}{\rho c_v} = \\
&= \frac{\sum_{i=1}^{N_D} U_i \omega_i\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)}{\rho c_v}
\end{aligned} \tag{2.52}$$

The last equality sign holds since the source term for the QSS species equals zero. This means that an error is introduced in the ODE for temperature, since the ODE only has contributions from the non QSS species. Hence, the QSS species must be chosen in a way that they do not affect the ODE for temperature too much. Species with large source terms and large internal energy will affect the energy equation more if they are set in QSS and vise versa.

When the system of ODE becomes a system of DAE, the product sign in eq(2.41) is replaced by two product signs, one for the species in the system of ODE and one for the species in the system of NAE.
Hence, the system of ODE is;

$$\frac{d(x_i(t))}{dt} = \omega_i(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)) = \sum_{k=1}^{N_R} \upsilon_{ik} r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)) = \\ = \sum_{k=1}^{N_R} \upsilon_{ik} \prod_{l=1}^{N_D} x_l^{\upsilon'_{lk}}(t) \cdot \prod_{m=1}^{N_A} x_m^{\upsilon'_{mk}}(t) \cdot K_k(T(t)), \quad i = 1, b \tag{2.53}$$

and the system of NAE is;

$$0 = \omega_i(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)) = \sum_{k=1}^{N_R} \upsilon_{ik} r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)) = \\ = \sum_{k=1}^{N_R} \upsilon_{ik} \prod_{l=1}^{N_D} x_{D,l}^{\upsilon'_{lk}}(t) \cdot \prod_{m=1}^{N_A} x_{A,m}^{\upsilon'_{mk}}(t) \cdot K_k(T(t)), \quad i = b+1, N_S \tag{2.54}$$

All the other variables and constants are described in section 2.3.5.1and 2.3.5.2. Equation (2.53), which is for the ODE, and (2.54), which is for the NAE, are the same with the exception that the time derivative is zero for equation (2.54).
The initial conditions;

$$\mathbf{x}_D(t=0) = \mathbf{x}_D^0 \\ \mathbf{x}_A(t=0) = \mathbf{x}_A^0 \\ T(t=0) = T_0 \tag{2.55}$$

are determined by the input-vector in the input/output model. The numerical methods suitable for this problem are presented in section 3.5 and the structure of the solver combination for this problem is presented in section 4.4. The system of ODE and the system of NAE are solved separately, but they are connected via the species concentration and temperature. The NQSS species and temperature will act as constants in the iteration of the system of NAE, while the QSS species will act as constants in the iteration for the system of ODE.

2.3.5.4.2. Errors due to the QSSA

If a species is set into QSS and its concentration is calculated from a NAE, there will be a difference compared to if the same species was calculated from an ODE. This difference vector of the QSS species, $\Delta\mathbf{x}_A$, is called the instantaneous error [2] of the QSS species and is calculated from;

$$\Delta\mathbf{x}_A = \mathbf{x}_A^{ODE} - \mathbf{x}_A^{NAE} \tag{2.56}$$

where $\mathbf{x}_A^{ODE}$ and $\mathbf{x}_A^{NAE}$ is the species vector of the QSS species when calculated from the system of ODE and NAE respectively. The time derivative of $\Delta\mathbf{x}_D$, which is the error in the concentration vector of the ODE species, and $\Delta\mathbf{x}_A$ is;

$$\begin{aligned}\frac{d\Delta\mathbf{x}_D}{dt} &= J^{DD}\Delta\mathbf{x}_D + J^{DA}\Delta\mathbf{x}_A \\ \frac{d\Delta\mathbf{x}_A}{dt} &= J^{AD}\Delta\mathbf{x}_D + J^{AA}\Delta\mathbf{x}_A\end{aligned} \tag{2.57}$$

if the partitioned Jacobian;

$$J = \begin{pmatrix} J^{DD} & J^{DA} \\ J^{AD} & J^{AA} \end{pmatrix} \tag{2.58}$$

is used. Assuming that the error in the species from the system of ODE is negligible;

$$\frac{d\Delta\mathbf{x}_A}{dt} = J^{AA}\Delta\mathbf{x}_A \tag{2.59}$$

Using the definition of the QSSA;

$$\frac{d\Delta\mathbf{x}_A}{dt} = \frac{d\mathbf{x}_A^{ODE}}{dt} - \frac{d\mathbf{x}_A^{NAE}}{dt} = \frac{d\mathbf{x}_A^{ODE}}{dt} \tag{2.60}$$

results in;

$$\frac{d\mathbf{x}_A}{dt} = J^{AA}\Delta\mathbf{x}_A \tag{2.61}$$

If the off diagonal terms are neglected eq(2.59) becomes;

$$\frac{d\mathrm{x}_{A,i}}{dt} = J_{\mathrm{ii}}^{AA}\Delta\mathrm{x}_{A,i} \tag{2.62}$$

Rearrangement gives;

$$\Delta x_{A,i} = \frac{1}{J_{ii}^{AA}} \frac{dx_{A,i}}{dt} \tag{2.63}$$

where

$$\tau_i = -1/\mathbf{J}_{ii} \tag{2.64}$$

is called the life time (LT) of species *i* and is found from the inverse of the diagonal element of the Jacobian. This is discussed further in section 5.4.2.1.1.1. Hence, the LT of a species is an indicator of the error the QSSA introduces in the species concentration.

2.3.5.5. The Ignition Process

This thesis focuses on the ignition process only, which excludes flames. Ignition simulations differ from steady state flame simulations. One major difference is that steady state flame simulations have a time independent solution, while the ignition simulations have time dependent solutions [1]. Also, the chain reactions, which are discussed in section 2.3.5.5.1, involved differ. Chain initiating and chain termination reactions are very important for the ignition process, but not that important for flames. Chain propagating and chain branching reactions are important for both ignition and flames. The different chain reactions are described below.
Examples of ignition processes are spark induced ignition, which occurs in gasoline engines induced by a spark and auto ignition, which occurs in Diesel engines.

In an ignition process, fuel reacts with O and O_2 to produce many different intermediate species, which in turn react via different reaction paths to give the end products. During this process there is a net release of heat due to the rearrangement of the chemical bonds. The heat release can be very large and very rapid at certain conditions and an ignition process can take place.

To accurately describe the complex behavior of the ignition process it is important that all significant species and reactions contributing to the ignition process are considered. However, some sub processes and species contribute more than others to the ignition process. An ignition process can have two main causes [1];

- Thermal ignition, which is due to the rapid increase in reaction rates with temperature.

- Chain branching ignition, which is due to the exponential growth of free radicals. Chain branching ignition occurs under circumstances when the chain branching reactions rates are fast and chain terminating reactions rates are slow.

The occurrence of IDT is typical [1] when chain branching reactions are involved, in contrast to thermal ignition processes where the temperature increases at once. The number of radicals increases at an exponential rate during the IDT. However, the amount of fuel consumed and thereby the amount of energy released into the system is almost undetectable. Hence, chain branching and chain propagating reactions occurs during the IDT, while the temperature is almost constant.
At some moment in time the number of radicals becomes large enough to consume a significant part of the fuel, which leads to a rapid ignition process. Even though the occurrence of the IDT is due chain reactions the IDT is strongly dependent on initial temperature. This is due to the chemical reactions dependence on the temperature, which is reflected in the exponential dependence seen in the Arrhenius expression.

The complex behavior of radicals and heat release gives rise to the explosion limits [1] for different temperatures and pressures. The second explosion limit is due to the competition of chain branching and chain termination reaction. The third explosion limit is due to the competition between heat production and heat loss.

2.3.5.5.1. Chain reactions

Combustion chemistry involves chain reactions, which lead from the fuel and oxygen to the end products via free radical intermediates. The free radicals are very reactive and are characterized by unpaired electrons. Chain reactions involve [1];

- Chain initiation reactions
- Chain propagation reactions
- Chain branching reactions
- Chain termination reactions

The most important reactions for the hydrogen-oxygen system are presented below [1];

Chain initiation reactions
Radicals, which are symbolized with black dots in this section, are formed in a chain initiation reaction. An example of a chain initiation reaction is;

$$H_2 + O_2 \rightarrow 2OH\bullet$$

Reactive species (radicals) are formed from stable species.

Chain propagation reactions
The ratio between the numbers of free radicals in the product to that in the reactant is equal to one for chain propagation reactions. An example of a chain propagating reaction is;

$$OH\bullet + H_2 \rightarrow H_2O + H\bullet$$

Reactive intermediate species react with stable species to form another reactive species.

Chain branching reactions
The ratio between the numbers of free radicals in the product to that in the reactant is greater than one for chain branching reactions. Examples of chain branching reactions are;

$$H\bullet + O_2 \rightarrow OH\bullet + O\bullet$$
$$O\bullet + H_2 \rightarrow OH\bullet + H\bullet$$

Reactive species react with stable species to form two reactive species.

Chain termination reactions
The radicals are destroyed in chain terminating reactions.
Examples of a chain terminating reactions are;

$$H\bullet + O_2 + M \rightarrow HO_2 + M$$
$$H\bullet + OH\bullet + M \rightarrow H_2O + M$$

where *M* is a third body. Reactive species form stable species.

There is a competition between the different types of chain reactions. At some conditions the chain branching reactions are fast, while the chain terminating reactions are slow, and vise versa. The free radicals increase exponentially under the conditions when the chain branching reactions are fast and the chain termination reactions are slow.
The reactions involving a third body, *M*, like the termination reaction in the example above, is pressure dependent. This means that the termination reaction in the example and other third body reactions are more likely at high pressures. It also means that exponential growth of free radicals due to chain branching reactions is more likely at some pressures than others. The chain branching is contributing to the ignition process.

2.3.5.5.2. Important molecules for the mechanisms in this thesis

This thesis simulates the ignition process of an N-Heptane mechanism and a Methane/Propane mechanism. Methane, Propane and N-Heptane are all paraffins/alkanes, which are saturated hydrocarbons with the general molecular formula C_nH_{2n+2}.
The paraffins can be straight chained or branched chained. Methane is the simplest paraffin with one carbon atom. Hence, the molecular formula CH_4. Propane is the has three carbon atoms. Hence, the molecular formula C_3H_8. N-Heptane is a straight chained hydrocarbon with seven carbon atoms. Hence, the molecular formula C_7H_{16}. N-Heptane is primarily used as a reference for octane number zero in combustion processes.

A simplified four step mechanism for oxidation of paraffins is described in section 5.3.1. However, the oxidation of Methane, Propane, N-Heptane and other paraffins is complicated and involves many intermediated steps and temporary species. Hence, to fully understand the combustion process of Methane, Propane and N-Heptane a detailed mechanism must be used.
A reference of the detailed Methane/Propane mechanism used in this thesis can be found in [8]. For a detailed description of the N-Heptane mechanism used in this thesis, see [4].

When the fuel molecules are subjected to oxygen the fuel molecules breaks down into oxygenated intermediates. Theses intermediates contain a smaller number of carbon atoms than the original fuel molecule. There are many reaction pathways for the intermediate species. Under ideal combustion conditions all reaction pathways sooner or later produce CO_2 and H_20.

All molecules are important in one way or the other in a combustion process. However, some species are considered more important than others depending on the application. Hence, the chemical species of interest for the ignition process in Chapter 6 in this thesis are;

- **Fuel**, is the starting point for the whole combustion process. The size of the fuel molecule determines the number of intermediates and reaction pathways.
  - **CH_4** (Methane), is the fuel in the Methane/Propane mechanism. Mechanisms of higher hydrocarbons usually contain the Methane mechanism.
  - **C_3H_8** (Propane), is the fuel in the Methane/Propane mechanism.
  - **C_7H_{16}** (N-Heptane), the fuel in the N-Heptane mechanism
- **O_2**, is the oxidizer that oxidizes the fuel and the fuel derivatives. Without the oxidizer the combustion process would die out or not start at all.
- **CO_2**, one of the final products. This molecule is very infamous for the greenhouse effect and global warming.
- **H_2O**, one of the final products.

- **CO**, a very toxic molecule which is unwanted in combustion. CO is produced especially in fuel rich conditions. Almost all the heat release in the combustion process comes from the oxidation of CO to CO_2.
- **CH_2O**, (Formaldehyde) is formed as an intermediate species in the preheating phase of hydrocarbon combustion. It is consumed in the reaction zone and is therefore well suited as an indicator of CF regions.
- **OH**, is an important radical. It is used as an indicator of the ignition point in this thesis, since it practically does not exist before the ignition point but reaches a maximum value at the ignition point. The radical is involved in chain branching and chain propagating reactions and is therefore very important for the IDT in ignition simulations.
- **HO_2**, is an important radical. It is used as an indicator for the CF in this thesis, since it has a peak at the CF. The radical is involved in chain terminating reactions and is therefore very important for the IDT in ignition simulations.
- **C_2H_2**, (Acetylene) is important for soot formation. Soot formation is not considered in the models this thesis is based on, but the models can be expanded to contain soot formation.
- **C_2H_4**, (Ethylene**)** is an important intermediate that is oxidized indirectly into CO and H_2.
- **H_2,** is an important intermediate which is oxidized into water with a high energy release. Hence, H_2 indicates the temperature increase.

2.4. Chapter References

[1] J. Warnatz, U. Maas, R.W. Dibble, *Combustion,* 4th edition Springer (2006)
[2] T. Turanyi, A.S. Tomlin, M.J. Pilling, *On the error of the Quasi-Steady-State Approximation J. Phys. Chem*. 1993, 97, 163-172
[3] T. Lövås, Automatic *Reduction procedures for Chemical mechanisms in Reactive Systems*, (2002), Doctoral Thesis, Combustion Physics, Lund University, Sweden, Lund reports on Physics, LRCP-78, ISRN LUTFD2/TFCP—78--SE
[4] S.S. Ahmed *A Detailed Modeling Study for Primary Reference Fuels and Fuel mixtures and Their use in Engineering applications* (2006), Doctoral Thesis, Combustion Physics, Lund University, Sweden, Lund reports on Physics, LRCP-115 ISSN 1102-8718 ISRN LUTFD2/TFCP—115—SE ISBN91-628-7013-0
[5] S. R. Turns, *An Introduction to Combustion Concepts and Applications* second edition, McGRAW-HILL INTERNATIONAL EDITIONS 2000
[6] R. Bellanca, *BlueBellMouse A tool for kinetic model development*, Doctoral Thesis, Combustion Physics, Lund University, Sweden, (2004), Lund reports on Physics, LRCP-92 ISRN LUTFD2/TFCP—92—SE ISSN 1102-8718
[7] P.W. Atkins, *Physical Chemistry* Fifth edition, Oxford University Press (1994)
[8] E.L. Petersen, D.M. Kalitan, S. Simmons, G. Bourque, H.J. Curran, and J.M. Simmie, (2006). Proc. Comb. Inst., vol. 31, Elsevier.
[9] N. Peters, *Turbulent Combustion*, Cambridge University Press (2000)

2.5. Chapter Appendix

A.2.1. Constant Volume Reactor (CVR)

This section shows the derivation of the equations for the CVR. Energy conservation for a CVR gives;

$$\frac{dU}{dt} = m\frac{du}{dt} = \dot{Q} \tag{A.2.1}$$

U is the internal energy of the system, m is the total mass of the system and $\dot{Q}$ is the instantaneous rate of heat transferred into the system. The specific internal energy is defined as;

$$u = U / m \tag{A.2.2}$$

The specific internal energy can be expressed as;

$$u = \sum_{i=1}^{N_S} Y_i \cdot u_i \tag{A.2.3}$$

where u_i is the specific internal energy of species i and the mass fraction of species i is;

$$Y_i = \frac{m_i}{\sum_{j=1}^{N_S} m_j} \tag{A.2.4}$$

where m_i is the mass of species i. Using eq.(A.2.3) the time derivate of the specific internal energy can be expressed is;

$$\begin{aligned} \frac{du}{dt} &= \sum_{i=1}^{N_S} \frac{dY_i}{dt} \cdot u_i + \sum_{i=1}^{N_S} \frac{du_i}{dt} Y_i = \sum_{i=1}^{N_S} \frac{dY_i}{dt} \cdot u_i + \sum_{i=1}^{N_S} \frac{\partial u_i}{\partial T} \frac{dT}{dt} Y_i = \\ &= \sum_{i=1}^{N_S} \frac{dY_i}{dt} \cdot u_i + \sum_{i=1}^{N_S} c_{v,i} \frac{dT}{dt} Y_i = \sum_{i=1}^{N_S} \frac{dY_i}{dt} \cdot u_i + c_v \frac{dT}{dt} \end{aligned} \tag{A.2.5}$$

where

$$\frac{du_i(T)}{dt} = \frac{\partial u_i(T)}{\partial T} \frac{dT}{dt} = c_{v,i} \frac{dT}{dt} \tag{A.2.6}$$

was used at the third equality sign. $c_{v,i}$ is constant volume heat capacity for species i.

Inserting eq.(A.2.4) into eq.(A.2.1) gives;

$$c_v \frac{dT}{dt} = \frac{\dot{Q}}{m} - \sum_{i=1}^{N_S} \frac{dY_i}{dt} \cdot u_i \tag{A.2.7}$$

The concentration of species *i* can be expressed as;

$$c_i = \frac{\rho}{W_i} Y_i \tag{A.2.8}$$

The time derivative of eq.(A.2.7) is (where ρ is constant since the mass and volume is constant);

$$\frac{dc_i}{dt} = \frac{\rho}{W_i} \frac{dY_i}{dt} = \omega_i \tag{A.2.9}$$

Inserting eq.(A.2.8) into eq.(A.2.6) gives;

$$c_v \frac{dT}{dt} = \frac{\dot{Q}}{m} - \sum_{i=1}^{N_S} \frac{\omega_i W_i}{\rho} \cdot u_i \tag{A.2.10}$$

The internal energy per mole for species *i* is;

$$U_i = W_i u_i \tag{A.2.11}$$

Inserting eq. (A.2.10) into eq.(A.2.9) finally gives;

$$\rho c_v \frac{dT}{dt} = \frac{\dot{Q}}{m} - \sum_{i=1}^{N_S} \omega_i U_i \tag{A.2.12}$$

A.2.2. Constant Pressure Reactor (CPR)

This section shows the derivation of the equations for the CPR. For a fixed mass system the energy equation gives;

$$\frac{dU}{dt} = m\frac{du}{dt} = \dot{Q} - \dot{W} \tag{A.2.13}$$

U is the internal energy of the system, m is the total mass of the system, $\dot{Q}$ is the instantaneous rate of heat transferred into the system and $\dot{W}$ is instantaneous rate of work done by the system. The specific internal energy is defined as;

$$u = U / m \tag{A.2.14}$$

The specific enthalpy is defined as;

$$h = H / m \tag{A.2.15}$$

The specific enthalpy relates to the specific internal energy by;

$$h \equiv u + Pv \tag{A.2.16}$$

The specific volume is defined as;

$$v = V / m \tag{A.2.17}$$

The work per mass is;

$$\frac{\dot{W}}{m} = P\frac{dv}{dt} \tag{A.2.18}$$

This results in;

$$\frac{\dot{Q}}{m} = \frac{dh}{dt} \tag{A.2.19}$$

The specific enthalpy can be expressed as;

$$h = \sum_{i=1}^{N_S} Y_i \cdot h_i \tag{A.2.20}$$

Where Y_i is the mass fraction of species i and h_i is the specific enthalpy of species i. Using eq.(A.2.20) the time derivate of the specific enthalpy can be expressed is;

$$\frac{dh}{dt}=\sum_{i=1}^{N_S}\frac{dY_i}{dt}\cdot h_i+\sum_{i=1}^{N_S}\frac{dh_i}{dt}Y_i=\sum_{i=1}^{N_S}\frac{dY_i}{dt}\cdot h_i+\sum_{i=1}^{N_S}\frac{\partial h_i}{\partial T}\frac{dT}{dt}Y_i=$$
$$=\sum_{i=1}^{N_S}\frac{dY_i}{dt}\cdot h_i+\sum_{i=1}^{N_S}c_{p,i}\frac{dT}{dt}Y_i=\sum_{i=1}^{N_S}\frac{dY_i}{dt}\cdot h_i+c_v\frac{dT}{dt} \qquad \text{(A.2.21)}$$

where

$$\frac{dh_i}{dt}=\frac{\partial h_i}{\partial T}\frac{dT}{dt}=c_{p,i}\frac{dT}{dt} \qquad \text{(A.2.22)}$$

was used at the third equality sign. $c_{p,i}$ is constant pressure heat capacity for species i. Inserting eq.(A.2.21) into eq.(A.2.19) gives;

$$c_p\frac{dT}{dt}=\frac{\dot{Q}}{m}-\sum_{i=1}^{N_S}\frac{dY_i}{dt}\cdot h_i \qquad \text{(A.2.23)}$$

The time derivative of the species mass fractions is;

$$\frac{dY_i}{dt}=\frac{\omega_i W_i}{\rho},\ i=1,...,N_S \qquad \text{(A.2.24)}$$

$$\frac{dN_i}{dt}\equiv V\omega_i \qquad \text{(A.2.25)}$$

The species mass fraction is;

$$Y_i=\frac{m_i}{\sum_{i=1}^{N_S}m_i} \qquad \text{(A.2.26)}$$

Where

$$m_i=N_i W_i \qquad \text{(A.2.27)}$$

Using eq(A.2.24) in eq(A.2.23) gives;

$$c_p\frac{dT}{dt}=\frac{\dot{Q}}{m}-\sum_{i=1}^{N_S}\frac{\omega_i W_i}{\rho}\cdot h_i \qquad \text{(A.2.28)}$$

Using;

$$H_i = W_i h_i \tag{A.2.29}$$

In eq(A.2.28) and rearrangement gives;

$$\rho c_p \frac{dT}{dt} = \frac{\dot{Q}}{V} - \sum_{i=1}^{N_S} \omega_i H_i \tag{A.2.30}$$

Hence, the system of ODE for the CPR is;

$$\rho c_p \frac{dT}{dt} = \frac{\dot{Q}}{V} - \sum_{i=1}^{N_S} \omega_i H_i \tag{A.2.31}$$

$$\frac{dY_i}{dt} = \frac{\omega_i W_i}{\rho}, \; i = 1,..., N_S \tag{A.2.32}$$

Chapter 3.

Numerical methods

3.1. Chapter introduction

There are many numerical methods to choose from when solving systems of ODE, NAE and DAE. The optimal numerical method for a given problem should be both fast and accurate. However, there is not a general answer to what the optimum numerical method is. The optimum numerical method depends on the character of the problem.
It should be noted that the optimized combinations of numerical methods in this thesis have been found from a combination of empirical knowledge and theoretical reasoning. For example, the optimized solver settings for the ODE system and NAE system solvers have been set from empirical knowledge, while the optimization of the NAE system solver was based on a global optimization method, which is discussed in section 4.3.3.3.

The objective of this thesis, as stated in Chapter 1, is to solve a system of stiff DAE as fast as possible with the accuracy of the solution maintained as the number of NAE increases. The strategy is to combine two numerical methods, one for the system of ODE and one for the system of NAE. The intention is to find the optimal combination of numerical methods for the system of ODE and the system of NAE, which is both faster and more accurate than other combinations. An increase of speed of the total combination of numerical methods can actually be achieved by the use of a slower but more accurate numerical method for the NAE. This is further elaborated on in section 3.3.4.1 and 3.5.

3.2. Integration of Ordinary Differential Equations (ODE)

In order to choose an appropriate numerical method for the system of ODE to be solved, the entire problem must be characterized. Problems involving ODE are not only specified by the system of equations. The initial and boundary conditions of the problem are also very important in determining how to attack the problem numerically. Problems with spatially resolved boundary conditions are outside the scope of this thesis. This thesis is based on an initial value problem for a system of ODE, which suggests the following well known numerical methods that might be able to solve the problem [1];

- Explicit Euler method
- Implicit Euler method
- RK method
- Newton based Predictor Corrector method

Some of the methods are not well suited for stiff nonlinear problems, like combustion problems. The advantages and disadvantages for each method are presented below.

3.2.1. Explicit and implicit methods

The numerical method for solving a system of ODE can be divided into;

- Explicit methods
- Implicit methods

Explicit methods calculate the new value explicitly in terms of the old value, which gives them the advantage that they are often simple to implement and that each time step demands low computational effort. However, these methods are unstable for certain systems if the time step is too large, which means that the time step size may have to be much smaller than what the accuracy of the solution demands in order to ensure stability. Hence the advantages of explicit methods when it comes to computational effort per time step are well compensated for by the number of time steps needed in order to keep a stable solution. Explicit methods are therefore not to recommend for stiff nonlinear problems. However, for large systems an explicit method can be preferable to an implicit method. This is further discussed below.
The simplest explicit method is the explicit Euler method, which is shown in the appendix for a linear problem. Runge-Kutta (RK) methods are basically more sophisticated versions of the explicit Euler method, which means that RK methods also must take small time steps due to stability demands. Explicit solvers like RK do not need to build or decompose a Jacobian. For this reason the CPU time scales linearly with the number of equations, i.e. species.

$$CPU_{time} = CPU^{0}_{time} + \alpha N_S \tag{3.1}$$

where CPU^{0}_{time} is the CPU time independent of the number of equations, N_S is the number of species and α is a constant. However, for stiff systems small time step sizes are needed, which makes the coefficient α large. This results in long CPU times. Hence, stiffness removal, which is described in section 3.2.2, is very important for explicit solvers.

Implicit methods calculate the new value based on the derivative of the **new** value, which gives them the advantage that they are always stable for systems of linear ODE no matter how large the time step is. The time step size in these methods applied to linear ODE is determined by the accuracy demands of the solution. An example of an implicit method is the implicit Euler method, which is shown in the chapter appendix for a linear problem. This method is not to recommend for nonlinear problems, since stability cannot be ensured for all time step sizes for systems of nonlinear ODE. Instead, a Predictor Corrector (PC) method (see section 3.2.3.1) based on an implicit method is preferable.

There are two major approaches for stiff problems. One approach is to use implicit solvers with an analytic Jacobian for the system of NAE. Another is to use explicit solvers with stiffness reduction. The performance of the methods depends on the size and stiffness of the system of ODE [5].

The CPU cost for an implicit method that involves matrix inversion is larger than an explicit method for each time step. The reason for this is that the implicit method scales as the cube of the number of equations, while the explicit method scales linearly with the number of equations. However, the number of time steps is much larger for an explicit method than for the implicit method due to the stiffness and stability demands described above. This implies that there is a breaking point somewhere, which is problem dependent, when the number of equations is large enough for the CPU time for the implicit method to exceed the CPU time for the explicit method. A schematic illustration of this is shown in Figure 3.1.
However, implicit methods are recommended for stiff non linear problems, since the explicit methods demands tiny time steps due to stability and sometimes do not converge at all in practice. Hence, an implicit method, described in section 3.3, was chosen for the work behind this thesis.

Most problems will benefit from higher-order methods. If the system of ODE is nonlinear and stiff the higher-order methods like PC methods, which are descendants of Gear's backward differentiation method are preferred [7,8]. Stiffness of a system of ODE is explained in section 3.2.2.

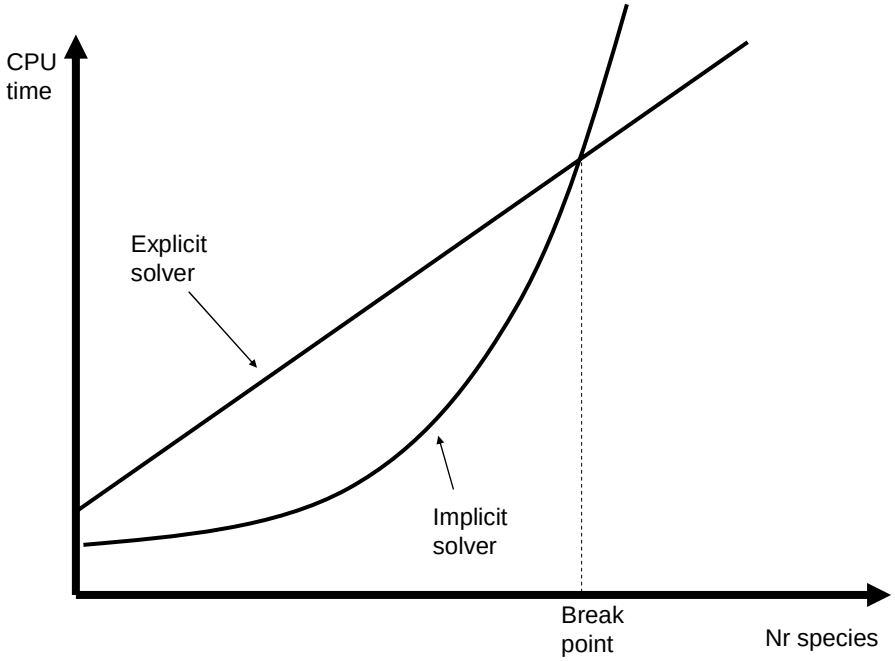


Figure 3.1. A schematic illustration of the CPU time vs number of species for implicit and explicit solvers. The decomposition of the Jacobian for implicit solver scales cubically, while the explicit solver scales linearly with the number of species. When the number of species is large enough a break point is reached where the CPU time of the implicit solver exceeds the CPU time of the explicit solver.

3.2.2. Stiffness

In order to choose the appropriate numerical method for a system of ODE or DAE, it is important to know if the system is stiff or not. The stiffness of a system of ODE depends on the nature of the physical system. A system is considered stiff if it has a wide range of timescales for the time period of interest. If many different time scales are involved, the time step of the numerical method and the numerical method itself must be chosen in a way that the smallest time scales are resolved. If not, the important details of the solution will be missed and the system can behave in a completely different way than expected. However, resolving the smallest timescales means long calculation times, which is the main problem with stiff systems and a motivation for using QSSA (see below).
The stiffness also depends on the initial conditions, since the solution trajectory is affected by them. The stiffness also depends on the actual processes taking place at hand. In a combustion problem the short time interval where the ignition takes place often has high stiffness, while following sequence when combustion is established has lower stiffness.

The timescales in combustion
Figure 3.2 shows the time scales involved in combustion. The shortest time scales that must be resolved by the numerical method are due to the short lived radicals, which have a lifetime of the order of 10^{-12} s [2]. The longest time scales in a combustion process is due to soot formation, which are of the order of 10^{0} s. Soot formation is not considered in the work behind this thesis.
The physical time scales, which are resolved in CFD calculations, ranges from 10^{-2} to 10^{-6} s. This difference between physical and chemical timescales makes it possible to solve the chemistry and CFD part separately, in CFD/chemistry interaction programs (see section 2.3.4.3).

Stiffness and condition number
The eigenvalues of the Jacobian of ODE systems reflect the time scales in the ODE system. The degree of stiffness can be estimated by the ratio between the largest and smallest eigenvalue of the Jacobian [2].
Hence, the stiffness of a system is linked to the Condition Number (CN) of the Jacobian matrix, which is discussed in detail in section 3.3.4.1.1. In a time interval where the CN of the Jacobian is high, the numerical method is forced to take smaller time steps in order to ensure the accuracy of the solution. This means that different regions have different stiffness.
The system of ODE describing combustion in the ACVR is very stiff in the region close to the ignition point, but much less stiff before and after the ignition point. The large stiffness close to the ignition point is due to the presence of short lived species.

Stiffness reduction
It is of great importance to reduce the stiffness of the system if possible in order to gain CPU time. One approach to accomplish this is to set fast reacting species into QSSA, which is explained in detail in section 2.3.5.4. The species that the QSSA is applied to are not involved in the system of ODE, resulting in a less stiff system of ODE, since it involves a smaller range of time scales. However, convergence problems for the resulting system of DAE can emerge due to the application of the QSSA. This can sometimes be interpreted as increased stiffness since the CPU time is affected.

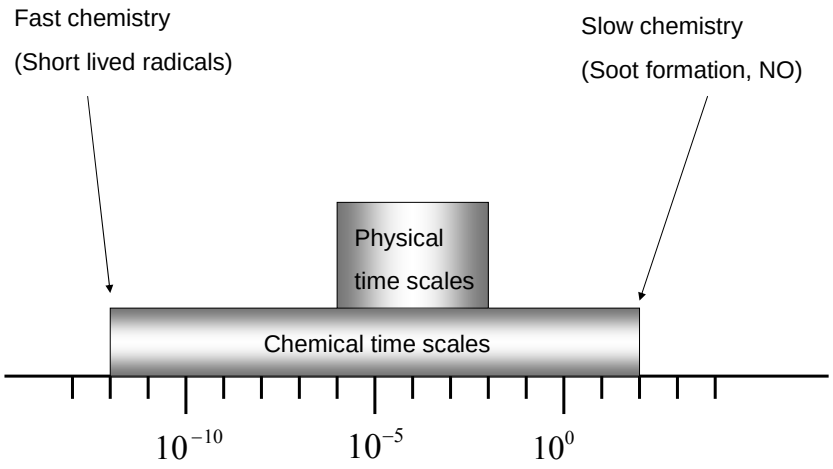


Figure 3.2. Physical and chemical time scales [2]. The chemical time scales range over a much larger time interval than the physical time scales.

3.2.3 Integration of stiff systems of ODE

In order to advance the solution of $y' = f(x, y)$ from x_n to x we have to evaluate;

$$y(x) = y_n + \int_{x_n}^{x} f(x', y)dx' \tag{3.2}$$

In a single step method the value of y_{n+1} at x_{n+1} only depends on y_n.
In a multi-step method, $f(x, y)$, is approximated by a polynomial that passes through several previous points $x_n, x_{n-1}, ...$ and possibly through x_{n+1}. Evaluating eq(3.2) at $x = x_{n+1}$ then gives [1];

$$y_{n+1} = y_n + h(\beta_0 y'_{n+1} + \beta_1 y'_n + \beta_2 y'_{n-1} + \beta_3 y'_{n-2} + ...) \tag{3.3}$$

using

$$y'_n = f(x_n, y_n) \tag{3.4}$$

The same time step size is used for all points in eq(3.3), which is generally not the case in practical problems. The numerical method applied in this thesis uses different time step sizes for the different points.
If $\beta_0 = 0$, the method is explicit and can be used as a predictor in a PC method, otherwise the method is implicit. The order of the method depends on how many previous step, that is the number of previous *y's*, that are used to get each new value of y.

An implicit method must be used for stiff problems if a very small step size shall be avoided. However, not all implicit methods works well for stiff problem, but some good ones are known. The Newton method was chosen as a basis for the work behind this thesis and is widely used for this type of problems [1]. The step size is not of great importance for the convergence of the Newton method, as long as the prediction is within the convergence radius of the Newton method (see section 3.3.1.1.). A predictor based on Gear's backward differentiation method provides an accurate prediction for the Newton method.
The numerical method used in this thesis is called "Implicit backward differencing modified Newton method", which is a PC method that uses Gear's backward differentiation method as a predictor and a modified Newton method as a corrector. The main reasons why this method was chosen are;

- The system to be solved is very stiff
- The method is implicit
- It is a PC method
- It is a higher order method
- Adaptive step size is used

The "Implicit backward differencing modified Newton method" is described in detail in section 3.3 and the principle behind PC methods is briefly described in section 3.2.3.1.

3.2.3.1. Predictor Corrector (PC) methods

A PC method is a two step algorithm. First, a rough approximation of the desired quantity is predicted by a "predictor" algorithm. Thereafter a "corrector" algorithm is used to refine the initial approximation of the desired quantity. Hence, the predictor acts as a starting value of the corrector iteration. The predictor can be a very simple method, like an explicit Euler method, but a more sophisticated predictor is preferable for complex systems, since the starting value of the corrector iteration must be within the convergence radius of the corrector method. The PC method used in work behind this thesis is described in section 3.3.

3.3. The numerical method for the system of ODE

When some species are set into QSSA the system will consist of differential and algebraic equations. The numerical method for this system of DAE consists of two parts, one part that solves the ODE and another part that solves the NAE. The two numerical methods interact with each other, which is described in section 3.5. The numerical method for the system of ODE is based on a PC method and is often referred to as "Implicit backward differencing modified Newton method".
The predictor is based on Gear's backward differencing scheme, while the corrector is a Newton solver. The basic Newton method is described in section 3.3.1, the corrector in section 3.3.2 and the predictor in section 3.3.3. The numerical method for the system of NAE is described in section 3.4.

3.3.1 The Basic Newton Method

The Newton method [3] is a method for finding zeroes of a function, $\mathbf{f}(\mathbf{x})$. If the function, $\mathbf{f}(\mathbf{x}) \neq 0$, is Taylor expanded to linear order around a certain **x**-value, that is

$$\mathbf{f}(\mathbf{x}+\mathbf{h}) = \mathbf{f}(\mathbf{x}) + \mathbf{J}(\mathbf{x}) \cdot \mathbf{h} + O(\mathbf{h}^2) \tag{3.5}$$

and

$$\mathbf{f}(\mathbf{x}+\mathbf{h}) = 0 \tag{3.6}$$

is demanded, the following linear system is obtained;

$$\mathbf{J}(\mathbf{x}) \cdot \mathbf{h} = -\mathbf{f}(\mathbf{x}) \tag{3.7}$$

This system is solved for **h**. The solution means that **h** is the distance we have to move from $\mathbf{f}(\mathbf{x})$ in the direction **h** in order to find $\mathbf{f}(\mathbf{x}+\mathbf{h}) = 0$. The function-value at $\mathbf{x}+\mathbf{h}$ is generally not zero, since only a linear approximation to the function at $\mathbf{f}(\mathbf{x}+\mathbf{h})$ was made. The whole process is thereafter repeated with $\mathbf{x} = \mathbf{x}+\mathbf{h}$. This iteration scheme is continued until some chosen stopping criterion is reached.

The stopping criterion in the solvers for the present work is a combination of absolute tolerance and relative tolerance for the **x**-values. This is discussed in detail in section 4.2.1. A schematic illustration of the Newton iteration is shown in Figure 3.3.

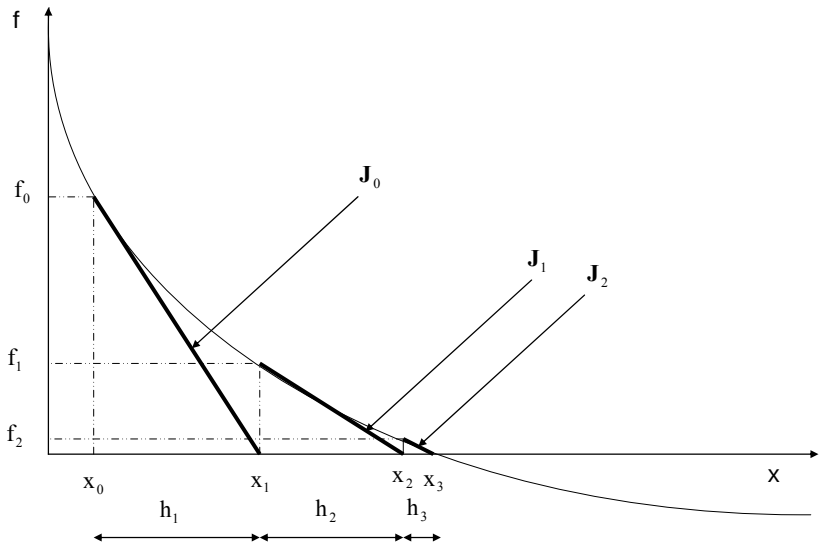


Figure 3.3. A schematic illustration of a Newton iteration in 1-D. f_0 is the function value of x_0, which is the starting point. The next x-value, x_1, is found when the tangent, J_0, cuts the x-axis. x_1 then equals x_0+h_1. f_1 is the function value of x_1. The tangent $J_1$is then found for f_1 and the process is repeated until the last x-value is found, which is when the iterated x-value is close enough to the point where the real function cuts the x-axis.

3.3.1.1. Convergence of the Newton method

The Newton iteration (now written in 1-D form) can be rewritten as an Fixed Point (FP) iteration [3];

$$x_{k+1} = g(x_k) \tag{3.8}$$

where

$$g(x) = x - \frac{f(x)}{f'(x)} \tag{3.9}$$

To examine the convergence the derivative is calculated;

$$g'(x) = \frac{f(x) \cdot f''(x)}{f'(x)^2} \tag{3.10}$$

$g'(x^*) = 0$ if x^* is a root, since $f(x^*) = 0$. Also, $f'(x^*) \neq 0$ must be fulfilled.
Hence, the Newton method is an extremely powerful technique and converges very fast close to the root. The convergence is in general quadratic [3], which means that the error is essentially squared at each iteration step.
This shows that the Newton method normally converges faster than an ordinary FP iteration, which is discussed in section 3.4.2.

A numerical method will converge towards the solution if the starting point of the iteration is within the convergence radius of the solution. This is illustrated in Figure 3.4. As long as $|g'(x)| < 1$ and if the starting value is within the convergence radius, the solution will converge to the point x^*.

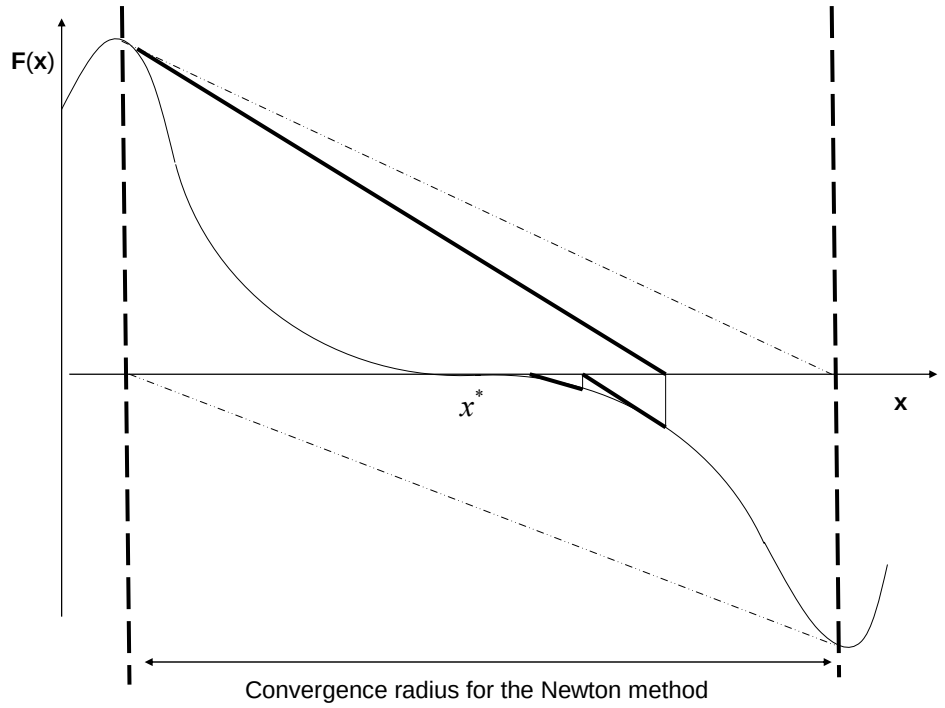


Figure 3.4. A schematic illustration of the convergence radius for the Newton method. The method converges if the starting value of the iteration is within the convergence radius. This is illustrated by the solid lines. The solid dotted lines show the border of the convergence radius, while the thinner dotted lines shown the tangent at the border of the convergence radius.

3.3.2 The corrector

The system of ODE can be expressed as;

$$\dot{\mathbf{x}}_D(t) = \boldsymbol{\omega}_D(\mathbf{x}_D(t)) \tag{3.11}$$

where $\mathbf{x}_D(t)$ is the time dependent concentration vector, $\dot{\mathbf{x}}_D(t)$ is the vector of the time derivative of the concentrations and $\boldsymbol{\omega}_D(\mathbf{x}_D(t))$ is the source term. The subscript *D* just denotes that the vector belongs to the system of ODE. (The subscripts *A* will be used below to denote that the vector belongs to the system of NAE).
The original problem can be reformulated as a zero value problem, that is, finding the roots of the function $\mathbf{g}_D(\mathbf{x}_D(t))$, which is defined as;

$$\mathbf{g}_D(\mathbf{x}_D(t)) = \dot{\mathbf{x}}_D(t) - \boldsymbol{\omega}_D(\mathbf{x}_D(t)) = 0 \tag{3.12}$$

When the roots to equation (3.12) are found, equation (3.11) is fulfilled. If equation (3.12) was fulfilled at time $t=t_n$ it means that $\mathbf{x}_D(t_n)$ was found so that $\mathbf{g}_D(\mathbf{x}_D(t_n)) = 0$. If the equation shall be fulfilled at time $t=t_{n+1}$, $\dot{\mathbf{x}}_D(t_{n+1})$ and $\mathbf{x}_D(t_{n+1})$ must be known. Unfortunately, $\dot{\mathbf{x}}_D(t_{n+1})$ and $\mathbf{x}_D(t_{n+1})$ are not known.
However, $\dot{\mathbf{x}}_D(t_{n+1})$ can be estimated from a polynomial of $\mathbf{x}_D$ for $t=t_{n+1}$ and previous time steps. The polynomial is derived via Taylor expansions at $t=t_{n+1}$, which are used to estimate $\mathbf{x}(t_n)$, $\mathbf{x}(t_{n-1})$, $\mathbf{x}(t_{n-2})$ and so on. A derivation for the third order Taylor expansions can be found in the chapter appendix.
The general expression of $\dot{\mathbf{x}}_D(t_{n+1})$ using a Taylor expansion of the order n_{ord} and n previous time points is [7,8];

$$\dot{\mathbf{x}}_D(t_{n+1}) = -\sum_{j=1}^{n}\left(\frac{\prod_{k=1,k\neq j}^{n_{ord}} \Delta t_k / \Delta t_j}{\prod_{k=1,k\neq j}^{n_{ord}} (\Delta t_k - \Delta t_j)} \cdot \mathbf{x}_D(t_{n+1-j})\right) + \left(\sum_{j=1}^{n}\frac{1}{\Delta t_j}\right)\cdot \mathbf{x}_D(t_{n+1}) \tag{3.13}$$

The meaning of the different Δt_i is shown in Figure 3.6. However, since the $\mathbf{x}_D(t_{n+1})$ do not exist, they must be predicted and inserted into eq (3.13) in order to get the predictor of $\dot{\mathbf{x}}_D(t_{n+1})$.
The predictor for $\mathbf{x}_D(t_{n+1})$ is found from a polynomial of $\mathbf{x}_D$ from previous time steps and is derived in a similar fashion as $\dot{\mathbf{x}}_D(t_{n+1})$. The predictor for $\mathbf{x}_D(t_{n+1})$ is discussed in detail section 3.3.3. Hence, both the predicted $\mathbf{x}_D$ values and the predicted time derivative of $\mathbf{x}_D$ are functions of $\mathbf{x}_D$ values from the previous time steps.

This means that eq (3.12) is solved via a PC method. First the predicted **x**-value, $\mathbf{x}_D^*(t_{n+1})$, is used to evaluate $\mathbf{g}_D(\mathbf{x}_D^*(t_{n+1}))$. Thereafter the **x**-value is corrected via the corrector algorithm, which is explained below, until $\mathbf{g}_D(\mathbf{x}_D(t_{n+1})) = 0$.

In order to solve the zero value problem the function, $\mathbf{g}_D(\mathbf{x}_D(t))$, is Taylor expanded to first order around the point $\mathbf{x}_D^i(t_{n+1})$ at time $t=t_{n+1}$. (The index, *i*, just indicates the iteration step).

$$\mathbf{g}_D(\mathbf{x}_D^{i+1}(t_{n+1})) = \mathbf{g}_D(\mathbf{x}_D^i(t_{n+1}) + \Delta\mathbf{x}_D^i(t_{n+1})) = \\ \mathbf{g}_D(\mathbf{x}_D^i(t_{n+1})) + \frac{\partial \mathbf{g}_D(\mathbf{x}_D^i(t_{n+1}))}{\partial \mathbf{x}_D}(\Delta\mathbf{x}_D^i(t_{n+1})) + O\left(\left[\Delta\mathbf{x}_D^i(t_{n+1})\right]^2\right) \tag{3.14}$$

where $\Delta\mathbf{x}_D^i(t_{n+1}) = \mathbf{x}_D^{i+1}(t_{n+1}) - \mathbf{x}_D^i(t_{n+1})$. In the following discussion $\mathbf{x}_D^i = \mathbf{x}_D^i(t_{n+1})$ in order to simplify the equations.

Truncation to linear order and using $\mathbf{J}_D^i(\mathbf{x}_D^i) = \frac{\partial \mathbf{g}_D(\mathbf{x}_D^i)}{\partial \mathbf{x}_D}$ gives;

$$\mathbf{g}_D(\mathbf{x}_D^{i+1}) = \mathbf{g}_D(\mathbf{x}_D^i) + \mathbf{J}_D^i(\mathbf{x}_D^i)\Delta\mathbf{x}_D^i \tag{3.15}$$

In order to find the roots $\mathbf{g}_D(\mathbf{x}_D(t))$ we demand that;

$$\mathbf{g}_D(\mathbf{x}_D^{i+1}) = 0 \tag{3.16}$$

This gives the following recursion equation;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i - \frac{\mathbf{g}_D(\mathbf{x}_D^i)}{\mathbf{J}_D^i(\mathbf{x}_D^i)} \tag{3.17}$$

For practical reasons we do not want to invert the Jacobian **J**. Instead the linear system is solved after the Jacobian has been decomposed via GE;

$$\mathbf{J}_D^i(\mathbf{x}_D^i)\cdot\Delta\mathbf{x}_D^i = -\mathbf{g}_D(\mathbf{x}_D^i) \tag{3.18}$$

and the $\mathbf{x}_D$ is updated according to;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i + \Delta\mathbf{x}_D^i \tag{3.19}$$

In practice, the same Jacobian can be reused for many iteration steps in order to save CPU time. This is illustrated in Figure 3.5. If the Jacobian from *k* previous iteration step is reused the recursion equation is;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i - \frac{\mathbf{g}_D(\mathbf{x}_D^i)}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k})} \tag{3.20}$$

The original system of ODE turns into a system of DAE when the QSSA is used for a part of the species. This means that the source term, $\boldsymbol{\omega}_D\left(\mathbf{x}_D(t), \mathbf{x}_A(t)\right)$, and thereby $\mathbf{g}_D\left(\mathbf{x}_D(t), \mathbf{x}_A(t)\right)$ is a function of the species concentrations from the system of NAE, $\mathbf{x}_A(t)$.
The system of NAE is iterated until convergence for each iteration step, *i*, of the Newton iteration for the system of ODE. Also, the system of NAE is iterated until convergence for each dimension when the Jacobian for the system of ODE is built, which is further discussed in section 3.3.4.1. Hence, the $\mathbf{x}_A$ in $\mathbf{g}_D\left(\mathbf{x}_D(t), \mathbf{x}_A(t)\right)$ will contain the latest updated value for iteration step *i* and will therefore have the corresponding index *i*. In other words, $\mathbf{x}_{A,i}(t_{n+1})$ is the final value of the last full iteration for the system of NAE at iteration step *i* and time step $t=t_{n+1}$. In the following discussion $\mathbf{x}_{A,i} = \mathbf{x}_{A,i}(t_{n+1})$ in order to simplify the equations. The recursion equation for the system of ODE then looks like;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i - \frac{\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i})}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k})} \tag{3.21}$$

And the linear system looks like;

$$\mathbf{J}_D^{i-k}(\mathbf{x}_D^i, \mathbf{x}_{A,i-k}) \cdot \Delta\mathbf{x}_D^i = -\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i}) \tag{3.22}$$

and the $\mathbf{x}_D$ are updated according to;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i + \Delta\mathbf{x}_D^i \tag{3.23}$$

The first iteration at time step $t=t_{n+1}$ starts with the predictor $\mathbf{x}_D^*(t_{n+1})$ and takes the form;

$$\mathbf{x}_D^1 = \mathbf{x}_D^* - \frac{\mathbf{g}_D(\mathbf{x}_D^*, \mathbf{x}_{A,*})}{\mathbf{J}_D^*(\mathbf{x}_D^*, \mathbf{x}_{A,*})} \tag{3.24}$$

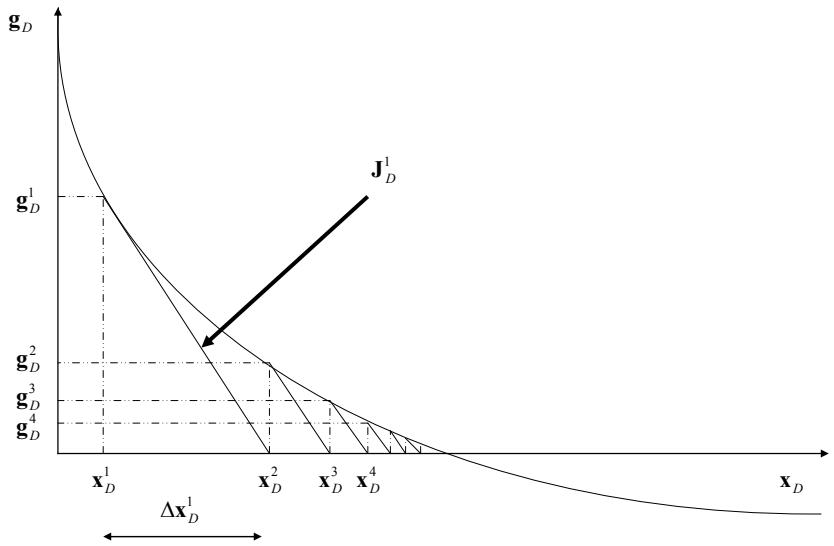


Figure 3.5. A schematic 1-D illustration of the Newton iteration for the system of ODE. The **x**- value and the corresponding **g**-value are shown for the first four iteration steps. The same Jacobian can be used for many iteration steps in order to save CPU time.

3.3.2.1. Damping of the Newton method

If $\left|\mathbf{g}\left(\mathbf{x}_D^{i+1}(t_{n+1})\right)\right| > \left|\mathbf{g}\left(\mathbf{x}_D^{i}(t_{n+1})\right)\right|$ the solution moves further and further away from zero, that is, the solution diverges. To overcome this problem a damping factor, $0 < \lambda < 1$, is introduced in the recursion equation (3.21) resulting in;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^{i} - \lambda \cdot \frac{\mathbf{g}_D(\mathbf{x}_D^{i}, \mathbf{x}_{A,i})}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k})} \tag{3.25}$$

Hence, the difference between $\mathbf{x}_D^{i+1}$ and $\mathbf{x}_D^{i}$ becomes smaller with the damping. A schematic illustration of the of the damping of the Newton method is shown in Figure 3.6.

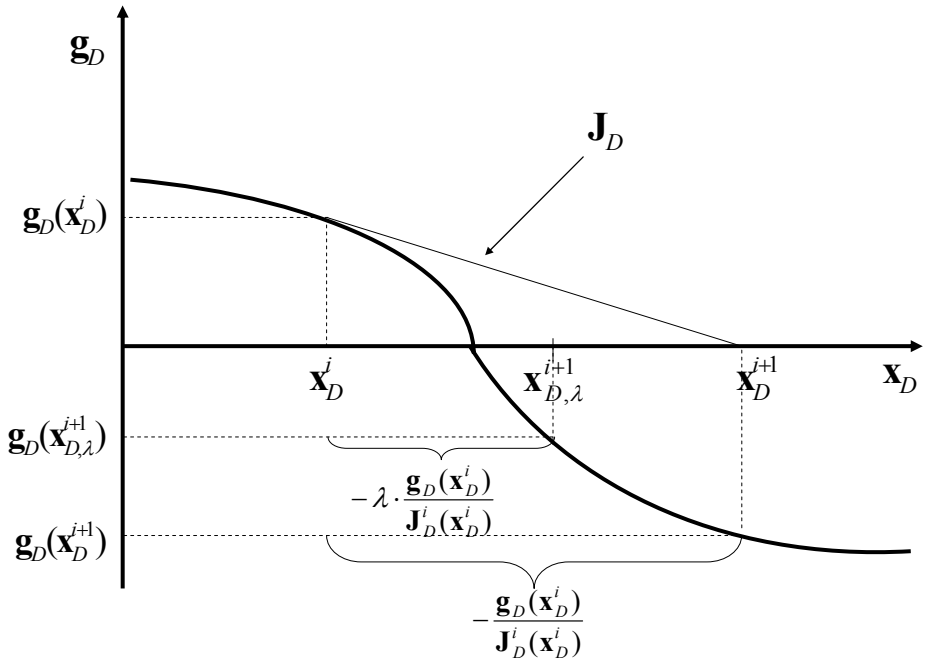


Figure 3.6. A schematic illustration of the damping of the Newton method. The damping is controlled by the parameter λ where $0 < \lambda < 1$. The method diverges without the damping since $\left|\mathbf{g}_D(\mathbf{x}_D^{i+1})\right| > \left|\mathbf{g}_D(\mathbf{x}_D^i)\right|$. However, the method converges with the damping since $\left|\mathbf{g}_D(\mathbf{x}_{D,\lambda}^{i+1})\right| < \left|\mathbf{g}_D(\mathbf{x}_D^i)\right|$.

3.3.3. The predictor

In order to find the $\mathbf{x}_D$ that fulfills eq(3.12) a "predictor" is needed as a first guess of the $\mathbf{x}_D$ then a "corrector" to find better and better approximations of the $\mathbf{x}_D$. This means that the system of ODE is solved by a PC method.

The $\mathbf{x}_D$-value at the next time step, $t=t_{n+1}$, can be predicted by extrapolating the $\mathbf{x}_D$ values from previous time steps. Or in other words, the predictor $\mathbf{x}_D^*(t_{n+1})$ is obtained by extrapolating a polynomial expression fitted to the points of the previous time steps, $\mathbf{x}_D(t_n)$, $\mathbf{x}_D(t_{n-1})$, $\mathbf{x}_D(t_{n-2})$ and so on.
The predictor must lie within the convergence radius, which is discussed in section 3.3.1.1, of the corrector if the $\mathbf{x}_D$ corresponding to, $\mathbf{g}_D(\mathbf{x}_D(t), \mathbf{x}_A(t)) = 0$, is to be found by the corrector. Both $\mathbf{x}_D$ and its time derivative must be predicted at time t_{n+1}, since $\mathbf{g}_D(\mathbf{x}_D(t), \mathbf{x}_A(t)) = \dot{\mathbf{x}}_D(t) - \boldsymbol{\omega}_D(\mathbf{x}_D(t), \mathbf{x}_A(t))$.
The $\mathbf{x}_A$ values from the previous time point are not considered in the calculation of the predictor for $\mathbf{x}_D$, since they have already contributed in the calculation of $\mathbf{x}_D$ at the previous time point.

3.3.3.1. Gear's Backward Differentiation Formulas (BDFs).

A derivation for a predictor with a third order Taylor expansion and three time points is found in the appendix. The general predictor of the $\mathbf{x}_D$ value at time point t_{n+1} is [7,8];

$$\mathbf{x}_D^*(t_{n+1}) = \sum_{j=1}^{n} \left(\frac{\prod_{k=1,k\neq j}^{n_{ord}} \Delta t_k}{\prod_{k=1,k\neq j}^{n_{ord}} (\Delta t_k - \Delta t_j)} \cdot \mathbf{x}_D(t_{n+1-j}) \right) \tag{3.26}$$

Here n represents the number of previous time steps used in the method and n_{ord} represents the order of the method. The truncation error of the Taylor expansion decreases with the order of the polynomial. The meaning of the different Δt_i is shown in Figure 3.6.
The general estimation of the time derivative of the $\mathbf{x}_D$ value at time point t_{n+1} that is used in the corrector is [7,8];

$$\dot{\mathbf{x}}_D(t_{n+1}) = -\sum_{j=1}^{n} \left(\frac{\prod_{k=1,k\neq j}^{n_{ord}} \Delta t_k / \Delta t_j}{\prod_{k=1,k\neq j}^{n_{ord}} (\Delta t_k - \Delta t_j)} \cdot \mathbf{x}_D(t_{n+1-j}) \right) + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) \cdot \mathbf{x}_D(t_{n+1}) \tag{3.27}$$

The predicted value from eq(3.26) is inserted into eq(3.27) to give a predictor for $\dot{\mathbf{x}}_D(t_{n+1})$. This means that both the predicted $\mathbf{x}_D$ values and the predicted time derivative are functions of the $\mathbf{x}_D$ values from the previous time steps.

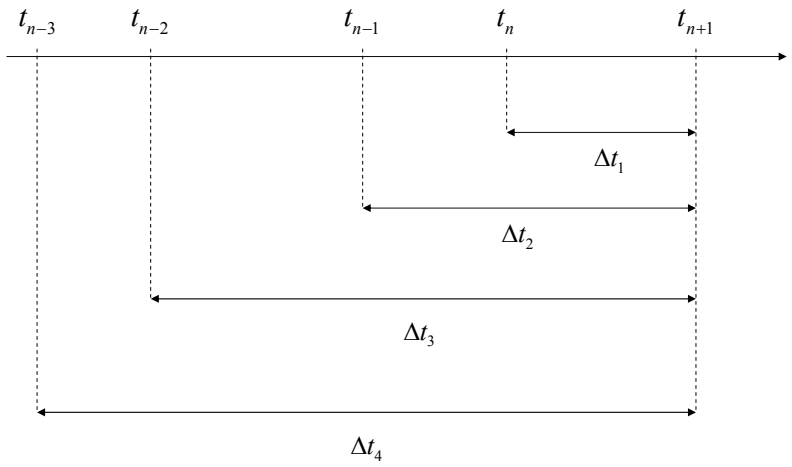


Figure 3.7. The meaning of the different Δt_i is shown. Note that Δt_1 is the difference between the present and future time point and that the other Δt_i is the difference between the previous and future time point.

3.3.3.2. Adaptive time step size

In order to save CPU time, the numerical method has to be able to vary the time step size and use a longer step length as soon as the accuracy of the solution permits it. In this thesis the time step is increased if the difference between the predictor and corrector for a given time step is small and the convergence is fast. The time step size will not be increased if the difference between the predictor and corrector for a given time step is small but the convergence slow and vice versa.

The difference between the predictor and corrector for a given time step can act as an indicator for the stiffness of the system. This is illustrated is Figure 3.8.
A large difference at the given time step indicates large changes in the system, while a small difference indicates small changes.
The new time step, Δt_1, is accepted if;

$$\left|\mathbf{x}_D^p(t_{n+1}) - \mathbf{x}_D^c(t_{n+1})\right| < \frac{\mathrm{Re}\,lativeTolerance}{\Delta t_1} \tag{3.28}$$

Where Relative tolerance in a number and $\mathbf{x}_D^p(t_{n+1})$ and $\mathbf{x}_D^c(t_{n+1})$ are the solution vectors for the predictor and corrector respectively. If the difference between the predictor and corrector for a given time step is too large the time step is decreased.
The time step is typically small at the ignition point, due to rapid changes of the concentrations and temperature, and typically large after the ignition point when the concentrations and temperature do not change much. Before the ignition point the time step size is of average length due to moderate changes in concentrations and temperature. The time step size affects;

- The acceptance of a new time step
- The CPU time of the simulation
- The condition number of the Jacobian
- The error in the predictor and the corrector

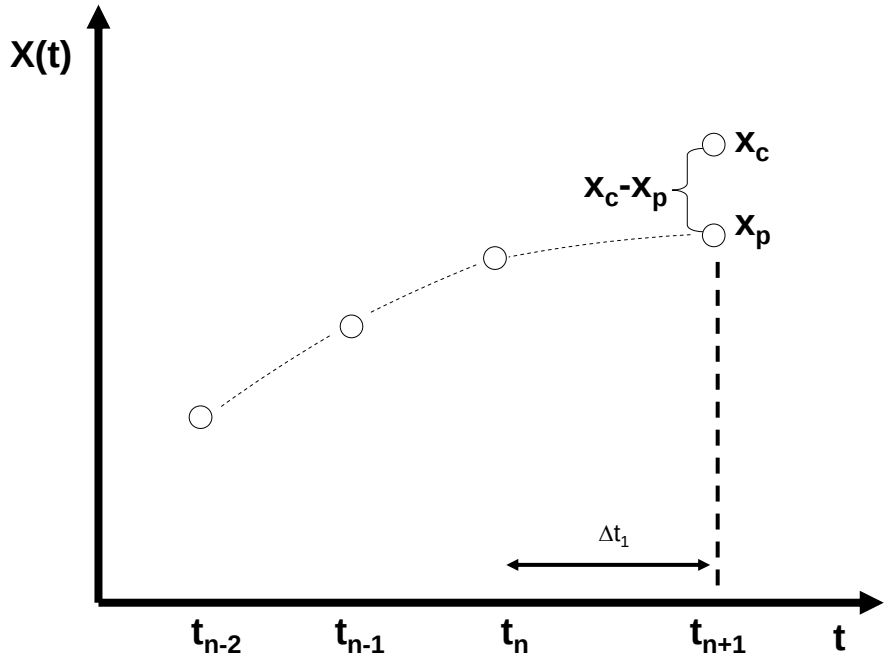


Figure 3.8. A schematic illustration of the predictor and corrector for the new time step Δt_1. The predictor for time step t_{n+1} is extrapolated from a polynomial from x-values of previous time points. The x-value is then corrected by the corrector. The difference between the predictor and corrector is used to accept or decline the new time step size. If the difference is small the time step size is accepted, if not the time step size is decreased.

3.3.4. The Newton method in detail

The Newton method for the system of ODE is computationally expensive, since the Jacobian must be built from finite differences and thereafter decomposed. The accuracy of the solution and the convergence of the Newton method are both affected by the Jacobian, which in turn is affected by the concentrations of the species in the system of NAE and the time step size. It is important to know this dependence in order to control the CPU time and the accuracy of the solution.

3.3.4.1. The Jacobian

The Jacobian in the corrector for the system of ODE deserves special attention, since it has a large effect on the CPU time, which is discussed in section 3.3.4.1.1. Also, the Jacobian for the system of ODE is affected by the time step length and the accuracy of the QSS species concentrations and is therefore a central part of the numerical method. The Jacobian in the corrector for the system of ODE can be written as;

$$\mathbf{J}_D\left(\mathbf{x}_D(t),\mathbf{x}_A(t)\right)=\frac{\partial\mathbf{g}\left(\mathbf{x}_D(t),\mathbf{x}_A(t)\right)}{\partial\mathbf{x}_D(t)}=\frac{\partial\left(\frac{d\mathbf{x}_D(t)}{dt}-\boldsymbol{\omega}\left(\mathbf{x}_D(t),\mathbf{x}_A(t)\right)\right)}{\partial\mathbf{x}_D(t)} \tag{3.29}$$

The Jacobian element, $J_{D,i}^{PQ}$, for time step $t=t_{n+1}$ and iteration step, i, is written as;

$$J_{D,i}^{PQ}\left(\mathbf{x}_D(t_{n+1}),\mathbf{x}_A(t_{n+1})\right)=\frac{\partial g_P\left(\mathbf{x}_D(t_{n+1}),\mathbf{x}_A(t_{n+1})\right)}{\partial x_{D,Q}(t_{n+1})}=\frac{\partial\left(\frac{dx_{D,P}(t_{n+1})}{dt}-\omega_P\left(\mathbf{x}_D(t_{n+1}),\mathbf{x}_A(t_{n+1})\right)\right)}{\partial x_{D,Q}(t_{n+1})} \tag{3.30}$$

The index D corresponds to the system of ODE, the index i corresponds to the i:th iteration step, the index P corresponds to the P:th source term and the index Q corresponds to the Q:th species concentration.
The first term on the RHS can be calculated directly using eq(3.27), which results in;

$$\frac{\partial\left(\frac{dx_{D,P}(t_{n+1})}{dt}\right)}{\partial x_{D,Q}(t_{n+1})}\cdot\delta_Q^P=\left(\sum_{j=1}^{n}\frac{1}{\Delta t_j}\right)\cdot\delta_Q^P \tag{3.31}$$

The $\delta_Q^P = \begin{cases} 1 ; P = Q \\ 0 ; P \neq Q \end{cases}$ means that only the diagonal elements in the Jacobian contains the term $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)$.

The second term in the Jacobian for the system of ODE is calculated from a finite difference in the computer code. The semi discrete form of the Jacobian element, $J_{D,i}^{PQ}$, for time step $t=t_{n+1}$ and iteration step, i, is then written as (from now on $\mathbf{x}_D(t_{n+1}) = \mathbf{x}_D$ and $\mathbf{x}_A(t_{n+1}) = \mathbf{x}_A$ in order to simplify the expressions);

$$J_{D,i}^{P,Q}(\mathbf{x}_D, \mathbf{x}_A) = \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) \cdot \delta_Q^P \cdot - \left(\frac{\omega_p(\mathbf{x}_{Di} + \Delta x_{D,Q} \cdot \hat{\mathbf{e}}_Q, \mathbf{x}_{A,i}) - (\omega_p(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}))}{\Delta x_{D,Q}} \right) \tag{3.32}$$

Hence, the Jacobian is affected by;

- $\mathbf{x}_D$
- $\mathbf{x}_A$
- $\Delta \mathbf{x}_D$
- $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)$ on the diagonal elements.

The influences they have of the CN of the Jacobian are described below.

Since the Jacobian is built from a finite difference, the source term must be calculated at $\boldsymbol{x}_D+\Delta\mathbf{x}_D$. The system of NAE is affected by the new $\boldsymbol{x}_D$ value, which means that the system of NAE must be evaluated at $\boldsymbol{x}_D+\Delta\mathbf{x}_D$. Hence, the system of NAE must be updated for each element of $\boldsymbol{x}_D$. And since the system of NAE must be iterated until convergence for each element of $\boldsymbol{x}_D$, the building of the Jacobian for the system of ODE is expensive in terms of CPU time and must be avoided if possible.
The accuracy of the concentrations from the system of NAE affects the number of Jacobians for the system of ODE needed during a simulation. This is discussed further in section 3.3.4.1.1 and 6.2.1.2.

3.3.4.1.1. The condition number of the Jacobian

A linear equation system $\mathbf{Ax} = \mathbf{b}$ is considered to be well conditioned if a small change in the matrix **A** or a small change in the right hand side vector **b** results in a small change in the solution vector **x**. Consequently, a system of equations is considered to be ill conditioned if a small change in **A** or a small change in **b** results in a large change in **x**. This condition of a system of equations is expressed with the Condition Number (CN) of the matrix.

The Condition Number (CN) associated with the linear equation system $\mathbf{Ax} = \mathbf{b}$ tells how much the solution **x** changes with respect to a change in **b**. If the condition number is large, even a small error in **b** will cause a large error in **x**. If the condition number is small, the error in **x** will be of the same size as the error in **b**. The condition number is a property of the matrix before effects of round-off errors, floating point precision and computer algorithms are taken into account.

The maximum value of the CN of a matrix **A** is;

$$\kappa(\mathbf{A}) = \left\|\mathbf{A}^{-1}\right\|_{\infty} \cdot \left\|\mathbf{A}\right\|_{\infty} \tag{3.33}$$

A derivation of this expression can be found in the chapter appendix.
If *A* is normal;

$$\kappa(\mathbf{A}) = \frac{\left|\lambda_{\max}(\mathbf{A})\right|}{\left|\lambda_{\min}(\mathbf{A})\right|} \tag{3.34}$$

where $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ are the maximum and minimum eigenvalues of **A** respectively. The Jacobians for the system of ODE for the chemical systems this thesis investigates are not symmetric, but almost. Hence, the CN can be estimated by eq(3.34).

The CN of the Jacobian matrix is related to the stiffness of the system, since the eigenvalue spectrum is proportional to the LT of the species. The large range of LT in combustion systems is reflected in high CN of the Jacobian.

The CN of a matrix is a property of the matrix itself. The values of the matrix elements of the Jacobian are functions of the chemical reaction rates, which differ very much depending on the involved species and their concentrations.

Also, the CN of the Jacobian changes with time, since the chemical reaction rates change with time. For ignition simulations, the CN is usually highest around the point of ignition due to the rapid changes of the system.

The Jacobian can be written as a sum of two matrices;

$$\mathbf{J}_D = \mathbf{J}_D^{time} + \mathbf{J}_D^{source} \tag{3.35}$$

where the matrix;

$$\mathbf{J}_D^{time} = \begin{pmatrix} \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j}\right) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j}\right) \end{pmatrix} = \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j}\right) \cdot \mathbf{1} \tag{3.36}$$

contains the time dependence on the diagonal elements and the matrix;

$$\mathbf{J}_D^{source} = -\begin{pmatrix} \left(\frac{\omega_{D,1}\left(x_D + \Delta x_{D,1}\right) - \omega_{D,1}\left(x_D\right)}{\Delta x_{D,1}}\right) & \cdots & \left(\frac{\omega_{D,1}\left(x_D + \Delta x_{D,N_D}\right) - \omega_{D,1}\left(x_D\right)}{\Delta x_{D,N_D}}\right) \\ \vdots & \ddots & \vdots \\ \left(\frac{\omega_{D,N_D}\left(x_D + \Delta x_{D,1}\right) - \omega_{D,N_D}\left(x_D\right)}{\Delta x_{D,1}}\right) & \cdots & \left(\frac{\omega_{D,N_D}\left(x_D + \Delta x_{D,N_D}\right) - \omega_{D,N_D}\left(x_D\right)}{\Delta x_{D,N_D}}\right) \end{pmatrix} \tag{3.37}$$

contains the concentration derivate of the source term.

Influence of the time step size
The effect of the time step size on the CN can be understood in the following way. Note that this reasoning only applies to symmetric (normal) matrices.
The Jacobian in this thesis is not symmetric. However, the Jacobian in this thesis is almost symmetric, which means that the following reasoning indicates the behavior of the Jacobian.

The eigenvalue equation [6] for $\mathbf{J}_D^{source}$ is;

$$\left(\mathbf{J}_D^{source}\right)\mathbf{v} = \lambda^{source}\mathbf{v} \tag{3.38}$$

Where λ^{source} is the eigenvalue and $\mathbf{v}$ is the eigenvector of $\mathbf{J}_D^{source}$.
The eigenvalue equation for $\mathbf{J}_D$ is;

$$\mathbf{J}_D v = \left(\mathbf{J}_D^{source} + \mathbf{J}_D^{time}\right)\mathbf{v} = \lambda^{source}\mathbf{v} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j}\right) \cdot \mathbf{1} \cdot \mathbf{v} = \left(\lambda^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j}\right)\right)\mathbf{v} \tag{3.39}$$

The eigenvectors for $\mathbf{J}_D^{source}$ and $\mathbf{J}_D$ are the same, since $\mathbf{J}_D^{time}$ is diagonal. Hence;

$$\lambda = \left(\lambda^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) \right) \tag{3.40}$$

$\mathbf{J}_D^{source}$ has the CN;

$$\kappa\left(\mathbf{J}_D^{source}\right) = \left| \frac{\lambda_{MAX}^{source}}{\lambda_{MIN}^{source}} \right| \tag{3.41}$$

and the CN of $\mathbf{J}_D$ is

$$\kappa\left(\mathbf{J}_D\right) = \left| \frac{\lambda_{MAX}}{\lambda_{MIN}} \right| = \left| \frac{\lambda_{MAX}^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)}{\lambda_{MIN}^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)} \right| \tag{3.42}$$

If λ_{MAX}^{source} and λ_{MIN}^{source} have the same sign as $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) > 0$, the relation in eq(3.43) holds.

Also, if λ_{MAX}^{source} and λ_{MIN}^{source} have a different sign than $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)$ but $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) < \left| \lambda_{MAX}^{source} \right|$ and $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) >> \left| \lambda_{MIN}^{source} \right|$ the relation in eq(3.43) holds.

For the simulations in this thesis, the typical values of $\left| \lambda_{MAX}^{source} \right|$ and $\left| \lambda_{MIN}^{source} \right|$ are about 10^{11}-10^{15} and 10^{-5}-10^{3} respectively. The time step size is typically 10^{-9}-10^{-5} sec and $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)$ is positive for the simulations in this thesis. Hence, $\left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right) >> \left| \lambda_{MIN}^{source} \right|$ and the inequality in eq.(3.43) holds.

$$\left| \frac{\lambda_{MAX}^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)}{\lambda_{MIN}^{source} + \left(\sum_{j=1}^{n} \frac{1}{\Delta t_j} \right)} \right| \leq \left| \frac{\lambda_{MAX}^{source}}{\lambda_{MIN}^{source}} \right| \tag{3.43}$$

Since $\left(\sum_{j=1}^{n}\frac{1}{\Delta t_j}\right)$ is added to both λ_{MAX}^{source} and λ_{MIN}^{source}, the inequality becomes larger the larger the term $\left(\sum_{j=1}^{n}\frac{1}{\Delta t_j}\right)$ gets, which occurs when the time steps sizes decreases.
Consequently, the difference between the largest and the smallest diagonal elements in the Jacobian, and thereby the CN, decreases when the time step sizes decreases. The opposite is true if the time step increases. Hence, the CN of the full Jacobian $\mathbf{J}_D$ becomes smaller than the CN of $\mathbf{J}_D^{source}$, since the same number is added to all the diagonal elements.

The number of previous time steps used in the predictor affects the CN of the Jacobian via the term $\left(\sum_{j=1}^{n}\frac{1}{\Delta t_j}\right)$. The *n* in the sum corresponds to the number of previous time steps. Hence, the sum increases with the number of previous time steps. The sum also increases if the Δt_j decreases. This means that many small consecutive time steps decrease the CN of the Jacobian. The latest Δt_j is always the smallest and affects the CN most.

When the Jacobian changes very much over the span of a few time steps, for example at the ignition point, it becomes very clear that the Δt_j needs to be small if the predictor shall be accurate and for the time step to be accepted.
This means that large time steps can only be taken when the $\mathbf{J}_D^{source}$ has a low CN in order to ensure that the $\mathbf{J}_D$ does not get a CN that is too large. This happens far away from the ignition point in an ignition simulation.

Influence of the finite difference
If the step size, $\Delta\mathbf{x}_D$, in the matrix, $\mathbf{J}_D^{source}$, is too small, $\Delta\boldsymbol{\omega}$ will be close to zero and cancellation errors will occur due to the computer systems limited floating point precision. Hence, the Jacobian matrix can contain a larger error due to round-off errors. If the step size, $\Delta\mathbf{x}_D$, is too large the round-off errors will not have a large impact on the accuracy of the matrix elements. However, the $\mathbf{J}_D^{source}$ will be evaluated at the wrong point in the **x**-space, which affects the convergence rate.
$\Delta\mathbf{x}_D$ is not varied at all in this thesis, instead a pre-determined value is used.

Influence of the QSS species
The species set into QSSA are often the species with the shortest LT, which corresponds to the largest eigenvalues. Hence, the difference between the largest and smallest eigenvalues in $\mathbf{J}_D^{source}$ decreases, which in turn gives a smaller CN of the Jacobian for the system of ODE.

It should be noted that the CPU time can increase, depending on the solver, when species are set into QSSA. This is due to convergence problems for the solver for the system of NAE. This effect can sometimes be interpreted as an increased stiffness of the system of DAE when QSSA is used.

Influence of the QSS species concentrations

The Jacobian for the system of ODE is affected by the QSS species concentrations, $\mathbf{x}_{A,i}$, via the source term $\boldsymbol{\omega}\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$. The $\boldsymbol{x_A}$ contain an error, $\boldsymbol{\varepsilon}_{\mathrm{A}}$, due to the tolerance of the solver for the NAE system. This means that $\boldsymbol{\varepsilon}_{\mathrm{A}}$ introduces an error in $\boldsymbol{\omega}\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$ and the Jacobian for the system of ODE. Since $\boldsymbol{x_D}$ is found from the linear eq(3.16), $\boldsymbol{\varepsilon}_{\mathrm{A}}$ will cause an error in $\boldsymbol{x_D}$ as well. Following the discussion above on the CN, a small $\boldsymbol{\varepsilon}_{\mathrm{A}}$ will then cause a large error in $\boldsymbol{x_D}$ if the CN of the Jacobian for the ODE system is large.
The $\boldsymbol{x_D}$ contain two errors, ε_D^{tol} and ε_D^A. ε_D^{tol} is the error in $\boldsymbol{x_D}$ due to the tolerance of the solver for the ODE system. ε_D^A is the error in $\boldsymbol{x_D}$ due to $\boldsymbol{\varepsilon}_{\mathrm{A}}$. Hence, the solution $\boldsymbol{x_D}$ from the corrector can be expressed as,

$$x_D = x_D^{exact} + \varepsilon_D^{tol} + \varepsilon_D^A \tag{3.44}$$

where x_D^{exact} is the hypothetical exact solution, with infinite precision from both solvers, from the corrector. (This exact solution cannot be obtained in practice).
It is pointless to have a tolerance in the solver for the ODE system that makes ε_D^{tol} smaller than ε_D^A, since the error in $\boldsymbol{x_D}$ is decided by the dominating error. Hence, the accuracy of the $\boldsymbol{x_A}$ must be high enough to compensate for the high CN of the Jacobian for the ODE system.
This is a motivation for high accuracy demands of the numerical method for the system of NAE. However, if the accuracy demands are too high the solver for the NAE system work in vain, which cost CPU time. Hence, a trade-off between accuracy and speed is needed for the solver for the NAE system.

As stated above, the CN of the Jacobian is a function of the system of ODE itself before effects of round-off errors, floating point precision and computer algorithms are taken into account. However, since $\boldsymbol{\varepsilon}_{\mathrm{A}}$ introduces an error in $\boldsymbol{\omega}_D\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$, $\boldsymbol{\varepsilon}_{\mathrm{A}}$ also introduces an error in the Jacobian elements. This error can affect the experienced stiffness of the equation system even though the real CN is not changed.
Eq.(3.45) and Figure 3.9 illustrate that an error $\boldsymbol{\varepsilon}_{\mathrm{A}}$, due to the inner solver tolerances, introduces an error $\boldsymbol{\varepsilon}_{\mathrm{J}}$ in the Jacobian, which can affect the convergence of the Newton method, which in turn affects the CPU time. The Newton method does not converge at all if the iteration ends up outside the convergence radius due to errors in the Jacobian. Even a small error in the Jacobian can cause the method to overshoot the convergence radius, depending on the landscape of the function **g**. This is illustrated in Figure 3.10.

$$\mathbf{J}_{D,i}^{source} \pm \boldsymbol{\varepsilon}_J\left(\boldsymbol{\varepsilon}_A\left(\mathbf{x}_{A,i}\right)\right) = -\left(\frac{\boldsymbol{\omega}_D\left(\mathbf{x}_{Di} + \Delta\mathbf{x}_{Di}, \mathbf{x}_{A,i} \pm \boldsymbol{\varepsilon}_A\left(\mathbf{x}_{A,i}\right)\right) - \boldsymbol{\omega}_D\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i} \pm \boldsymbol{\varepsilon}_A\left(\mathbf{x}_{A,i}\right)\right)}{\Delta\mathbf{x}_D}\right) \quad (3.45)$$

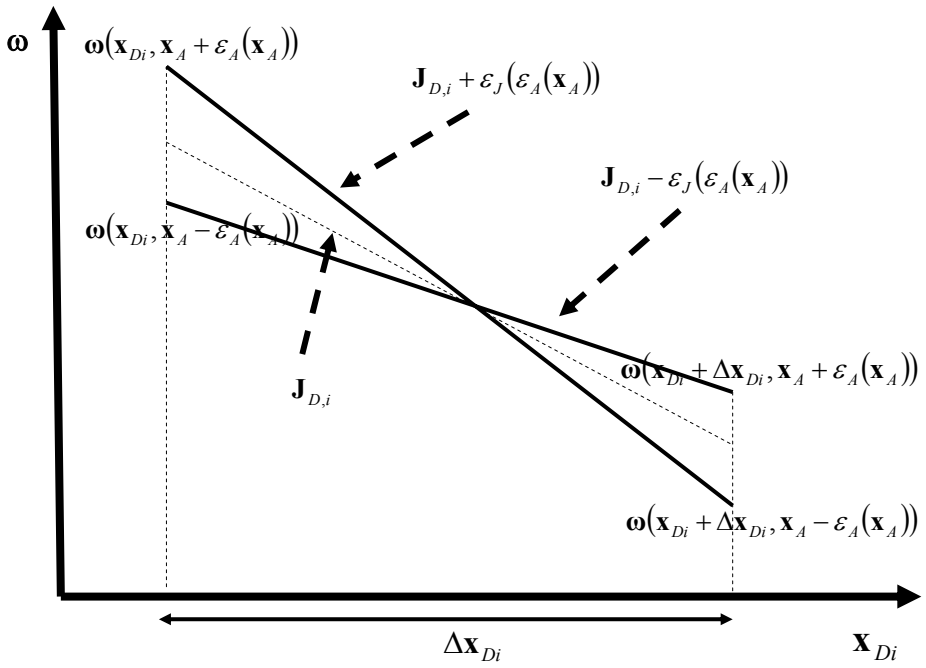


Figure 3.9. The QSS species concentrations, $\mathbf{x}_A$, introduce an error in $\boldsymbol{\omega}_D\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$, which in turn introduces an error in the Jacobian for the system of ODE. Two extreme values of the Jacobian are illustrated with thick lines, while the Jacobian without error is a dotted line in between. The convergence of the Newton method for the system of ODE is affected if the error in the Jacobian is large. Hence, the error in $\mathbf{x}_A$ must be low in order to save CPU time.

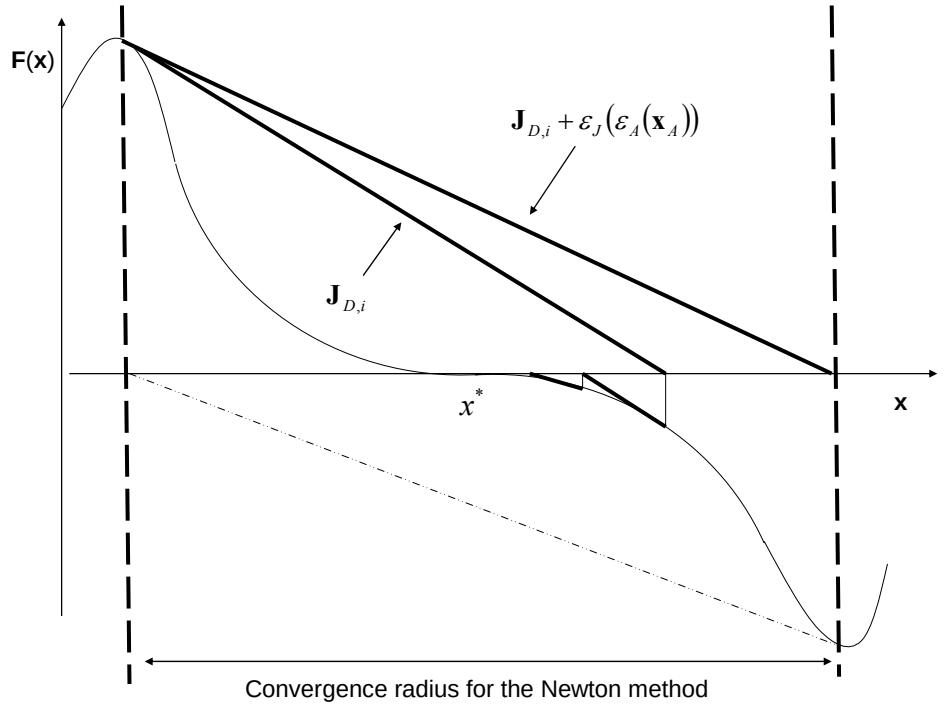


Figure 3.10. A schematic illustration of the effect of an error in the Jacobian due to an error in $\mathbf{x}_A$. The new function value can end up outside the convergence radius of the Newton method if a Jacobian with an error is used, which means that the iteration will not converge.

High accuracy of $\mathbf{x}_A$ can lower the CPU time for the entire solver for the reasons stated above. However, high accuracy demands on the $\mathbf{x}_A$ costs CPU time. So, the question is if there is a numerical method for the system of NAE that can be used in combination with the numerical method for the system of ODE that lowers the entire CPU time?
Such a method exists and is presented below. The simulation results are presented in chapter 6.

The error in $\mathbf{x}_A$ is controlled by the choice of and tolerance level in the numerical method for the system of NAE. It is therefore very important to have a low tolerance level in the solver for the system of NAE and to choose a numerical method that converges, so that the error in $\mathbf{x}_A$ is minimized. The Newton solver for the NAE in section 3.4.1 has the following good qualities in this context;

- The Jacobian is calculated analytically, in contrast to a finite difference. This decreases the error in $\mathbf{x}_A$.
- $\mathbf{x}_A$ is iterated until convergence for each iteration step of $\mathbf{x}_D$. This decreases the error in $\mathbf{x}_D$ due to the error in $\mathbf{x}_A$.
- The method uses the gradient information, which facilitates the convergence

If a FP solver is used for calculation of $\mathbf{x}_A$, the $\mathbf{x}_A$ are found from equations like eq(3.77). If the species in the denominator have very low concentrations, a division with approximately zero is made, which can give large errors in $\mathbf{x}_A$. This happens especially at low temperatures for combustion simulations.

The FP solver does not have any tools, like gradient information, to improve the convergence in order to get a solution. Hence, the convergence rate is generally slower than that of the Newton method. The FP iteration does not have any guaranties of convergence, which can introduce large errors in $\mathbf{x}_A$ and thereby in $\boldsymbol{\omega}_D\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$. This influences the convergence of the Newton method for the ODE system for the reasons stated above.

Similar arguments that were used for the Jacobian for the system of ODE can be used for the Jacobian for the system of NAE. The source term for the species that are solved from the NAE, and thereby the values of the matrix elements of the Jacobian for the system of NAE, are affected by $\mathbf{x}_D$. This means that an error in $\mathbf{x}_D$ introduces an error in the Jacobian for the system of NAE. This is further discussed in section 3.4.1.

3.3.4.2 Ordinary Gaussian Elimination (GE)

In every iteration step for a Newton method, a linear system must be solved. The normal way to solve a linear system is to perform a back-substitution procedure. This implies that the matrix must be upper triangular. The normal way to make a matrix into an upper triangular matrix is to perform a LU-factorization procedure or a Gaussian elimination procedure. The solver this thesis is based on uses a Gaussian elimination procedure.

The ordinary Gaussian elimination makes all column elements below the diagonal of the matrix equal to zero, in order for the matrix to be “upper triangular”. Mathematically the Gaussian elimination can be expressed as [3];

$$m_{ik} = \frac{a_{ik}}{a_{kk}} \qquad i = k+1,\ldots,n \tag{3.46}$$

$$a'_{ij} = a_{ij} - m_{ik} \cdot a_{kj}, \qquad j = k+1,\ldots,n, \qquad i = k+1,\ldots,n \tag{3.47}$$

$$b'_i = b_i - m_{ik} \cdot b_k, \qquad i = k+1,\ldots,n \tag{3.48}$$

where a_{ij} is the matrix element for row *i* and column *j* and b_i is the RHS vector for row *i*. Index *k*, runs from 1 to n-1 and m_{ik} is the ratio between the matrix element, a_{ik}, in column, *k*, that shall be set to zero and the matrix element, a_{kk}, on the diagonal of

column, *k*. a_{ij} is the matrix element on row, *i*, and column, *j*, and a'_{ij} is the same matrix element after subtraction. b_i is the element belonging to the *i*:th row in the *b*-vector and b'_i is the same element after subtraction. This procedure makes the *i*:th row element of column *k* zero;

$$a'_{ik} = a_{ik} - m_{ik} \cdot a_{kk} = a_{ik} - \frac{a_{ik}}{a_{kk}} a_{kk} = 0 \tag{3.49}$$

and the *i:th* row element of column *j* non zero (it can be zero by coincidence);

$$a'_{ij} = a_{ij} - m_{ik} \cdot a_{kj} = a_{ij} - \frac{a_{ik}}{a_{kk}} a_{kj} \neq 0 \tag{3.50}$$

The procedure starts with *k=1* and eliminates all a_{i1}, which are not equal to zero from the beginning, in column 1. The elimination of column 1 also affects the other matrix elements and the *b*-vector, which are located at the same rows as the eliminated elements in column 1.
In the next step *k*=2 and all a_{i2}, which are not equal to zero from the beginning, are eliminated from column 2. The matrix elements in the rest of the matrix and the *b*-vector, which are at the same row as the eliminated elements in column 2 are affected. This procedure continues until the last column is reached and outputs an upper triangular matrix, which can be used for back substitution.

Transformation of a linear system into upper triangular form, with *n* unknowns, demands approximately $\frac{2}{3}n^3$ operations [3], since the Gaussian elimination uses three loops, which are proportional to the size of the matrix.

3.3.4.3. Back Substitution (BS) of a triangular system.

If the matrix A is upper triangular, it can be solved by back substitution;
First the lowest *x*-value is found by;

$$x_n = b_n / a_{nn} \tag{3.51}$$

Then the second lowest x-value is found, using the lowest *x*-value, by;

$$x_i = \left(b_i - \sum_{j=i+1}^{n} a_{ij} \cdot x_j \right) / a_{ii}, \quad i = n-1,..,1 \tag{3.52}$$

This equation is then repeated until all *x*-values are found. In other words, the first *x* – value that is found is x_n by the equation $x_n = b_n / a_{nn}$. The second *x*-value to be found is x_{n-1} by the equation $x_{n-1} = \left(b_{n-1} - a_{n-1,n} \cdot x_n\right) / a_{n-1,n-1}$. The third *x*-value to be found is x_{n-2} by the equation $x_{n-2} = \left(b_{n-2} - a_{n-2,n-1} \cdot x_{n-1} - a_{n-2,n} \cdot x_n\right) / a_{n-2,n-2}$ and so on.

It takes about n^2 operations to solve a triangular system with *n* unknowns [3].

3.3.4.4. CPU cost for the Newton method

The cost of transforming a system of linear equations of dimension, *n*, into triangular form, by Gaussian elimination, is approximately $\frac{2}{3}n^3$ operations. The cost of solving a triangular equation system, by back substitution, is approximately n^2 operations. If the same Jacobian can be reused for several iteration steps, the number of operations for a full iteration of the Newton method can be expressed as follows;

$$\text{number of operations per full iteration} = \sum_{i=1}^{NI} \left[P_i \left(\frac{2}{3} \cdot n^3 \right) + \left(1 \cdot n^2\right) \right]_i \tag{3.53}$$

where *NI* is the number of iteration steps, P_i is a number that is either zero or one. If P_i is one the Jacobian is decomposed at iteration step, *i*. If P_i is zero the Jacobian from the previous time step is reused. This is discussed further in section 3.5.3

3.4. Two numerical methods for systems of NAE

This section presents two numerical methods for solving a system of NAE. The first method is the Newton method and the second is the FP method. The Newton solver for the system of NAE, which is based on the Newton method presented below, is optimized for speed and is presented in section 4.3.1.

3.4.1. The Newton method

The method is the same as the method described in section 3.3.2.
The system of NAE can be expressed as;

$$0 = \boldsymbol{\omega}_A(\mathbf{x}_A(t)) \tag{3.54}$$

where $\mathbf{x}_A(t)$ is the time dependent concentration vector, $\boldsymbol{\omega}_A(\mathbf{x}_A(t))$ is the source term. The subscript *A* just denotes that vector belongs to the system of NAE. This is a zero value problem, that is, the task is to find the roots of the function $\boldsymbol{\omega}_A(\mathbf{x}_A(t))$. Taylor expansion gives;

$$\begin{aligned}&\boldsymbol{\omega}_A(\mathbf{x}_A^{j+1}(t_{n+1})) = \boldsymbol{\omega}_A(\mathbf{x}_A^j(t_{n+1}) + \Delta\mathbf{x}_A^j(t_{n+1})) = \\ &\boldsymbol{\omega}_A(\mathbf{x}_A^j(t_{n+1})) + \frac{\partial\boldsymbol{\omega}_A(\mathbf{x}_A^j(t_{n+1}))}{\partial\mathbf{x}_A}(\Delta\mathbf{x}_A^j(t_{n+1})) + O\left(\left[\Delta\mathbf{x}_A^j(t_{n+1})\right]^2\right)\end{aligned} \tag{3.55}$$

where $\Delta\mathbf{x}_A^j(t_{n+1}) = \mathbf{x}_A^{j+1}(t_{n+1}) - \mathbf{x}_A^j(t_{n+1})$. In the following discussion $\mathbf{x}_A^j = \mathbf{x}_A^j(t_{n+1})$ in order to simplify the equations.
Truncation to linear order gives;

$$\boldsymbol{\omega}_A(\mathbf{x}_A^{j+1}) = \boldsymbol{\omega}_A(\mathbf{x}_A^j) + \mathbf{J}_{A,i}^j(\mathbf{x}_A^j)\Delta\mathbf{x}_A^j \tag{3.56}$$

In order to find the roots $\boldsymbol{\omega}_A(\mathbf{x}_A(t))$ we demand that;

$$\boldsymbol{\omega}_A(\mathbf{x}_A^{j+1}) = 0 \tag{3.57}$$

This gives the following recursion equation;

$$\mathbf{x}_A^{j+1} = \mathbf{x}_A^j - \frac{\boldsymbol{\omega}_A(\mathbf{x}_A^j)}{\mathbf{J}_{A,i}^j(\mathbf{x}_A^j)} \tag{3.58}$$

For practical reasons we do not want to invert the Jacobian $\boldsymbol{J_A}$. Instead the linear system is solved after the Jacobian has been decomposed via GE;

$$\mathbf{J}_{A,i}^{j}(\mathbf{x}_A^j)\cdot\Delta\mathbf{x}_A^j = -\boldsymbol{\omega}_A(\mathbf{x}_A^j) \tag{3.59}$$

and the $\mathbf{x}_A$ is updated according to;

$$\mathbf{x}_A^{j+1} = \mathbf{x}_A^j + \Delta\mathbf{x}_A^j \tag{3.60}$$

In practice, the same Jacobian can be reused for many iteration steps in order to save CPU time. This is illustrated in Figure 3.11. If the Jacobian from *m* previous iteration step is reused the recursion equation is;

$$\mathbf{x}_A^{j+1} = \mathbf{x}_A^j - \frac{\boldsymbol{\omega}_A(\mathbf{x}_A^j)}{\mathbf{J}_{A,i}^{j-m}(\mathbf{x}_A^{j-m})} \tag{3.61}$$

The original system of ODE turns into a system of DAE when the QSSA is used for a part of the species. This means that the source term, $\boldsymbol{\omega}_A\left(\mathbf{x}_A(t),\mathbf{x}_D(t)\right)$, is a function of the species concentrations from the system of ODE, $\mathbf{x}_D(t)$.
The system of NAE is iterated until convergence for each iteration step, *i*, of the Newton iteration for the system of ODE. Also, the system of NAE is iterated until convergence for each element in $\boldsymbol{x_D}$ when the Jacobian for the system of ODE is built, which is further discussed in section 3.3.4.1. Hence, $\mathbf{x}_D(t)$ is a constant throughout each full iteration for the system of NAE.
In the following discussion $\mathbf{x}_{A,i}^{j} = \mathbf{x}_{A,i}^{j}(t_{n+1})$ in order to simplify the equations. The recursion equation for the system of NAE then looks like;

$$\mathbf{x}_{A,i}^{j+1} = \mathbf{x}_{A,i}^{j} - \frac{\boldsymbol{\omega}_A(\mathbf{x}_{A,i}^{j},\mathbf{x}_D^{i})}{\mathbf{J}_{A,i}^{j-m}(\mathbf{x}_{A,i}^{j-m},\mathbf{x}_D^{i})} \tag{3.62}$$

And the linear system looks like;

$$\mathbf{J}_{A,i}^{j-m}(\mathbf{x}_{A,i}^{j-m},\mathbf{x}_D^{i})\cdot\Delta\mathbf{x}_A^j = -\boldsymbol{\omega}_A(\mathbf{x}_{A,i}^{j},\mathbf{x}_D^{i}) \tag{3.63}$$

and the $\mathbf{x}_A$ are updated according to;

$$\mathbf{x}_A^{j+1} = \mathbf{x}_A^j + \Delta\mathbf{x}_A^j \tag{3.64}$$

A schematic Newton iteration is illustrated in Figure 3.11. The iteration starts with a starting value $\mathbf{x}_A^1$, which has a corresponding function value $\mathbf{f}_A^1$. The iteration thereafter continues and produces new function-values, each associated with an $\mathbf{x_A}$-value. In this particular illustration the iteration converges toward the zero, which means that the starting value was within the convergence radius of the method.

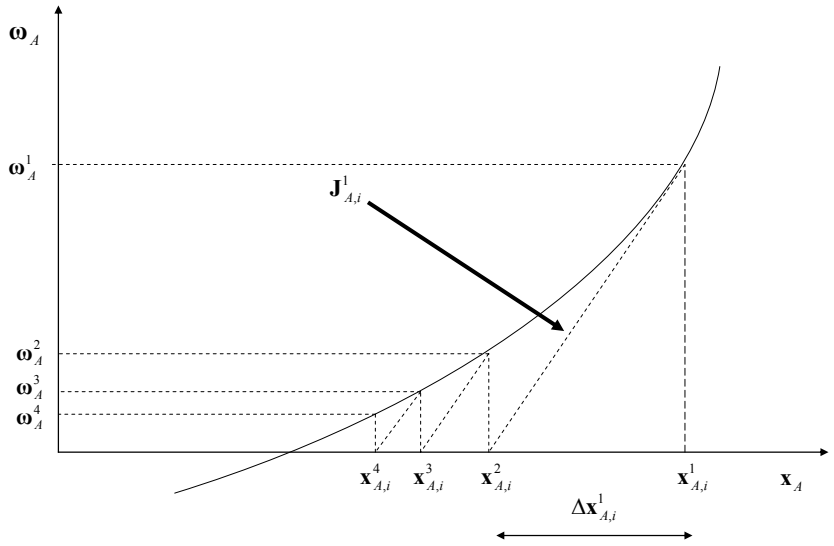


Figure 3.11. A schematic 1-D illustration of the Newton iteration for the system of NAE. The **x**- value and the corresponding **ω**-value are shown for the first four iteration steps. The same Jacobian can be used for many iteration steps.

The Newton iteration for the system of NAE does not use the same kind of predictor as the Newton iteration for the system of ODE does, since the NAE does not have a time derivative. Instead, the final value, $\mathbf{x}_{A,i}^{j\ final}$, of the full iteration for the system of NAE for $\mathbf{x}_D^i$ can act as a predictor for the next inner iteration for $\mathbf{x}_D^{i+1}$, since $\mathbf{x}_D$ appears as a constant in $\boldsymbol{\omega}_A(\mathbf{x}_{A,i}^j(t_{n+1}), \mathbf{x}_D^i(t_{n+1}))$ for each full iteration of the system of NAE.
The reason for this is that the function-landscape for the system of NAE changes for each new $\mathbf{x}_D$ and that $\mathbf{x}_{A,i}^{j\ final}$ for that reason does not correspond to a zero anymore.
In other words, $\mathbf{x}_D$ changes between two iteration steps for the outer solver so that $\boldsymbol{\omega}_A(\mathbf{x}_{A,i}^{j\ final}(t_{n+1}), \mathbf{x}_D^{i+1}(t_{n+1})) \neq 0$ even though $\boldsymbol{\omega}_A(\mathbf{x}_{A,i}^{j\ final}(t_{n+1}), \mathbf{x}_D^i(t_{n+1})) = 0$, which will have the effect that $\mathbf{x}_{A,i}^{j\ final}(t_{n+1})$ from the last full inner iteration will act as a predictor for the new iteration for the system of NAE. This is schematically illustrated for the Newton method in Figure 3.12.

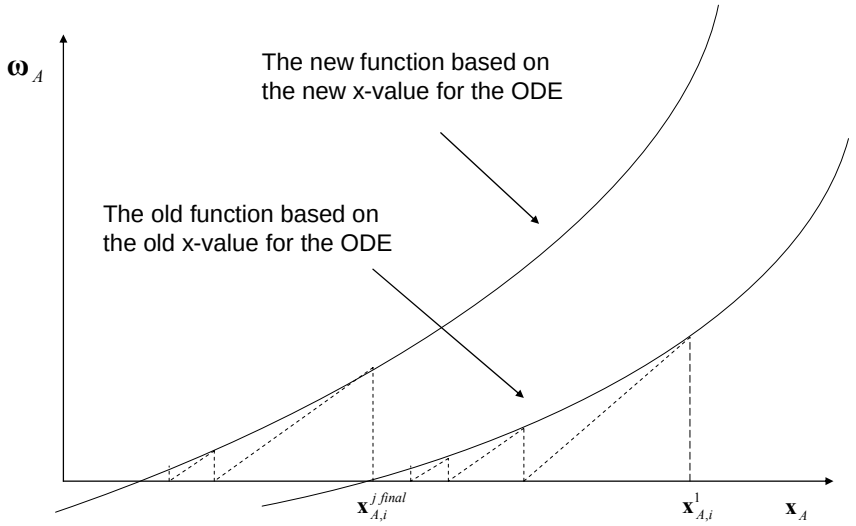


Figure 3.12. A schematic illustration of the effect of a new $\mathbf{x_D}$ on the function-landscape for the system NAE. The $\mathbf{x_D}$ appears as a constant in the function for the system of NAE but still affects the function value. The final x-value for the last full iteration for the system of NAE, $\mathbf{x}_A^{j\ final}$, based on the old $\boldsymbol{x_D}$ generally does not correspond to the zero of the function based on the new $\boldsymbol{x_D}$. Hence, the $\mathbf{x}_A^{j\ final}$ based on the old $\mathbf{x_D}$ acts as a predictor for the new iteration for the NAE.

3.4.1.1. The Jacobian for the system of NAE

Evaluation of the Jacobian can be performed either numerically or analytically. The numerical evaluation is time consuming, while the analytical evaluation is not. The Jacobian for the system of ODE must be evaluated numerically, since the temperature in the exponential term in the Arrhenius expression is a function of the predictor, which in turn is a function of the concentration of the species in the system of ODE.
The numerical evaluation is expensive since all reaction rates need to be evaluated for each perturbed species concentration. The Jacobian for the system of ODE must therefore evaluate the concentrations for the system of NAE for each perturbed species concentration in the system of ODE.
However, the Jacobian for the system of NAE can be evaluated analytically, since the temperature appears as a constant in the source term for the NAE each time the system of NAE is solved.

The Jacobian for the system of NAE can be written as;

$$\mathbf{J}_A\left(\mathbf{x}_D(t),\mathbf{x}_A(t),T(t)\right)=\frac{\partial\boldsymbol{\omega}_A\left(\mathbf{x}_D(t),\mathbf{x}_A(t),T(t)\right)}{\partial\mathbf{x}_A(t)} \tag{3.65}$$

The P:th source term for the system of NAE for time step $t=t_{n+1}$, outer iteration step, i, and inner iteration step j, is written as; (After the first equality sign $\mathbf{x}_A=\mathbf{x}_A(t_{n+1})$, $\mathbf{x}_D=\mathbf{x}_D(t_{n+1})$ and $T=T(t_{n+1})$ in order to simplify the equations. The index i and j are dropped after the second equality sign in order to simplify the expression.)

$$\begin{aligned}&\omega_{A,P}\left(\mathbf{x}_D^i(t_{n+1}),\mathbf{x}_{A,i}^j(t_{n+1}),T(t_{n+1})\right)=\sum_{k=1}^{N_R}\upsilon_{Pk}r_k\left(\mathbf{x}_D^i,\mathbf{x}_{A,i}^j,T\right)=\\&=\sum_{k=1}^{N_R}\upsilon_{Pk}\prod_{l=1}^{N_D}x_{D,l}^{\upsilon_{lk}}\cdot\prod_{m=1}^{N_A}x_{A,m}^{\upsilon_{mk}}\cdot K_k(T),\quad P=k+1,N_S\end{aligned} \tag{3.67}$$

The Jacobian element, $J_{A,i}^{j,PQ}$, for time step $t=t_{n+1}$, outer iteration step, i, and inner iteration step j, is written as; (After the first equality sign $\mathbf{x}_A=\mathbf{x}_A(t_{n+1})$, $\mathbf{x}_D=\mathbf{x}_D(t_{n+1})$ and $T=T(t_{n+1})$ in order to simplify the equations. The index i and j are dropped after the second equality sign in order to simplify the expression.)

$$\begin{aligned}&J_{A,i}^{j,PQ}\left(\mathbf{x}_D(t_{n+1}),\mathbf{x}_A(t_{n+1}),T(t_{n+1})\right)=\frac{\partial\omega_{A,p}\left(\mathbf{x}_D^i,\mathbf{x}_{A,i}^j,T(t)\right)}{\partial x_{A,Q}}=\\&=\sum_{k=1}^{N_R}\upsilon_{pk}\prod_{l=1}^{N_D}x_{D,l}^{\upsilon_{lk}}\cdot\prod_{\substack{m=1\\m\neq Q}}^{N_A}x_{A,m}^{\upsilon_{mk}}\cdot K_k(T)\cdot\left(x_{A,Q}^{\nu_{Qk}-1}\cdot\nu_{Qk}\right)=\\&=\sum_{k=1}^{N_R}\upsilon_{pk}\prod_{l=1}^{N_D}x_{D,l}^{\upsilon_{lk}}\cdot\prod_{m=1}^{N_A}x_{A,m}^{\upsilon_{mk}}\cdot K_k(T)\cdot\frac{1}{x_{A,Q}}\cdot\nu_{Qk}\end{aligned} \tag{3.68}$$

N_R, N_D and N_A are the number of reactions, number of species in the system of ODE and number of species in the system of NAE respectively.
$\mathbf{x}_A$ is the variable, while $\mathbf{x}_D$ is regarded as a constant for inner iteration step, j.

The sum in equation (3.68) is over all reactions N_R including both forward and backward since they are treated separately. The first product symbol is for the species in the system of ODE, while the second product symbol is for the species in the system of NAE. However, only a maximum of three species are involved in one and each reaction, which means that most v_{lk}=0 and v_{mk}=0. Hence, most $x_{D,l}^{\upsilon_{lk}}(t)=1$ and $x_{A,m}^{\upsilon_{mk}}(t)=1$. Also, an arbitrary species p is involved in part of all reactions only. Hence, many υ_{pk} are zero in the sum over reactions.
If the species Q is not present in a particular ω_P, all v_{Qk} are zero for that particular ω_P, which means that the corresponding Jacobian element is zero. Most species, Q, are only involved in a part of all ω_P, which means that the Jacobian is more or less sparse. This

will be further discussed and utilised when the solver for the system of NAE is optimized for speed in section 4.3.3.

3.4.2 The FP iteration method

A value x which satisfies the equation:

$$x = g(x) \tag{3.69}$$

is called a fixed point of the function g, since x is unchanged when g is applied to it.
A nonlinear function $f(x) = 0$ can be rewritten as a FP problem, $x = g(x)$. Many iterative algorithms for solving nonlinear equations are based on iteration schemes of the form

$$x_{k+1} = g(x_k) \tag{3.70}$$

where g is a function chosen so that its fixed points are solutions for $f(x) = 0$. Such a procedure is called a FP iteration, since g is applied repeatedly to an initial starting value x_0. The iteration starts with an initial guess x_0 and is then iterated until convergence, while producing a series of $x_1, x_2, \ldots, x_N$. The fixed point is reached when;

$$x^* = g(x^*) \tag{3.71}$$

where x^* is the fixed point.

Convergence.
A FP iteration only converges if $|g'(x^*)| < 1$ and if x_0 is close enough to the fixed point [4]. The fastest possible convergence for a FP iteration is when $|g'(x^*)| = 0$. One way to achieve this is through the Newton method described in section 3.3.1.

CPU cost for the FP method
The CPU time scales linearly with the number of equations, n, in the equation system that is to be solved. The constant, C_{FP}, depends on the number of iterations that was needed in order to achieve convergence.

$$CPU_{time} = C_{FP} \cdot n \tag{3.72}$$

3.4.2.1. The FP method applied to the system of NAE

The FP method applied to the NAE is described below.
The nonlinear function $0 = \boldsymbol{\omega}_A(\mathbf{x}_A(t))$ can be rewritten as a FP problem,

$$\mathbf{x}_A(t) = \mathbf{H}_A(\mathbf{x}_A(t)) \tag{3.73}$$

This gives the recursion equation;

$$\mathbf{x}_A^{j+1}(t_{n+1}) = \mathbf{H}_A(\mathbf{x}_A^j(t_{n+1})) \tag{3.74}$$

The original system of ODE turns into a system of DAE when the QSSA is used for a part of the species. This means that the source term, $\boldsymbol{\omega}_A(\mathbf{x}_A(t), \mathbf{x}_D(t))$, is a function of the species concentrations from the system of ODE, $\mathbf{x}_D(t)$.
The system of NAE is iterated until convergence for each iteration step, *i*, of the Newton iteration for the system of ODE. Also, the system of NAE is iterated until convergence for each element in $\boldsymbol{x_D}$ when the Jacobian for the system of ODE is built, which is further discussed in section 3.3.4.1. Hence, $\mathbf{x}_D(t)$ is a constant throughout each full iteration for the system of NAE.
In the following discussion $\mathbf{x}_{A,i}^j = \mathbf{x}_{A,i}^j(t_{n+1})$ and $\mathbf{x}_D^i = \mathbf{x}_D^i(t_{n+1})$ in order to simplify the equations. The recursion equation for the system of NAE then looks like;

$$\mathbf{x}_{A,i}^{j+1} = \mathbf{H}_A(\mathbf{x}_{A,i}^j, \mathbf{x}_D^i) \tag{3.75}$$

The function $\mathbf{H}_A(\mathbf{x}_{A,i}^j, \mathbf{x}_D^i)$ is explained as follows.
The source term of the *g:th* QSS species, x_A^g, can be found from;

$$\begin{aligned} 0 = \omega_{A,g} &= \sum_{k=1}^{N_R} \upsilon_{gk} r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)) = \\ &= \sum_{k=1}^{N_R^{Prod}} \upsilon_{gk} r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t))^{Prod} + \sum_{k=1}^{N_R^{Cons}} \upsilon_{gk} r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t))^{Cons} \end{aligned} \tag{3.76}$$

where N_R^{Prod} and N_R^{Cons} is the number of producing and consuming reactions of x_A^g respectively, $r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t))^{Prod}$ is the producing reactions of x_A^g, while $r_k(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t))^{Cons}$ is the consuming reactions of x_A^g.
Hence, x_A^g can be found from;

$$x_A^g = \frac{\sum_{k=1}^{N_R^{Prod}} \upsilon_{gk} r_k\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)^{Prod}}{\sum_{k=1}^{N_R^{Cons}} \upsilon_{gk} r_k^{\#}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)^{Cons}} = H_A^g(\mathbf{x}_A, \mathbf{x}_D) \tag{3.77}$$

where $H_A^g(\mathbf{x}_A, \mathbf{x}_D)$ is the *g:th* term of $\mathbf{H}_A(\mathbf{x}_{A,i}^j, \mathbf{x}_D^i)$ and

$$r_k^{\#}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)^{Cons} = \frac{r_k\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)^{Cons}}{x_A^g} \tag{3.78}$$

that is, the consuming reactions of x_A^g divided by the x_A^g concentration.
If the species in $r_k^{\#}\left(\mathbf{x}_D(t), \mathbf{x}_A(t), T(t)\right)^{Cons}$ have very low concentrations, a division with approximately zero is made, which can give large errors in x_A^g. This happens especially at low temperatures for combustion simulations.

The FP solver does not have any tools, like gradient information, to improve the convergence in order to get a solution. Hence, the convergence rate is generally slower than that of the Newton method. The FP iteration does not have any guaranties of convergence, which can introduce large errors in $\mathbf{x}_A$ and thereby in $\boldsymbol{\omega}_D\left(\mathbf{x}_{D,i}, \mathbf{x}_{A,i}\right)$.

A schematic FP iteration is illustrated in Figure 3.11. The iteration starts with a starting value $\mathbf{x}_A^1$, which has a corresponding function value $\mathbf{H}_A^1$. The iteration thereafter continues and produces new function-values, each associated with an **x**-value. In this particular illustration the iteration converges toward the FP, which means that the absolute value of the derivative of the function is less than one at the FP, $\left|\mathbf{H}'\left(\mathbf{x}^*\right)\right| < 1$. If $\left|\mathbf{H}'\left(\mathbf{x}^*\right)\right| > 1$ the iteration can get into higher order cycles or even have a chaotic behavior [4].

The FP iteration for the system of NAE does not use the same kind of predictor as the Newton iteration for the system of ODE does, since the NAE does not have a time derivative. Instead the FP iterations start with zero. The last converged value for the FP iteration has also been tested as a starting value for the next FP iteration but with poorer results.

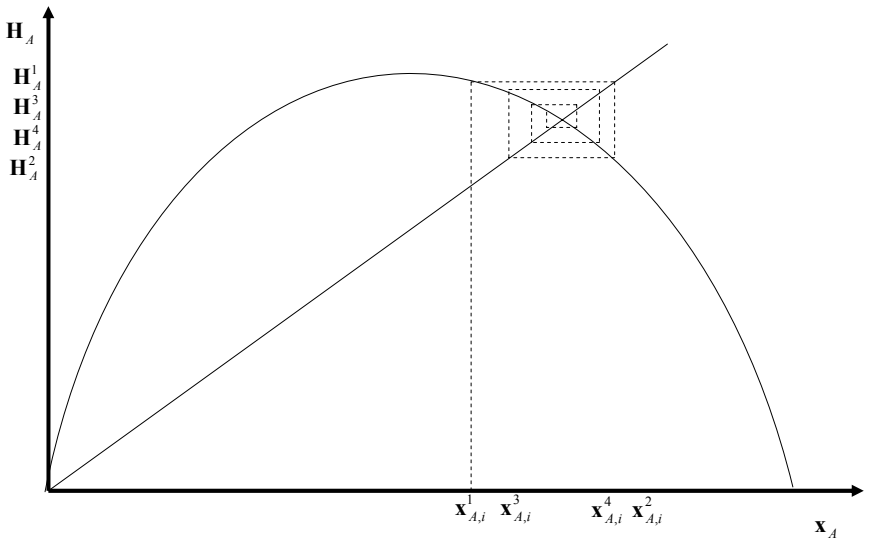


Figure 3.13. A schematic illustration of an FP iteration. Each function value, $\mathbf{H_A}$, is associated with an ***x***-value. In this illustration the iteration converges towards a FP.

3.5. Two combinations of numerical methods for stiff systems of DAE

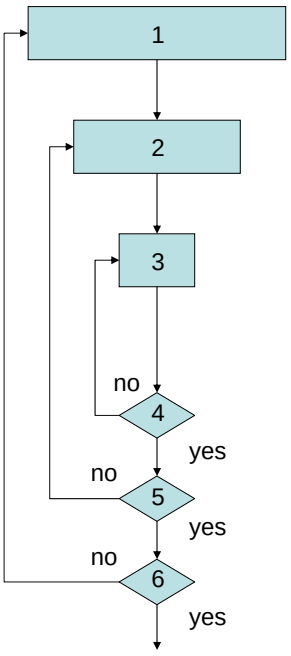
The original system of ODE turns into a system of DAE when the QSSA is used for a part of the species. This is discussed in detail in section 2.3.5.4.1.
The source term for the system of ODE, $\boldsymbol{\omega}_D\left(\mathbf{x}_A(t),\mathbf{x}_D(t)\right)$, and the source term for the system of NAE, $\boldsymbol{\omega}_A\left(\mathbf{x}_A(t),\mathbf{x}_D(t)\right)$, are both functions of $\mathbf{x}_D(t)$ and $\mathbf{x}_A(t)$. This dependence connects the system of ODE and the system of NAE to each other. Hence, the numerical methods for the system of ODE and NAE are also connected to each other. The system of DAE becomes less accurate compared to the original system of ODE when the number of QSS species increases. This indicates that the recursion equations of the two numerical methods must be combined in a fashion that compensates for the loss of accuracy.

In this thesis the Newton iteration for the system of ODE is iterated until convergence for each time step. The system of NAE is iterated until convergence for each iteration step, *i*, of the Newton iteration for the system of ODE. Also, the system of NAE is iterated until convergence for each element in $\mathbf{x}_D(t)$ when the Jacobian for the system of ODE is built, which is further discussed in section 3.3.4.1. Hence, the system of NAE is iterated until convergence many more times than the system of ODE is.

The numerical method for the system of ODE uses the latest converged value of $\mathbf{x}_A$ at each iteration step, while the numerical method for the system of NAE uses the latest iterated value of $\mathbf{x}_D$, which acts as a constant, throughout each full iteration for the system of NAE.
In this thesis two methods based on the combination of numerical methods are tested;

- Newton-Newton
- Newton-FP

A simple schematic flow chart of the combination of numerical methods is shown in Figure 3.14. A more detailed flow chart is discussed in section 4.4. The two different combinations of numerical methods are described below and only differ in the numerical methods for the NAE.



1. New time step
2. New iteration step for the system of differential equations
3. New iteration step for the system of algebraic equations
4. Convergence for the system of algebraic equations?
5. Convergence for the system of differential equations?
6. Final time step?

Figure 3.14. A simplified flow chart of the combined solver for a system of DAE. The outer loop is a loop over time. For each time step there is an iteration loop for the system of ODE. Finally there is an inner iteration loop for the system of NAE at each iteration step for the system of ODE.

3.5.1. Newton-FP method

This combination of numerical methods uses the Newton method described in section 3.3.2 for the system of ODE and the FP method described in section 3.4.2 for the system of NAE.

The combined numerical method for the system of DAE
The recursion equation for the ODE and the recursion equation for the NAE are connected to each other. The source term of the system of ODE and the source term of the system of NAE both depend on the concentration vector of the algebraic species, $\mathbf{x}_A$, as well as the concentration vector of the differential species, $\mathbf{x}_D$.
For each iteration step of the recursion equation for the system of ODE, the recursion equation for the system of NAE is iterated until convergence and the value of the index *j* is then *j final*. The concentration-vector of the corresponding to *j final*, $\mathbf{x}_{A,i}^{j\,final}(t_{n+1})$, is then inserted into the recursion equation for the system of ODE. The recursion equation for the system of ODE is then iterated one step and the new value, $\mathbf{x}_D^{i+1}(t_{n+1})$, is then inserted into the recursion equation for the system of NAE, which is iterated until convergence again.
This procedure is repeated until the recursion equation for the system of ODE converge and reaches the final value, $\mathbf{x}_D^{i\,final}(t_{n+1})$. Thereafter the predictor for the system of ODE is calculated for the new time step and the whole procedure described above is done all over again for the new time step. In the following discussion $\mathbf{x}_D^i = \mathbf{x}_D^i(t_{n+1})$ and $\mathbf{x}_{A,i}^j = \mathbf{x}_{A,i}^j(t_{n+1})$ is used in order to simplify the equations. The recursion equation for the ODE system is;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i - \frac{\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j\,final})}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k}^{j\,final})} \tag{3.79}$$

The recursion equation for the NAE system is;

$$\mathbf{x}_{A,i}^{j+1} = \mathbf{H}_A(\mathbf{x}_D^i, \mathbf{x}_{A,i}^j)) \tag{3.80}$$

For practical reasons we do not want to invert the Jacobian $\boldsymbol{J}_D$. Instead the linear system is solved after the Jacobian has been decomposed via GE;

$$\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k}^{j\,final}) \cdot \Delta\mathbf{x}_D^i = -\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j\,final}) \tag{3.81}$$

and the $\mathbf{x}_D$ is updated according to;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i + \Delta\mathbf{x}_D^i \tag{3.82}$$

A schematic illustration of the Newton-FP method is shown in Figure 3.15.

In order to get the first iterated value, $\mathbf{x}_D^1(t_{n+1})$, in the recursion equation (3.79) for the system of ODE at time step, t_{n+1}, the predicted value of, $\mathbf{x}_D^*(t_{n+1})$, is used as a starting value of $\mathbf{x}_D$ and $\mathbf{g}_D$ is a function of the predicted value of $\mathbf{x}_D$ and the starting value of $\mathbf{x}_A$, which is the last value of the previous full algebraic iteration, $\mathbf{x}_{A,*}^{j\,final}(t_{n+1})$.

$$\mathbf{x}_D^1 = \mathbf{x}_D^* - \frac{\mathbf{g}_D(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final})}{\mathbf{J}_D^*(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final})} \tag{3.83}$$

The first value of $\mathbf{x}_A$ using the predictor as $\mathbf{x}_D$ and the final value $\mathbf{x}_A$ from the previous time step, t_n, is;

$$\mathbf{x}_{A,*}^1 = \mathbf{H}_A(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final}) \tag{3.84}$$

The final value of $\mathbf{x}_A$ using the predictor as $\mathbf{x}_D$ is;

$$\mathbf{x}_{A,*}^{j\,final} = \mathbf{H}_A(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final-1}) \tag{3.85}$$

The final value of $\mathbf{x}_D$ using the final value of $\mathbf{x}_A$ is;

$$\mathbf{x}_D^{i\,final} = \mathbf{x}_D^{i\,final-1} - \frac{\mathbf{g}_D(\mathbf{x}_D^{i\,final-1}, \mathbf{x}_{A,i}^{j\,final})}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k}^{j\,final})} \tag{3.86}$$

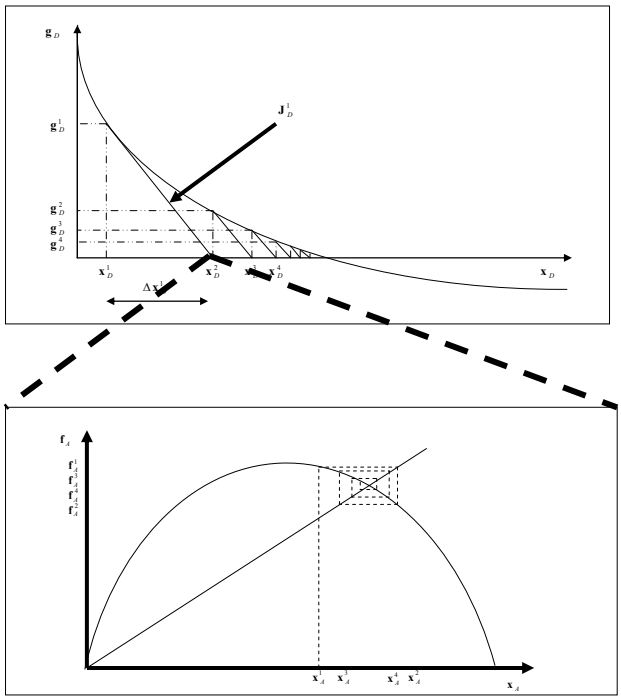


Figure 3.15. The “Newton-FP” method. Above is a schematic illustration, which is the same as Figure 3.5, of the Newton-iteration for the system of ODE. Below is a schematic illustration, which is the same as Figure 3.13, of the FP-iteration for the system of NAE. The FP-iteration is iterated until convergence at each iteration step of the Newton iteration for the system of ODE.

As stated previously, the FP method for the system of NAE is iterated until convergence at each iteration step of the Newton method for the system of ODE. In addition to that, the FP method for the system of NAE is iterated until convergence for each element in $\mathbf{x}_D$ when the Jacobian of the system of ODE is built. This is illustrated in Figure 3.16 and discussed in detail in section 3.3.4.1.

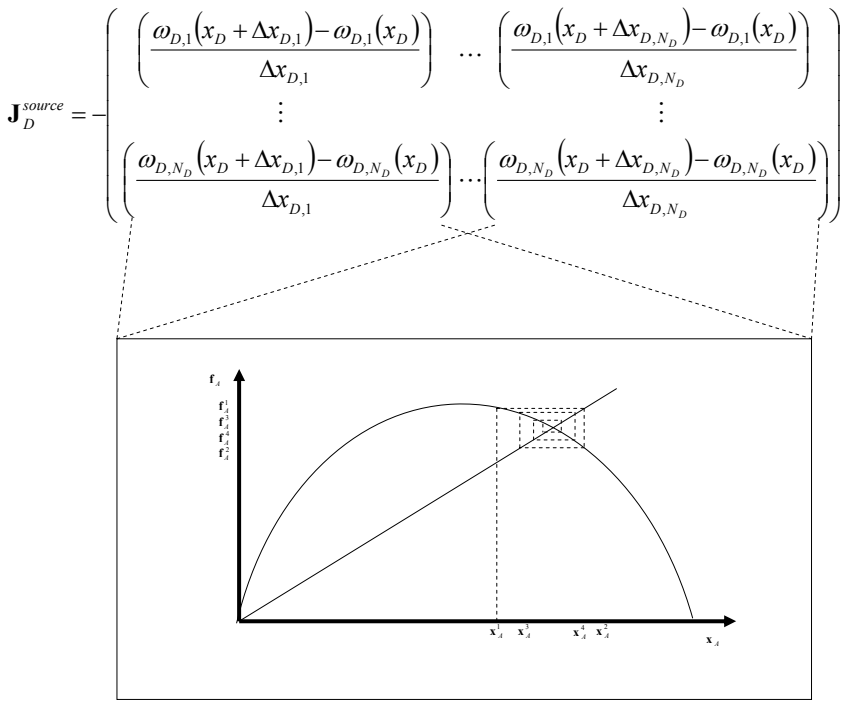


Figure 3.16. The figure illustrates that a full FP iteration, which is the same as Figure 3.13, is performed for each element in $\mathbf{x}_D$ when the Jacobian for the system of ODE is built. This results in N_D full iteration for the system of NAE. N_D is the number of species in the ODE system.

3.5.2. Newton-Newton method

This combination of numerical methods uses the Newton method described in section 3.3.2 for the system of ODE and the Newton method described in section 3.4.1 for the system of NAE.

The combined numerical method for the system of DAE
The recursion equation for the ODE and the recursion equation for the NAE are connected to each other. The source term of the system of ODE and the source term of the system of NAE both depend on the concentration vector of the algebraic species, $\mathbf{x}_A$, as well as the concentration vector of the differential species, $\mathbf{x}_D$.
For each iteration step of the recursion equation for the system of ODE, the recursion equation for the system of NAE is iterated until convergence and the value of the index *j* is then *j final*. The concentration-vector corresponding to *j final*, $\mathbf{x}_{A,i}^{j\,final}(t_{n+1})$, is then inserted into the recursion equation for the system of ODE. The recursion equation for the system of ODE is then iterated one step and the new value, $\mathbf{x}_D^{i+1}(t_{n+1})$, is then inserted into the recursion equation for the system of NAE, which are iterated until convergence again.

This procedure is repeated until the recursion equation for the system of ODE converge and reaches the final value, $\mathbf{x}_D^{i\,final}(t_{n+1})$. Thereafter the predictor for the system of ODE is calculated for the new time step and the whole procedure described above is done all over again for the new time step. In the following discussion $\mathbf{x}_D^i = \mathbf{x}_D^i(t_{n+1})$ and $\mathbf{x}_{A,i}^j = \mathbf{x}_{A,i}^j(t_{n+1})$ is used in order to simplify the equations. The recursion equation for the ODE system is;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i - \frac{\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j\,final})}{\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k}^{j\,final})} \tag{3.87}$$

The recursion equation for the NAE system is;

$$\mathbf{x}_{A,i+1}^{j+1} = \mathbf{x}_{A,i+1}^j - \frac{\mathbf{g}_A(\mathbf{x}_D^i, \mathbf{x}_{A,i}^j)}{\mathbf{J}_{A,i}^{j-m}(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j-m})} \tag{3.88}$$

For practical reasons we do not want to invert the Jacobians $\boldsymbol{J}_{\mathrm{D}}$ **and** $\mathbf{J}_{\mathrm{A}}$. Instead the linear systems are solved after the Jacobians has been decomposed via GE;

$$\mathbf{J}_D^{i-k}(\mathbf{x}_D^{i-k}, \mathbf{x}_{A,i-k}^{j\,final}) \cdot \Delta\mathbf{x}_D^i = -\mathbf{g}_D(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j\,final}) \tag{3.89}$$

$$\mathbf{J}_{A,i}^{j-m}(\mathbf{x}_D^i, \mathbf{x}_{A,i}^{j-m}) \cdot \Delta\mathbf{x}_A^j = -\mathbf{f}_A(\mathbf{x}_A^j, \mathbf{x}_D^i) \tag{3.90}$$

and the $\mathbf{x}_{\mathrm{D}}$ and $\mathbf{x}_{\mathrm{A}}$ are updated according to;

$$\mathbf{x}_D^{i+1} = \mathbf{x}_D^i + \Delta\mathbf{x}_D^i \tag{3.91}$$

and

$$\mathbf{x}_A^{j+1} = \mathbf{x}_A^j + \Delta\mathbf{x}_A^j \tag{3.92}$$

A schematic illustration of the Newton-Newton method is shown in Figure 3.17.

In order to get the first iterated value, $\mathbf{x}_D^1(t_{n+1})$, in the recursion equation (3.87) for the system of ODE at time step, t_{n+1}, the predicted value, $\mathbf{x}_D^*(t_{n+1})$, is used as a starting value of $\mathbf{x}_{\mathrm{D}}$. $\mathbf{g}_{\mathrm{D}}$ and $\boldsymbol{J}_{\boldsymbol{D}}$ are functions of $\mathbf{x}_D^*(t_{n+1})$ and the starting value of $\mathbf{x}_{\mathrm{A}}$, which is the last value of the previous full algebraic iteration, $\mathbf{x}_{A,*}^{j\,final}(t_{n+1})$. Hence,

$$\mathbf{x}_D^1 = \mathbf{x}_D^* - \frac{\mathbf{g}_D(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final})}{\mathbf{J}_D(\mathbf{x}_D^*, \mathbf{x}_{A,*}^{j\,final})} \tag{3.93}$$

The first value of $\mathbf{x}_A$ using the predictor as $\mathbf{x}_D$ and the final value $\mathbf{x}_A$ from the previous time step, t_n, is;

$$\mathbf{x}_{A,1}^{1}(t_{n+1}) = \mathbf{x}_{A,i\ final}^{j\ final}(t_n) - \frac{\mathbf{g}_A(\mathbf{x}_D^*(t_{n+1}), \mathbf{x}_{A,*}^{j\ final}(t_{n+1}))}{\mathbf{J}_A(\mathbf{x}_D^*(t_{n+1}), \mathbf{x}_{A,*}^{j\ final}(t_{n+1}))} \tag{3.94}$$

The final value of $\mathbf{x}_A$ using the predictor as $\mathbf{x}_D$ is;

$$\mathbf{x}_{A,*}^{j\ final}(t_{n+1}) = \mathbf{x}_{A,*}^{j\ final-1}(t_{n+1}) - \frac{\mathbf{g}_A(\mathbf{x}_{A,*}^{j\ final-1}(t_{n+1}), \mathbf{x}_D^*(t_{n+1}))}{\mathbf{J}_A(\mathbf{x}_{A,*}^{j\ final-1}(t_{n+1}), \mathbf{x}_D^*(t_{n+1}))} \tag{3.95}$$

The final value of $\mathbf{x}_D$ using the final value of $\mathbf{x}_A$ is;

$$\mathbf{x}_D^{i\ final}(t_{n+1}) = \mathbf{x}_D^{i\ final-1}(t_{n+1}) - \frac{\mathbf{g}_D(\mathbf{x}_D^{i\ final-1}(t_{n+1}), \mathbf{x}_{A,i}^{j\ final}(t_{n+1}))}{\mathbf{J}_D(\mathbf{x}_D^{i\ final-1}(t_{n+1}), \mathbf{x}_{A,i}^{j\ final}(t_{n+1}))} \tag{3.96}$$

.

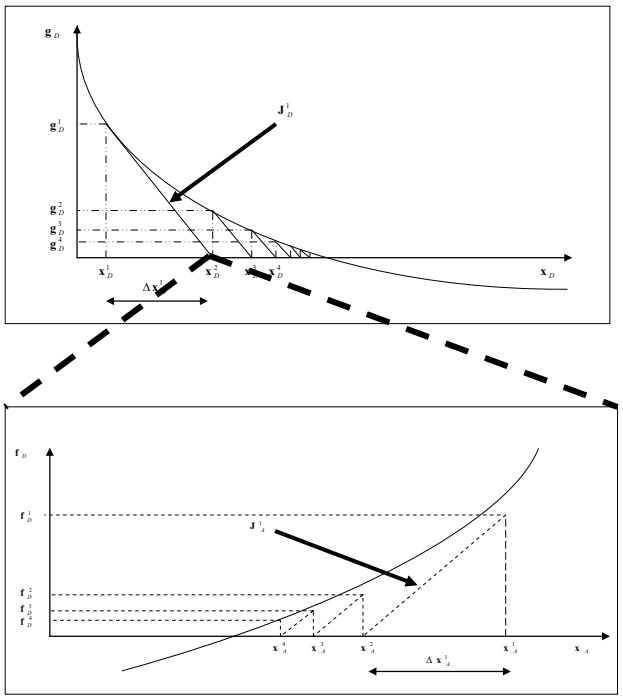


Figure 3.17. The "Newton-Newton" method. Above is a schematic illustration, which is the same as Figure 3.5, of the Newton-iteration for the system of ODE. Below is a schematic illustration, which is the same as Figure 3.11, of the Newton-iteration for the system of NAE. The Newton-iteration for the system of NAE is iterated until convergence at each iteration step of the Newton iteration for the system of ODE.

As stated previously, the Newton method for the system of NAE is iterated until convergence at each iteration step of the Newton method for the system of ODE. In addition to that, the Newton method for the system of NAE is iterated until convergence for each element in $\mathbf{x}_D$ when the Jacobian of the system of ODE is built. This is illustrated in Figure 3.18 and discussed in detail in section 3.3.4.1.

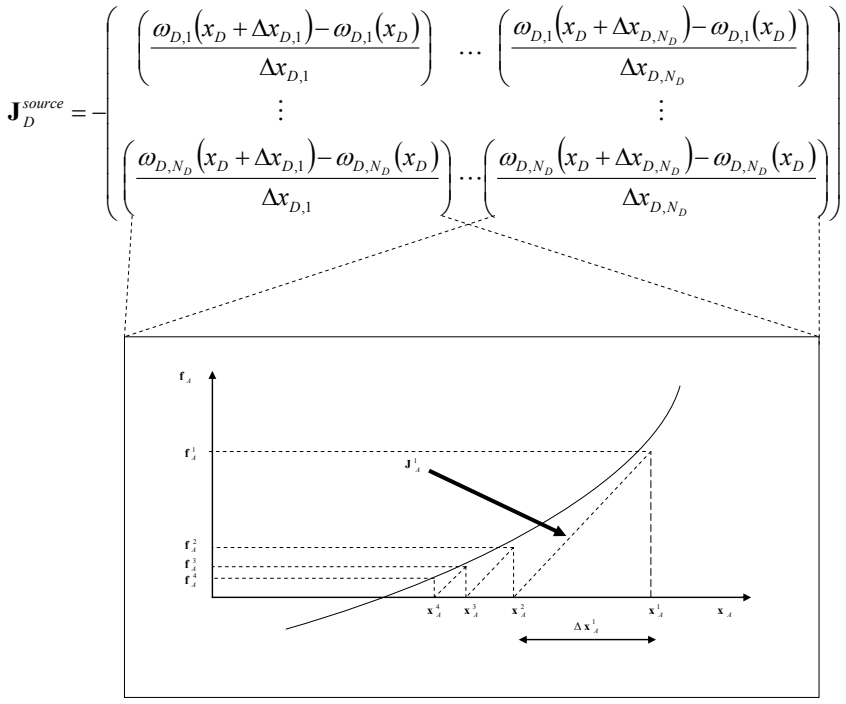


Figure 3.18. The figure illustrates that a full Newton iteration for the system of NAE, which is the same as Figure 3.11, is performed for each element in $\mathbf{x}_D$ when the Jacobian for the system of ODE is built. This results in N_D full iteration for the system of NAE. N_D is the number of species in the ODE system.

3.5.2.1 Newton-Newton method with Operator Splitting

An operator splitting method, with the same outer and inner solvers as in the Newton-Newton method, has also been tested. However, the coupling between the methods separates the operator splitting method from the Newton-Newton method. Instead of a full iteration of the inner solver for each iteration step of the outer solver, the operator splitting method only makes one iteration step for the inner solver for each iteration step of the outer solver. Hence, the operator splitting method uses less iteration steps for the system of NAE than the Newton-Newton method.

However, this affected the accuracy of the QSS species and thereby the convergence of the operator splitting method. This forced the operator splitting method to use small time steps in order to converge, which lead to long CPU times. Also, some species profiles were not as accurate as expected. The results from the operator splitting method are not presented in this thesis due to the poor results.

3.5.3. CPU cost for the Newton-Newton method and the Newton-FP method

The section aims at making a simple estimate of the CPU cost for the Newton-Newton method and the Newton-FP method.

Newton-Newton solver
The total CPU time for the Newton-Newton method depends on both the numerical method for the system of ODE and the numerical method for the system of NAE. The solvers associated with the numerical method for the system of ODE and NAE are from now on referred to as the outer and inner solver respectively. A simple estimate of the total CPU time needed for the simulation of a combustion system can be given;

$$CPU_{total}\left(N_D, N_A\right) = \sum_{k=1}^{N_{TS}} CTS_k\left(N_D, N_A\right) + \beta_k\left(N_D, N_A\right) \tag{3.97}$$

where N_D and N_A are the number of species for the system of ODE and NAE respectively. The index k runs over the time steps and N_{TS} is the number of time steps in the simulation. $\beta_k\left(N_D, N_A\right)$ is the CPU time at time step k that is not directly associated with the solver.
CTS_k is the CPU time per time step and can be expressed as follows:

$$CTS_k\left(N_D, N_A\right) = \sum_{i=1}^{N_{OI}^k} \left[COI_i^k\left(N_D\right) + \left(\sum_{j=1}^{N_{II}^{k,i}} CII_j^{k,i}\left(N_A\right) \right)_i \right]_{k,i} \tag{3.98}$$

Each time step contains a full Newton iteration of the outer solver. The index i runs over the outer iterations and N_{OI}^k is the number of outer iteration at time step k. $COI_i^k\left(N_D\right)$ is the CPU time for the outer solver at iteration step i and time step k.
Each outer iteration step contains a full Newton iteration of the inner solver. Hence, the index j runs over the inner iterations and $N_{II}^{k,i}$ is the number of inner iterations at outer iteration step i and time step k. $N_{II}^{k,i}$ is the sum of two parts. One part is associated with the building of the Jacobian for the outer solver and the other part is associated with the BS of the outer solver.

$CII_j^{k,i}(N_A)$ is the CPU time for the inner solver at iteration step *j* of the inner solver and at iteration step *i* of the outer solver and time step *k*.
$COI_i^k(N_D)$ and $CII_j^{k,i}(N_A)$ are in turn expressed as;

$$COI_i^k(N_D) = \left(P_{Di} \cdot \left(A \cdot N_D^3\right) + \left(B \cdot N_D^2\right)\right)_i^k \tag{3.99}$$

and

$$CII_j^{k,i}(N_A) = \left(P_{A,j}^i \cdot \left(A \cdot N_A^3\right) + \left(B \cdot N_A^2\right)\right)_j^{k,i} \tag{3.100}$$

where P_{Di} is the probability that a Jacobian is built and decomposed for the system of ODE at outer iteration step *i* and $P_{A,j}^i$ is the probability that a Jacobian is built and decomposed for the system of NAE at outer iteration step *i* and inner iteration step *j*. P_{Di} and $P_{A,j}^i$ are either zero or one for the particular iteration step. P_{Di} and $P_{A,j}^i$ are in reality determined by the convergence of the outer and inner solvers respectively. Good convergence means low probability and vise versa.

The probability to build and decompose new Jacobians, i.e. P_D and P_A, varies during the simulation of the combustion process. They are very low far away from the ignition point but increases rapidly when approaching the onset of ignition, since both solvers must compensate for the convergence problems caused by steep concentration and temperature gradients around this point. However, since the ignition region is very narrow in time and small compared to the entire simulation, most of the CPU time still comes from the BS, which suggests that both the inner and outer Newton iterations approximately scales as the number of species squared.

The terms $\left(A \cdot N_D^3\right)$ and $\left(B \cdot N_D^2\right)$ are the CPU times for the GE and BS for the system of ODE, while $\left(A \cdot N_A^3\right)$ and $\left(B \cdot N_A^2\right)$ are the CPU times for the GE and BS for the system of NAE. The constants *A* and *B* can be determined exactly using well known mathematical procedures [3], but the approximate values $A \approx 2/3$, and $B \approx 1$ are used in this thesis. Hence, the $CTS_k(N_D, N_A)$ for the Newton-Newton solver is;

$$CTS_k(N_D, N_A) = \sum_{i=1}^{N_{OI}^k} \left[P_{Di} \cdot \left(A \cdot N_D^3\right) + \left(B \cdot N_D^2\right) + \sum_{j=1}^{N_{II}^{k,i}} \left(P_{A,j}^i \cdot \left(A \cdot N_A^3\right) + \left(B \cdot N_A^2\right)\right)_j \right]_{k,i} \tag{3.101}$$

For each particular iteration step of the outer solver, many inner iteration steps are performed. Hence, the total of the number of iteration steps of the inner solver is much larger than the total number of outer iteration steps, which motivates the effort to speed up the inner solver in order to decrease the total CPU time. In symbol language a speed

up, which comes from an optimization algorithm, of the Newton-Newton solver would be expressed as;

$$CTS_k\left(N_D,N_A\right)=\sum_{i=1}^{N_{OI}^k}\left[P_{Di}\cdot\left(A\cdot N_D^3\right)+\left(B\cdot N_D^2\right)+\alpha\left(N_A\right)\cdot\sum_{j=1}^{N_{II}^{k,i}}\left(P_{A,j}^i\cdot\left(A\cdot N_A^3\right)+\left(B\cdot N_A^2\right)\right)_j\right]_{k,i} \tag{3.102}$$

which is the same as eq(3.101) except from $\alpha\left(N_A\right)$. The task is to make $\alpha\left(N_A\right)$, which takes values between zero and one, as small as possible. If $\alpha\left(N_A\right)=1$ eq(3.102) equals eq(3.101), which symbolizes no speed up. If $\alpha\left(N_A\right)=0$, eq(3.102) symbolizes an hypothetical infinitely fast inner solver. The speed up of the inner solver is discussed in detail in section 4.3.3.

Hypothetical Newton-Newton solver

Using the assumption that the inner solver is infinitely fast and that few Jacobians are built and decomposed during a simulation, the CPU time scales approximately quadratically with N_D. Hence, the CPU time per time step an hypothetical infinitely fast inner solver can be approximated by;

$$CTS_k\left(N_D,N_A\right)\approx\sum_{i=1}^{N_{OI}^k}\left[\left(B\cdot N_D^2\right)\right]_{k,i} \tag{3.103}$$

The total CPU time for such a solver would be;

$$CPU_{total}\left(N_D\right)\approx\sum_{k=1}^{N_{TS}}\left(\sum_{i=1}^{N_{OI}^k}\left[\left(B\cdot N_D^2\right)\right]_{k,i}+\beta\left(N_D\right)\right)\approx N_{TS}\cdot N_{OI}^{AV}\cdot\left(B\cdot N_D^2\right) \tag{3.104}$$

Assuming that $\beta\left(N_D\right)$ is small enough to be approximated by zero and N_{OI}^{AV} is the average number of outer iteration per time step.

$$CPU_{NORM,total}^{Approx}\left(N_D\right)=\frac{CPU_{total}^{Approx}\left(N_D\right)}{CPU_{total}^{Approx}\left(N_{D,\max}\right)}=\frac{N_{TS}\cdot N_{OI}^{AV}\cdot\left(B\cdot N_D^2\right)}{N_{TS}\cdot N_{OI}^{AV}\cdot\left(B\cdot N_{D,Max}^2\right)}=\frac{N_D^2}{N_{D,Max}^2} \tag{3.105}$$

Where $N_{D,\max}$ is the maximum number of species for the system of ODE, which corresponds to zero QSS species. The normalized simulated CPU time of the Newton-Newton solver will be compared to $CPU_{NORM,total}^{Approx}\left(N_D\right)$ in section 6.2.1.3. The $CPU_{NORM,total}^{Approx}\left(N_D\right)$ serves as the lower limit in CPU time the can be achieved with a Newton-Newton solver.

Newton-FP solver

The CPU time for a FP solver scales linearly with N_A with the constant of proportionality C_{FP}. Hence;

$$CII_j^{k,i}\left(N_A\right)=\left(C_{FP}\cdot N_A\right)_j^{i,k} \tag{3.106}$$

Hence, the $CTS_k\left(N_D,N_A\right)$ for the Newton-FP solver is;

$$CTS_k\left(N_D,N_A\right)=\sum_{i=1}^{N_{QI}^k}\left[P_{Di}\cdot\left(A\cdot N_D^3\right)+\left(B\cdot N_D^2\right)+\sum_{j=1}^{N_{II}^{k,i}}\left(C_{FP}\cdot N_A\right)_j\right]_{k,i} \tag{3.107}$$

As a first thought, one would expect the Newton-FP method, which contains a numerical method for the NAE that scales linearly with the number of NAE, to be faster than a Newton-Newton method, which contains a numerical method for the NAE that scales quadratically or cubically with the number of NAE.
However, the accuracy of the solution from the numerical method for the NAE affects the convergence and CPU time of the numerical method for the ODE. In order to reach convergence and thereby high enough accuracy of the QSS species concentrations the total number of inner iterations may need to be much larger for the FP solver than for the inner Newton solver. The reason for this is the lack of gradient information in the FP method. This interaction between the numerical methods, which is discussed in detail in section 3.5, may result in the Newton-Newton method being faster than the Newton-FP method. The results in section 6.2.1.1.2 show that this indeed is the case.

3.6. Chapter References

[1] S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in C*, Second edition, William H Press (1999)
[2] J. Warnatz, U. Maas, R.W. Dibble, *Combustion,* 4th edition*,* Springer (2006)
[3] L. Elden, L. Wittmeyer-Koch, *Numeriska beräkningar*, Studentlitteratur (2001)
[4] S.H. Strogatz, *Nonlinear Dynamics and Chaos*, Perseus books (1994)
[5] T. Lu, C.K. Law, *Toward accommodating realistic fuel chemistry in large-scale computations*, Progress in Energy and Combustion Science 35 (2009) 192-215
[6] K.G. Andersson, *Lineär algebra*, Studentlitteratur (2000)
[7] R. Bellanca, *BlueBellMouse A tool for kinetic model development*, Doctoral Thesis, Combustion Physics, Lund University, Sweden, (2004), Lunds reports on Physics, LRCP-92 ISRN LUTFD2/TFCP—92—SE ISSN 1102-8718
[8] F. Schmitt, *Motorinterne Masnahmen zur Stickstoffoxireduction im Dieselmotor*. University Kaiserslautern Fachbereich Maschinenwesen (1993).

3.7. Chapter Appendix

A.3.1. Examples of the explicit and implicit Euler method

To see the difference between explicit and implicit methods we solve the linear test equation (with $\lambda > 0$);

$$\dot{y} = -\lambda y \tag{A.3.1}$$

with the explicit Euler method and the implicit Euler method.

Explicit Euler
The explicit Euler method gives;

$$y_{n+1} = y_n + y'_n \cdot h = (1 - \lambda h) y_n \tag{A.3.2}$$

where y_n is the present function value, y_{n+1} is the new function value, y' is the derivative at the present function value and *h* is the step length.
The condition for this method to be stable is $|(1 - \lambda h)| < 1$ which gives that $|y_n| \to 0$ when $n \to \infty$ if the step size is;

$$h < \frac{2}{\lambda} \tag{A.3.3}$$

Implicit Euler
The implicit Euler method gives;

$$y_{n+1} = y_n + y'_{n+1} \cdot h = \frac{y_n}{(1 + \lambda h)} \tag{A.3.4}$$

Where y_n is the present function value, y_{n+1} is the new function value, y'_{n+1} is the derivative at the new function value and *h* is the step length.
The condition for this method to be stable is $\frac{1}{(1 + \lambda h)} < 1$ which gives that $|y_n| \to 0$ when $n \to \infty$ for any step size.

A.3.2. Estimation of the condition number

If *e* is the error in *b*, the error in the solution $A^{-1}b$ is $A^{-1}e$. The ratio of the relative error in the solution and the relative error in b is;

$$\frac{\left\|A^{-1}e\right\|/\left\|A^{-1}b\right\|}{\|e\|/\|b\|} \tag{A.3.5}$$

which can be written as

$$\left(\frac{\left\|A^{-1}e\right\|}{\|e\|}\right)\cdot\left(\frac{\|b\|}{\left\|A^{-1}b\right\|}\right) \tag{A.3.6}$$

The maximum value is

$$\kappa(A)=\left\|A^{-1}\right\|\cdot\|A\| \tag{A.3.7}$$

If A is normal then;

$$\kappa(A)=\left|\frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}\right| \tag{A.3.8}$$

where $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ are the maximum and minimum eigenvalues of A respectively.

A.3.3. Gear's Backward Differentiation Formulas (BDFs).

The Taylor theorem is used to estimate $\mathbf{x}(t_n)$, $\mathbf{x}(t_{n-1})$, $\mathbf{x}(t_{n-2})$ and so on. The meaning of the different Δt_i is explained in Figure 3.7. The Taylor series are expanded at $\mathbf{x}(t_{n+1})$. $\mathbf{x}(t_n)$ can be expressed as [8];

$$\mathbf{x}(t_n) = \mathbf{x}(t_{n+1}) - \frac{d\mathbf{x}(t_{n+1})}{dt}\Delta t_1 + \frac{d^2\mathbf{x}(t_{n+1})}{dt^2}\frac{\Delta t_1^2}{2} - \frac{d^3\mathbf{x}(t_{n+1})}{dt^3}\frac{\Delta t_1^3}{6} + \frac{d^4\mathbf{x}(t_{n+1})}{dt^4}\frac{\Delta t_1^4}{24} - ... + ... \quad \text{(A.3.9)}$$

$\mathbf{x}(t_{n-1})$ can be expressed as;

$$\mathbf{x}(t_{n-1}) = \mathbf{x}(t_{n+1}) - \frac{d\mathbf{x}(t_{n+1})}{dt}\Delta t_2 + \frac{d^2\mathbf{x}(t_{n+1})}{dt^2}\frac{\Delta t_2^2}{2} - \frac{d^3\mathbf{x}(t_{n+1})}{dt^3}\frac{\Delta t_2^3}{6} + \frac{d^4\mathbf{x}(t_{n+1})}{dt^4}\frac{\Delta t_2^4}{24} - ... + ... \quad \text{(A.3.10)}$$

$\mathbf{x}(t_{n-2})$ can be expressed as;

$$\mathbf{x}(t_{n-2}) = \mathbf{x}(t_{n+1}) - \frac{d\mathbf{x}(t_{n+1})}{dt}\Delta t_3 + \frac{d^2\mathbf{x}(t_{n+1})}{dt^2}\frac{\Delta t_3^2}{2} - \frac{d^3\mathbf{x}(t_{n+1})}{dt^3}\frac{\Delta t_3^3}{6} + \frac{d^4\mathbf{x}(t_{n+1})}{dt^4}\frac{\Delta t_3^4}{24} - ... + ... \quad \text{(A.3.11)}$$

The time derivative is at $t=t_{n+1}$ is approximated by;

$$\frac{d\mathbf{x}(t_{n+1})}{dt} = \frac{\mathbf{x}(t_{n+1}) - \mathbf{x}(t_n)}{\Delta t_1} + \left[\frac{d^2\mathbf{x}(t_{n+1})}{dt^2}\frac{\Delta t_1}{2} + ...\right] \quad \text{(A.3.12)}$$

Insertion of $\frac{d^2\mathbf{x}(t_{n+1})}{dt^2}$ from equation (A.3.10) gives;

$$\frac{d\mathbf{x}(t_{n+1})}{dt} = \mathbf{x}(t_{n-1})\frac{\Delta t_1}{\Delta t_2(\Delta t_2 - \Delta t_1)} - \mathbf{x}(t_n)\frac{\Delta t_2}{\Delta t_1(\Delta t_2 - \Delta t_1)} + \mathbf{x}(t_{n+1})\frac{\Delta t_1 + \Delta t_2}{\Delta t_2 \Delta t_1} + \left[\frac{d^3\mathbf{x}(t_{n+1})}{dt^3}\frac{\Delta t_1 \Delta t_2}{6} + ...\right] \quad \text{(A.3.13)}$$

Insertion of $\frac{d^3\mathbf{x}(t_{n+1})}{dt^3}$ from equation (A.3.11) gives;

$$\frac{d\mathbf{x}(t_{n+1})}{dt} = \mathbf{x}(t_{n-2})\frac{\Delta t_1 \Delta t_2}{\Delta t_3(\Delta t_3 - \Delta t_1)(\Delta t_3 - \Delta t_2)} + \mathbf{x}(t_{n-1})\frac{\Delta t_1 \Delta t_3}{\Delta t_2(\Delta t_2 - \Delta t_1)(\Delta t_3 - \Delta t_2)}$$
$$- \mathbf{x}(t_n)\frac{\Delta t_2 \Delta t_3}{\Delta t_1(\Delta t_2 - \Delta t_1)(\Delta t_3 - \Delta t_1)} + \mathbf{x}(t_{n+1})\frac{\Delta t_1 \Delta t_2 + \Delta t_1 \Delta t_3 + \Delta t_2 \Delta t_3}{\Delta t_2 \Delta t_1 \Delta t_3} + \quad (A.3.14)$$
$$+ \left[\frac{d^4\mathbf{x}(t_{n+1})}{dt^4}\frac{\Delta t_1 \Delta t_2 \Delta t_3}{24} + \ldots\right]$$

This is the difference equation for the time derivative, which is substituted for $\frac{d\mathbf{x}(t_{n+1})}{dt}$ in equation (3.12). The equation contains an error of the order $\left[\frac{d^4\mathbf{x}(t_{n+1})}{dt^4}\frac{\Delta t_1 \Delta t_2 \Delta t_3}{24} + \ldots\right]$. The error is dependent of the order of the Taylor expansion.
The difference equation for the time derivative contains the value for $\mathbf{x}(t_{n+1})$, which will be found by the corrector. The predicted value of the **x**-vector, $\mathbf{x}^*(t_{n+1})$, must be inserted in order to get the first value for difference approximation for $\frac{d\mathbf{x}(t_{n+1})}{dt}$.

The predictor of the **x**-vector is found from equation (A.3.9); is approximated by;

$$\mathbf{x}^*(t_{n+1}) = \mathbf{x}(t_n) + \left[\frac{d\mathbf{x}(t_{n+1})}{dt}\Delta t_1 + \ldots\right] \quad (A.3.15)$$

Using equation (A.3.10) $\frac{d\mathbf{x}(t_{n+1})}{dt}$ is eliminated;

$$\mathbf{x}^*(t_{n+1}) = \mathbf{x}(t_n) + \frac{\left(\mathbf{x}(t_n) - \mathbf{x}(t_{n-1})\right)}{\Delta t_2 - \Delta t_1}\Delta t_1 + \left[\frac{d^2\mathbf{x}(t_{n+1})}{dt^2}\frac{\Delta t_1 \Delta t_2}{2} + \ldots\right] \quad (A.3.16)$$

Using equation (A.3.11), $\frac{d^2\mathbf{x}(t_{n+1})}{dt^2}$ is eliminated, which gives the predictor for the **x**-vector;

$$\mathbf{x}^*(t_{n+1}) = \mathbf{x}(t_n)\frac{\Delta t_2 \Delta t_3}{(\Delta t_1 - \Delta t_2)(\Delta t_1 - \Delta t_3)} + \mathbf{x}(t_{n-1})\frac{\Delta t_1 \Delta t_3}{(\Delta t_2 - \Delta t_1)(\Delta t_2 - \Delta t_3)}$$
$$+ \mathbf{x}(t_{n-2})\frac{\Delta t_1 \Delta t_2}{(\Delta t_3 - \Delta t_1)(\Delta t_3 - \Delta t_2)} + \left[\frac{d^3\mathbf{x}(t_{n+1})}{dt^3}\frac{\Delta t_1 \Delta t_2 \Delta t_3}{6} + \ldots\right] \quad (A.3.17)$$

This expression is inserted into equation (A.3.14) to give the first approximation for difference equation of the time derivative.

Chapter 4.

Solver Combinations

4.1. An overview of the solver combinations

The solver combinations used in this thesis are the Newton-Newton and Newton-FP solver combinations. The Newton-Newton and Newton-FP methods are described in section 3.5.
The solver combinations use the following procedure;

1. Read input
2. Solve the equations
3. Produce output

4.1.1. Input

The input that the solvers in this thesis use are;

- initial conditions
  - initial species concentrations
  - Initial temperature
  - Initial pressure
  - Initial mixture fractions
  - Initial fuel/air ratio
- solver settings
  - absolute tolerances
  - relative tolerances
  - initial time step size

4.1.2. Output

The output that the solvers in this thesis produce are;

- Species concentration profiles
- Temperature profiles
- Pressure profiles
- CPU time
- Ignition delay time
- Solver information

4.1.3. Solver structure

The solver combinations consist of;

- A time loop
- An iteration loop to solve the system of equations at each time step

The solver structure to prefer depends on the system that is to be investigated and the problem to be solved. If the system of equations consists of one system of ODE and one system of NAE, there are two possibilities.
One possibility is that at each time step an entire iteration loop is performed for the system of ODE and an entire iteration loop is performed for the system of NAE. This approach has a low computational cost but also provides a solution with poor accuracy. Sometimes the accuracy from the system of NAE is so poor that the system does not converge at all with this approach.

Another possibility is that at each time step an entire iteration loop is performed for the system of ODE. At each iteration step of the solver for the system of ODE, an entire iteration loop of the solver for the system of NAE is performed. This is the solver structure used in this thesis. This approach has a high computational cost but also provides a solution with high accuracy.
Modifications of the inner solver in order to decrease the computational cost are discussed in detail in section 4.3.3. The high accuracy can actually reduced the overall computational cost. This is discussed in detail in section 3.3.4.1.1 and section 6.2.1.2.

4.1.4. Simplified flow chart for the solver combinations for the system of DAE

A simplified flow chart for solver combinations with adaptive step size can be seen in Figure 4.2. The action for each node is explained in Figure 4.1.

0) Inputs for the solver
1) Choose a new time step size. Go to 2.
2) Solve the combined system of DAE. Go to 3.
3) Did the solver for the ODE converge? If yes, go to 4. If no, go to 5.
4) The solver for the ODE did converge. Is this the last time step? If yes, go to 7. If no, go to 1.
5) The solver for the ODE did not converge. Decrease the time step? If yes, go to 6. If no, go to 2.
6) Decrease the time step. Go to 2.
7) Print the output

Figure 4.1. Explanation of the function of each node in figure 4.2.

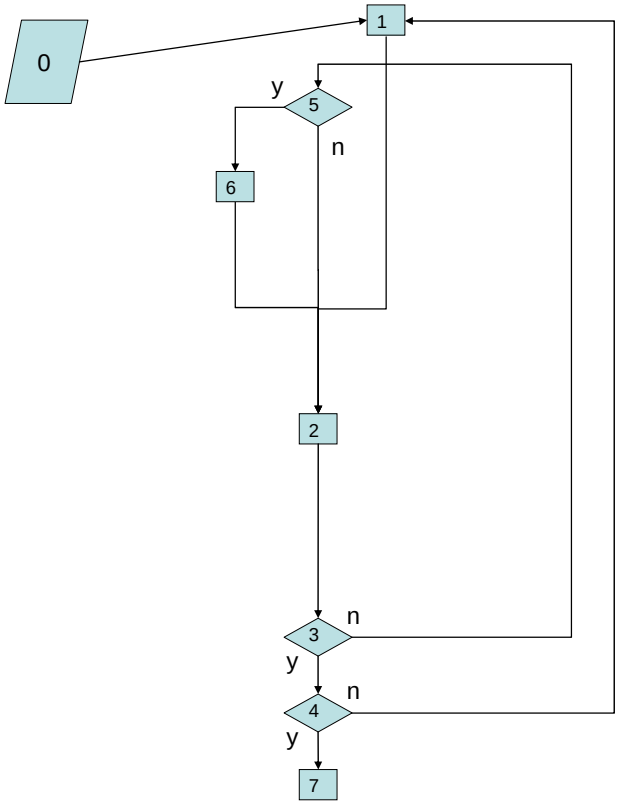


Figure 4.2. Flow chart of the entire solver, which excludes the sub flow chart for the solver of the system of ODE and the sub flow chart of the solver for the system of NAE. The meaning of each node is explained in figure 4.1.

4.2. The Newton solver for the system of ODE

There are two main paths for the Newton solver, depending on if the old Jacobian can be reused or if a new Jacobian must be built. If a new Jacobian must be built the operations described in the “Full outer iteration step” must be performed.

Full outer iteration step

- Build the Jacobian
- Decompose the Jacobian (GE)
- Solve the system of NAE
- Update the source-vector
- Solve the linear system (BS)

If the Jacobian from the previous iteration step can be reused, the operations described in the “Simple outer iteration step” must be performed.

Simple outer iteration step

- Solve the system of NAE
- Update the source-vector
- Solve the linear system (BS)

A typical full outer Newton iteration for a simulation of a CVR involves few full iteration steps and many simple iteration steps until convergence is reached. The number of full iteration steps that is used depends on the point in time in the simulation. The number of full iteration steps is higher close to the ignition point where the partial derivatives changes faster and lower far away from the ignition point where the partial derivatives changes slower.
Simple iteration steps saves much CPU time, since building and decomposing the Jacobian is expensive compared to the other operations. The cost of each operation is described below.

Building the Jacobian
Each element in the Jacobian is found from a finite difference of the source-vector, see section 3.3.4.1. This means that the source-vector must be calculated at another point in concentration-space and as a consequence the system of NAE must be solved. When the concentrations for the system of ODE are changed the concentrations for the system of NAE must be changed as well, since they are coupled via the chemical reactions. This means that the system of NAE must be solved once for each species when the Jacobian is built. Hence, the system of NAE is solved n times if the system of ODE consists of n species.

Decomposing the Jacobian (GE)
Decomposing the Jacobian is expensive, since the cost scales roughly as $(2/3)* n^3$ (see section 3.3.4.2.).

Solve the system of NAE
In order to update the source-vector the system of NAE must be solved. This happens at each iteration step of the Newton solver for the system of ODE. Also, the source vector must be calculated at $\boldsymbol{x}+\Delta\boldsymbol{x}$ every time a new Jacobian is built from a finite difference. Hence, the system of NAE must be solved and the source vector must be updated for each dimension of the Jacobian. The cost each time the solver for the system of NAE is called can be found in section 3.5.3.

Update the source-vector
The source-vector must be updated before the linear system can be solved. The source - vector must undergo the same row subtractions as the Jacobian does during the GE, which suggests that the cost scales roughly as n^2

Solve the linear system (BS)
The concentration-vector is found from BS, which is discussed in section 3.3.4.3. The cost scales roughly as n^2.

4.2.1. Convergence

The speed and accuracy of the solver combination is affected by the solver settings of both the inner and outer solver. This is discussed in detail in section 6.2.1.2. Both the inner and outer solver has an *Absolute tolerance* and a *Relative tolerance*. Each solver is said to have converged if;

$$X_NormMax < \mathrm{Re}\,lative\,tolerance \tag{4.1}$$

$X_NormMax$ is the maximum value of all X_Norm_i, which is defined as:

$$X_Norm_i = \left| \frac{\Delta x_i}{x_i + Absolute\,tolerance} \right|, i = 1,...n \tag{4.2}$$

Where x_i is the concentration of species i, Δx_i is the difference in x_i between two consecutive iteration steps and n is the number of equations in the solver. The inequality basically says that the solver has converged when the species with the largest relative change in concentration is less than the relative tolerance.
The *Absolute tolerance* decides the concentration sizes, and thereby the species, that are allowed to influence the convergence. If the *Absolute tolerance* is small the small species concentrations will influence the X_Norm_i, which means that the species with small concentrations and small Δx_i can determine if the solver converged or not. Contrary, if the *Absolute tolerance* is large the small species concentrations will not influence the X_Norm_i.

The values of the absolute tolerance and relative tolerance for the outer solver are;

- Absolute tolerance=$1*10^{-10}$
- Relative tolerance=$1*10^{-6}$

Theses values have been chosen from previous experience at the division of Combustion Physics at Lund University and not varied in this thesis.

4.2.2. Flow chart for the Newton solver for the system of ODE

A sub flow chart for the Newton solver for the system of ODE can be seen in Figure 4.4. The action for each node is explained in Figure 4.3.
The solver for the system of NAE, i.e. nr 4, can be replaced with the sub flow chart in Figure 4.6. If this replacement is done, the whole sub flow chart replaces the entire solver combination for the DAE system, which in turn replaces nr 2, in Figure 4.2.

1) Run the Newton solver for the system of ODE with or without building a new Jacobian. If a new Jacobian must be built, go to 2. If not, go to 4.
2) Build a new Jacobian for the system of ODE. In order to do that the source vector must be calculated at $\boldsymbol{x}+\Delta\boldsymbol{x}$. The system of NAE must be solved and the source vector must be updated for each dimension of the Jacobian. Go to 4.
3) Decompose the Jacobian. Go to 4.
4) Solve the system of NAE. Go to 5.
5) Update the source vector. IF within build Jacobian loop: Go to 6. ELSE Go to 7.
6) Have all dimensions in the Jacobian been updated? If yes, go to 3. If no, go to 2.
7) Solve the linear system for the ODE. Go to 8.
8) Did the ODE solver converge or not? If yes, exit ODE flow chart, if no, go to 4 or 9.
9) The ODE solver did not converge. Does a new Jacobian need to be built? If yes, go to 2. If no, go to 4.

Figure 4.3. Explanation of the function of each node in Figure 4.4.

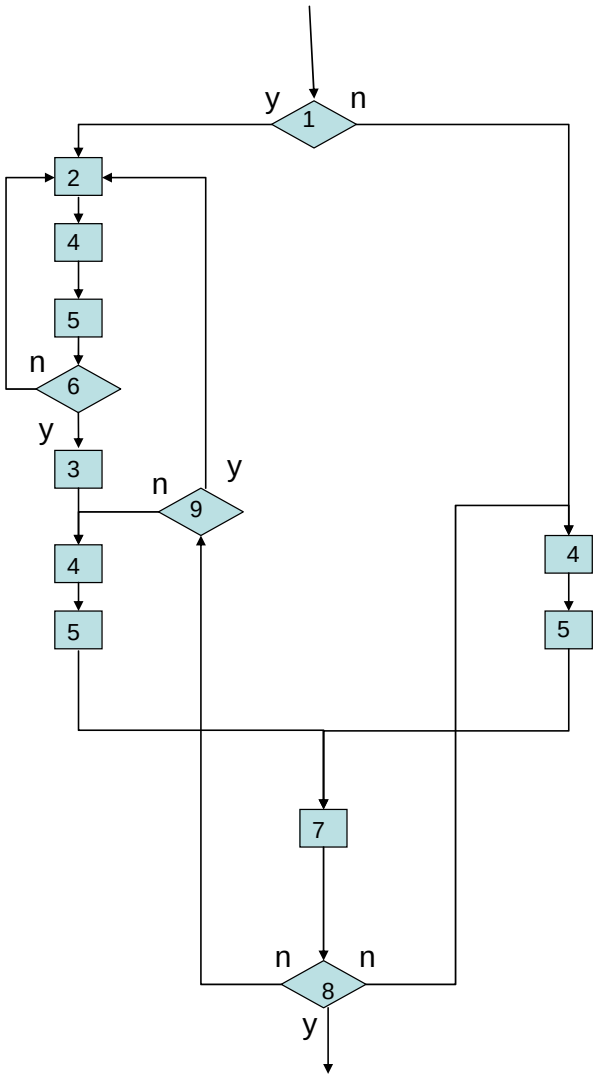


Figure 4.4. Flow chart of the solver for the system of ODE. This flow chart is a sub flow chart of the entire solver. The meaning of each node is explained in Figure 4.3.

4.3. The solver for the system of NAE

The two combined numerical methods that were presented in section 3.4 were the Newton-Newton method and the Newton-FP method. The two methods differ because of the inner numerical method/solver. This section describes both inner solvers.
Also, the modifications made in order to speed up the Newton-Newton method are described.

4.3.1. The Newton solver for the system of NAE

The Newton solver for the system of NAE is very similar to the Newton solver for the system of ODE. However there are some differences. The Newton solver for the system of NAE does not need a predictor and does not need to bother about the time derivative, since it is zero. The reason why a predictor is not needed is explained in section 3.4.1.
The inner solver can at each iteration step chose between building a new Jacobian and decompose it or reusing the old decomposed Jacobian, depending on the convergence, just as the Newton solver for the system of ODE.
There are two main paths for the Newton solver, depending on if the old Jacobian can be reused or if a new Jacobian must be built. If a new Jacobian must be built the operations described in the “Full inner iteration step” must be performed.

Full inner iteration step

- Build the Jacobian
- Decompose the Jacobian (GE)
- Update the source-vector
- Solve the linear system (BS)

The building and decomposition of the Jacobian are done once, while the update of the source-vector and the BS are repeated until convergence of the iteration is reached.
If a new Jacobian must be built the operations described in the “Simple inner iteration step” must be performed.

Simple inner iteration step

- Update the source–vector
- Solve the linear system (BS)

A typical full inner Newton iteration for a simulation of a CVR involves few full iteration steps and many simple iteration steps until convergence is reached. The number of full iteration steps used depends on the point in time in the simulation. The number of full

iteration steps is higher close to the ignition point where the partial derivatives change faster and lower far away from the ignition point where the partial derivatives changes slower.
Simple iteration steps save much CPU time, since building and decomposing the Jacobian is expensive compared to the other operations. The cost of each operation is described below.

Building the Jacobian
Each element in the Jacobian is found from an exact derivative of the source-vector, see section 3.4.1.1. Building the Jacobian of the system of NAE is less costly than building the Jacobian of the system of ODE, since the system of NAE does not need to be solved for each element in the Jacobian of the system of NAE.

Decomposing the Jacobian (GE)
Decomposing the Jacobian is done via sparse GE, which is discussed in section 4.3.3.2.2. The cost scales roughly as $(2/3)*n^3$.

Update the source-vector
The source-vector must be updated before the linear system can be solved. The source vector must undergo the same decomposing operations as the Jacobian, which suggests that the cost scales roughly as n^2, since the number of row subtractions in the GE scales as n^2.

Solve the linear system (BS)
The concentration vector is found from sparse BS, which is discussed in section 4.3.3.2.3. The cost scales roughly as n^2.

4.3.1.1. Convergence

The convergence criterion of the inner solver is the same as for the outer solver, which is discussed in section 4.2.1. However, the values of the absolute tolerance and relative tolerance differ for the inner and outer solver. These tolerances are much smaller for the inner solver than for the outer solver. The reason for this is that much higher demands are put on the accuracy of the concentrations for the system of NAE, since this minimizes the number of Jacobians for the system of ODE, which in turn minimizes the total CPU time. This is discussed in detail in section 6.2.1.2 and section 3.3.4.1.1.

The values of the absolute tolerance and relative tolerance of the inner solver are varied in section 6.2.1.2. The optimum values were found to be;

- Absolute tolerance=$1*10^{-15}$
- Relative tolerance=$1*10^{-7}$

It should be noted that the optimum solver setting depends on the stiffness of the problem and thereby on the chemical mechanism.

4.3.1.2. Flow chart for the Newton solver for the system of NAE

A sub flow chart for the Newton solver for the system of NAE can be seen in Figure 4.6. The action for each node is explained in Figure 4.5.

1) Run the Newton solver for the system of NAE with or without building a new Jacobian. If a new Jacobian must be built, go to 2. If not, go to 4.
2) Build a new Jacobian. Go to 3.
3) Decompose the Jacobian. Go to 4.
4) Update the source vector. Go to 5.
5) Solve the system of NAE. Go to 6.
6) Did the solver converge or not? If yes, exit sub flow chart. If no, go to 4 or 7.
7) Does the inner solver need to build a new Jacobian in order to converge or not? If yes, go to 2. If no, go to 4.

Figure 4.5. Explanation of the function of each node in Figure 4.6.

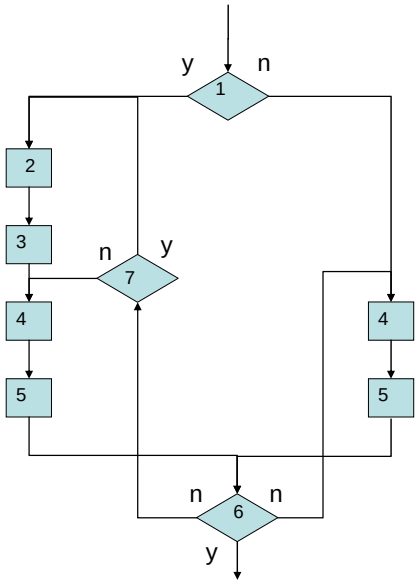


Figure 4.6. Flow chart of the Newton solver for the NAE system. This flow chart is a sub flow chart of the entire solver. The meaning of each node is explained in Figure 4.5.

4.3.2. Arguments for using a modified Newton solver for the system of NAE

The problem is to get a solver combination that is fast and still maintains the accuracy of the solution. The main cost for the system of ODE is building the Jacobian, which means that the number of Jacobians must be kept low. A high accuracy of the concentrations of the inner solver results in fewer Jacobians and better convergence for the outer solver. This is discussed in detail in section 6.2.1.2 section 3.3.4.1.1. Hence, a reasonable approach to solve the problem would be to get a very accurate inner solver, to avoid building new Jacobians of the outer solver, and somehow try to make the inner solver as fast as possible.

A Newton solver was chosen as a solver for the NAE because of its high accuracy. According to section 3.5 the Newton solver for the NAE is iterated until convergence at

each iteration step of the outer solver and that the Newton solver for the NAE is iterated until convergence for each element in $\mathbf{x}_D$ when building the Jacobian for the system of ODE. This means that the Newton solver for the NAE is called many more times than the outer solver. So in order to keep the total CPU time for the entire solver combination, which is described in section 3.5.3, as low as possible, while maintaining the accuracy of the solution, the Newton solver for the NAE must be speeded up somehow. The way to modify the Newton solver in order to get a speed up is described below.

4.3.3. Modifications of the Newton solver

The computational cost for the Newton for the system of NAE consists of; building the Jacobian, decomposing the Jacobian by GE, updating the source vector and solving the linear system by BS. The computational cost for each part is shown in section 4.3.1.
The computer code for the operations described above is normally made up of DO-loops, of the size of the number of species involved, and IF-statements. The Newton solver for the system of NAE can therefore be speeded up in the following ways;

- Calculate the Jacobian elements analytically.
- Only use the Non Zero Elements (NZE) in the GE and the BS. See section 4.3.3.2.2 and 4.3.3.2.3.
- Minimize the number of operations in the GE and the BS by optimizing the sparseness pattern of the Jacobian. See section 4.3.3.3.
- Write the whole Building of the Jacobian, GE, updating of the source vector and the BS in long files in a pre-processing step in order to avoid expensive DO-loop and IF statements in the solver code. See section 4.3.3.1.1.

The last two points can only be done if the **sparseness pattern of the Jacobian is constant in time** and will be explained in detail below.

4.3.3.1. Fixed sparseness pattern

The Jacobian for the system of NAE can be found analytically, which is explained in detail in section 3.4.1.1. The Jacobian elements in eq(3.68).are non zero only if the source term, $\omega_{A,p}$, depends on $x_{A,Q}$. Hence, the locations of the NZE are fixed and constant in time, since the chemical reaction scheme does not change with time. In other words, the sparseness pattern of the Jacobian is constant in time. This means that only the NZE of the Jacobian must be calculated. Figure 4.7 shows the sparseness pattern of the Jacobian with 90 QSS species. The NZE are black and the zero elements are white.
However, the values of the non zero Jacobian element do change with time, since the concentrations and thereby the source term change with time. Hence, the non zero matrix elements can be written as a function of the reaction rates, which in turn are functions of

the concentrations, in a file and calculated without expensive DO-loops and IF-statements. An example of the computer code for calculation of the analytical Jacobian for the system of NAE is shown in Figure 4.8.
It should be noted that the non zero Jacobian element can be zero for a point in time if the reaction rates in the source term are zero.

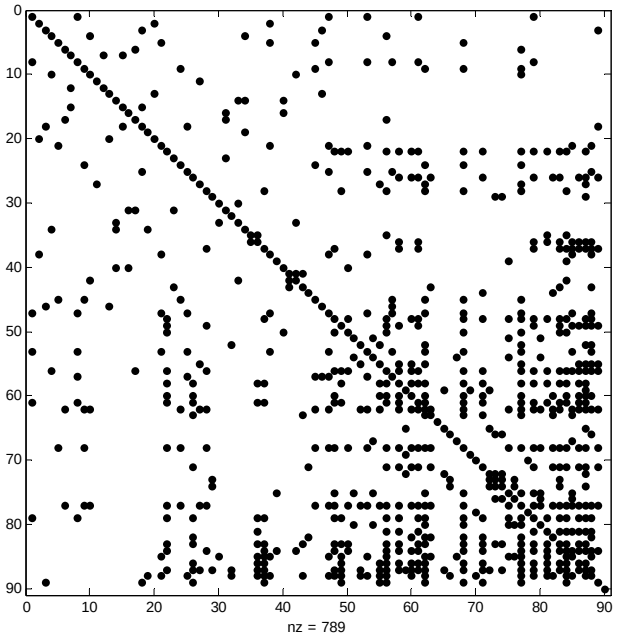


Figure 4.7. Fixed sparseness pattern of the Jacobian for the system of NAE. The pattern stays constant in time, while the values of the non zero matrix elements change with time. The picture comes from the optimized Methane/Propane mechanism with 90 QSS species.

```
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1058)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1060)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1062)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1064)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1066)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1068)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1070)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1072)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1074)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1076)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1) +  RateR( 1077)*(-1)
 SSJacobian(   1,   1)=SSJacobian(   1,   1)/MAX(c( 102),1.d-40)
 SparseValueMatrix(   1,   1)=SSJacobian(   1,   1)
!------------------------------------------------------------------------------

 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1036)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1038)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1040)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1042)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1044)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1046)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1048)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1050)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1052)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1054)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2) +  RateR( 1055)*(-1)
 SSJacobian(   2,   2)=SSJacobian(   2,   2)/MAX(c( 100),1.d-40)
 SparseValueMatrix(   2,   1)=SSJacobian(   2,   2)
!------------------------------------------------------------------------------

 SSJacobian(   3,   3)=SSJacobian(   3,   3) +  RateR( 1099)*(-1)
 SSJacobian(   3,   3)=SSJacobian(   3,   3)/MAX(c( 104),1.d-40)
 SparseValueMatrix(   3,   1)=SSJacobian(   3,   3)
!------------------------------------------------------------------------------

 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1080)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1082)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1084)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1086)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1088)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1090)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1092)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1094)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1096)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4) +  RateR( 1098)*(-1)
 SSJacobian(   4,   4)=SSJacobian(   4,   4)/MAX(c( 107),1.d-40)
 SparseValueMatrix(   4,   1)=SSJacobian(   4,   4)
```

Figure 4.8. Computer code for calculation of the analytical Jacobian for the system of NAE. The calculations correspond to eq(3.68). The expressions are constant in time, while the values change with time. “SSJacobian(i, j)” is the Jacobian element (i,j). “RateR(x)” is the reaction rate for reaction x. “c(y)” is the concentration of species y. “SparseValueMatrix(i, j)” is the compressed matrix of linked list *i* and box *j*. The compressed matrix is explained in detail in section 4.3.3.2.3.

4.3.3.1.1. Pre-processing

The pre-processor is a program called REDKIN. This program chooses the level of reduction, which gives the number of QSS species, and the program outputs many subroutines that are all functions of the chosen level of reduction. Another program, IGNITION, uses these subroutines from REDKIN and computes the combustion simulation. This is illustrated in Figure 4.9.

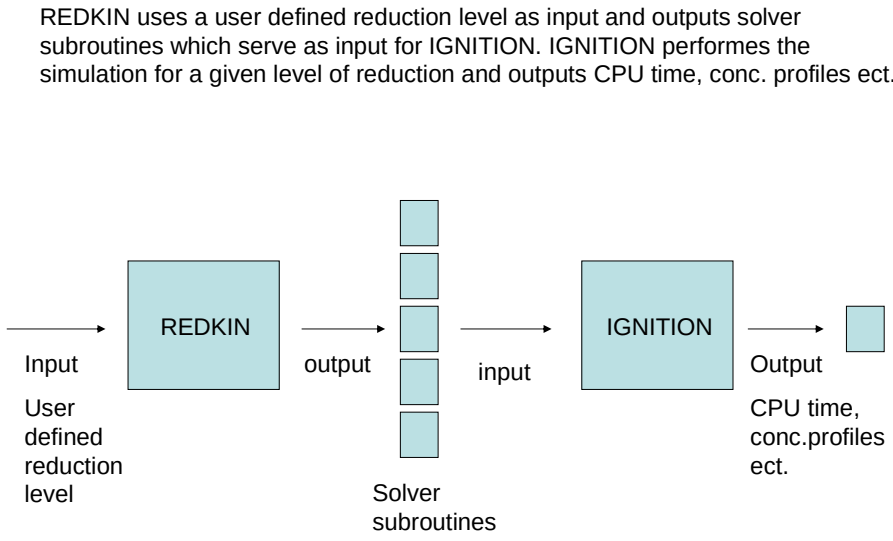


Figure 4.9. The cooperation between the two programs; REDKIN and IGNITION. REDKIN outputs solver subroutines that are used by IGNITION.

The IGNITION program mostly consists of the solver for the ODE and parts of the solver for the NAE. However, parts of the inner solver are subroutines generated from REDKIN. How can this be?

Since the sparseness pattern of the Jacobian of the NAE is constant in time and only the values of the Jacobian change with time, every operation done on the Jacobian will also be independent of time. In other words, it will always be the same matrix elements that are involved when an operation is done on the Jacobian.
Hence, every operation in the GE can be written in long subroutines instead of the usual GE, which involves a couple of DO-loops and IF-statements. Getting rid of these DO-loops and IF-statements saves CPU time.
The same thing can be done with the source vector. The source vector must be multiplied with all the operation in the GE as well before it can be used in the solution of the linear system by BS.

Also, the whole BS can be written in long subroutines without DO-loops and IF-statements since the same matrix elements are used independent of time and only the values of the matrix elements and the source vector elements change with time.

So, the analytical calculation of the Jacobian, the GE, update of the source vector and BS ca all be written in files since the sparseness pattern is constant in time.

4.3.3.2. Using the sparseness pattern

This section explains in detail how the sparseness pattern is optimized and the algorithm behind the optimization. Also, the data structures used in practice are mentioned.
All of the optimization takes place in the REDKIN program.

4.3.3.2.1. Linked lists

A liked list is a chain of boxes where the order of the boxes can be changed. Each of the boxes contains, except for the obvious contains of the box, two pointers that point to the addresses of other boxes. An example of this is shown in Figure 4.10. One of the pointers of a given box in the chain points to the address of the box previous to it in the chain and the other pointer points to the address of the box in front of it in the chain. To change the orders of the boxes one simply changes the addresses of pointers. To insert a new box into a chain, one must change the addresses of three boxes and four pointers in total. An example of this is shown in Figure 4.11.

The program that generates the GE and BS for the inner solver uses liked lists, one for each row of the Jacobian. The number of boxes in each list is the same as the number of NZE in the corresponding row. Each box contains the Column Index (CI) of the NZE. When the GE is performed the lower row is subtracted by a part of the upper row. Sometimes this means that a NZE is transported from the upper row into the lower row. Hence, a box must be inserted into the liked list corresponding to the lower row. A more detailed discussion of the GE using liked lists is found in section 4.3.3.2.2.

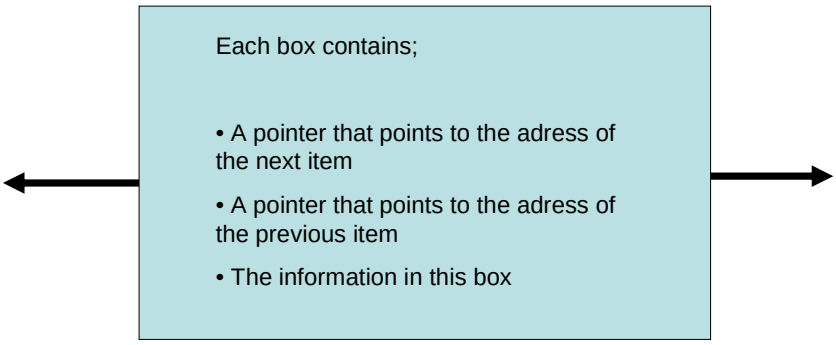


Figure 4.10. Characteristics of a box in a linked list.

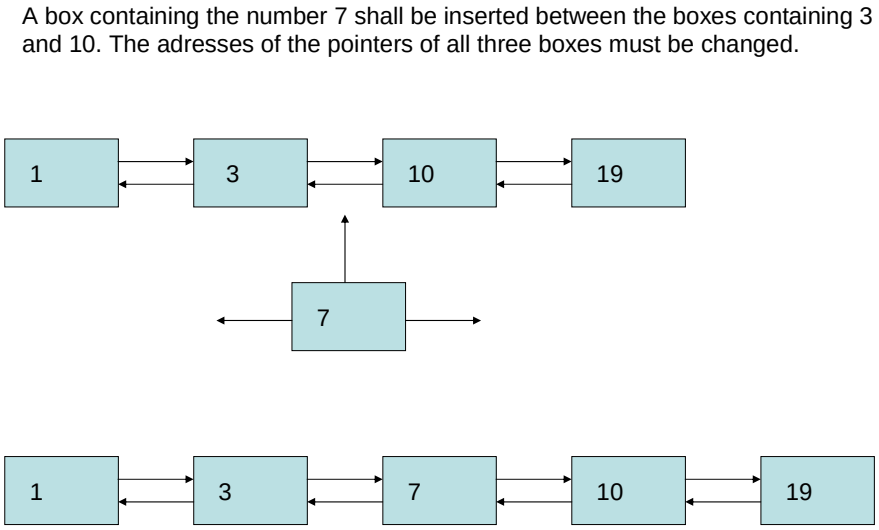


Figure 4.11. Insertion of a box into a linked list of boxes. A linked list corresponds to a particular row in the Jacobian. The number of boxes in each linked list corresponds to the number of NZE in the row. The number in the box is the CI. The figure shows the lower row in the GE. This particular procedure in the figure corresponds to the case when the upper row in the GE contains a number different from zero in column seven and the lower row does not. The number seven must therefore be inserted into the lower row.

4.3.3.2.2. Sparse GE

The sparse Newton solver only uses the NZE of the Jacobian in the GE and the NZE of the decomposed Jacobian in the BS. The details of the modified GE and the modified BS are described below.
In order to reduce CPU time for the entire ignition procedure, only the operations of the NZE of the Jacobian should be considered. This complicates the GE algorithm considerably.
The preprocessor, that generates the hard coded GE subroutine, stores the column indices for the NZE, for a certain row in the Jacobian, in a linked list. One linked list is used for each row in the Jacobian. The column indices are stored in an increasing order in each linked list. An example of this is shown in Figure 4.12. Now the GE is performed on the "compressed Jacobian", that is, the liked lists. An example of this is shown in Figure 4.13. Whenever there is a CI in the upper row that also exists in the lower row the lower row element should just be subtracted by a multiple of the upper row, like in a normal GE. Whenever there is a CI in the upper row that doesn't exist in the lower row, a multiple of the element should be inserted in the lower row. This is easily done with linked lists. When a column is "zeroed" in the lower row of a normal GE, it just corresponds to the elimination of the "box" containing that particular CI in the linked list corresponding to the lower row.
Only those operations in the GE that are candidates to be written out hard coded in files are those involved with row subtractions that have the same first CI on the upper and lower row (if the first CI of the upper and lower rows is not the same, there shall not be a row subtraction at all). And from these candidates only those operations that have the same CI in both rows are written out. Also, those operations that have a CI in the upper row that doesn't exist in the lower row are written out. All the operations that are written out must be done in the same order as in the preprocessing GE.

- Store the CI of the NZE in the boxes of the linked lists. Use one linked list for each row of the matrix. The first NZE in the first box, the second NZE on the second box, and so on…
- Perform the GE. The first box of the lower row is deleted when the value of the box becomes zero. If an index exists in the upper row but not in the lower, a box must be added to the linked of the lower row.

4.3.3.2.3. Sparse BS

When the whole GE is done, the linked list contains only the indices that correspond to an upper triangular matrix. An example of this is shown in Figure 4.14. Now BS is performed involving only the NZE of the upper triangular matrix. For a certain linked list, the value that corresponds to a particular CI is multiplied by the element in the solution-vector with the same index as that particular CI. In this way, only the operations involving the NZE of the Gaussian eliminated Jacobian are written out in the BS subroutine. An example of the sparse BS is shown in Figure 4.15.

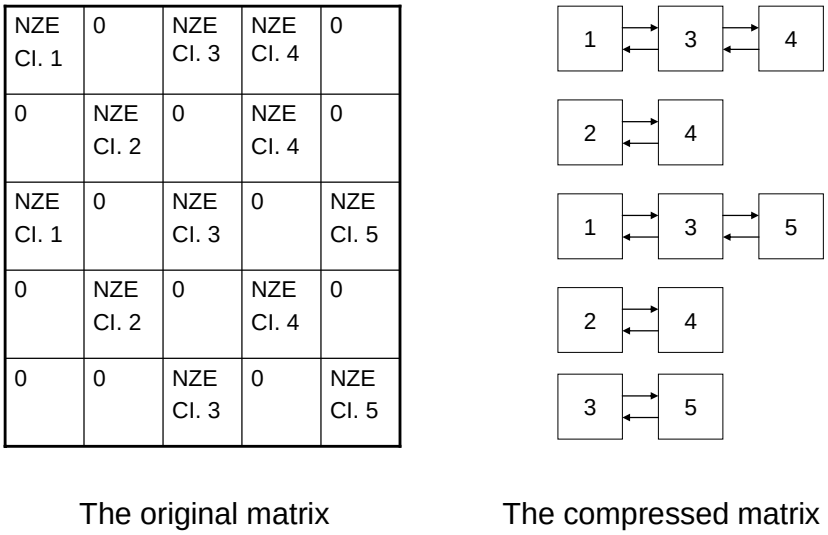


Figure 4.12. Example of an original matrix being compressed into linked lists. The original matrix to the left contains the Column Index (CI)of the NZE. The location of the NZE is constant in time but the value changes with time. Each row has its own liked list and stores the CI of the NZE, starting from the left, but does not store the value of the matrix element. Only the NZE of the original matrix is involved in the GE BS.

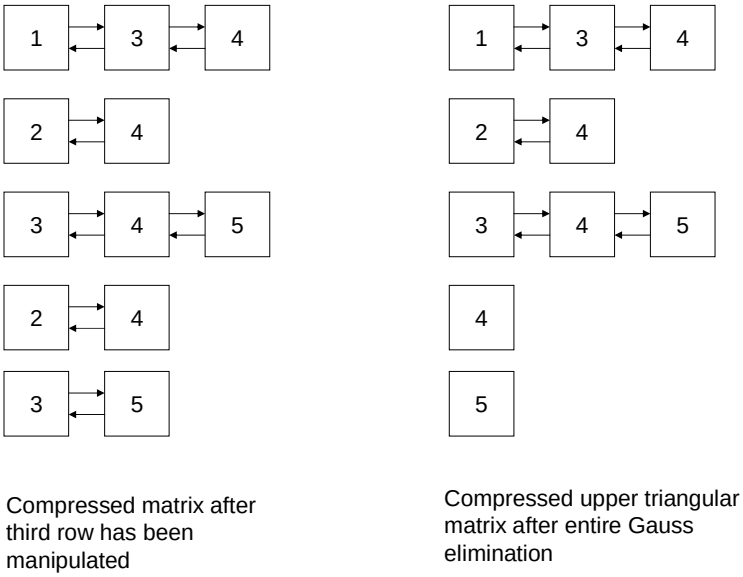


Figure 4.13. To the left is the compressed matrix after the third row has been manipulated by the first row. This means that the CI, 1, has been removed from and CI, 4, has been inserted into row 3. To the right is the compressed matrix after the entire GE. This means that the CI, 2, has been removed from row 4 and CI, 3, has been removed from row 5. Only the NZE of the original matrix is involved in the GE.

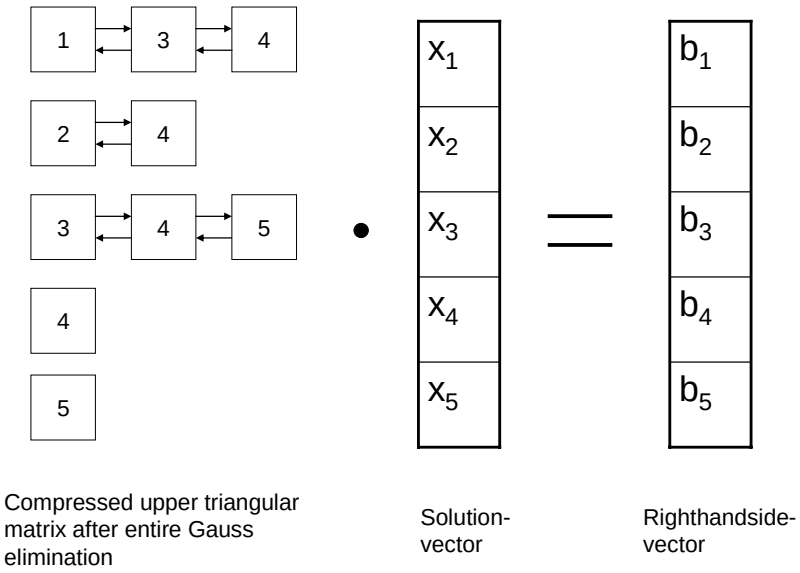


Figure 4.14. The compressed matrix after GE times the solution-vector equals the right hand side-vector. BS must be performed in order to find the solution vector and can be seen in Figure 4.15.

$$x(5) = \frac{b(5)}{A(5,5)}$$

$$x(4) = \frac{b(4)}{A(4,4)}$$

$$x(3) = \frac{\left[b(3) - \left(A(3,4)\cdot x(4) + A(3,5)\cdot x(5)\right)\right]}{A(3,3)}$$

$$x(2) = \frac{\left[b(2) - \left(A(2,4)\cdot x(4)\right)\right]}{A(2,2)}$$

$$x(1) = \frac{\left[b(1) - \left(A(1,3)\cdot x(3) + A(1,4)\cdot x(4)\right)\right]}{A(1,1)}$$

Figure 4.15. BS in order to find the solution vector. Only the NZE of the original matrix is involved in the BS. First x(5) is found, then x(4) is found, then x(3) is found, using x(4) and x(5), thereafter x(2) is found, using x(4), and finally x(1) is found, using x(3) and x(4).

4.3.3.3. Sparseness optimization

The sparseness pattern of the Jacobian can be optimized in the sense that the combined number of operations in the GE and BS can be minimized. A minimized number of operations in the solver for the NAE in turn results in a minimized CPU time for the combined solver for the DAE. Since it is very hard to predict which sparseness pattern of the Jacobian that will minimize the number of operations, a simulated annealing process is used. The principals of simulated annealing if described in section 4.3.3.3.1 and the simulated annealing applied to optimization of the Jacobian is described in section 4.3.3.3.3.

4.3.3.3.1. Simulated annealing

Simulated annealing [1] is a global optimization method which is very useful on complex systems with many local minima in the solution landscape. An example of this is the famous “Travelling Salesman Problem”. The algorithm is constructed in such a way that the solution should be able to get out of local minima and travel towards the global minima or a deep local minima with time. The path to the global minimum is controlled by *acceptance probabilities*, which in turn depends on a time varying parameter called the “temperature” and on the new and old function values. This is discussed in more detail in section 4.3.3.3.2. There is however no guarantee that the global minimum is found, but there is a high probability that a deep local minimum is found. An example of a solution landscape with many local minima of the “Energy”-function, $E(\mathbf{x})$, as a function of state-vector, $\mathbf{x}$, is shown in Figure 4.16.

An essential feature of the simulated annealing methods is that the temperature is gradually reduced as the simulation proceeds. The temperature is initially set very high and then decreases at each step of the algorithm. How the temperature decreases is specified by the user, but the procedure must end with the temperature equaling zero. In this way the algorithm is expected to initially search a broad region of the state space and ignore smaller features of the function landscape. At lower temperatures the algorithm is expected to drift towards deep local minima and finally just move downhill when the temperature equals zero, since the probability of moving out of local minima also goes to zero when the temperature goes to zero.

The cooling rate of the temperature is a key parameter for finding the global minimum. If the cooling rate is too slow the algorithm becomes computationally expensive. If the cooling rate is too high there is a risk the solution gets stuck in a local minimum.

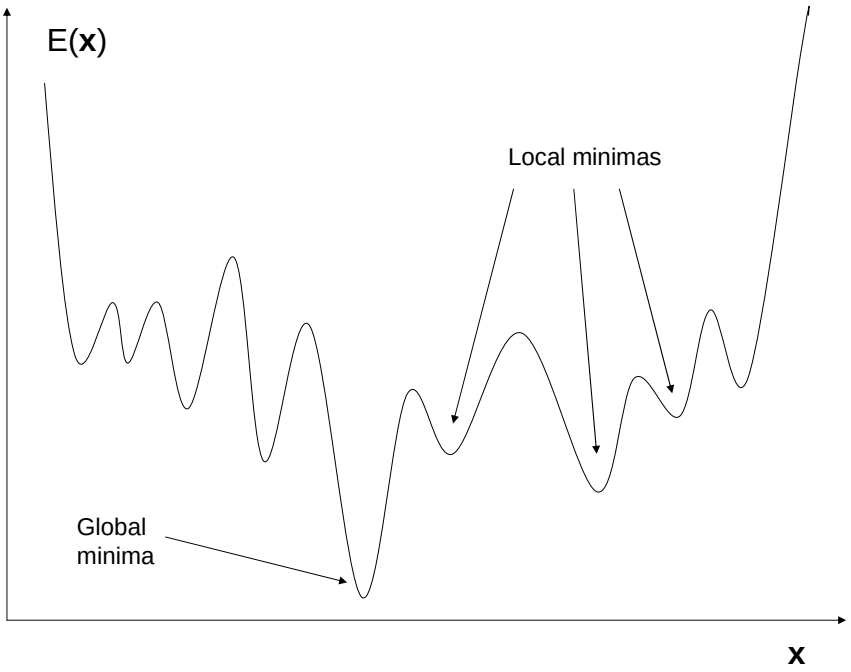


Figure 4.16. The "Energy"-function, *E(**x**)*, as a function of ***x***. There are usually many local minima in the energy-landscape when simulated annealing is powerful tool to use. The simulated annealing algorithm is created to be able to get of local minima and find the global minimum or deep local minima.

The principle of simulated annealing can be applied to an input/output model of a complex system. This is illustrated in Figure 4.17.
The input or parameter vector is changed (often randomly). Thereafter the simulation is performed and the output before and after the change is compared. If the result after the change was better then the result before the change the new input vector is kept, which moves the solution "downhill" in the solution landscape. If the result after the change was worse then the result before the change the new input vector is kept with an *acceptance probability* which depends on the difference in the output values as well as the temperature. This makes it possible for the solution to move "uphill" in the solution landscape, that is, get out of local minima. The probability to get out of deep local minima is less than the probability to get out of shallow local minima. The acceptance probabilities in this thesis are based on the Metropolis algorithm [2], which is discussed in section 4.3.3.3.2.

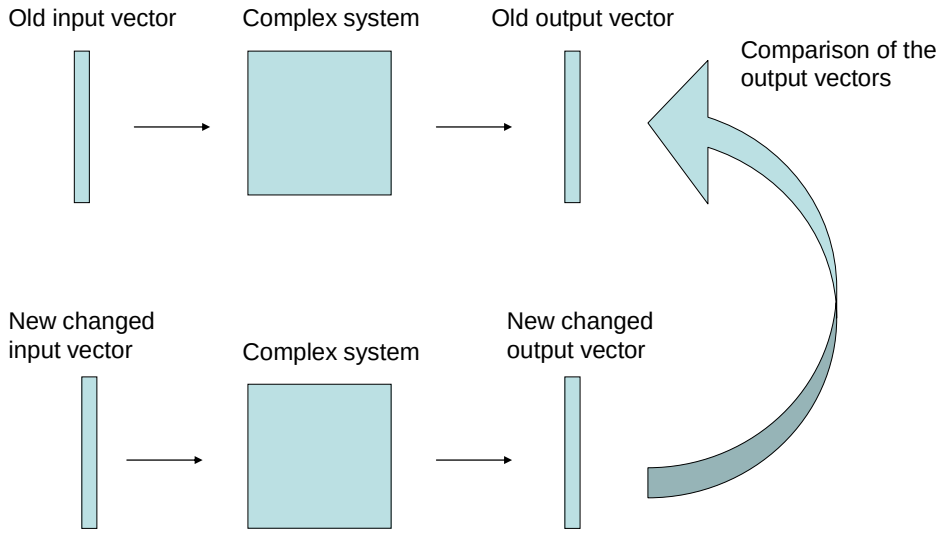
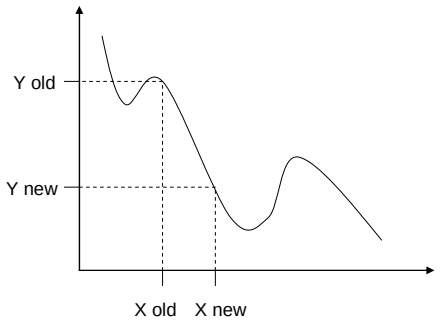


Figure 4.17. Principals of the simulated annealing applied to an input/output model of a complex system. A small change in the input vector can give large changes in the output vector since the system is complex. The new and old output vectors are compared in order to accept or decline the new change in the input vector. The logic for acceptance or not can be found in Figure 4.18.

If the new output vector, Y new, is better than the old output vector, Y old, keep the new input vector, x new, with propability P=1.



If the new output vector is worse than the old output vector, keep the new input vector with propability

$$P(Y_{new} - Y_{old}) = \exp\left(-\left(\frac{Y_{new} - Y_{old}}{T}\right)\right)$$

Otherwise keep the old input vector

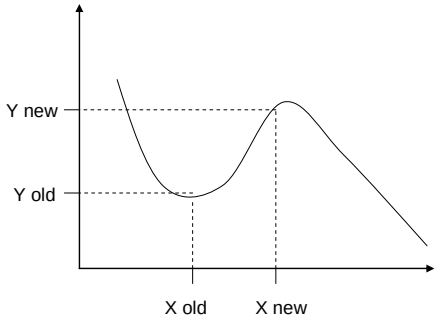


Figure 4.18. Principals of the logic for acceptance or not for a new change in the input vector in a simulated annealing procedure. A new change is accepted with different probabilities if the new output vector is better or worse than the old output vector. The acceptance probabilities based on the Metropolis algorithm.

Repeat the MBSA algorithm and decrease T in every cycle.

Continue the procedure until the global minimum is reached or a deep enogh local minimum is reached.

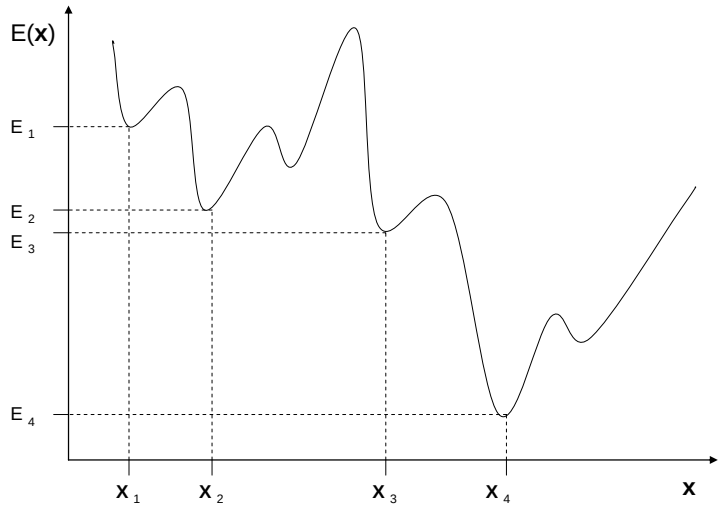


Figure 4.19. Principal of simulated annealing. Each function-value, *E(x)*, corresponds to an *x*-value. The function-value reaches lower and lower minima with time, since there is always a probability go get out of local minima. The "temperature" is lowered with time, which means that the probability to get up from local minima decreases with time.

4.3.3.3.2 The Metropolis Based Simulated Annealing (MBSA) algorithm

It's hard to reach the global minima in a landscape which contain many local minima if the starting-point isn't close to the global minima. The solution must not be stuck in a local minimum once it gets there. Therefore there must be a possibility to move up from a local minimum and continue to the global minima. On the other hand must there be a (low) probability for the solution to move up from the global minima once it gets there. The acceptance probabilities used in the MBSA algorithm, which are based on the Metropolis algorithm, are illustrated in Figure 4.18. The pseudo-MBSA algorithm is constructed as follows;

Pseudo-MBSA algorithm

DO WHILE (stopping criteria)

1. Generate new x-vector
2. Calculate new "energy-function"
3. Calculate the difference in the energy-function, ΔE, between the old and new energy-function values. That is, $\Delta E = E_{new} - E_{old}$

IF ($\Delta E < 0$) **THEN**

4a. Accept the new x-vector

ELSE

4b. Accept new x-vector with acceptance probability;

$$P_{ACC}(\Delta E, T) = \exp\left(-\left(\frac{\Delta E}{T}\right)\right)$$

END IF

5. Decrease T

END WHILE

If ΔE is less than zero, the solution will move down in the energy landscape with probability one. If ΔE is positive there is a probability that the solution will move upwards in the energy landscape, the larger ΔE is the smaller the probability is for a given T. This means that it's relatively easy to get up from a shallow minima and hard to get out of a deep minima. The decrease of T with time (or iterations step in the WHILE-loop) makes the probability to move upwards smaller with time, which means that the solution will have a larger probability to settle down in a nearby minimum as time passes. There is no guarantee that the solution will be in the global minimum with such an algorithm, but the probability to reach a deep minimum increases with time.
The starting temperature and decrease of the temperature can be optimized depending on the landscape.
The stopping criteria in usually that $T=0$ or a fixed number of iterations.
A schematic illustration of the MBSA algorithm is shown in Figure 4.19.

4.3.3.3.3. Simulated Annealing applied to sparseness optimization

The aim is to minimize the sum of the operations in the GE and the BS in order to make the CPU time of the solver for the NAE lower, which in turn means lower CPU time for the entire system of DAE. If the following changes are made to the MBSA algorithm in section 4.3.3.3.2 the simulated annealing process is applied to the sparseness optimization of the Jacobian;

- The "Energy"-function in the MBSA algorithm is exchanged for Sum of OPerations (SOP), which is the sum of the operations in the GE and the BS.
- The ***x***-vector is exchanged for the QSS species order-vector.

This means that a new QSS species order-vector is generated by randomly changing the location of two species in the vector. For example, the QSS species located at place 13 in the vector changes place with the QSS species located at place 46 in the vector. This will result in a changed sparseness pattern of the Jacobian. This is illustrated in Figure 4.20. The GE is performed and the number of row subtractions made during the GE was counted. The number of non zero Jacobian elements in the upper triangular matrix after the GE was performed is also counted. The sum of these numbers, in the following called SOP, now corresponds to the Energy-function and shall be minimized.
If the Energy-function corresponding to the new QSS species order-vector is **lower** than the Energy-function corresponding to the old QSS species order-vector the new QSS species order-vector is accepted with probability one. If the Energy-function corresponding to the new QSS species order-vector is **higher** than the Energy-function corresponding to the old QSS species order-vector the new QSS species order-vector is accepted with probability;

$$p(\Delta E) = \exp\left(-\left(\frac{\Delta E}{T}\right)\right) \tag{4.3}$$

The "fictive" temperature is now lowered a bit and a new QSS species order-vector is thereafter generated and the whole process is repeated.

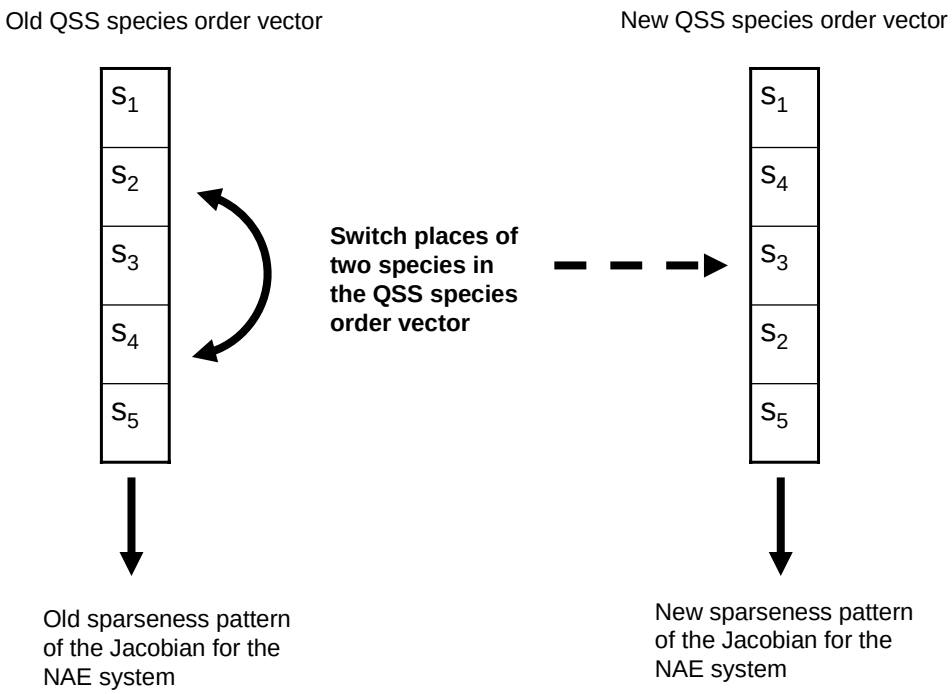


Figure 4.20. The old sparseness pattern of the Jacobian for the NAE system corresponds to the old QSS species order list. A new QSS species order list is created by switching places of two QSS species. The new QSS species order list gives rise to a new sparseness pattern of the Jacobian for the NAE system.

In order to find the optimum order of the QSS species order-vector, which corresponds to the minimum number of operations in the inner solver, the *MBSA* algorithm is employed. The pseudo algorithm for minimization of the SOP is outlined as follows:

The MBSA algorithm for optimizing the number of operations in the solver

Initialize the QSS species order-vector for the inner solver
Initialize T
Repeat
1. Change the places of two randomly chosen species in the QSS species order-vector
2. Create new Jacobian and perform GE
3. Calculate the SOP_{NEW} (based on the new QSS species order-vector)
4. Calculate the $\Delta SOP=SOP_{NEW}-SOP_{OLD}$, based on the new and old QSS species order-vector.
IF($\Delta SOP<1$):

5a. Accept the new QSS species order-vector with acceptance probability

$$P_{ACC}(\Delta \mathrm{SOP}, T) = 1$$

ELSE

5b. Accept the new QSS species order-vector with acceptance probability

$$P_{ACC}(\Delta \mathrm{SOP}, T) = \exp\left(-\left(\frac{\Delta \mathrm{SOP}}{T}\right)\right)$$

END IF
6. Decrease T
Until terminating condition is fulfilled

The *MBSA* algorithm is constructed in such a fashion that the solution can get up from local minima because of the probability to accept $\Delta SOP > 0$, and travel towards deeper local minima. This algorithm will not guarantee that the global minimum is found, but in most cases a "deep enough" local minimum, corresponding to a very low number of operations in the inner solver, is found. Hence, the total CPU time for the integration of the system of DAE at high reduction levels will be close to minimized, since the inner solver is dominating the total CPU time at for high reduction levels.
It should be noted that the final value of SOP does not necessary correspond to the deepest minimum. For this reason, the QSS concentration vector which corresponds to the deepest minimum during the entire optimization procedure is saved instead in this thesis. The initial value of T and the function used in order to decrease of T can be optimized for each case. In this thesis a linear decrease of T is used and the termination condition is a fixed number of iterations. Another terminating condition would be $T=0$, which means that the solution cannot move uphill.

The sparseness pattern of the Jacobian changes for each iteration of the *MBSA* algorithm. The sparseness pattern of the optimized Jacobian is very hard to predict beforehand, since a change of two species can have little or dramatic effect on the sparseness pattern and thereby on the SOP.

An example of the optimized Jacobian before and after the GE can be seen in Figure 4.21. The black dots symbolize the NZE.

Figure 4.22 shows the "GE-matrix", which shows the rows that have been involved during the GE. A NZE in position (i,j) in the GE-matrix means that a fraction of row *i* was subtracted from row *j* in order to produce a zero in position (j,i) in the Jacobian matrix.
An example of this is the following: If row 7 in column 2, that is element (7,2), in the Jacobian contains an NZE, a fraction of the diagonal element of row 2 is subtracted from the NZE in row 7 in order to produce a zero. This results in a NZE at position (2,7) in the GE matrix.
The number of NZE in the GE matrix corresponds to the number of row subtraction during the GE. However, the number of operations involved in each row subtraction depends on the sparseness pattern and are not considered in the GE-matrix. Instead each row subtraction corresponds to one NZE, independent of the number of operations involved in the row subtraction.

Hence, the sum of NZE of the Jacobian after GE and the NZE of GE-matrix corresponds to the SOP.

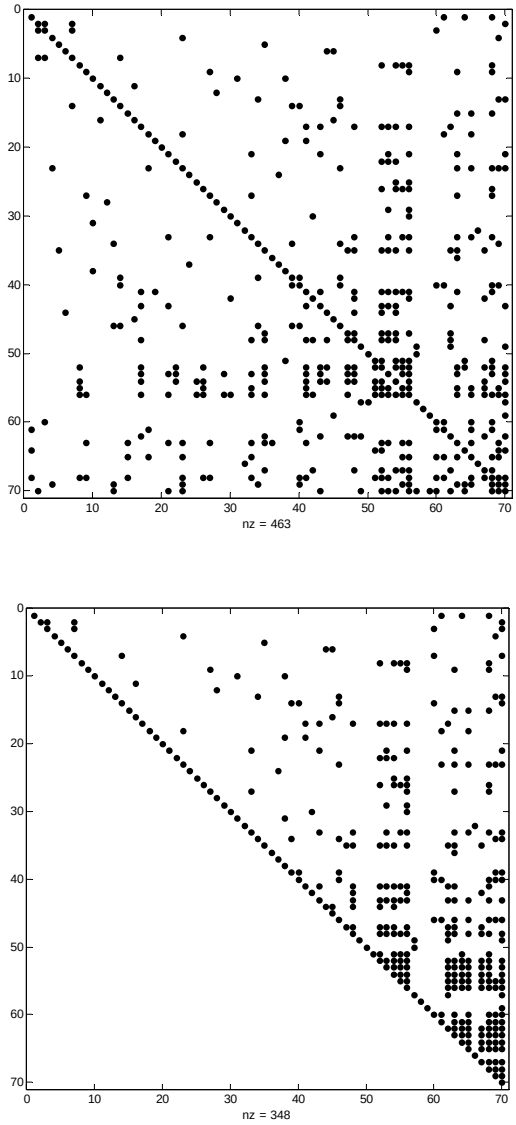


Figure 4.21. The optimized Jacobian before (top) and after (bottom) GE. The latter matrix is used in the BS. The pictures come from the optimized N-Heptane mechanism with 70 QSS species described in section 6.2.1.3.1.

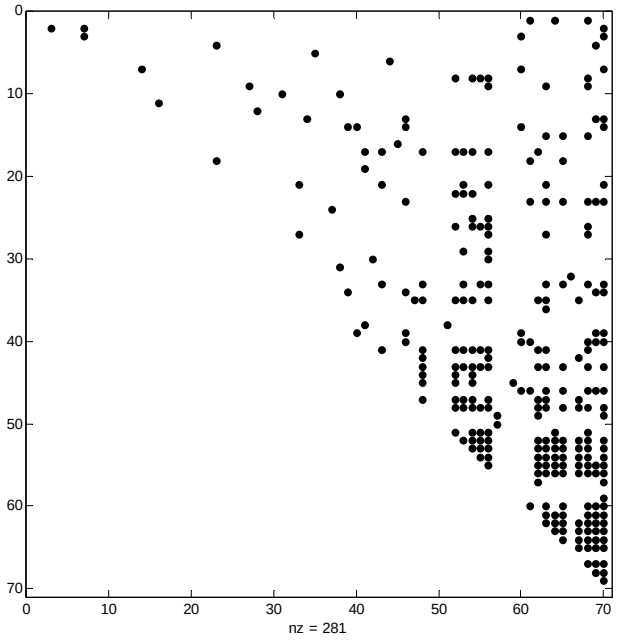


Figure 4.22. The figure shows the "GE-matrix", which shows the rows that have been involved during the GE. A NZE in position (i,j) in the GE-matrix means that a fraction of row *i* was subtracted from row *j* in order to produce a zero in position (j,i) in the Jacobian matrix. The picture comes from the optimized N-Heptane mechanism with 70 QSS species described in section 6.2.1.3.1.

4.3.4. The FP solver for the system of NAE

The FP solver is much simpler than the Newton solver, since no gradient information is used. This means that a Jacobian does not have to be built or decomposed. The FP method is explained in detail in section 3.4.2.
The FP solver starts with a starting value of $1*10^{-40}$ for all QSS species and is iterated until convergence or continues the iteration a pre-defined number of iteration steps until a final value is reached. The final value of the iteration is then accepted as the QSS concentration vector used by the outer solver.
The same convergence criterion that is used for the Newton solver is used for the FP solver. The same optimized solver setting that was used for the inner Newton solver was also used for the FP solver during the simulations.

4.3.4.1. Flow chart for the FP solver for the system of NAE

A sub flow chart for the FP solver for the system of NAE can be seen in Figure 4.24. The action for each node is explained in Figure 4.23.

1) Initialize the FP solver. Go to 2.
2) Take a new iteration step. Go to 3.
3) Did the FP solver converge? If yes exit FP solver. If no go to 2.

Figure 4.23. Explanation of the function of each node in Figure 4.24.

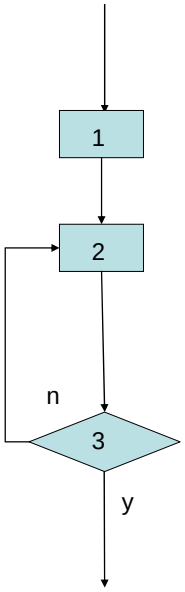


Figure 4.24. Flow chart of the FP solver for the NAE system. This flow chart is a sub flow chart of the entire solver. The meaning of each node is explained in Figure 4.23.

4.4. The combined solvers for the system of DAE

This section describes the full flow charts of the two solver combinations;

- Newton-Newton solver
- Newton-FP solver

4.4.1. Newton-Newton solver

First the time step size is chosen. If the solver converged well in the last time step a longer time step may be tried, but if the solver did not converge a shorter time step may be tried.
Thereafter the solver chooses between reusing the old Jacobian and building a new Jacobian for the system of ODE. Weather or not to build a new Jacobian in the outer solver depends on how the last time step was executed. If the solver converged well in the last time step a new Jacobian is less likely to be built than if the solver did not converge.
At each iteration step of the outer solver the inner solver is called, which in turn chooses between reusing the old Jacobian and building a new Jacobian. The number of BS for both the inner and outer solver depends on weather or not a new NAE Jacobian was built. If a new NAE Jacobian was built a smaller number of BS are needed.
In order for the system of DAE to converge the inner solver is free to adjust itself any way it wants depending on weather or not a new Jacobian was chosen for the system of ODE. If a new Jacobian was built for the system of ODE the inner solver may not need to build a new Jacobian, but if a new Jacobian was not built for the system of ODE the inner solver is more likely to build a new Jacobian in order for the combined system of equations to converge.

4.4.1.1. Flow chart for the Newton-Newton solver

A flow chart for the Newton-Newton solver can be seen in Figure 4.2. Box nr 2 in Figure 4.2 corresponds to the solver for the DAE system. Box nr 2 can be replaced with the Newton solver for the ODE system shown in Figure 4.4.
Box nr 4 in Figure 4.4 corresponds to the solver for the NAE system. Box nr 4 can therefore be replaced with the Newton solver for the NAE system shown in Figure 4.6.

4.4.2. Newton-FP solver

First the time step size is chosen. If the solver converged well in the last time step a longer time step may be tried, but if the solver did not converge a shorter time step may be tried.
Then the solver chooses between reusing the old Jacobian and building a new Jacobian for the system of ODE. Weather or not to build a new Jacobian in the outer solver depends on how the last time step was executed. If the solver converged well in the last time step a new Jacobian is less likely to be built than if the solver did not converge.
At each iteration step of the outer solver the inner solver (FP solver) is called and iterated until convergence. The number of BS for the outer solver depends on weather or not a new Jacobian was built. If a new Jacobian was built a smaller number of BS are needed.
In order for the combined system of DAE to converge the inner solver is **not** free to adjust itself any way it wants depending on weather or not a new Jacobian was chosen for the system of ODE. The FP solver is iterated a pre-defined number of times or until convergence is reached and cannot adjust itself by building a new Jacobian like the Newton solver, since it does not use any gradient information.

4.4.2.1. Flow chart for the Newton-FP solver

A flow chart for the Newton-FP solver can be seen in Figure 4.2. Box nr 2 in Figure 4.2 corresponds to the solver for the DAE system. Box nr 2 can be replaced with the Newton solver for the ODE system shown in Figure 4.4.
Box nr 4 in Figure 4.4 corresponds to the solver for the NAE system. Box nr 4 can therefore be replaced with the FP solver for the NAE system shown in Figure 4.24.

4.5. Chapter References

[1] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, *Optimization by Simulated Annealing*, (1983) Science. New Series 220 (4598): 671-680

[2] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E.Teller, (1953). *Equations of State Calculations by Fast Computing Machines*. Journal of Chemical Physics 21 (6): 1087–1092.

Chapter 5.

Mechanism Reduction

5.1. Chapter introduction

Many detailed mechanisms for higher hydrocarbons have been developed in recent years. In order to capture the all the physical situations of the combustion process the mechanisms have become larger, stiffer and more complicated. They must for instance include both low and high temperature chemistry. Many of the detailed mechanisms that have been developed are simply too large and stiff to be used in realistic simulation scenarios, since the CPU time increases strongly with the size and stiffness of the mechanism. (The exact dependence of the CPU time on the mechanism size is solver dependent. This is discussed in section 3.2 and 3.5.3).
This means that size and stiffness reduction of the mechanisms is of great importance in order to perform realistic simulations in reasonable time. However, the accuracy reduced mechanism must not deviate too much from the detailed mechanism. Hence, the reduction methods must be chosen wisely. In this thesis the QSSA is used in order to reduce the mechanisms. The selection of the QSS species can be done in various ways. This thesis examines a couple of methods, which are discussed below, and introduces an Automatic Reduction Tool (ART).

5.2. The goal of the mechanism reduction

The goal is to produce a reduced mechanism that is as;

- Small as possible, in order to save CPU time and memory. But with the boundary condition that the solution, based on the reduced mechanism, does not deviate too much from the solution based on the detailed mechanism.

In CFD application, CPU time and memory are saved with the usage of a smaller mechanism. The reason for this is that fewer species needs to be transported, which

means that fewer calculations need to be done, and less species concentrations need to be saved. The stiffness of the chemical system is usually reduced when the mechanism is reduced, which also saves CPU time. The reason for this is that longer time steps can be taken if the species with the shortest timescales can be eliminated from the mechanism.

The method used in this thesis in order to produce a reduced mechanism is to first generate an accurate detailed mechanism and then to reduce it while maintaining the accuracy within acceptable (user defined) limits. When the size of the mechanism is reduced some deviation in accuracy must be accepted. The deviation in accuracy usually increases as the reduction increases. This means that there will be a trade off between size and accuracy of the reduced mechanism. The size of the accuracy deviations that are accepted is user defined and case dependent.

The total reduction procedure consists of several reduction steps and corresponding mechanisms, which are all described below. However, the work behind this thesis focuses on the last step only in the total reduction procedure, which is application of the QSSA.

5.3. Mechanism generation

There are two ways to generate a mechanism, either to use the simplest possible mechanism and build it up step by step by adding more species and reactions, or to use all known species and reactions to produce a mechanism that is as detailed as possible, and then reduce it. The first option is presented in section 5.3.1 and the second option is presented in section 5.3.2. This work behind this thesis is only based on reduction of detailed mechanisms.

5.3.1. Global and semi global mechanisms

The simplest semi-global mechanism is the one-step methane oxidation.

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O \tag{5.1}$$

This does not involve CO, which is a weakness. CO is a key molecule for combustion and emission processes since oxidation of CO releases much energy into the system. A two-step mechanism that takes into account CO oxidation is;

$$\begin{aligned} CH_4 + \frac{3}{2}O_2 &\rightarrow CO + 2H_2O \\ CO + \frac{1}{2}O_2 &\rightarrow CO_2 \end{aligned} \tag{5.2}$$

A three-step mechanism, that allows separate oxidation of H_2, is;

$$\begin{aligned} CH_4 + \frac{1}{2}O_2 &\rightarrow CO + 2H_2 \\ CO + \frac{1}{2}O_2 &\rightarrow CO_2 \\ H_2 + \frac{1}{2}O_2 &\rightarrow H_2O \end{aligned} \qquad (5.3)$$

H_2 is a key molecule for combustion processes since oxidation of H_2 releases much energy into the system.
For the higher hydrocarbons, like n-alkanes, a four step mechanism has been proposed [20]. It involves an initial decomposition of C_mH_{2m+2} to C_2H_4 and H_2.

$$\begin{aligned} C_mH_{2m+2} &\rightarrow \frac{m}{2}C_2H_4 + H_2 \\ C_2H_4 + O_2 &\rightarrow 2CO + 2H_2 \\ CO + \frac{1}{2}O_2 &\rightarrow CO_2 \\ H_2 + \frac{1}{2}O_2 &\rightarrow H_2O \end{aligned} \qquad (5.4)$$

These mechanisms have been popular to use in certain CFD programs. The positive aspect of the small semi-global mechanisms is that they do not use much CPU time. The negative aspect is that they do not contain many important molecules, which results in a less accurate solution compared to a detailed mechanism.

5.3.2. Detailed mechanisms

Detailed mechanisms are the basis for reduced mechanisms and are validated against experimental data. When a detailed mechanism is built, all possible species and reaction are added to the mechanism, which means that the can become quite large.
The simplest detailed carbohydrate mechanism is the methane mechanism. Mechanisms of higher hydrocarbons normally contain the methane mechanism. The detailed N-Heptane mechanism [17] used in this thesis consists of 203 species and 1624 reactions. The mechanism contains both low and high temperature chemistry, which means that the NTC region is well predicted by the simulation.
When detailed mechanisms of higher hydrocarbons are built up they normally contain the species and reactions of the methane mechanism.

The positive aspect of the large detailed mechanisms is that they contain all possible information about the molecules and reactions, which gives a detailed and accurate solution.
The negative aspect is that the large mechanisms use a lot of CPU time. Hence, the small semi-global and large detailed mechanisms have opposite strengths and weaknesses.

5.4. Mechanism reduction

Today, detailed chemical reaction mechanisms for model fuels include up to 1000 species and 10.000 reactions [1-5], while CFD applications aim at applying mechanisms with a maximum of 100 species. Hence, a 90 % reduction in size of the chemical reaction mechanism is demanded. This can only be reached through a sequence of reduction methods [6] i.e. chemical lumping [7], species removal [8] and application of the QSSA [9]. The mechanism reduction must however be performed with the boundary conditions that the solution obtained with reduced mechanism does not deviate too much from the solution of the detailed mechanism. For this reason, the reduced mechanism will be valid only for a limited set of physical conditions.
The reduction method in this thesis is the QSSA. The QSSA was applied to a detailed Methane/Propane mechanism [21] and to a skeletal N-Heptane mechanism [16] that was previously reduced in several steps by the use of different reduction techniques, which are presented below.

5.4.1. The total reduction procedure

The procedure for generating a reduced mechanism following the Chemistry Guided Reduction (CGR) [16] is shown in Figure 5.1. First a detailed mechanism is generated and thereafter the reduction starts. A lumping procedure is applied to the detailed mechanism to generate the lumped mechanism as the first step in the reduction. Reaction flow analysis and/or sensitivity analysis is then applied to the lumped mechanism to generate the skeleton mechanism as the second step in the reduction.
After the CGR method was applied, the QSSA is applied to the skeleton mechanism to generate the reduced mechanism as the third and final step in the reduction. This is done for the N-Heptane mechanism in chapter 6.
Each mechanism is validated against the mechanism in the previous step. Hence, the reduced mechanism is validated against the skeleton mechanism, the skeleton mechanism is validated against the lumped mechanism and the lumped mechanism is validated against the detailed mechanism. An alternative is that the skeleton mechanism is skipped and QSSA is applied directly to the lumped mechanism to generate the reduced mechanism.

Normally the reduction procedure shown in Figure 5.2 is used. The reaction flow analysis and/or sensitivity analysis is applied directly to the detailed mechanism to generate the skeleton mechanism. Thereafter the QSSA is applied to the skeletal mechanism to give the reduced mechanism. An alternative is that the skeleton mechanism is skipped and QSSA is applied directly to the detailed mechanism to generate the reduced mechanism. This is done for the Methane/Propane mechanism in chapter 6.

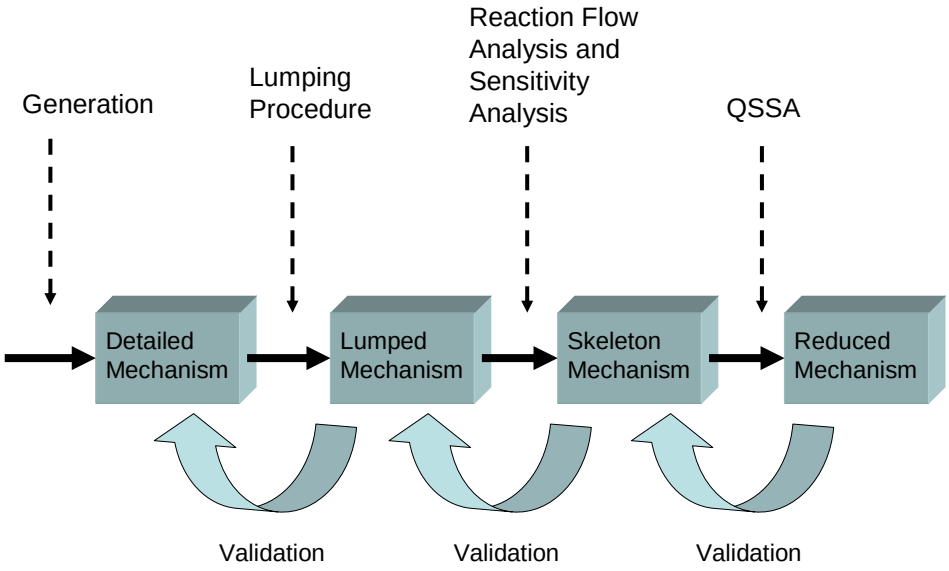


Figure 5.1. The procedure of generation a reduced mechanism. First a detailed mechanism is produced, which then is lumped. Thereafter a skeleton mechanism is produced by applying reaction flow analysis and sensitivity analysis to the lumped mechanism. Finally the reduced mechanism is produced by applying QSSA to the skeleton mechanism. All mechanisms are validated towards the previous mechanism.

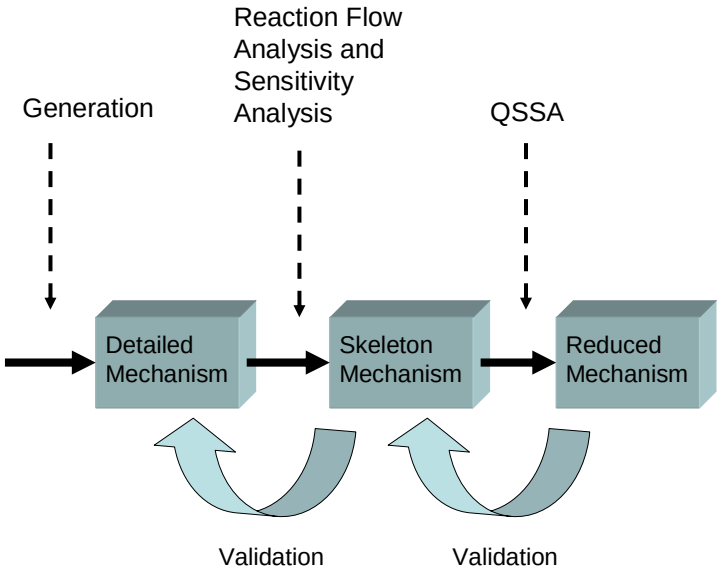


Figure 5.2. The figure is very similar to Figure 5.1 with the exception that the lumped mechanism is skipped and the reaction flow and sensitivity analysis is applied directly to the detailed mechanism and QSSA is applied to the skeletal mechanism.

5.4.1.1. Chemical Lumping and lumped mechanism

Chemical lumping is a technique to reduce the complexity of a chemical model and results in a lumped mechanism. In linear chemical lumping hydrocarbon species with the same functional groups, the same distance between functional groups and the same degree of branching can be represented by only one species.
This procedure can reduce the complexity significantly but with the cost of lower accuracy of the lumped mechanism compared to the detailed mechanism. The decrease in accuracy accepted is decided by the user.
The CPU time saved is a function of the number of ODEs that are removed, i.e. the number of species lumped. The CPU time decreases approximately quadratically if a Newton solver is used for the system of ODE. This is discussed in detail in section 3.5.3.

5.4.1.2. Species removal and skeletal mechanism

The skeletal mechanism is constructed from the lumped mechanism. Some species of the lumped mechanism are completely removed to give the skeletal mechanism. The species removal procedure involves determination of the fluxes in the mechanism and elimination of the species involved in the minor paths. This is done by Reaction Flow Analysis and sensitivity analysis. [19]
The species removal procedure can reduce the complexity significantly but with the cost of lower accuracy of the skeletal mechanism compared to the lumped mechanism.
The CPU time that is saved is a function to the number of ODEs that are removed, i.e. the number of species removed. The CPU time decreases approximately quadratically if a Newton solver is used for the system of ODE. This is discussed in detail in section 3.5.3.

The skeletal N-Heptane mechanism [16] used in this thesis consists of 110 species and 1170 reactions. This skeletal mechanism, which is referred to as the original mechanism in chapter 6, is the starting point for the reduction in this thesis.

5.4.2. The reduced mechanism

The reduced mechanism is constructed from the skeletal or lumped mechanism. The QSSA is applied to some species, giving the reduced mechanism. (The partial equilibrium assumption can be applied to some reactions to reduce the mechanism also, but this method is not used in this work).
This QSSA can reduce the complexity significantly but with the cost of lower accuracy of the reduced mechanism compared to the skeletal mechanism. The decrease in accuracy that is accepted is decided by the user.
The CPU time for the system of ODE decreases approximately quadratically if a Newton solver is used for the system of ODE. This is discussed in detail in section 3.5.3.
However, the CPU time for the system of NAE increases approximately quadratically if a Newton solver is used for the system of NAE. This complex dependence of the CPU time on the number of species set in QSS is explained in detail in section 3.5.3. The approach to reduce the CPU time for the system of NAE is discussed in detail in section 4.3.3.

5.4.2.1. QSS species selection

The QSSA is used for some species in order to reduce the lumped or skeletal mechanism. A detailed discussion of QSSA can be found in section 2.3.5.4.
QSS species can be selected by various methods. The methods used in this thesis are;

- Life Time Analysis
- LOI
- Concentration
- LTC

All methods are presented below and aim at ranking the QSS species in such a way that the species with low ranking are supposed to be appropriate QSS species, while the species with high ranking are supposed to be non appropriate QSS species. All species have ranking values between zero and one.
All methods have advantages and disadvantages. Some of the methods rank the QSS species well, others do not. Some are expensive in terms of CPU time, while others are cheap. The optimal methods rank the QSS species well to a low CPU cost. The methods presented below are all tested and compared in section 6.2.2.5.

None of the methods used in order to rank the QSS species works perfect. Hence, sometimes non appropriate QSS species can have a low ranking and vise versa. This means that only a few species on the ranking list can be set into QSS and that many appropriate QSS species are missed. This motivates the ART, which is discussed in section 5.4.2.2.

5.4.2.1.1. Time scale separation methods

Chemical reactions occur on time scales ranging over several orders of magnitude, leading to stiff systems of ODEs, which are very time consuming to solve. The stiffness can be reduced by means of a time scale analysis, which identifies the fast and slow reacting species to further reduce the original mechanism. The manner in which the fast and slow sub-mechanisms are identified has been the subject of intensive research over the last years [6,10-12]. Such procedures rely on the QSSA for species reacting on the shortest chemical time scales, replacing ODE for these species with simpler NAE.

The time scale separation method used in this thesis is LT analysis, which is described in detail below. However, there are other methods based on time scale separation that can be used to select QSS species other than the one used in this thesis. Two of those methods are CSP and ILDM [14]. The CSP method identifies and separates fast and slow modes by repeated basis rotation. The fast modes are thought of as exhausted and therefore results as a set of NAE.
The fast and slow mode separation must be done locally at each time step, which means that the CSP method has a high cost due to the refinement procedure that leads to the

mode separation. This cost increases with the size of the mechanism. The CSP method is sometimes combined with the concentration of the species, which acts as a kind of sensitivity measure. However, the concentration calculation cost much less than the sensitivity calculations.
The LOI measure, described below, instead uses the low cost LT instead of the high cost CSP in order to separate the time scales. However, the sensitivity analysis in LOI is computationally expensive.

5.4.2.1.1.1. Life Time (LT) Analysis

The LT of a species can be used as an indicator whether or not a species is a good or bad QSSA species. The LT of a species, at a certain moment in time, can be found from the diagonal element, $\mathbf{J}_{ii}$, of the Jacobian. The motivation for this is found in section 2.3.5.4.2. The LT of species, *i*, is [19];

$$\tau_i = -1/\mathbf{J}_{ii} = -1/\frac{\partial \omega_i}{\partial x_{D,i}} = -\frac{x_{D,i}}{\sum_{k=1}^{N_R} \nu_{ik} \cdot r'_k \cdot \nu'_{ik}} \tag{5.5}$$

where, ω_i, is the source term for species, *i*, and, $x_{D,i}$ is the concentration of species *i*. The denominator is the sum of the consuming reactions of species *i*. The LT can therefore be understood as a measure of how fast a particular species is consumed after being produced.
Hence, small LT gives small instantaneous error of the QSS species while large LT gives large instantaneous error of the QSS species. This is a motivation for using LT as an indicator for QSS species. Hence, short LT indicates that the species is a good QSS species, while long LT indicates a bad QSS species.

The LT of a species varies with time since the reaction rates and the species concentrations vary with time. Hence, the LT can be observed at all point is time during the combustion process. Which LT should be used? There are a couple of choices. Work done by previous members of the Division of Combustion Physics at Lund University has shown that LT at the maximum concentration of the species gives good results. Hence, this thesis also uses LT at the maximum concentration of the species. The short notation for this will be LT at MaxS. Also, LT at specific points in time like CF and HF is used in this thesis.

The LT of a species can also be used in combination with sensitivity and concentration to create the LOI and LTC index respectively. Both the LOI and LTC are in turn used for indicating whether or not a species is an appropriate or non appropriate QSSA species.

5.4.2.1.2. Sensitivity analysis

Sensitivity analysis is used to eliminate unimportant reactions and/or species from the detailed mechanism. However, sensitivity analysis is used only for species in this thesis. Sensitivity can be used as an indicator to classify species as “appropriate” or “non appropriate” QSS species. Low sensitivity indicates an appropriate QSS species, while high sensitivity indicates a non appropriate QSS species.
The sensitivity analysis is an expensive method compared to the others presented in this thesis.

Sensitivity can also be combined with other parameters to give new parameters. An example of this is LOI (see section 5.4.2.1.3.).
Sensitivity analysis means to determine the influence of one parameter on another selected parameter. An example of this is how much a change in species concentrations affects another species concentration or temperature. Another example of this is how much a change in reaction coefficients affects species concentration or temperature. The sensitivity of a parameter, k, on a selected parameter, Q, is;

$$S_{Q,k} = \frac{\partial Q}{\partial k} \tag{5.6}$$

The OH and HO_2 concentrations are important parameters for the combustion process. The sensitivity of other species on OH and HO_2 concentration are;

$$S_{OH,x_j} = \frac{\partial [OH]}{\partial x_j}, \quad j = 1,...,N_S - 1 \tag{5.7}$$

$$S_{HO_2,x_j} = \frac{\partial [HO_2]}{\partial x_j}, \; j = 1,...,N_S - 1 \tag{5.8}$$

where, x_j, are the other species concentrations and N_S is the number of species. The sensitivity for OH and HO_2 are used in this thesis and are further discussed in section 6.2.2.5.
Although sensitivity analysis is quite simple to apply, it is very time consuming due to the large number of equations to be solved. The sensitivity analysis is produced by running the IGNITION program (see section 5.4.2.2.) with the detailed mechanism in a pre-processing step.
A derivation of the sensitivity equations is found in the chapter appendix.

5.4.2.1.3. Level Of Importance (LOI)

The LOI list is produced by running the IGNITION program (see section 5.4.2.2.) with the detailed mechanism in a pre-processing step. LOI is a measure used to classify species as “Appropriate” or “Non Appropriate” QSS species. The LOI measure is a product of LT and sensitivity, S. Hence, the LOI is an expensive method compared to the others presented in this thesis, since it involves sensitivity analysis.

The LOI for species, *i*, with sensitivity taken with respect to the specific quantity, *Q*, is;

$$\left(LOI\right)_i^Q = \tau_i \cdot S_i^Q \tag{5.9}$$

where τ_i is the LT of species *i* and S_i^Q is the sensitivity of species *i* on species *Q*. This means that species with short LT and low sensitivity will have low LOI values, while species with long LT and high sensitivity will have high LOI values. Species with short LT but high sensitivity and vise versa will have medium LOI values. The LOI value ranges from zero to one and generally relates to QSSA in the following way;

- The species with the lowest LOI are the most appropriate species for the QSSA.
- The species with the highest LOI are the least appropriate species for the QSSA.
- The species with intermediate LOI value can be either appropriate or non appropriate QSS species and are hard to classify.

The relation between LOI and QSSA is illustrated in Figure 5.4. The relation between QSSA, LOI, LT and sensitivity is shown in Table 5.1.

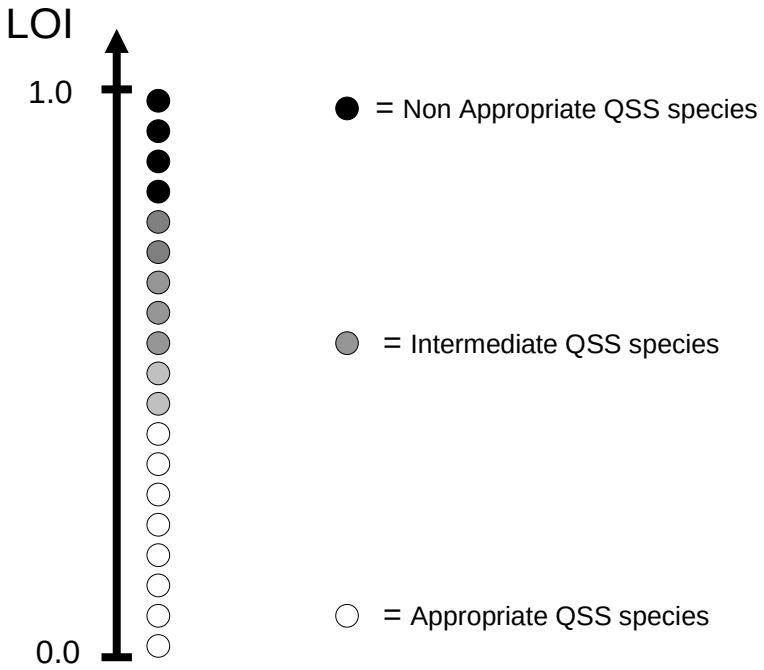


Figure 5.4. A schematic illustration of the LOI measure. The LOI values goes from zero to one. The appropriate QSS species have low LOI values and are represented by white circles, while the non appropriate QSS species have high LOI value and are represented by black circles. The intermediate QSS species have LOI value somewhere in between and are represented by grey circles.

	Long LT	**Short LT**
High Sensitivity	**High LOI** *Non Appropriate QSS Species*	**Medium LOI** *intermediate QSS Species Hard to classify*
Low Sensitivity	**Medium LOI** *intermediate QSS Species Hard to classify*	**Low LOI** *Appropriate QSS Species*

Table 5.1. The table shows the LOI value and its relation to the QSSA for different combinations of LT and sensitivity.

5.4.2.1.4. Concentration

Whenever a species is set in QSS eq(5.10) to eq(5.13) are affected. The total enthalpy or internal energy and element mass fraction must always be constant. The total enthalpy is used for a CPR and the internal energy is used for a CVR.

$$\sum_{i=1}^{N_D} Y_i = 1 \tag{5.10}$$

$$\sum_{i=1}^{N_D} h_i Y_i = H_{TOT} \tag{5.11}$$

$$\sum_{i=1}^{N_D} u_i Y_i = U_{TOT} \tag{5.12}$$

$$\sum_{i=1}^{N_D} \eta_i^a Y_i = const.,\ a = 1,..,N_{atoms} \tag{5.13}$$

where η_i^a is the number of atoms of atom *a* in species *i*. In this thesis *a* is either O, H, C and N. Hence, N_{atoms}=4.
Whenever a species is set in QSS the species is excluded from the ODE system. This means that the element mass fraction of the species with the same atoms as the excluded QSS species are increased in order to compensate for the excluded QSS species so that eq(5.13) is fulfilled. This in turn affects the mass fractions of other species as well, which leads to a chain of events. An example is shown in the chapter appendix.

Hence, QSS species with high species mass fractions affect the system of ODE much and are therefore not recommended as QSS species [19]. The opposite is true for species with low species mass fractions.
Also, the mass fractions of the non QSS species are increased in order to compensate for the enthalpy and internal energy of the QSS species excluded from the ODE system. This means that the temperature of the system is affected via c_p or c_v, since it is calculated from the enthalpy or internal energy.
Hence, QSS species with high species mass fractions and high enthalpy or internal energy affect the system of ODE much and are therefore not recommended as QSS species [19]. The opposite is true for species with low species mass fractions and low enthalpy or internal energy.
Hence, the mass fraction and thereby concentration is an indicator of how much the system is affected by the QSSA.

Also, whenever a species is set as QSS an error is introduced in eq.(2.52) to (2.54). The energy equation is affected since the source term is zero for the QSS species. Hence,

species with large source terms and large internal energy will affect the energy equation more if they are set in QSS and vise versa.
The ODEs for the species concentrations are affected since the source term includes QSS species concentrations.

A reason for the concentration to be a good measure for QSS species ranking is that the source term of the ODEs is a function of the species concentrations according to eq.(2.53). The larger a specific species concentration is and the more reactions the species is involved in, the more it affects the system of ODE.
Hence, if the species concentrations are observed at a specific point in time, the concentrations will be proportional to and thereby rank the impact the species have on the system of ODE at that specific point in time. This means that species with low concentrations will be appropriate QSS species, since they will have low impact on the system of ODE. The opposite is true for species with high concentrations.

The calculation of species concentrations must be done anyway and does not cost any extra CPU time. Hence, it is a cheap method of selecting QSS species.

5.4.2.1.5. LTC

LTC is the product of LT and concentration,

$$(LTC)_i = \tau_i \cdot Concentration_i \tag{5.14}$$

Hence, the LTC has a low CPU cost since both LT and concentration are cheap to calculate.
The use of and reasoning behind LTC is the same as for LOI. The only difference is that concentration is used instead of sensitivity. The relation between QSSA, LTC, LT and concentration is shown in Table 5.2.

	Long LT	**Short LT**
High concentration	**High LTC** *Non Appropriate QSS Species*	**Medium LTC** *intermediate QSS Species* *Hard to classify*
Low concentration	**Medium LTC** *intermediate QSS Species* *Hard to classify*	**Low LTC** *Appropriate QSS Species*

Table 5.2. The table shows the LTC value and its relation to the QSSA for different combinations of LT and concentration.

5.4.2.2. The Automatic Reduction Tool (ART)

The ART is based on a PERL-program that repeatedly (in a looping fashion) calls the preprocessor and the solver (written in FORTRAN 90/95) at different reduction levels. The ART thereby automatically builds up a QSS species list that is validated for user defined ranges of physical conditions and user defined demands on the ART Evaluation Targets (ART ET). The physical conditions used in this thesis are temperature interval, pressure interval and fuel/air ratio interval.
The ART ET controls the deviation between the reduced mechanism and the detailed mechanism for important features of the ignition process such as ignition delay time, concentration profiles, temperature profiles and CPU time. The ART ETs used in this thesis are discussed in detail in section 5.4.2.2.2.

Figure 5.5 illustrates how the ART works. The figure shows the input the ART uses and the output the ART produces. The input is the user defined ranges of physical conditions and the user defined demands on the ART ET, for which the reduced mechanism shall be valid. The output is a reduced mechanism, which is valid for the input parameters.

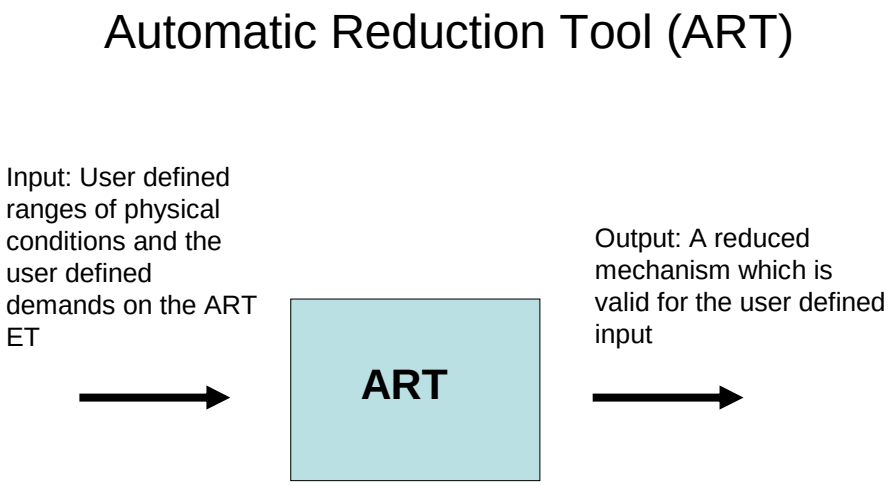


Figure 5.5. The input/output model of the ART. The input is the user defined ranges of physical conditions and the user defined demands on the ART ET, for which the reduced mechanism shall be valid. The output is a reduced mechanism, which is valid for the input parameters.

The ART is built up of two programs, “REDKIN” and “IGNITION”, which cooperate in a looping-fashion that is illustrated in Figure 5.6. REDKIN chooses the reduction level and produces output files in the form of solver subroutines used by IGNITION. The solver subroutines are the “hard-coded” and optimized GE and BS for the inner solver, which are discussed in detail in section 4.3. The IGNITION program then produces output files, which are used as input for REDKIN. These output files tell weather or not the calculation based on the reduced mechanism was within the accuracy limits of the validation parameters. The loop is then started again when REDKIN chooses a new reduction level and produces new output files that are used by IGNITION.

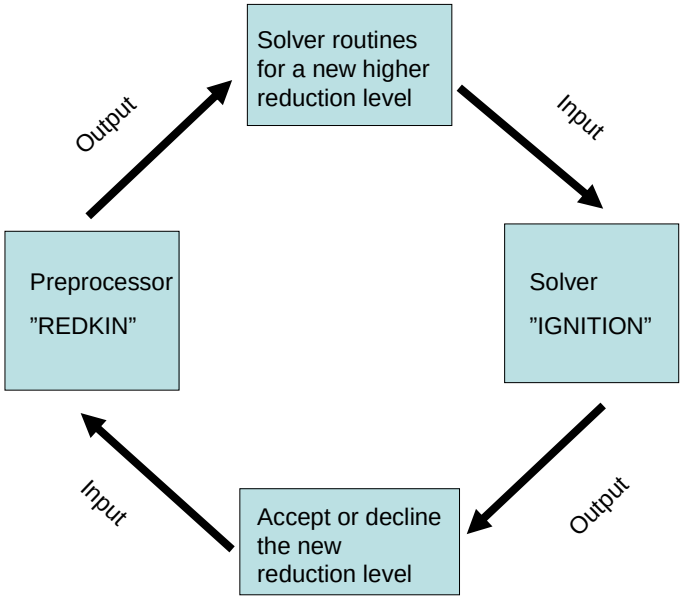


Figure 5.6. The cooperation between the programs “REDKIN” and “IGNITION” in a looping-fashion. The output files from REDKIN, which are solver subroutines based on a higher reduction level than before, acts as input for IGNITION. The output files from IGNITION, which tell weather or not the new reduction level was accepted, acts as input for REDKIN.

The reduced mechanism can be generated by setting the species with low LOI value in QSS and stop somewhere in the intermediate region where the accuracy criteria are no longer fulfilled. But the LOI measure is not perfect. Sometimes species with low LOI value are actually very non appropriate QSS species, while species with high LOI value are appropriate QSS species. For this reason the reduction ART wants to skip the species with low LOI value that are actually very non appropriate QSS species and thereafter continue to add QSS species to the reduced mechanism based on the LOI list.
Figure 5.7 shows an illustration of a typical LOI list. By using only the LOI measure and no manual manipulation, the number of appropriate QSS species (white circle) in Figure 5.7 would be only the number of white circles up to the first non appropriate QSS species

(black circle). What this number is in a real case depends on the mechanism and the parameter range the LOI has been validated for.
If the first non appropriate QSS species automatically could be taken away from the LOI list the number of appropriate QSS species would increase until the next non appropriate QSS species came along. If this next non appropriate QSS species automatically could be taken away from the LOI list the number of appropriate QSS species would increase until the yet another non appropriate QSS species came along and so on. In this way all the non appropriate QSS species are taken away and the QSS lists is only made up of the appropriate QSS species.

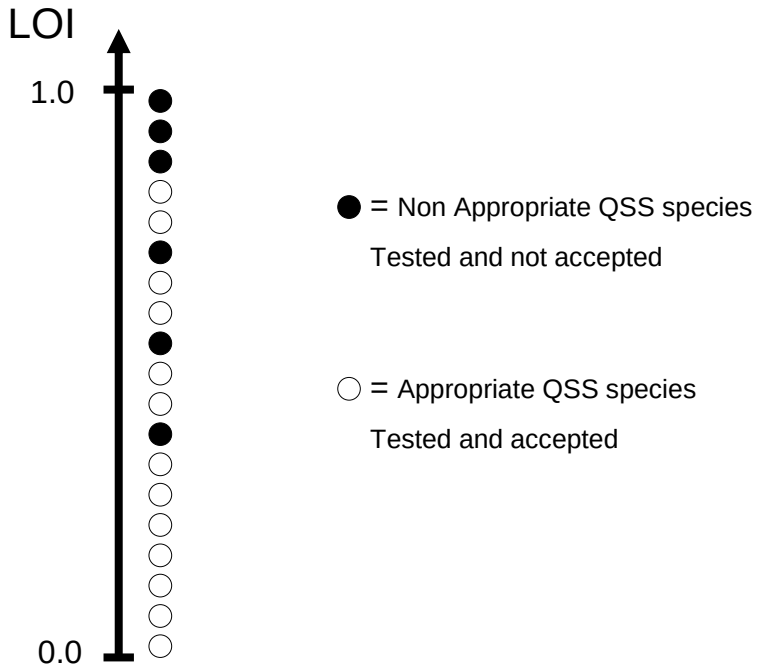


Figure 5.7. A schematic illustration of the incomplete relation ship between LOI and QSS species. The appropriate QSS species are supposed to have low LOI values, while the non appropriate QSS species are supposed to have high LOI value. Sometimes species with quite low LOI value are non appropriate QSS species, which motivates the ART.

5.4.2.2.1. The ART-algorithm

The ART-algorithm goes as follows:

Begin ART-loop

1. In the program REDKIN: Select the species with the lowest LOI-value that has not been tried yet and add it to the QSS species list.
2. In the program REDKIN: Run the program REDKIN and produce the solver subroutines based on the QSS species list.
3. Copy and transfer the subroutines from REDKIN to the program IGNITION.
4. In the program IGNITION: Run the program IGNITION and produce a diagnosis file on whether or not the QSS species list fulfills the demands on the ART ET.
5. Copy and transfer the diagnosis file from IGNITION to REDKIN.
6. In the program REDKIN: Check diagnosis file:
7. If the diagnosis was OK; keep the selected QSS species list. Go to point 1.
8. If the diagnosis was NOT OK; remove the last species from the QSS species list. Go to point 1.

End ART-loop

The flow chart for the ART is presented in Figure 5.8.

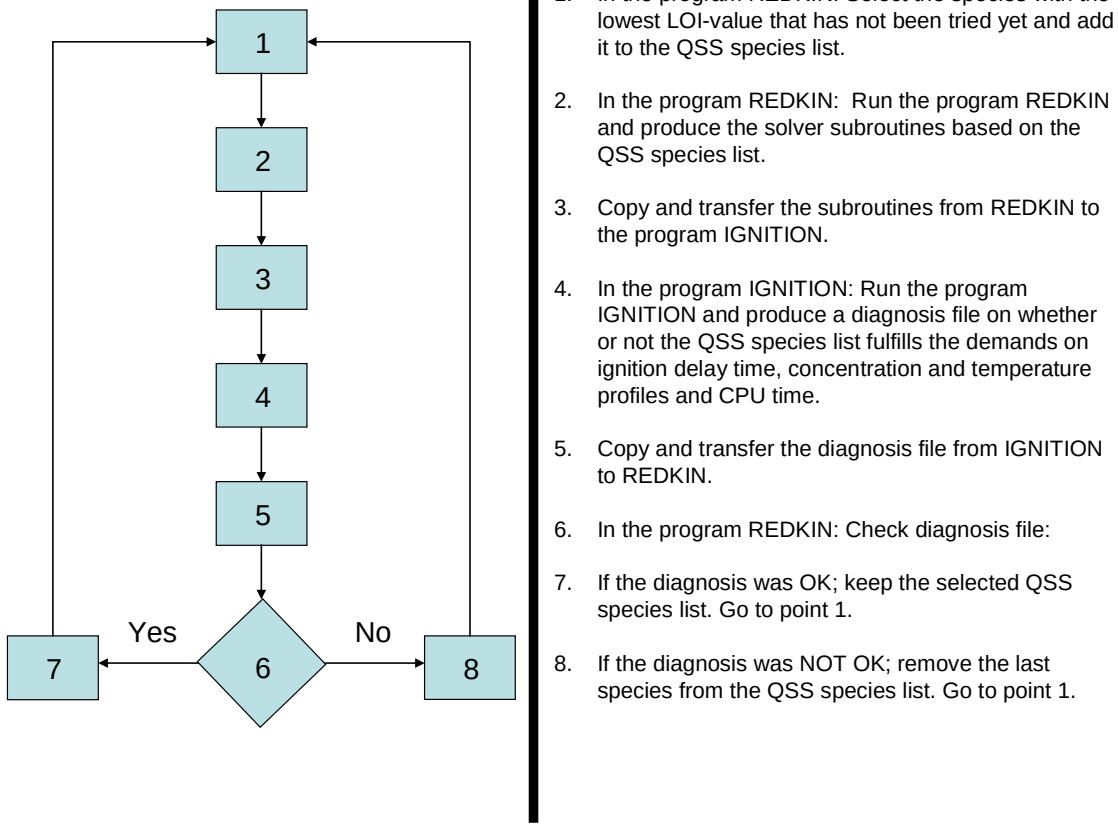


Figure 5.8. Flow chart for the ART that shows the cooperation between the programs REDKIN and IGNITION.

5.4.2.2.2. Mechanism validation by the ART

The reduced mechanism must be compared to the non reduced mechanism to see if it is valid or not. The validation always involves a user defined range of physical conditions for which the reduced mechanism must be valid. The range of physical conditions consists of used in this thesis are;

- A temperature interval
- A pressure interval
- A fuel/air ratio interval

Often an interval consists of a single point only.

The mechanism must also be validated for the ART ET, which are used to control the accuracy of the solution of the reduced mechanism for all physical conditions. The ART ETs are chosen in a manner that controls the part of the solution the user finds interesting. Each ART ET is in turn controlled by ART ET limits, which are set by the user. This is discussed further in section 6.1. The ART ETs in this thesis are;

IDT CF
This is the ignition delay time for the first ignition corresponding to the CF. In this thesis it is measured by the time for Max HO2 CF, which is explained below.

IDT HF
This is the ignition delay time for the second or main ignition. In this thesis it is measured by the time for Max OH HF, which is explained below.

Max HO2 CF
This is the maximum value of the HO_2 concentration at the first peak corresponding to the first ignition.

Max OH HF
This is the maximum value of the OH concentration corresponding to the second ignition.

Chemical species
This is the maximum value of the concentration of the selected chemical species. The selected chemical species can be everything from one to all species.

There are some additional ART ETs that can be used to control the solution. However, these have not been used in this thesis. The additional ART ETs are;

CPU time
This controls the total CPU time for a reduced mechanism.

Max temperature
This controls the maximum temperature of the reduced mechanism.

QSS Jacobian
This controls the number of Jacobians for the system of NAE, which means that it controls the convergence of the system of NAE. Hence, it indirectly controls the total CPU time for a reduced mechanism.

5.5. Chapter References

[1] E. Ranzi, P. Gaffuri, T. Faravelli, P. Dagaut, Combust. Flame 103 (1995) 91-106.
[2] F. Battin-Leclerc, Prog Energy Combust Sci 34 (2008) 440-498.
[3] H.J. Curran, W.J. Pitz, C.K. Westbrook, Abstracts of papers of the American chemical society 215 (1998) U162-U162
[4] G. Moreac, E.S. Blurock, F. Mauss, Combust. Sci. Technol. 178 (2006) 2025-2038
[5] S.S. Ahmed, F. Mauss, G. Moreac, T. Zeuch, PCCP 9 (2007) 1107-1126
[6]A.S. Tomlin, T. Turnyi, M.J. Pilling, in: M.J. Pilling (Ed.), *Low-Temperature Combustion and Autoignition*, Comprehensive Chemical Kinetics, Vol. 35, Elsevier, 1997
[7] S.S. Ahmed, F. Mauss, G. Moreac, T. Zeuch, PCCP 9 (2007) 1107-1126
[8] T. Lövås, Combust. Flame. 156 (2009) 1348-1358
[9] T. Lövås, D. Nilsson, F. Mauss, Proc. Combust. Instit. 28 (2000) 1809-1815
[10] T. Turányi, New J. Chem 14 (1990) 795–803.
[11] F. Mauss, N. Peters, in: N. Peters, B. Rogg, (Eds.), *Reduced Kinetic Mechanisms for Application in Combustion Systems*, Lecture Notes in Physics, New Series, m 15, Springer Verlag, 1993, p. 58-75.
[12] U. Maas, S.B. Pope, *Laminar flame caluculations using simplified chemical kinetics based on intrinsic low-dimensional manifolds,* Proc. Comb. Inst. 25 (1994) 1349-1356.
[13] S.H. Lam, D.A. Goussis, *The CSP Method for Simplifying Kinetics*, International Journal of Chemical Kinetics 26 (1994), 461-486.
[14] T. Lu, C.K. Law, *Toward accommodating realistic fuel chemistry in large-scale computations*, Progress in Energy and Combustion Science 35 (2009) 192-215
[16] S.S. Ahmed, F. Mauss, G. Moreac, T. Zeuch, Combustion and Flame Elsevier 155 (2008) 651-674
[17] S.S. Ahmed *A Detailed Modeling Study for Primary Reference Fuels and Fuel mixtures and Their use in Engineering applications* (2006), Doctoral Thesis, Combustion Physics, Lund University, Sweden, Lund reports on Physics, LRCP-115 ISSN 1102-8718 ISRN LUTFD2/TFCP—115—SE ISBN91-628-7013-0
[18] T. Turanyi, A.S. Tomlin, M.J. Pilling, *On the error of the Quasi-Steady-State Approximation* J. Phys. Chem. 1993, 97, 163-172
[19] T. Lövås, Automatic *Reduction procedures for Chemical mechanisms in Reactive Systems*, (2002), Doctoral Thesis, Combustion Physics, Lund University, Sweden, Lund reports on Physics, LRCP-78, ISRN LUTFD2/TFCP—78—SE
[20] S. R. Turns, *An Introduction to Combustion Concepts and Applications* second edition, McGRAW-HILL INTERNATIONAL EDITIONS 2000
[21] E.L. Petersen, D.M. Kalitan, S. Simmons, G. Bourque, H.J. Curran, and J.M. Simmie, (2006). Proc. Comb. Inst., vol. 31, Elsevier.

5.6. Chapter Appendix

A.5.1. Sensitivity

If the original set of equations are;

$$\frac{d\mathbf{x}}{dt} = \mathbf{g}(\mathbf{x},\mathbf{k}) \tag{A.5.1}$$

Where **k** can be a vector of constants or be equal to the **x**-vector.
The sensitivity matrix is;

$$\mathbf{S} = \frac{\partial \mathbf{x}}{\partial \mathbf{k}} \tag{A.5.2}$$

And each matrix element is;

$$\mathbf{S}_{i,k_j} = \frac{\partial x_i}{\partial k_j} \tag{A.5.3}$$

The time derivative of the sensitivity matrix is;

$$\frac{\partial \mathbf{S}}{\partial t} = \frac{\partial}{\partial t}\left(\frac{\partial \mathbf{x}}{\partial \mathbf{k}}\right) = \frac{\partial}{\partial \mathbf{k}}\left(\frac{\partial \mathbf{x}}{\partial t}\right) = \frac{\partial \mathbf{g}(\mathbf{x},\mathbf{k})}{\partial \mathbf{k}} = \frac{\partial \mathbf{g}}{\partial \mathbf{k}} + \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{k}} = \frac{\partial \mathbf{g}}{\partial \mathbf{k}} + \mathbf{J} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{k}} = \frac{\partial \mathbf{g}}{\partial \mathbf{k}} + \mathbf{J} \cdot \mathbf{S} \tag{A.5.4}$$

where

$$\mathbf{J} = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \tag{A.5.5}$$

is the Jacobian.

A.5.2. Element mass fraction

Whenever a species is set in QSS the species is excluded from the ODE system. This means that the element mass fraction of the species with the same atoms as the excluded QSS species are increased in order to compensate for the excluded QSS species so that eq(5.13) is fulfilled. This in turn affects the mass fractions of other species as well, which leads to a chain of events.

For example, if H is set as QSS, the mass fraction of species like OH that include H atoms must increase in order to fulfill the element mass fraction conservation of the H atoms.
This in turn means that the mass fraction of species like O_2 that include O atoms must decrease in order to fulfill the element mass fraction conservation of the O atoms.

This in turn means that the mass fraction of species like CO that include C atoms must decrease in order to fulfill the element mass fraction conservation of the C atoms.

This in turn means that the mass fraction of species like CH_4 that include C atoms must increase in order to fulfill the element mass fraction conservation of the C atom.

Chapter 6.

Results and Discussion

6.1. Chapter introduction

This chapter will investigate the performance of the solver combinations and the performance of the ART on an N-Heptane and Methane/Propane mechanism. The investigation based on N-Heptane will be more rigorous, since is a more complex mechanism. Both mechanisms will be investigated for various physical conditions.

The purpose of the ART is to produce a heavily reduced mechanism with high accuracy and low CPU time compared to the original mechanism. For this reason the reduction level, CPU time and accuracy must be investigated at the same time. The user can then choose the reduced mechanism, which has the best combination of reduction level, CPU time and accuracy for the particular application of interest.

The CPU time and the accuracy of the reduced mechanisms are investigated in order to determine the performance of the solver combinations. The accuracy of the reduced mechanisms is evaluated by the ART for a set of Evaluation Targets (ART ET). The ART ET is represented by IDT HF, IDT CF, Max HO2 CF and Max OH HF for all cases. In section 6.2.2.4, some extra ETs for chemical species are added to the others.

The ART compares the deviation in the ART ET of the reduced mechanism to the previously reduced mechanism in order to determine whether or not the reduced mechanism shall be accepted or not. If

$$\Delta\, ART\ ET_{RED}\,(\%) < ART\ ET_{LIMIT}^{RED} \tag{6.1}$$

the reduced mechanism is accepted and kept and a new reduced mechanism will be tested next. If the opposite is true the reduced mechanism is not accepted, which means that the previously reduced mechanism is kept and a new reduced mechanism will be tested next. $ART\ ET_{LIMIT}^{RED}$ is the ART ET Limit when the comparison is made with the previously reduced mechanism. $\Delta\, ART\ ET_{RED}\,(\%)$ is explained below. The deviation of the ART ET that is compared to the ART ET limits is calculated as follows;

$$\Delta\, ART\ ET_{RED}\,(\%) = \frac{\left(ART\ ET_{RED} - ART\ ET_{PREVRED}\right)}{ART\ ET_{PREVRED}} \cdot 100 \tag{6.2}$$

The $ART\ ET_{RED}$ and $ART\ ET_{PREVRED}$ are the values of a particular ART ET for the reduced and previously reduced mechanism respectively.

The ART also has the option to compare the reduced mechanism to the original mechanism. If

$$\Delta\, ART\ ET_{ORG}\,(\%) < ART\ ET_{LIMIT}^{ORG} \tag{6.3}$$

the reduced mechanism is accepted and kept and a new reduced mechanism will be tested next. If the opposite is true the reduced mechanism is not accepted, which means that the previously reduced mechanism is kept and a new reduced mechanism will be tested next. $ART\ ET_{LIMIT}^{ORG}$ is the ART ET Limit when the comparison is made with the original mechanism. $\Delta\, ART\ ET_{ORG}\,(\%)$ is explained below. The deviation of the ART ET is calculated as follows;

$$\Delta\, ART\ ET_{ORG}\,(\%) = \frac{\left(ART\ ET_{RED} - ART\ ET_{ORG}\right)}{ART\ ET_{ORG}} \cdot 100 \tag{6.4}$$

The $ART\ ET_{RED}$ and $ART\ ET_{ORG}$ are the values of a particular ART ET for the reduced and original mechanism respectively.

The deviation of the ART ET of the reduced mechanism compared to the original mechanism is of interest when the accuracy of the entire reduced mechanism is investigated. Hence, all figures that involves deviation of some ART ET will be based on eq.(6.4) if nothing else is stated in the text. The $\Delta\, ART\ ET_{ORG}\,(\%)$ used in this thesis are called Δ IDT HF (%), Δ IDT HF (%), Δ Max HO2 CF (%) and Δ Max OH HF (%).

The ART also has a “second chance” option which is useful when some of the species are just on the border of being accepted the first time they are tested. Hence, some of the species that were on the “wrong side” of the border can be on the “right side” of the border when they are tested for the second time. This happens if $ART\ ET_{RED}$ and $ART\ ET_{PREVRED}$ in eq.(6.2) are changed favorable compared to the first attempt.

In almost all the figures that show a normalized quantity of some sort, the quantity of the reduced mechanisms was normalized with the corresponding quantity of the original mechanism with 0 QSS species.

The only exception are the figures that show quantities from the system of NAE, where the normalization was done with a mechanism containing two QSS species. The reason for this is that normalization with the original mechanism and a reduced mechanism with only one QSS species does not make any sense for the number of Jacobians and BS of the system of NAE.

6.2. N-Heptane mechanism

The N-Heptane mechanism [1] used in the following simulations is a skeletal mechanism that contains 110 species and 1170 reactions.
The N-Heptane mechanism does contain low temperature chemistry and a CF exists for most physical conditions in the following simulations. However, the CF does not exist for all physical conditions in the following simulations. For this reason the IDT CF instead represents the time of the maximum HO_2 peak for some physical conditions (with high initial temperature). Hence, the Max HO2 CF represents the Max HO2 HF.
The IDT CF and IDT HF are then almost but not exactly identical. The reason for this is that the HO_2 and OH profiles have their peaks at slightly different points in time and that they are affected differently by the QSS species.

The ART was applied to different physical conditions, which all contain an initial temperature range, an initial pressure range and a fuel/air equivalence ratio range. This gives a set of cases shown in Table 6.1.

Table 6.1. The table shows the physical conditions for four different cases.

Case:	Temperature [K]	Pressure [bar]	Equivalence ratio [-]
Case I	900	40	1
Case II	625-1300	40	1
Case III	625-1300	40	0.5-2.0

The ART used the same LOI list for all cases when the reduced mechanisms were generated. The LOI list that was used was generated in a pre-processing step for Case I. This LOI list corresponds to "**Max (LOI MaxS Sens$_{OH}$, LOI MaxHO2 CF Sens$_{HO2}$)**", which is described in section 6.2.2.5.

All simulations for the N-Heptane-mechanism are made for Case I if nothing else is stated.

6.2.1 Variation of inner solver

This section investigates the performance of two solver combinations on the reduced mechanisms that the ART produces for the following physical conditions (Case I);

- Temperature point: 900 K
- Pressure point: 40 bar
- Fuel/air ratio point: 1.0

The following ART ET limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 1 %

6.2.1.1 Time step size variation

This section investigates the performance of two different solver combinations and their reaction to different outer time step sizes, which are 10^{-6}, 10^{-5} and 10^{-4} s. Both solver combinations use a modified Newton solver as an outer solver, while the inner solver, which is either a Newton solver or an FP solver, differs. The solver combination that uses an FP solver as an inner solver will in the following be referred to as Newton-FP solver, while the solver combination that uses a Newton solver as an inner solver will in the following be referred to as Newton-Newton solver. Both solver combinations are expected to give similar results for the same outer time step size. Also, each solver combination is expected to give similar results if the outer time step size varies with several orders of magnitude, since each solver combination uses adaptive time step size.

Figure 6.1 shows number of QSS species vs LOI rank. The most reduced mechanism of 73 QSS species is exactly the same for all outer time step sizes and both solver combinations. Whenever a species with a certain LOI rank is not accepted as a QSS species the curve takes a step to the right. (This behavior is the same whenever this kind of figure is shown below). Hence, the first species that is not accepted has LOI rank 47. This number depends on the QSS species ranking lists and ART ET limits.

Figure 6.2 shows the normalized CPU time vs number of QSS species for the two solver combinations using an outer time step size of 10^{-6}, 10^{-5} and 10^{-4}. There is a clear distinction in normalized CPU time between the two solver combinations but a very small distinction between the different outer time steps for each solver combination, which shows robustness in both solver combinations.

The ratio in CPU time between the two solver combinations for all outer time steps is shown in Figure 6.3. The ratio increases dramatically beyond 56 QSS species and reaches more than a factor 120 for the most reduced mechanism. The dramatic increase is mainly due to the inclusion of "difficult" QSS species in the system of NAE, which affects the convergence of the Newton-FP solver negatively, but also due to the size of the system of NAE. The dramatic increase that can be seen for the CPU time can also be seen for the number of FP iterations, which is shown in Figure 6.4. Hence, the convergence problem in the FP solver is responsible for the increase in CPU time.

The species responsible for the dramatic increase in CPU time for the Newton-FP solver, is the species "HCCO", which is the 57:th accepted QSS species with LOI rank 66 (see Table 6.2). If the ART is set not to accept QSS species that contributes to enormous CPU time changes, the species HCCO will not be accepted, which results in other reduced mechanisms and other CPU times. This will result in a more "fair" comparison between the two solver combinations, since the QSSA for one particular species will not dominate the CPU time for the Newton-FP solver.

When difficult QSS species are included in the system of NAE, a numerical method which uses gradient information, i.e. the Newton solver, in order to converge is more powerful and converges faster than a numerical method that does not use gradient information, i.e. the FP solver. Hence, the Newton solver uses a smaller number of iteration steps than the FP solver does. However, a Newton method that reuses the Jacobian approximately scales as n^2, where n is the number of equations, for each iteration step, while the FP solver only scales as n each iteration step. This extra CPU cost for the Newton method, is minimized by the use of the MBSA algorithm and hard coding of the entire inner Newton solver.

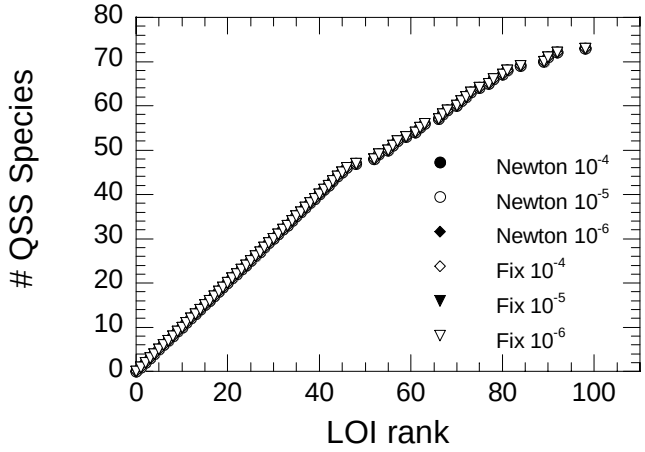


Figure 6.1. Number of QSS species vs LOI rank for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to the inner solver being a Newton solver and an FP solver respectively.

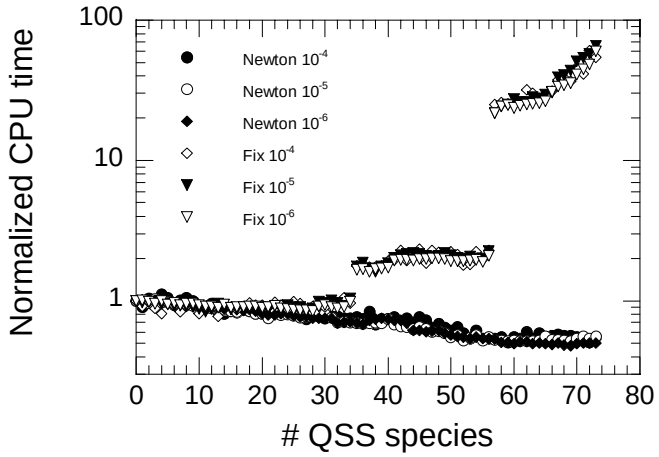


Figure 6.2. Normalized CPU time vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

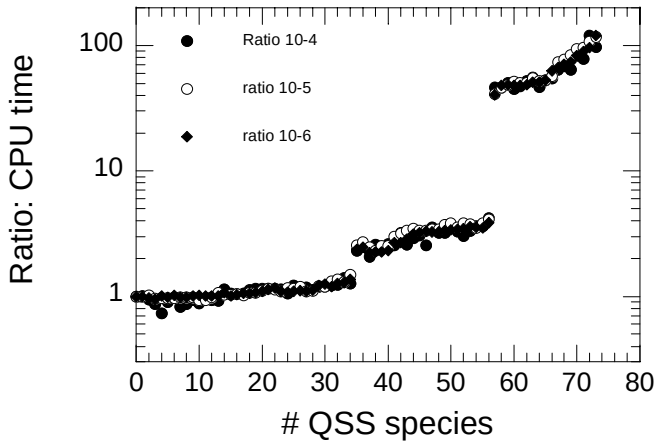


Figure 6.3. Ratio of CPU time between the two solver combinations for vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers.

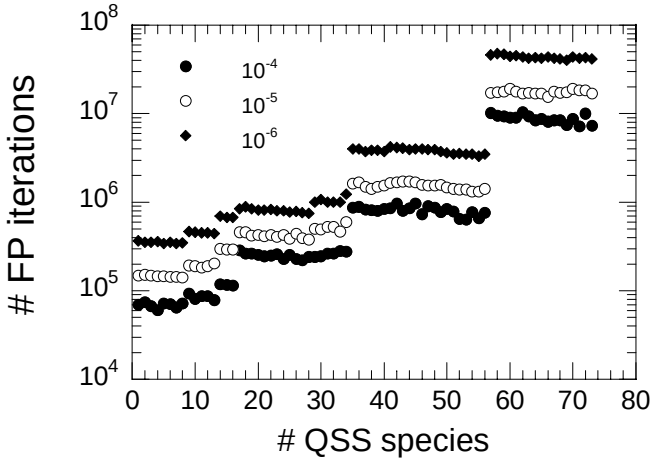


Figure 6.4. Number of FP iterations vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for the Newton-FP solver.

6.2.1.1.1 Reduction level of the reduced mechanisms

Figure 6.5 shows the number of QSS species vs LOI rank for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. The ART could identify identical (minus the species "HCCO" for the Newton-FP solver) systems of NAE, which included 73 and 72 QSS species when the Newton-Newton and Newton-FP method was used respectively, for all outer time step sizes. These simulations show that the outer time step size does not affect the final mechanism (using the stated ART-setting). These results are expected since the solver combination uses adaptive time step size.

Table 6.2 shows the species names and the corresponding LOI, LT and sensitivity value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted and the reason for the failure for the two solver combinations. A blank means that the species was accepted, while a letter and number combination means that the species failed to be accepted. The meaning of the letters can be seen in the table heading. The numbers represent the deviation for each ET in percent. The reason for failure and the numbers are equal for both solver combinations as expected. The reason for failure only differs for a few species (nr 91, 94 and 105). The reason for this is that limit where the inner solver is said not to converge is different for the Newton and FP solver. Also notable is that most species fail for several reasons. One reason for this is that the species that affect the IDT CF also affect the IDT HF. The species with higher LOI rank tend to fail for more reasons and have larger deviations than the species with lower LOI rank. The reason for this is that species with higher LOI rank affect the system more.

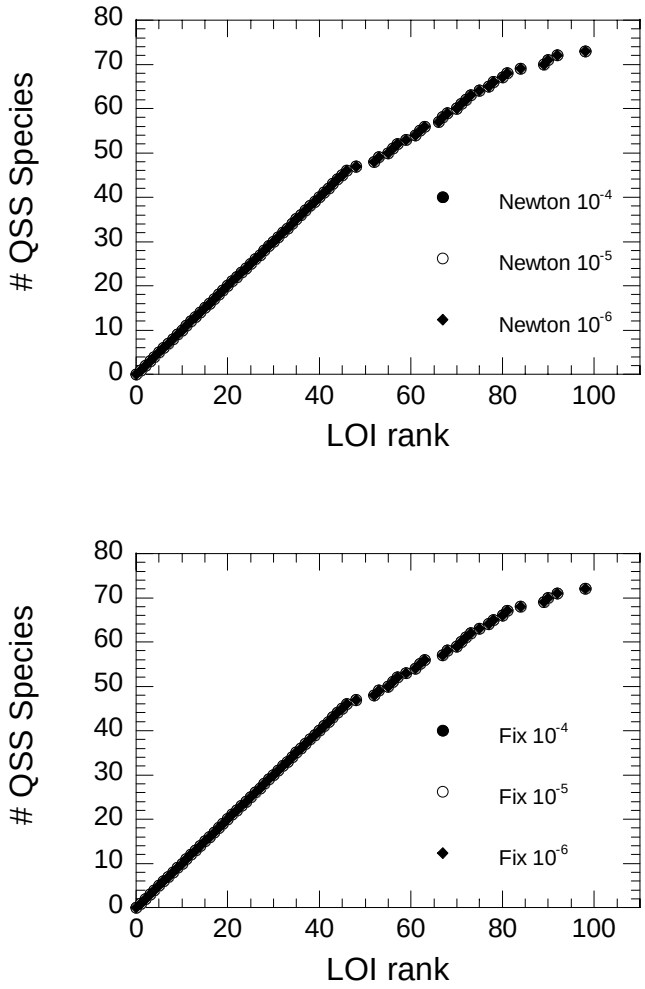


Figure 6.5. Number of QSS species vs LOI rank for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. Newton and Fix corresponds to the inner solver being a Newton solver and an FP solver respectively.

Table 6.2. The table shows the LOI rank of the species, the species names and the corresponding LOI value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted and the reason for the failure for the two solver combinations; Newton-Newton and Newton-FP. A blank means that the species was accepted, while a letter and number combination means that the species failed to be accepted. The numbers represent the deviation according to eq(6.2) for each ET in percent. The letters mean failure due to; A) IDT HF limit B) IDT CF limit C) Max HO2 CF limit D) Max OH HF limit E) No convergence in the inner solver F) The species is forbidden as a QSS species in advance G) There was no ignition H) Temperature limit I) CPU time limit

	Species	**LOI**	**LT**	**Sensitivity**	**Newton-Newton**	**Newton-FP**
1	CH2CH2COCH3	0.57992E-23	2.5041e-14	2.3159e-10		
2	C2H5COCH2	0.81611E-23	1.8689e-10	4.3668e-14		
3	CH2CH2CHO	0.30747E-22	4.6781e-08	6.5725e-16		
4	CH3CHCOCH3	0.67565E-22	9.2183e-14	7.3294e-10		
5	N-C3H7CO	0.29803E-21	2.5867e-08	1.1522e-14		
6	C2H5O	0.30558E-20	1.6356e-11	1.8683e-10		
7	5R-HEOOH-P	0.10122E-19	2.7216e-09	3.7191e-12		
8	HOCH2O	0.16819E-19	5.6222e-12	2.9915e-09		
9	N-C3H7COC2H4-1	0.10651E-17	5.2170e-07	2.0416e-12		
10	C2H5CO	0.11704E-17	3.9799e-09	2.9408e-10		
11	7R-HEOOH-P	0.24758E-17	1.4230e-08	1.7398e-10		
12	6R-HEOOH-P	0.36896E-17	1.1414e-08	3.2325e-10		
13	C2H4O2H	0.39947E-17	3.1542e-08	1.2665e-10		
14	5R-HEOOHO2-P	0.86579E-17	6.1318e-07	1.4120e-11		
15	7R-HEOOH-S	0.18380E-16	2.3132e-08	7.9457e-10		
16	1-C5H11	0.19356E-16	4.8102e-08	4.0239e-10		
17	2-C5H11	0.25297E-16	5.7478e-08	4.4012e-10		
18	5R-HEOOH-S	0.35434E-16	1.5629e-08	2.2672e-09		
19	N-C3H7	0.50926E-16	1.8957e-06	2.6864e-11		
20	CH3COCH2	0.58997E-16	1.8053e-06	3.2680e-11		
21	1-C4H9	0.66697E-16	5.4601e-09	1.2215e-08		
22	6R-HEOOH-S	0.10622E-15	1.6981e-08	6.2552e-09		
23	CH	0.18024E-15	3.2427e-10	5.5583e-07		
24	C6H10	0.21326E-15	5.7347e-06	3.7188e-11		
25	7R-O-HEPOOH-P	0.44094E-15	9.5563e-06	4.6141e-11		
26	1-C2H4COC2H5	0.71797E-15	6.0328e-06	1.1901e-10		
27	I-C3H7	0.99270E-15	9.4589e-08	1.0495e-08		
28	CH3O	0.10338E-14	8.5661e-09	1.2069e-07		
29	6R-HEOOHO2-P	0.16628E-14	4.5979e-07	3.6164e-09		
30	C2H	0.18632E-14	7.1760e-09	2.5964e-07		
31	HO2CHO	0.19006E-14	9.5540e-05	1.9893e-11		
32	1-CH2	0.25684E-14	9.3895e-09	2.7354e-07		
33	L-C7H15	0.26253E-14	4.5924e-08	5.7166e-08		
34	7R-HEOOHO2-P	0.27514E-14	7.6694e-07	3.5875e-09		
35	OCHO	0.35343E-14	9.1604e-05	3.8582e-11		
36	2-C4H8	0.36433E-14	2.6455e-06	1.3772e-09		
37	C6H5O	0.50284E-14	5.9863e-07	8.3998e-09		
38	7R-O-HEPOOH-S	0.71393E-14	9.5563e-06	7.4708e-10		
39	C2H5O2	0.10700E-13	3.8423e-06	2.7848e-09		
40	C4H7	0.11039E-13	7.6887e-08	1.4357e-07		
41	7R-HEOOHO2-S	0.12656E-13	7.6310e-07	1.6585e-08		
42	CH3O2H	0.14152E-13	7.6732e-05	1.8443e-10		
43	C7H13	0.16824E-13	1.6580e-06	1.0147e-08		
44	A1-	0.18662E-13	2.2945e-07	8.1334e-08		
45	C6H8	0.24974E-13	4.2682e-08	5.8512e-07		
46	5R-HEOOHO2-S	0.25519E-13	7.8898e-07	3.2344e-08		
47	6R-O-HEPOOH-P	0.33649E-13	1.9112e-05	1.7606e-09	B=-6.6	B=-6.6
48	C4H612	0.49074E-13	2.4345e-06	2.0158e-08		
49	6R-HEOOHO2-S	0.58707E-13	6.6151e-07	8.8747e-08	B=-7.2	B=-7.2

50	CH3O2	0.70054E-13	2.2574e-06	3.1033e-08	A=-6.1 B=-24.6	A=-6.1 B=-24.6
51	5R-O-HEPOOH-S	0.71260E-13	1.9112e-05	3.7285e-09	A=-3.3 B=-21.0 C=6.3	A=-3.3 B=-21.0 C=6.3
52	N-C4H3	0.75976E-13	1.8321e-07	4.1469e-07		
53	A1	0.86912E-13	2.2738e-06	3.8223e-08		
54	O2CHO	0.87279E-13	0.00023643	3.6915e-10	A=-11.9 B= -7.1 C=17.0	A=-11.8 B= -7.1 C=17.0
55	C4H10	0.11237E-12	6.7250e-06	1.6709e-08		
56	C5H9	0.14757E-12	2.2339e-06	6.6059e-08		
57	N-C4H9COCH2	0.17086E-12	0.00038328	4.4578e-10		
58	6R-O-HEPOOH-S	0.22825E-12	9.5572e-06	2.3883e-08	A=-5.3 B=-28.7 C=7.2	A=-5.3 B=-28.7 C=7.2
59	N-C4H5	0.23363E-12	3.8285e-07	6.1024e-07		
60	C2H5O2H	0.27242E-12	0.00034807	7.8266e-10	C=9.4	C=9.4
61	C2H6	0.33900E-12	1.3576e-05	2.4971e-08		
62	I-C4H3	0.45295E-12	3.8345e-07	1.1812e-06		
63	HCO	0.46041E-12	5.8900e-09	7.8168e-05		
64	N-C3H7COCH2	0.65550E-12	0.0024301	2.6974e-10	C=9.3	C=9.3
65	L-C7H15O2	0.66422E-12	6.2105e-07	1.0695e-06	A=-4.1 B=-20.7 C=6.1	A=-4.1 B=-20.7 C=6.1
66	HCCO	0.68469E-12	1.2506e-09	0.00054749		I
67	3-CH2	0.14667E-11	1.2123e-07	1.2098e-05		
68	I-C4H5	0.17894E-11	6.0625e-07	2.9516e-06		
69	CH3OH	0.34782E-11	0.0038866	8.9492e-10	A=5.7 C=-10.8	A=5.7 C=-10.8
70	CH2CHO	0.36697E-11	1.8655e-06	1.9671e-06		
71	HOCHO	0.49648E-11	0.00037895	1.3101e-08		
72	C4H4	0.62090E-11	1.1780e-06	5.2708e-06		
73	C4H6	0.63781E-11	6.1404e-06	1.0387e-06		
74	1-C4H8	0.81147E-11	0.0045500	1.7835e-09	A=9.7 B=17.1 C=-21.4	A=9.7 B=17.1 C=-21.4
75	C3H8	0.98504E-11	1.2254e-05	8.0385e-07		
76	5R-C7H14O	0.98797E-11	0.00082719	1.1944e-08	A=16.2 B=32.5 C=-26.2	A=16.2 B=32.5 C=-26.2
77	CH2OH	0.16731E-10	1.3693e-07	0.00012219		
78	C2H5	0.23923E-10	7.9534e-07	3.0079e-05		
79	C3H6	0.27350E-10	0.0041283	6.6250e-09	A=16.0 B=25.6 C=-20.5	A=16.0 B=25.6 C=-20.5
80	C2H3	0.28788E-10	1.1232e-07	0.00025630		
81	CH3CO	0.40888E-10	2.2299e-07	0.00018336		
82	1-C5H10	0.42325E-10	0.00046564	9.0896e-08	A=20.7 B=9.8 C=-23.4	A=20.7 B=9.8 C=-23.4
83	CH3CHO	0.47077E-10	0.0045347	1.0382e-08	B=17.1 C=-23.7	B=17.1 C=-23.7
84	C3H5	0.53180E-10	3.7938e-07	0.00014018		
85	N-C3H7CHO	0.60902E-10	0.0016152	3.7706e-08	A=18.2 B=26.8 C=-32.0	A=18.2 B=26.8 C=-32.0
86	H2O2	0.78482E-10	0.0060066	1.3066e-08	A=-26.4 B=-9.8	A=-26.4 B=-9.8
87	C2H5CHO	0.87104E-10	0.0051944	1.6769e-08	A=-6.4 B=23.2 C=-20.5	A=-6.4 B=23.2 C=-20.5
88	C3H4P	0.11307E-09	2.4980e-05	4.5264e-06	A=-4.2	A=-4.2
89	C3H4	0.13220E-09	2.0135e-06	6.5657e-05		
90	C3H3	0.18911E-09	4.7807e-07	0.00039557		
91	L-C7H14	0.21697E-09	0.017846	1.2158e-08	E	A=-18.5 B=12.2 C=-37.8

92	C2H2	0.23131E-09	1.2584e-06	0.00018381		
93	HO2	0.25436E-09	8.7777e-07	0.00028978	F	F
94	O	0.74893E-09	1.1707e-07	0.0063973	E	D=3.4
95	CH2CO	0.83539E-09	1.2467e-05	6.7008e-05	A=-19.8 B=36.6 C=-20.7	A=-19.8 B=36.6 C=-20.7
96	CH2O	0.95470E-09	0.00035468	2.6917e-06	A=-43.7 B=155.9 C=217.7	A=-43.7 B=155.9 C=217.4
97	C2H5COCH3	0.14892E-08	2.7037e-07	0.0055080	A=-99.8 B=865.9 C>1000 D>1000 G H	A=-99.8 B=865.9 C>1000 D>1000 G H
98	CH3	0.35709E-08	1.0282e-07	0.034730		
99	CH4	0.64018E-08	7.4762e-06	0.00085629	F	F
100	C2H4	0.65893E-08	3.4746e-05	0.00018964	A=-44.7 B=-12.2	A=-44.7 B=-12.2
101	OH	0.30677E-07	1.8253e-07	0.16807	F	F
102	CO2	0.92212E-07	1.5508e-05	0.0059461	F	F
103	H	0.10853E-06	1.3732e-06	0.079034	D=2.2	D=2.2
104	N-C7H16	0.26023E-06	0.00096877	0.00026862	F	F
105	H2	0.27517E-06	5.6302e-05	0.0048874	E	D=3.7
106	CO	0.31018E-06	9.3823e-07	0.33060	A=-25.2 C=-5.8 D=27.3 H	A=-25.2 C=-5.8 D=27.0 H
107	H2O	0.49953E-04	4.8320e-06	10.338	F	F
108	O2	0.62892E-04	0.0046001	0.013672	F	F
109	N2	0.10000E+01	1.0000	1.0000	F	F
110	AR	0.10000E+01	1.0000	1.0000	F	F

6.2.1.1.2 CPU time of the reduced mechanisms

Figure 6.6 shows the normalized CPU time vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. The figure shows that there is a clear difference in CPU time for all outer time step sizes between the two solver combinations.

The ratio in CPU time between the two solver combinations for each time step size is shown in Figure 6.7. The ratio increases less dramatically without the species HCCO, but the ratio reaches about a factor 40 for the most reduced mechanism. The increase beyond 65 QSS species is due to both the inclusion of “difficult” QSS species in the system of NAE, which affects the convergence of the Newton-FP solver negatively, and the size of the system of NAE alone.

When these difficult QSS species are included in the system of NAE, a numerical method which uses gradient information in order to converge, like the Newton method, is more powerful and converges faster than a numerical method that does not use gradient information, like the FP method.

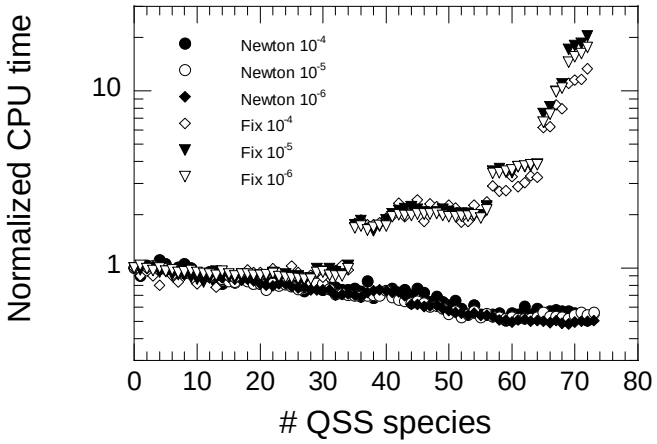


Figure 6.6. Normalized CPU time vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

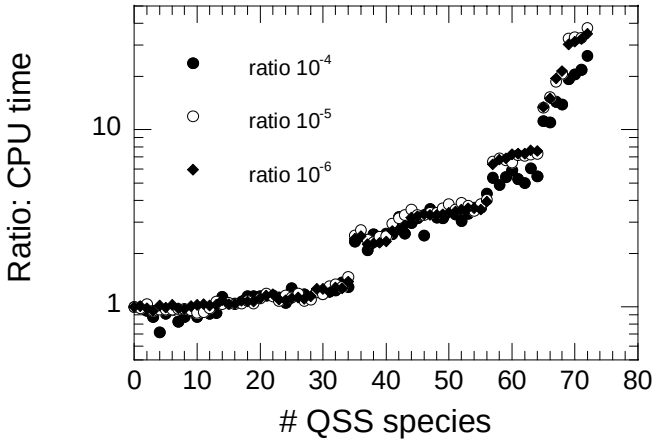


Figure 6.7. Ratio of CPU time between the two solver combinations for vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers.

6.2.1.1.3 Solver information

The average time step size vs number of QSS species is shown in Figure 6.8. The average time step size is different for the different outer time step sizes even though adaptive time step size is used. This is expected since each solver combination uses very short time steps during the ignition and longer time steps after ignition. The figure also shows that the average time step size is independent of the inner solver and almost independent of the reduction level.

Figure 6.9 shows the number of Jacobians vs number of QSS species for the system of ODE for different outer time steps and inner solvers. There is a clear difference in the number of Jacobians for the different outer time step sizes. This is expected since a new Jacobian is built for each outer time step. The figure also shows that the number of Jacobians is independent of the inner solver and almost independent of the reduction level.

Figure 6.10 shows the number of BS vs number of QSS species for the system of ODE for different outer time steps and inner solvers. There is a clear difference in the number of BS for the different outer time step sizes. The figure also shows that the number of BS is independent of the inner solver and almost independent of the reduction level.
Hence, the average time step size, the number of Jacobians and BS for the system of ODE are independent of the solver combination. Consequently, the difference in CPU time for the two solver combinations is entirely due to the inner solver.

Figure 6.11 shows the number of Jacobians vs number of QSS species for the system of NAE for different outer time steps. There is a clear difference in the number of Jacobians for the different outer time step sizes. The number of Jacobians is increases with the reduction level for all outer time step sizes. The reason for this is that the system of NAE becomes more difficult to solver with increasing number of QSS and that the difficult QSS species, which affects the convergence negatively, have high LOI rank and are accepted at a later stage of the reduction process.

Figure 6.12 shows the number of BS vs number of QSS species for the system of NAE for different outer time steps. There is a clear difference in the number of BS for the different outer time step sizes. The number of BS decreases with the reduction level for all outer time step sizes. The reason for this is that the inner solver already compensated for the increasing convergence problems (with increasing number of QSS species) by an increasing number of Jacobians for the system of NAE. Consequently, a lower number of BS is needed for convergence.
If the number of ODE decreases, the number of species in the Jacobian for the system of ODE decreases linearly. When the Jacobian for the system of ODE is built, the inner solver is called once for each species in order to create the finite difference (see section 3.3.4.1. and 4.2). Hence, the number of calls to the inner solver also decreases linearly when the number of ODE decreases. This means that the number of Jacobians and BS for the system of NAE is expected to decrease linearly as well, under the assumption that the convergence is not changed.

However, this assumption is not accurate, since the size of the system of NAE increases, which normally affects the convergence negatively, at the same time as the system of ODE decreases. Also, species with higher LOI rank, which affects the convergence negatively, are included in the system of NAE. If the increase in the number of Jacobians for the system of NAE is seen in this light, it is a strong indicator that the system of NAE becomes more difficult to solve with increasing number of QSS species.

Figure 6.13 shows the number of FP iterations vs number of QSS species for the system of NAE for different outer time steps. There is a clear difference in the number of FP iterations for the different outer time step sizes. The curve for number of FP iterations increases with the reduction level and clearly resembles the curve for the CPU time, which is expected since the FP solver uses most of the CPU time in the solver combination.

Figure 6.14 shows the ratio between the number of FP iterations for the Newton-FP solver and the number of BS for the Newton-Newton solver for different outer time steps. The ratio is very similar for all outer time step sizes and reaches more than a factor 10^2 for the most reduced mechanisms. This is the reason for the performance difference between the Newton-Newton solver and the Newton-FP solver.

The main conclusion from this investigation is that the performance difference between the Newton-Newton solver and the Newton-FP solver is entirely due to the inner solver.

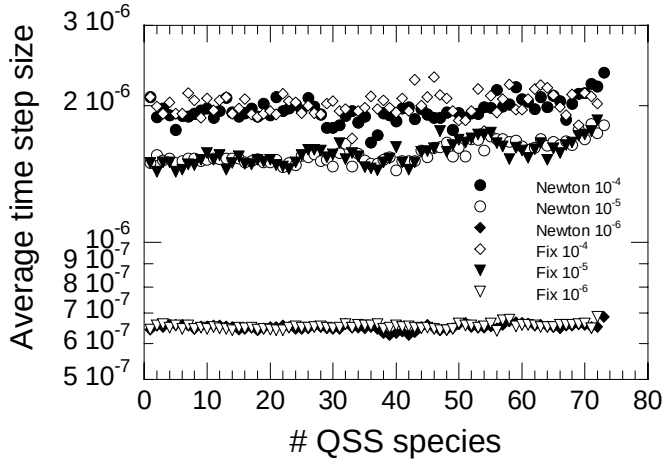


Figure 6.8. Average time step size vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

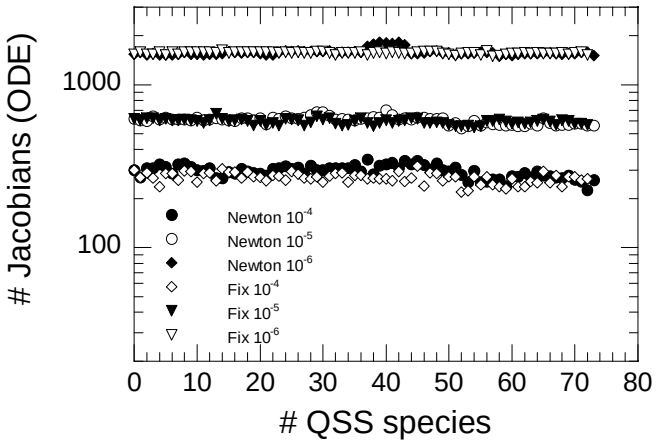


Figure 6.9. Number of Jacobians (ODE) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

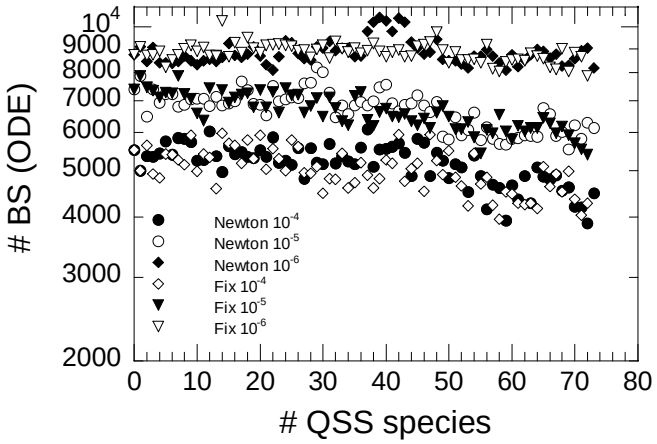


Figure 6.10. Number of BS (ODE) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both inner solvers. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

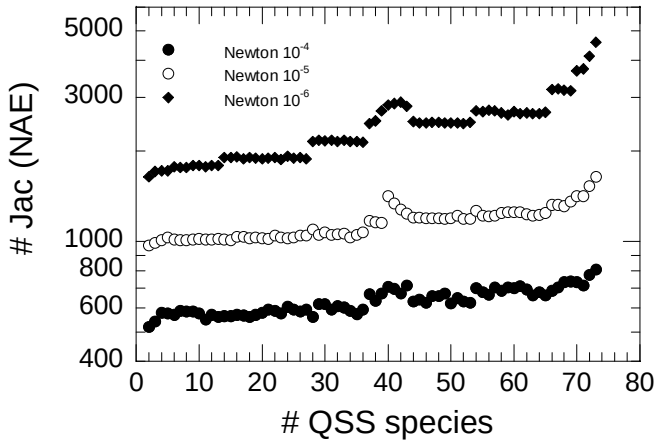


Figure 6.11. Number of Jacobians (NAE) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4}.

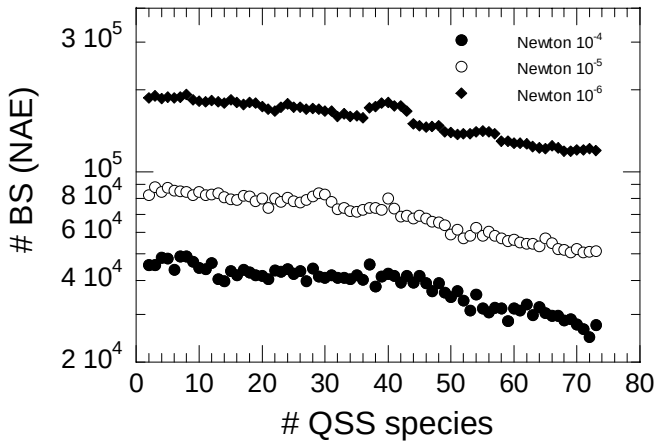


Figure 6.12. Number of BS (NAE) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4}.

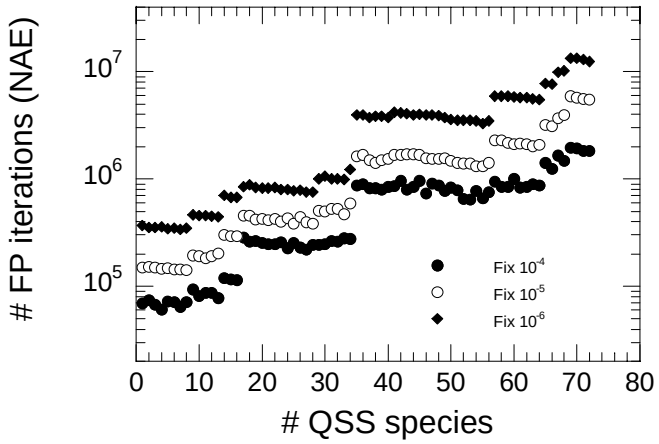


Figure 6.13. Number of FP iterations (NAE) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4}.

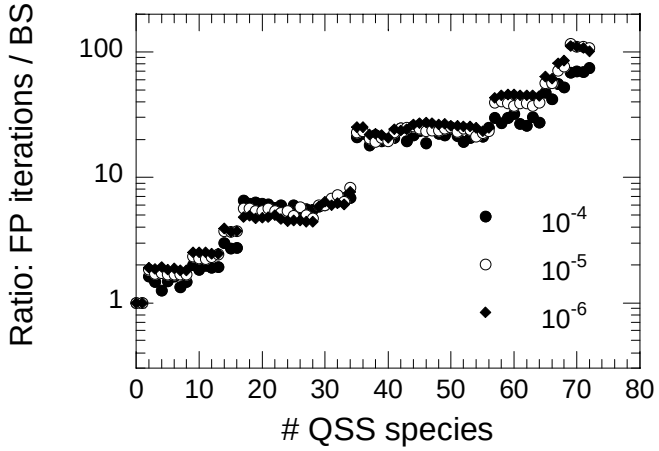


Figure 6.14. Ratio between the number of FP iterations for the Newton-FP solver and number of BS for the Newton-Newton solver vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4}.

6.2.1.1.4 Accuracy of the reduced mechanisms

Figures 6.15 to 6.18 show that there is a small difference in IDT HF, IDT CF, Max HO2 CF and Max OH HF for the different outer time step sizes. This can be explained by the fact that different solution trajectories are obtained for the different outer time step sizes. However, the IDT HF, IDT CF, Max HO2 CF and Max OH HF are very similar for both solver combinations for all outer time step sizes. Hence, the solution trajectories are basically the same for both solver combinations, which means that the difference in CPU time is due to the number of operations in each solver combination. It is thereby safe to claim that the Newton-Newton solver combination is much faster than the solver combination that uses the Newton-FP solver combination.

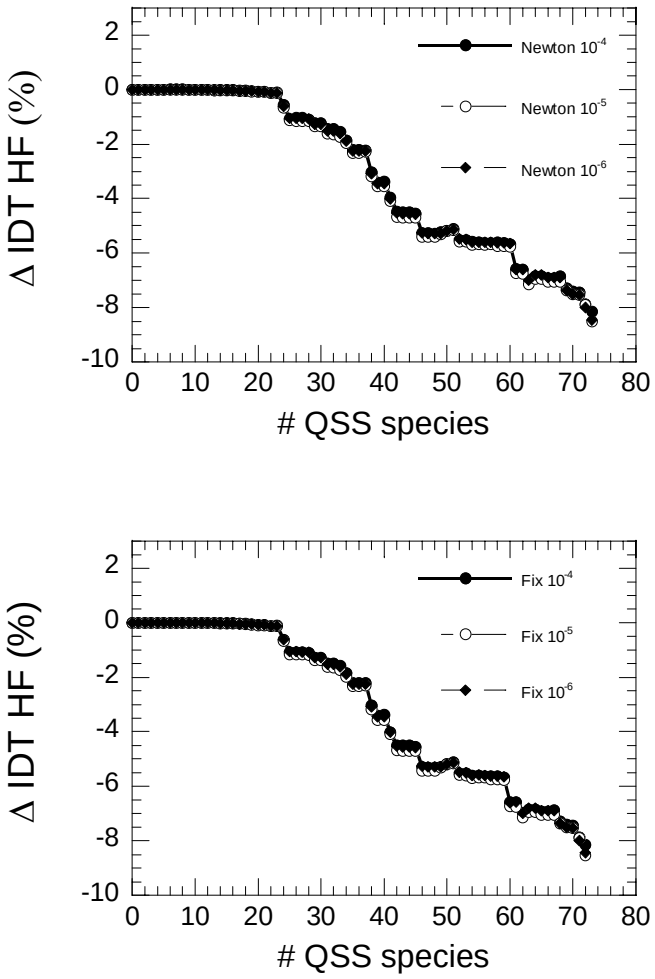


Figure 6.15. Δ IDT HF (%) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

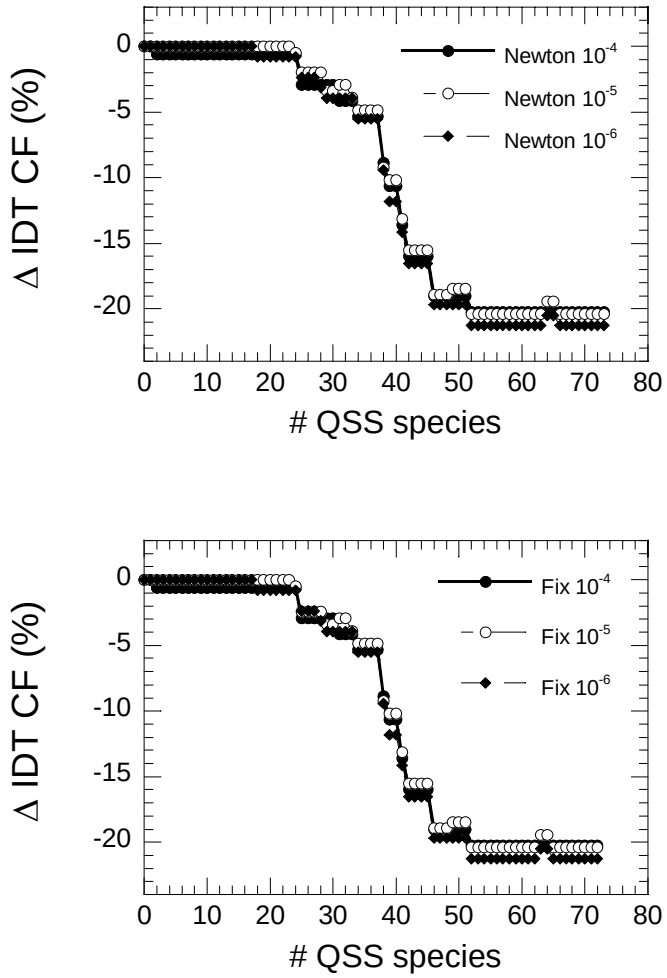


Figure 6.16. Δ IDT CF (%) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

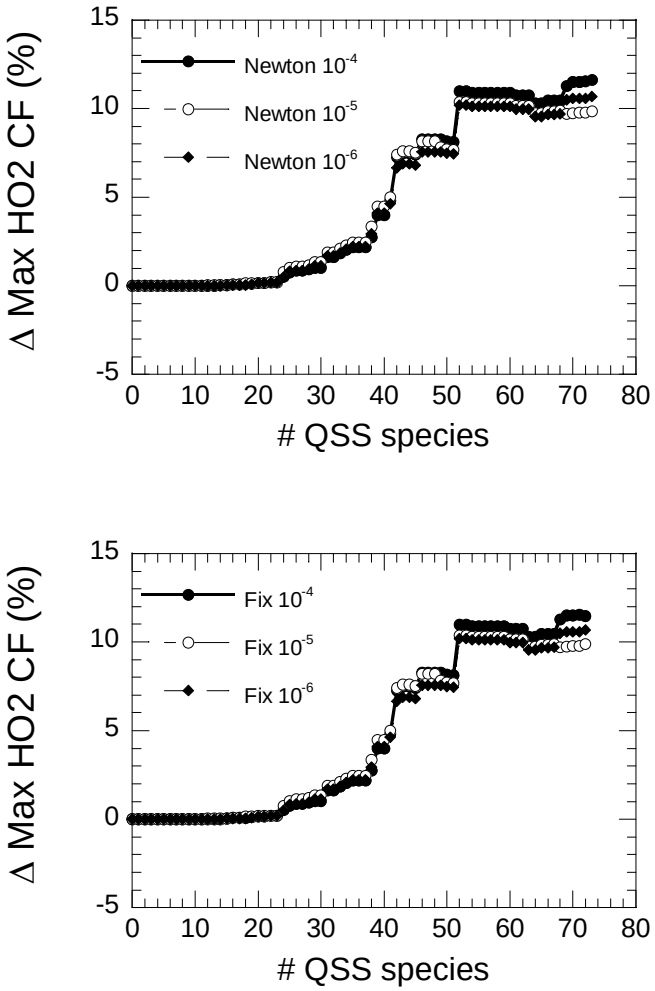


Figure 6.17. Δ Max HO_2 CF (%) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively.

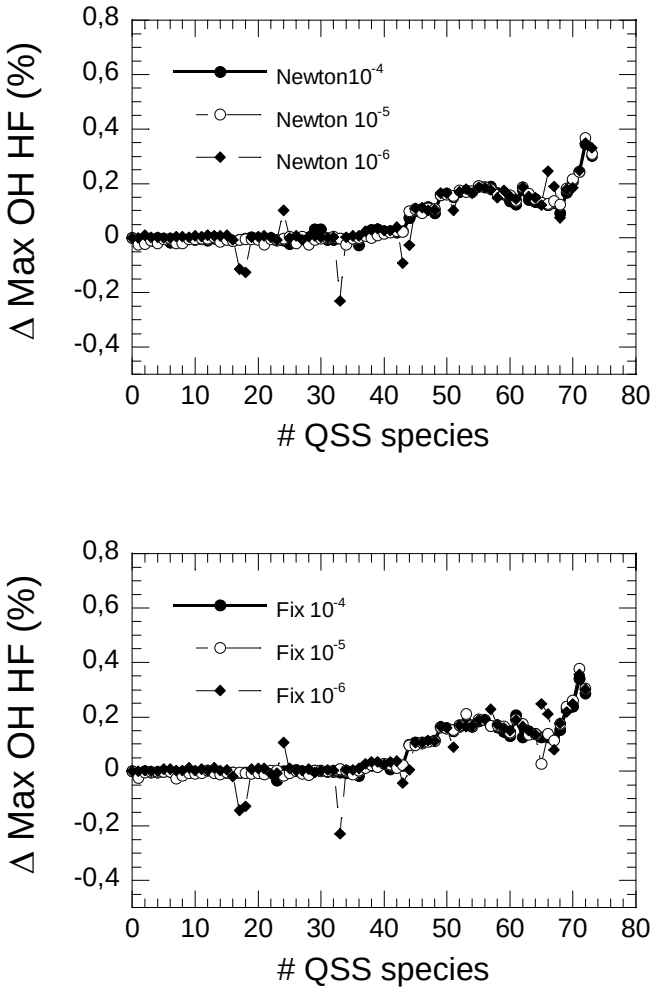


Figure 6.18. Δ Max OH HF (%) vs number of QSS species for outer time step sizes of 10^{-6}, 10^{-5} and 10^{-4} for both solver combinations. Newton and Fix corresponds to solver combinations where the inner solver is a Newton solver and an FP solver respectively. The jumps in the Δ Max OH HF (%) for some reduced mechanisms are probably due to sampling of the OH profile.

6.2.1.1.5 Species profiles

Figures 6.19 to 6.21 show mass fraction vs time for some important species as well as temperature and pressure vs time. All figures show the original mechanism with 0 QSS species, the most reduced mechanism with 73 QSS species using the Newton-Newton solver combination and the most reduced mechanism with 72 QSS species using Newton-FP solver combination. Both solvers used an outer time step of 10^{-5} s for all figures. It is clear that the Newton-Newton and Newton-FP solver combinations give very similar results, which further strengthens what was stated above, that is, the difference in CPU time solely comes from the number of operations in the two solver combinations and that the solution trajectories are basically the same for both solver combinations. The temperature, pressure and species like OH, HO_2, N-C_7H_{16}, O_2, CO_2 and H_2O do not deviate much from the original mechanism. However, the species CH_4, CO, C_2H_4, CH_2O and H_2 have larger deviation compared to the original mechanism than the others, which is natural since OH was the only ET for chemical species during the simulations. If all of the species must have a deviation as small as the one for OH, the ART user must set all of them as ET for chemical species, which will result in a less reduced mechanism. A simulation using many ET for chemical species is shown in section 6.2.2.4. The ET for chemical species that are appropriate to select depends on what features the simulation focuses on.

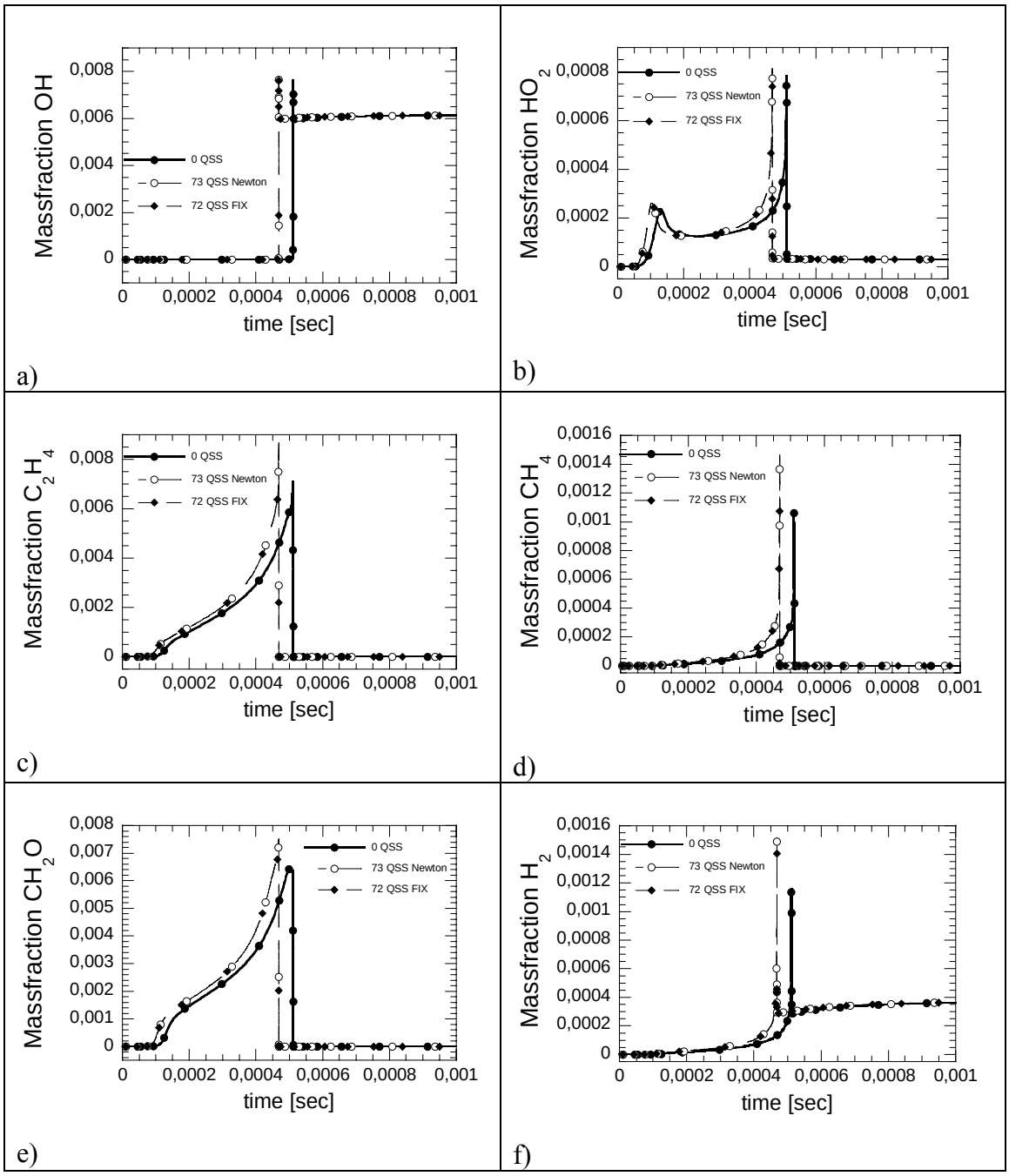


Figure 6.19. Mass fraction of important species vs time for 0 QSS, 73 QSS with the Newton-Newton solver combination and 72 QSS with the Newton-FP solver combination. The outer time step size was 10^{-5} for all.
a) OH b) HO_2 c) C_2H_4 d) CH_4 e) CH_2O f) H_2

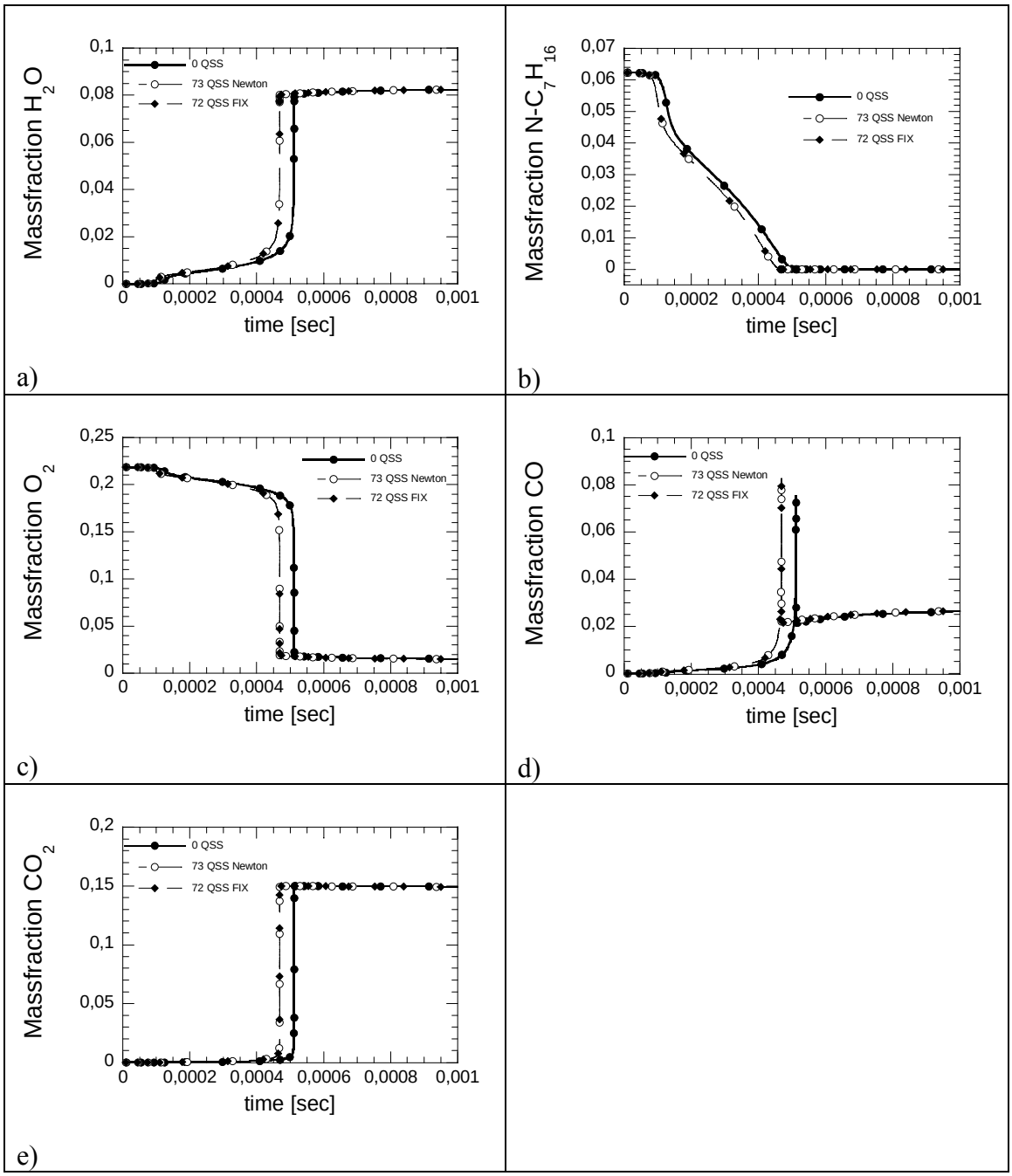


Figure 6.20. Mass fraction of important species vs time for 0 QSS, 73 QSS with the Newton-Newton solver combination and 72 QSS with the Newton-FP solver combination. The outer time step size was 10^{-5} for all.
a) H_2O b) N-C_7H_{16} c) O_2 d) CO e) CO_2

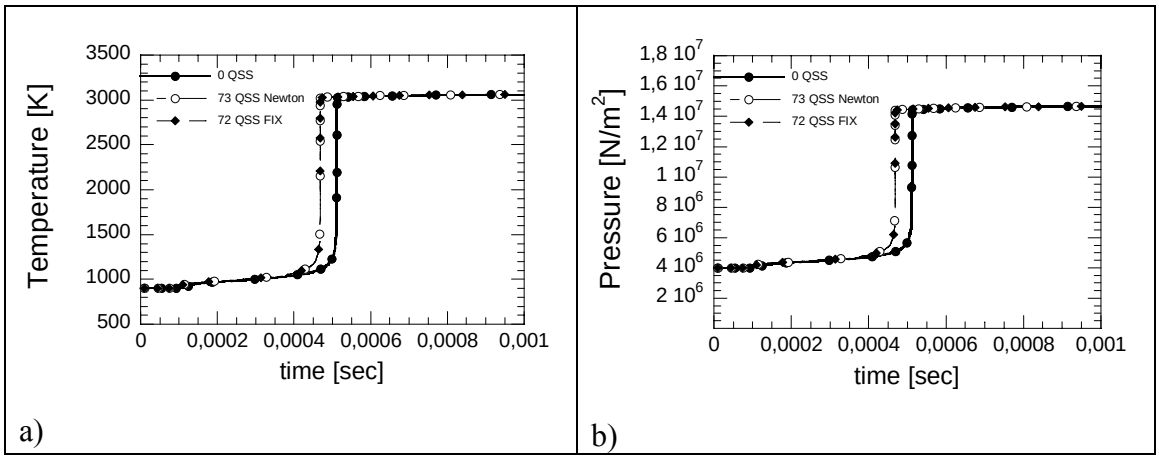


Figure 6.21. Temperature and pressure vs time for 0 QSS, 73 QSS with the Newton-Newton solver combination and 72 QSS with the Newton-FP solver combination. The outer time step size was 10^{-5} for all.
a) Temperature b) Pressure

Concluding remarks

The Newton-Newton solver combination is faster than the Newton-FP solver combination. The reason for this is that the CPU time for the Newton-Newton solver is up to 40 times lower than for the Newton-FP solver for all outer time step sizes, while the accuracy of the ART ET and the species profiles and thereby the solution trajectories of the two solver combinations are the same.

The solver information shows that the difference in CPU time solely comes from the inner solver, since the average time step size and number of Jacobians and BS for the outer solver is the same for both solver combinations. Also, the performance of the Newton-FP solver is more sensitive to some difficult QSS species.

6.2.1.2 Variation of solver settings

The speed and accuracy of the solver combination is affected by the solver settings of both the inner and outer solver. Both the inner and outer solver have an *Absolute tolerance* and a *Relative tolerance*. Each solver is said to have converged if

$$X_NormMax < \mathrm{Re}\,lative\,tolerance \tag{6.6}$$

$X_NormMax$ is the maximum value of all X_Norm_i, which is defined as:

$$X_Norm_i = \left| \frac{\Delta x_i}{x_i + Absolute\,tolerance} \right|, i = 1,...n \tag{6.7}$$

where x_i is the concentration of species i, Δx_i is the difference in x_i between two consecutive iteration steps and n is the number of equations in the solver. The inequality basically says that the solver has converged when the species with the largest relative change in concentration is less than the relative tolerance.

The *Absolute tolerance* decides the concentration sizes, and thereby the species, that are allowed to influence the convergence. If the *Absolute tolerance* is small the small species concentrations will influence the X_Norm_i. Hence, the species with small concentrations and small Δx_i can determine if the solver converged or not. Contrary, if the *Absolute tolerance* is large the small species concentrations will not influence the X_Norm_i. Hence, if a high *Absolute tolerance* is used in the inner solver and if the convergence of the outer solver is sensitive for the accuracy of the QSS species with small concentrations, the outer solver will have convergence problems. This will in turn lead to longer CPU times. The opposite is true for low *Absolute tolerance*.

A low *Relative tolerance* provides QSS species with high accuracy but forces the inner solver to work hard, which leads to long CPU times. The opposite is true for high *Relative tolerance*. This is summarized in Table 6.3.

The absolute tolerance of the inner solver that allows enough QSS species to affect the outer solver depends on the solver setting of the outer solver. If not enough QSS species are allowed to have high accuracy by the absolute tolerance of the inner solver, the convergence of the outer solver will be affected negatively, which in turn leads to longer CPU times.

The relative tolerance of the inner solver that provides enough accuracy in the QSS species concentrations to the outer solver depends on the solver setting of the outer solver. If the accuracy of the QSS species concentrations is too low, the convergence of the outer solver will be affected negatively, which in turn leads to longer CPU times.

Conversely, if the relative tolerance of the inner solver is too low and thereby provides QSS species concentrations with too high accuracy, the inner solver works in vain. This results in longer CPU times.

However, the relative tolerance of the inner solver cannot be lower than or the same as the relative tolerance of the outer solver. The reason for this is that the error in the QSS species concentrations would cause the source term and the Jacobian of the outer solver to be inaccurate enough to cause convergence problems of the outer solver, which in turn lead to longer CPU times. This is discussed in detail in section 3.3.4.1.1.

An illustrative and detailed example of the concentrations effect on the inequality in eq(6.6) and eq(6.7) is shown in the chapter appendix.

Table 6.3. Shows four different combinations of Absolute and Relative tolerances of the inner solver and the effect they have on the convergence of the outer solver and the entire CPU time.

	Low Absolute tolerance	High Absolute tolerance
Low Relative tolerance	• Outer solver has **no** convergence problems due to inaccurate QSS species concentrations. • Inner solver is slow • Medium CPU time	• Outer solver has convergence problems due to inaccurate QSS species concentrations. • Inner solver is slow • High CPU time
High Relative tolerance	• Outer solver has **no** convergence problems due to inaccurate QSS species concentrations. • Inner solver is fast • Low CPU time	• Outer solver has convergence problems due to inaccurate QSS species concentrations. • Inner solver is fast • Medium CPU time

In order to investigate the optimum solver settings of the inner solver given the solver settings of the outer solver, the solver settings of the outer solver were fixed: Absolute tolerance=10^{-10} and a Relative tolerance=10^{-6}. The solver settings of the inner solver were varied in the following way. The Absolute tolerance was varied from 10^{-8} to10^{-15}, while the relative tolerance was varied over a wide range of values for each absolute tolerance.

6.2.1.2.1 CPU time of the reduced mechanisms

Figure 6.22 and 6.23 show the normalized CPU time vs number of QSS species for different combinations of Absolute tolerance and Relative tolerance of the inner solver. The same trend concerning the relative tolerance is observed for all absolute tolerances. A high relative tolerance gives high CPU time for all QSS species, while a low relative tolerance gives an increase in CPU time after a minimum at about 50 QSS species.
A medium high relative tolerance value, which depends on the value of the absolute tolerance, gives the lowest CPU time for all QSS species. A minimum in CPU time for all QSS species is reached for Absolute tolerance=10^{-15} and a Relative tolerance=10^{-7}.
Hence, the following solver setting was chosen for all other simulations in this thesis;

- Outer solver
  - Absolute tolerance: 10^{-10}
  - Relative tolerance: 10^{-6}
- Inner solver
  - Absolute tolerance: 10^{-15}
  - Relative tolerance: 10^{-7}

6.2.1.2.2 Solver information

Figure 6.24 to 6.28 show the solver data for absolute tolerance 10^{-15}. The average time step size is small for high relative tolerance and larger for low relative tolerance.
The normalized numbers of Jacobians and BS for the system of ODE have high numbers for high relative tolerances and low numbers for lower relative tolerances.
However, the normalized number of Jacobians and BS for the system of NAE behave in the opposite way, that is, high number for low relative tolerances and low numbers for higher relative tolerances. The same behavior is observed for other absolute tolerances as well. These results can be explained in the following way.

If the solver settings of the inner solver have large values for both the Absolute tolerance and Relative tolerance, the inner solver supplies QSS species concentrations with low accuracy to the source term of the system of ODE (see section 3.3.4.1.1). This can cause convergence problems for the outer solver, which responds by increasing the number of Jacobians and BS. Hence, the CPU time of the solver combination increases.
However, if the solver settings of the inner solver have small values for both the Absolute tolerance and Relative tolerance, the inner solver must increase the number of Jacobians and BS in order to reach the high demands. Hence, the CPU time of the solver combination increases as well. This argumentation proposes that a minimum in CPU time exists for the best combination of solver settings of the inner solver. This best combination is dependent on the solver settings of the outer solver.

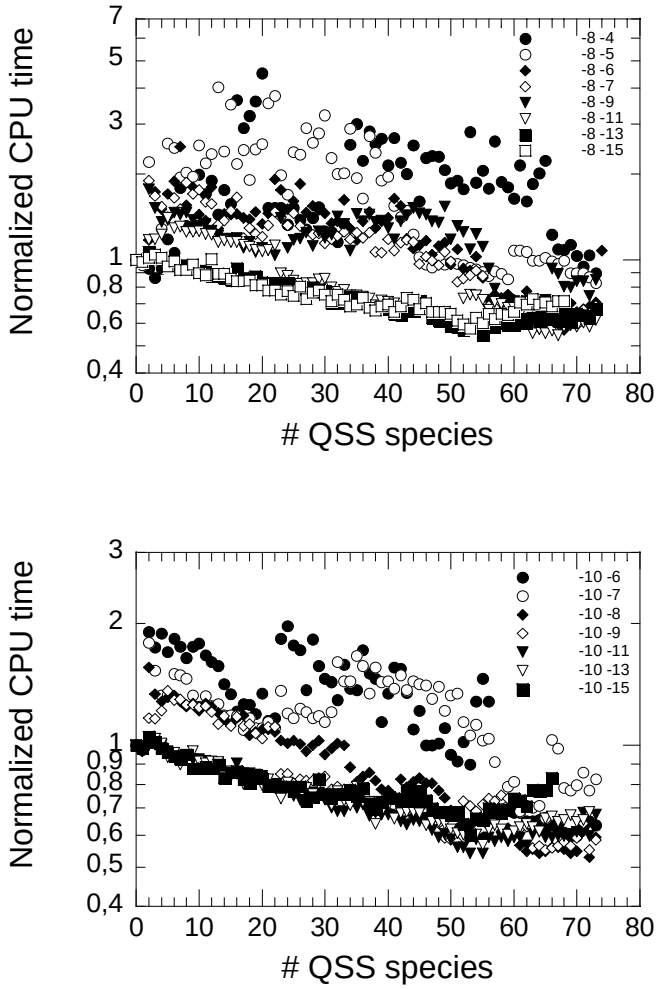


Figure 6.22. Normalized CPU time vs number of QSS species. The first and second number in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

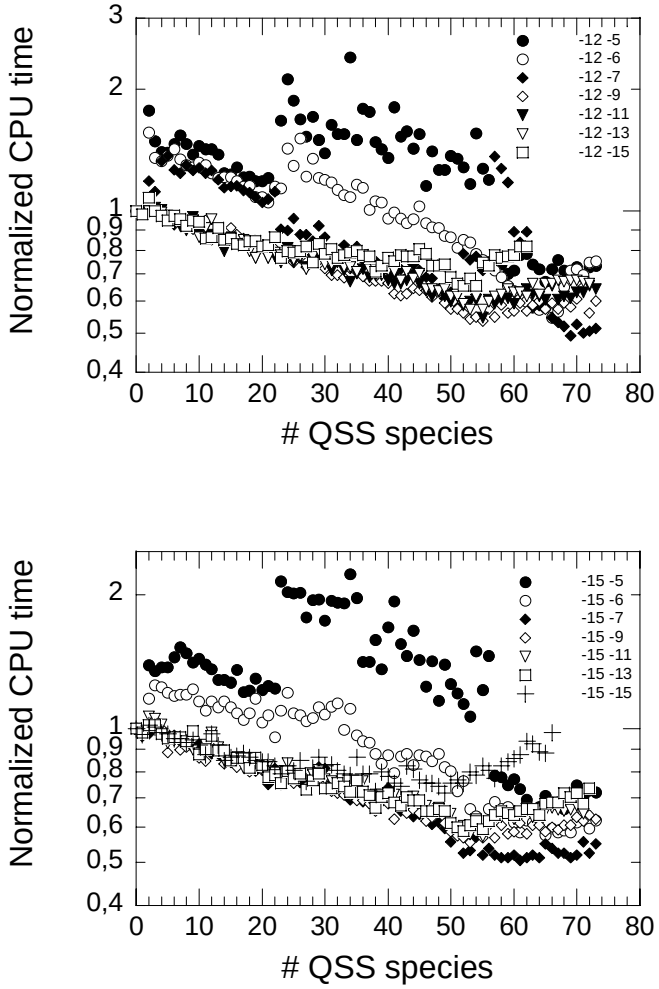


Figure 6.23. Normalized CPU time vs number of QSS species. The first and second numbers in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

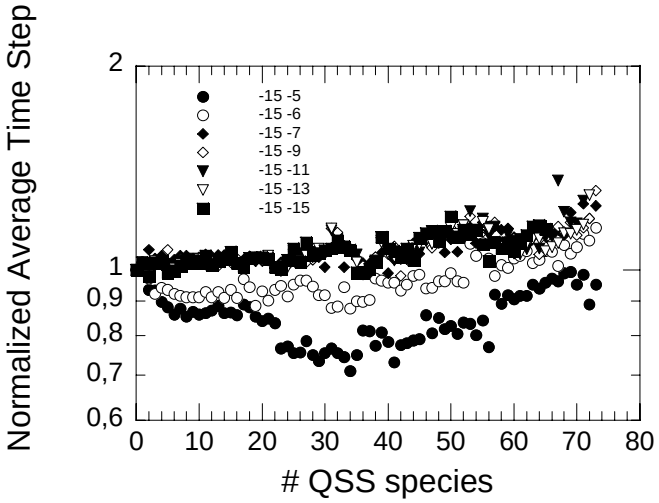


Figure 6.24. Normalized average time step size vs number of QSS species for the absolute tolerance of 10^{-15}. The first and second numbers in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

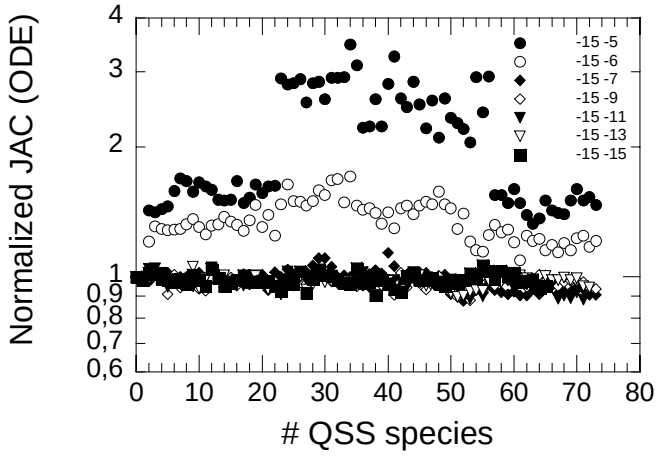


Figure 6.25. Normalized number of Jacobians for the system of ODE vs number of QSS species for the absolute tolerance of 10^{-15}. The first and second numbers in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

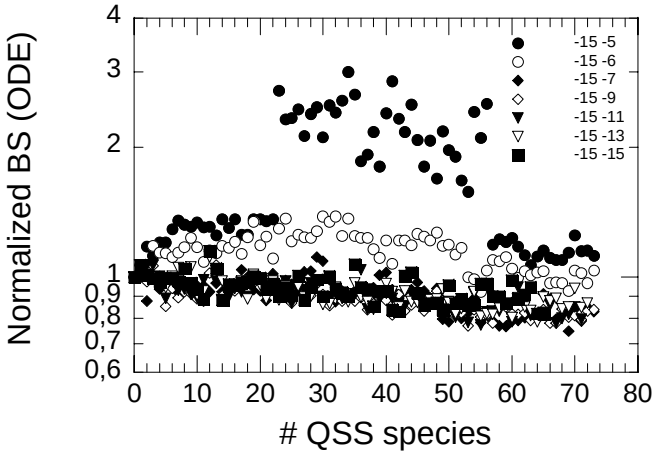


Figure 6.26. Normalized number of BS for the system of ODE vs number of QSS species for the absolute tolerance of 10^{-15}. The first and second numbers in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

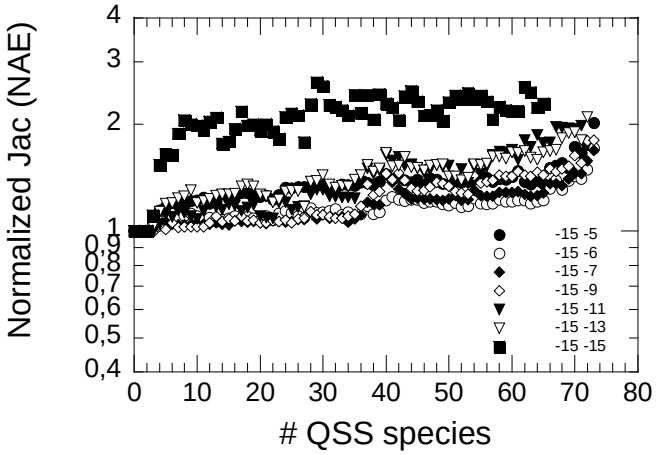


Figure 6.27. Normalized number of Jacobians for the system of NAE vs number of QSS species for the absolute tolerance of 10^{-15}. The first and second numbers in the legend correspond to the exponent in the absolute and relative tolerance respectively for the inner solver.

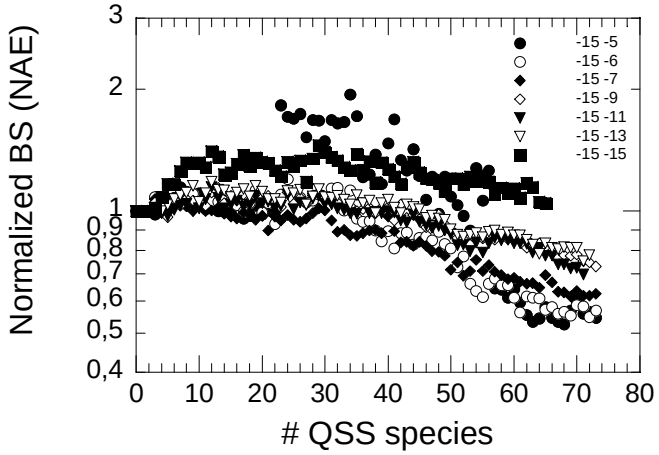


Figure 6.28. Normalized number of BS for the system of NAE vs number of QSS species for the absolute tolerance of 10^{-15}. The first and second number in the legend corresponds to the exponent in the absolute and relative tolerance respectively for the inner solver.

Concluding remarks
The Absolute tolerance and the Relative tolerance of the inner solver affects the CPU time. An optimum in CPU time is found for high relative tolerance and low absolute tolerance. The reason for this is that the low absolute tolerance makes QSS species with low concentration important, which means that they can facilitate the convergence of the outer solver. Also, a high relative tolerance minimizes the number of iterations in the inner solver. This decreases the CPU time, since the inner solver is used many more times than the outer solver.

6.2.1.3. Minimization and Maximization of CPU time

The CPU time of the Newton-Newton solver combination is proportional to the number of operations in the inner solver, which can be either minimized or maximized depending on the settings in the MBSA algorithm. The minimum number of operations is obtained by the used of the MBSA algorithm in section 4.3.3.3.3. However, if the maximum number of operations is wanted the sign in the algorithm is simply switched.

Figure 6.29 shows the normalized CPU time vs number of QSS species for minimized and maximized number of operations in the inner solver. The two curves show the impact of the MBSA algorithm on the speed up of the solver combination. The two curves start to differ at about 55 QSS species and the ratio between them reaches almost a factor two for the most reduced mechanism. This difference between the CPU time curves is a consequence of the difference in the curves for the number of operations in the GE and BS of the inner solver, which can be seen in Figure 6.30. The maximized number of operations increases more dramatically beyond about 55 QSS species, which is a consequence of the inclusion of species with higher LOI rank in the system of NAE.

Figure 6.31 shows the number of reactions per QSS species vs number of QSS species. Even though the curve is oscillating, the number of reactions per QSS species increases beyond about 55 QSS species. Hence, the species with higher LOI rank have a tendency to react with many species, including both QSS and non QSS species. Consequently, the sparseness pattern of the Jacobian of the inner solver becomes less sparse when species with higher LOI rank are included in the system of NAE. The denser sparseness pattern in turn increases the number of operations in the GE and BS of the inner solver. These results show the importance of optimization of the sparseness pattern in order to decrease the CPU time, especially for heavily reduced mechanisms.

The most reduced mechanism with 73 QSS species has about 800 operations in the GE and BS for the minimized case, which is about 15 % of the theoretical maximum number of operations. This shows the efficiency of the MBSA algorithm. The theoretical maximum corresponds to the number of operations in an algorithm that does not use any sparseness advantages and instead just eliminates the entire sub diagonal part of the Jacobian during the GE and uses every element in the upper triangular matrix, resulting from the GE, during the BS.

However, in order to ensure that the difference in CPU time between the minimized and maximized case solely results from the number of operations in the inner solver, the solver information must also be investigated. Figure 6.32 to 6.36 show the normalized number of Jacobians for the outer solver, the normalized number of BS for the outer solver, the normalized number of Jacobians for the inner solver, the normalized number of BS for the inner solver and the average time step size respectively. All figures show that the solver data is equal for the minimized and maximized case. Hence, the difference in CPU time between the minimized and maximized case solely results from the number of operations in the inner solver.

Figure 6.29 also shows the normalized CPU time for a hypothetical Newton-Newton solver, which has an infinitely fast inner solver. The hypothetical Newton-Newton solver is discussed in section 3.5.3. Such a solver decreases approximately quadratically with the number of QSS species, since the Jacobian of the outer solver is assumed to be reused. The difference between the hypothetical solver and the minimized Newton-Newton solver increases with the number of QSS species but the difference is of the same order as the difference between the minimized and maximized Newton-Newton solver for high reduction levels. This difference is much smaller than the difference between the Newton-Newton solver and the Newton-FP solver. Hence, the performance of the minimized Newton-Newton solver is relatively close to the hypothetical solver.

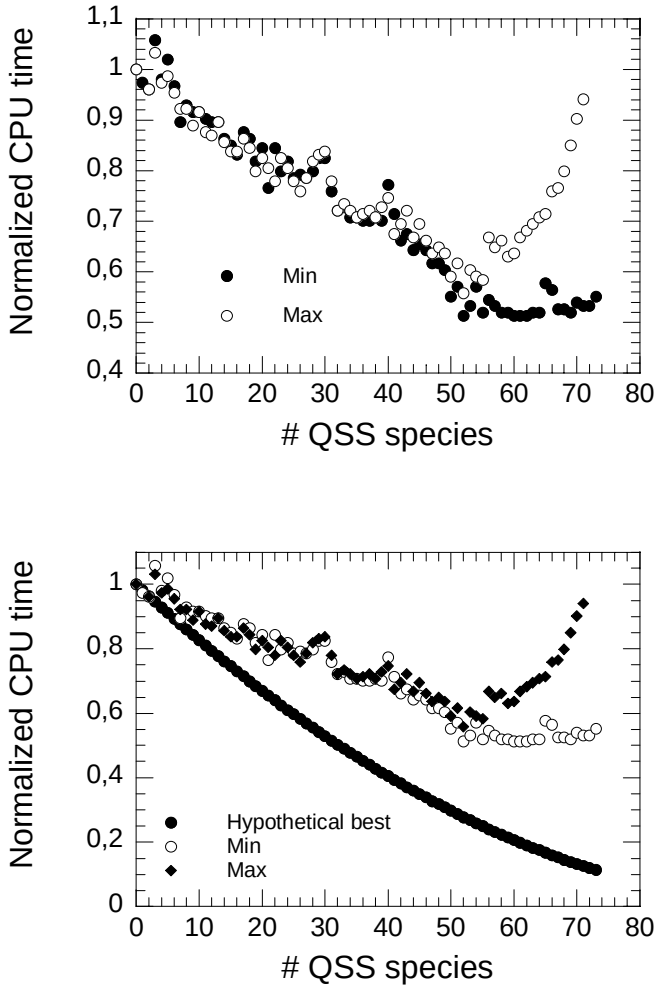


Figure 6.29. Normalized CPU time vs number of QSS species for minimized and maximized number of operations in the inner solver.

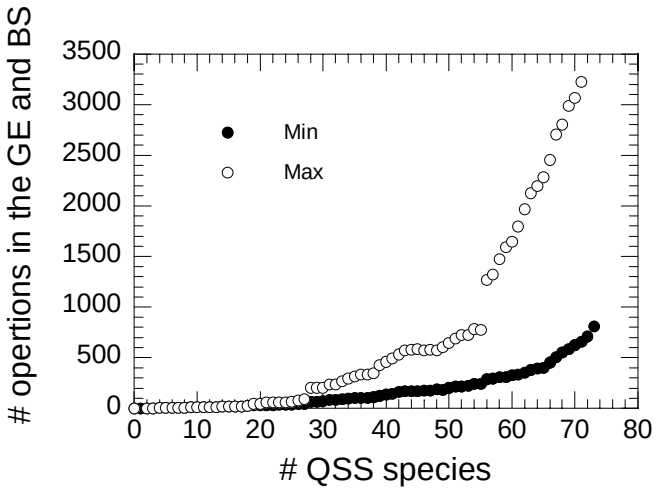


Figure 6.30. Minimized and maximized number of operations in the GE and BS of the inner solver vs number of QSS species.

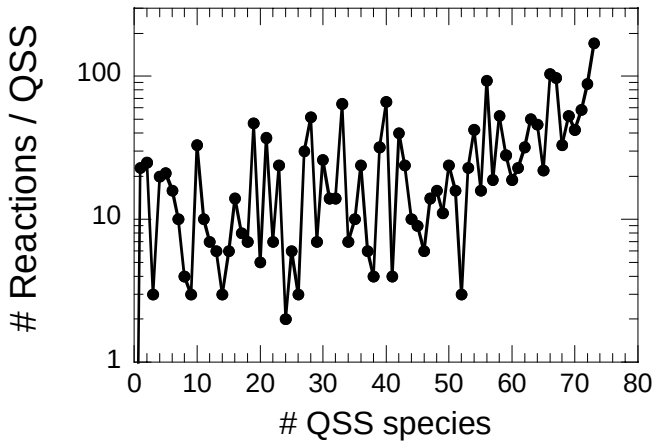


Figure 6.31. Number of reactions per QSS species vs number of QSS species.

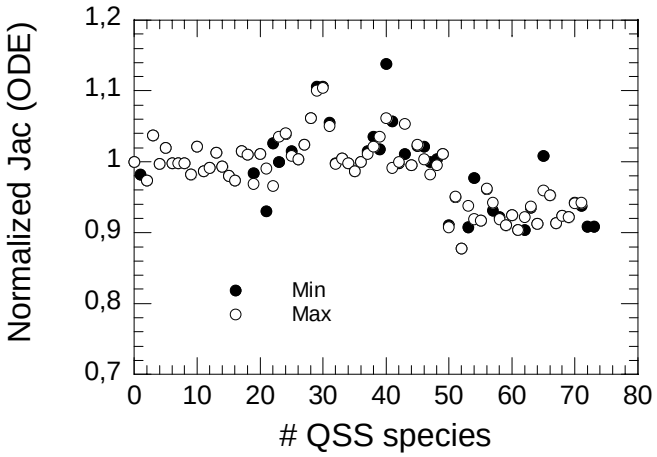


Figure 6.32. Normalized number of Jacobians (ODE) vs number of QSS species for minimized and maximized number of operations in the inner solver.

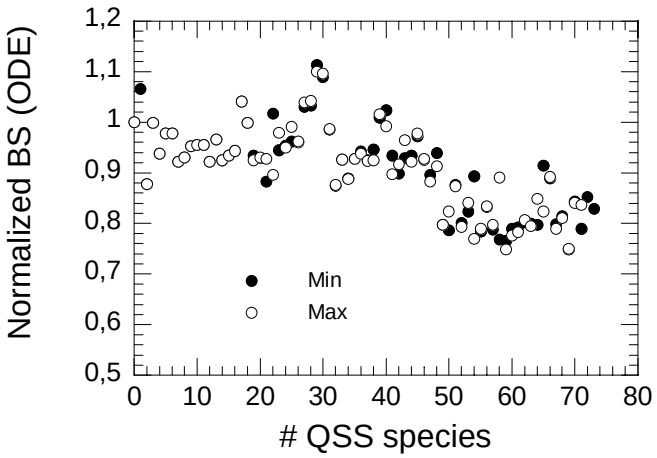


Figure 6.33. Normalized number of BS (ODE) vs number of QSS species for minimized and maximized number of operations in the inner solver.

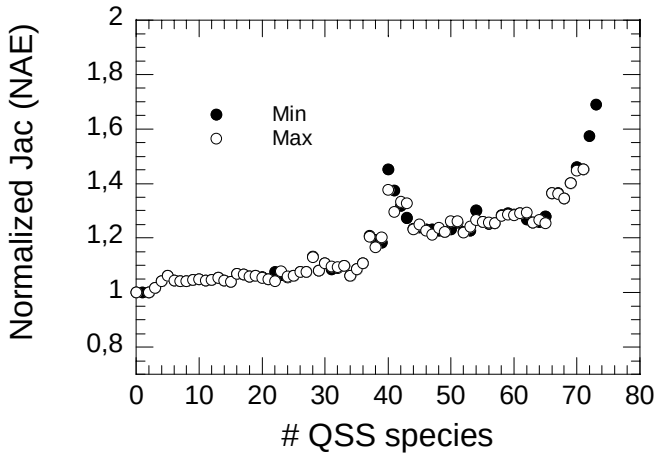


Figure 6.34. Normalized number of Jacobians (NAE) vs number of QSS species for minimized and maximized number of operations in the inner solver.

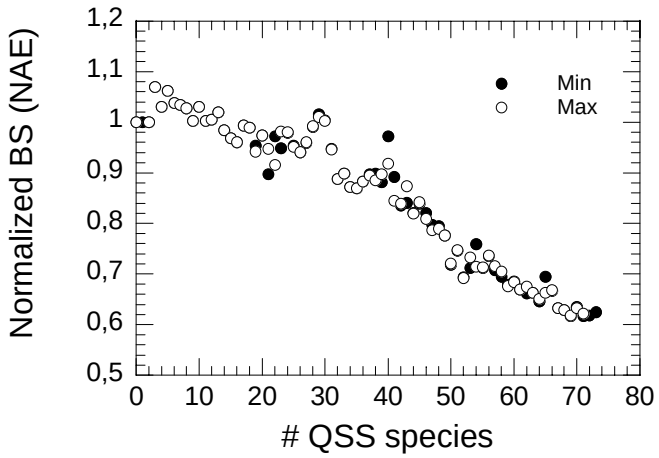


Figure 6.35. Normalized number of BS (NAE) vs number of QSS species for minimized and maximized number of operations in the inner solver.

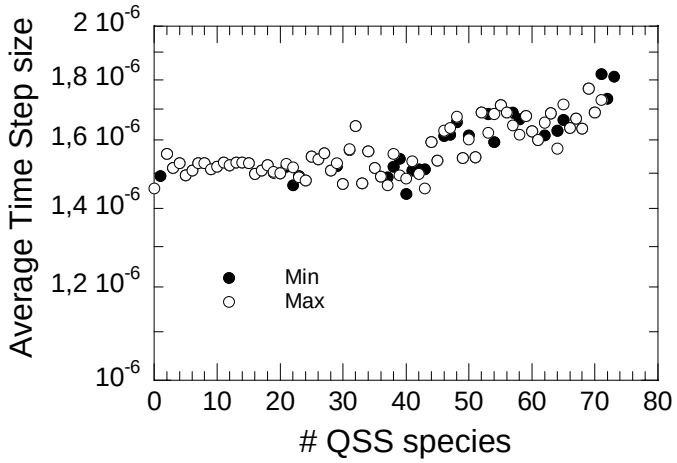


Figure 6.36. Average time step size vs number of QSS species for minimized and maximized number of operations in the inner solver.

6.2.1.3.1 Sparseness pattern of the NAE Jacobian

Figure 6.37 shows the minimized and maximized Jacobian before GE, Figure 6.38 shows the minimized and maximized Jacobian after GE and Figure 6.39 shows minimized and maximized "GE-matrix". The interpretation of the GE-matrix is explained in section 4.3.3.3.3. All the figures correspond to 70 QSS species.

There is a clear difference in the number Non Zero Elements (NZE) in the Jacobian after GE, which corresponds to the number of operations in the BS, between the minimized and maximized Jacobian. The minimized and maximized Jacobians have 348 and 1504 NZE respectively.
There is also a clear difference in the number NZE in the GE matrix, which corresponds to the number of operations in the GE, between the minimized and maximized Jacobian. The minimized and maximized Jacobians have 281 and 1562 NZE in the GE matrix respectively. The number of NZE of the Jacobian after GE and number of NZE of the GE matrix are added in order to get the total number of NZE (and thereby operations) shown in Figure 6.30.

The sparseness pattern of the minimized and maximized Jacobian before GE both contains 463 NZE, which means that they contain the same species. The sparseness patterns look quite similar at a first glance, except for the minimized Jacobian being more dens in the lower right corner and the maximized Jacobian being more dense in the upper and left side.
It is hard to understand just by looking at the sparseness pattern if the Jacobian that corresponds to the minimized sparseness pattern will minimize or maximize the number of NZE in the GE and BS. It is nearly impossible to foresee the permutations of the QSS species vector that would minimize the number of operations in the inner solver, since a

given permutation can have little of huge effect on the pattern and that the pattern is a function of the history of permutations. Hence, the MBSA algorithm is used for optimization of the sparseness pattern.

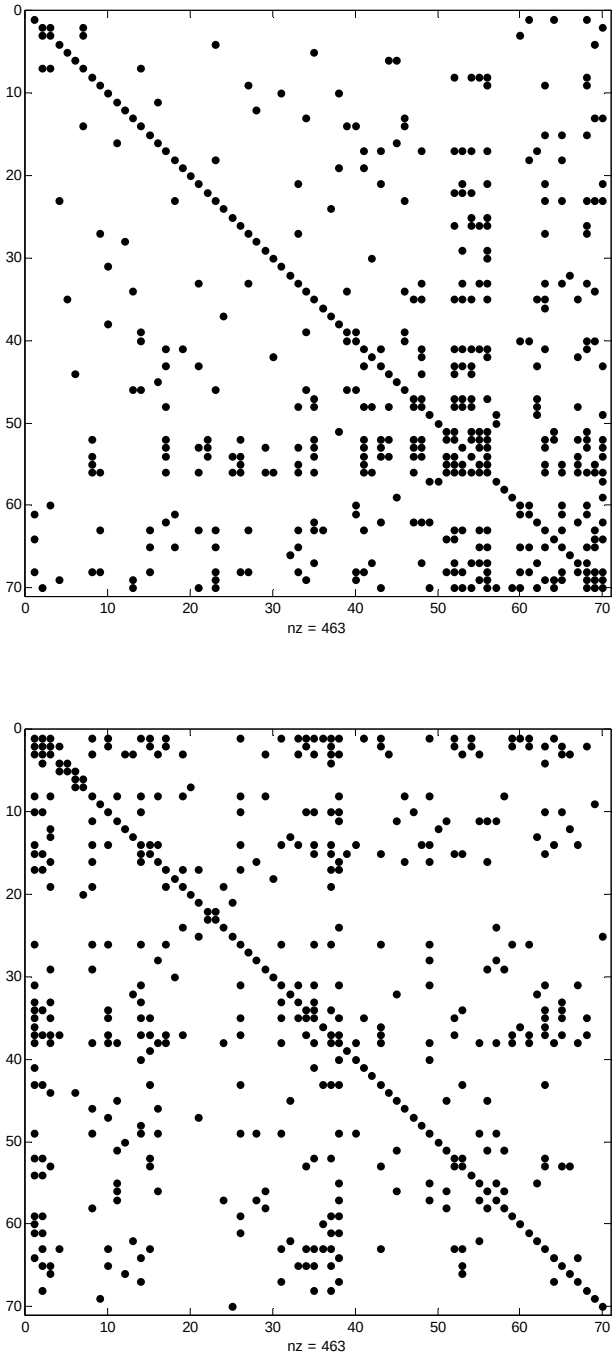


Figure 6.37. The minimized and maximized Jacobian before GE for 70 QSS species. The number of non zeros is 463 for both Jacobians, which means that the same species are involved.

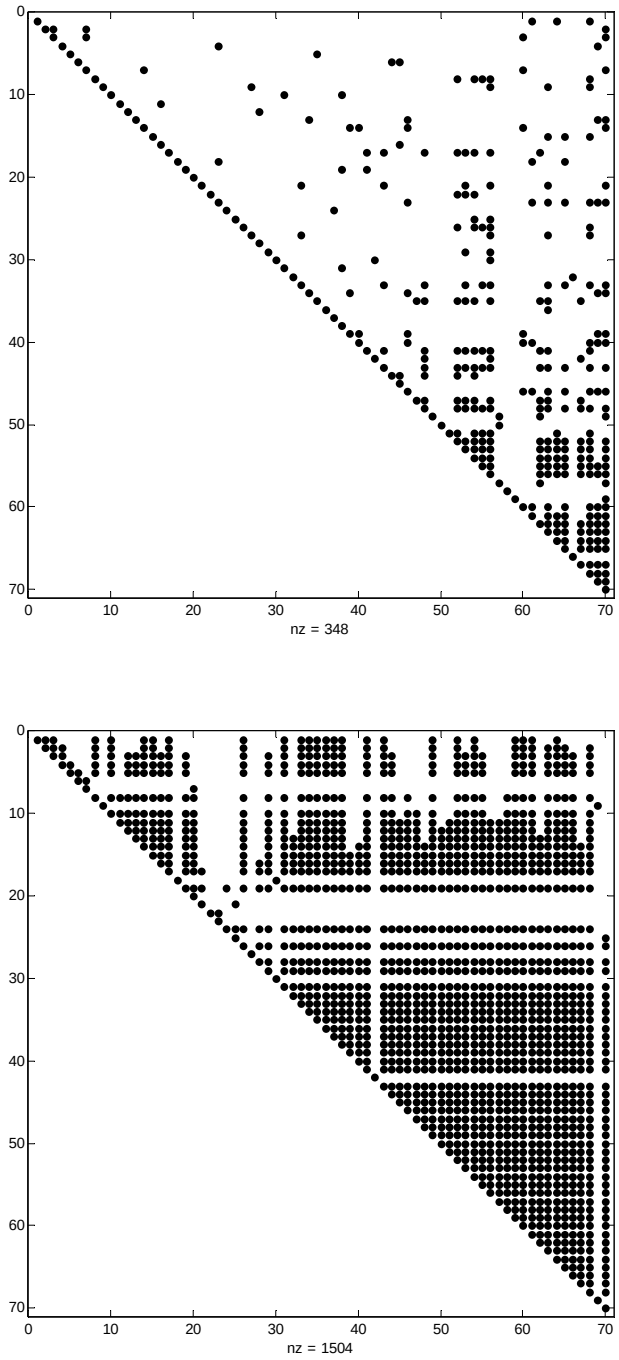


Figure 6.38. The minimized and maximized Jacobian after GE for 70 QSS species. The number of non zeros for the minimized and maximized case is 348 and 1504 respectively.

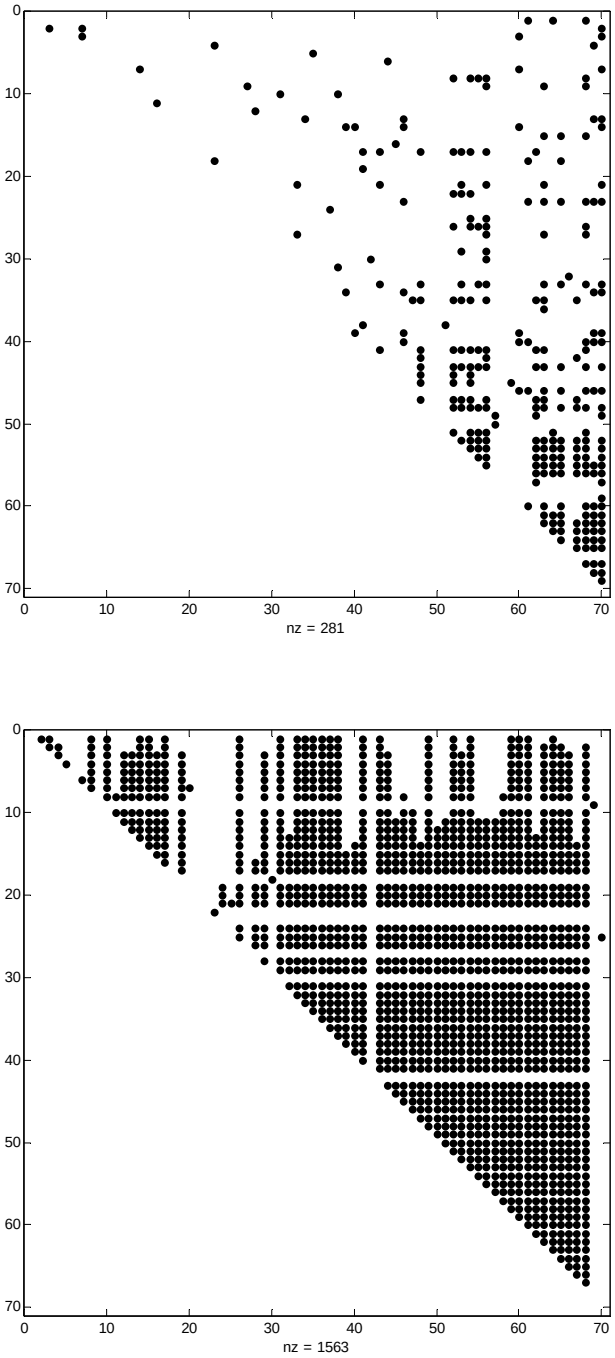


Figure 6.39. The minimized and maximized “GE-matrix” for 70 QSS species. The number of non zeros for the minimized and maximized case is 281 and 1563 respectively.

Concluding remarks
There is a clear difference in CPU time between the minimized and maximized number of operations in the inner solver, which shows the benefits of the MBSA algorithm. It is very hard to predict the sparseness pattern of the Jacobian that minimizes or maximizes the number of operation during and after the GE. For such problems the MBSA algorithm is very powerful.
Also, the minimized Newton-Newton solver is close to the hypothetical best solver, which has an infinitely fast inner solver. This means that the inner solver and its interaction with the outer solver cannot be improved much more. Hence, the minimized Newton-Newton solver is a very fast solver combination for stiff DAE systems based on chemical kinetics.

6.2.2 Variation of ART ET

This section investigates the effect of variation of ART ET. The ART ETs that are varied are IDT HF, IDT CF and Max HO2 CF. Each ART ET limit varied while the others are kept constant. All possible permutations cannot be shown. Hence, only the "relevant" (which is subjective) permutations of the ART ET limits are shown.

This investigation uses the same physical conditions and LOI list as Case I, which is shown in Table 6.1.

6.2.2.1 Variation of IDT HF limit

Simulations were performed when the IDT HF limit was 1, 2, 3, 4 and 5 %. The IDT CF limit, the Max HO2 CF limit and the Max OH HF limit were fixed at 5 %, 5% and 1 % respectively during the simulations.

6.2.2.1.1 Reduction level of the reduced mechanisms

Figure 6.40 shows that the ART could identify 73, 73, 73, 73 and 74 QSS species for 1, 2, 3, 4 and 5 % respectively. This means that 1/3 of the original mechanism is still described by ODE and 2/3 of the original mechanism instead is described by NAE. The curves corresponding to 1, 2, 3 and 4 % are equal and start to differ to the curve corresponding to 5 % at 70 QSS species, since the species corresponding to the LOI rank 88, which is C3H4P according to Table 6.2, was accepted for the 5 % case and not by the others. These simulations show that the IDT HF limit does not affect the final mechanism much when the other limits were set as stated above.

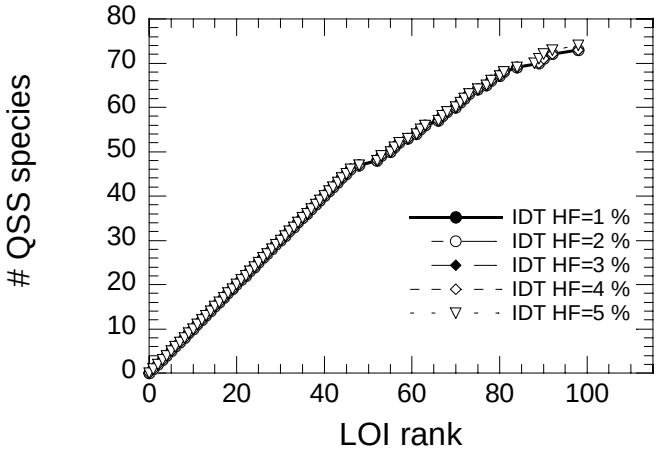


Figure 6.40. Number of QSS species vs LOI rank for IDT HF limits of 1, 2, 3, 4 and 5 %.

6.2.2.1.2 CPU time of the reduced mechanisms

Figure 6.41 shows the normalized CPU time vs the number of QSS species. The normalized CPU time decreases as the number of QSS species increases to a minimum point of about 0,5 at about 60 QSS species. Thereafter the normalized CPU time fluctuates between 0,5 and 0,55. The mechanisms corresponding to all limits are identical up to 70 QSS species as explained above. Therefore the CPU time is expected to be identical for all limits. However, the CPU time shows some small variations for each reduced mechanism. This variation can be explained by noise due to other processes in the computer.

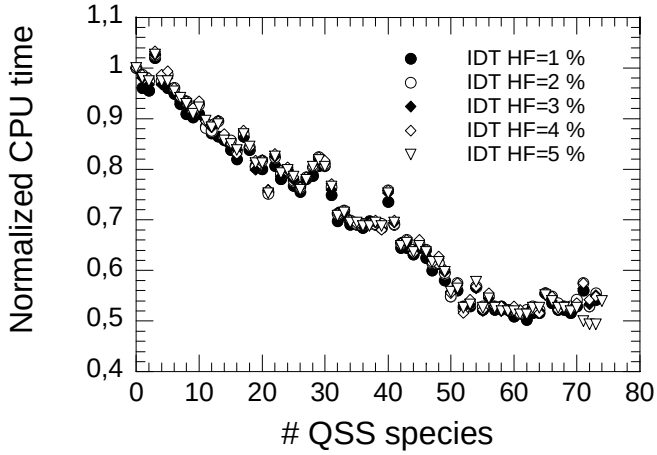


Figure 6.41. Normalized CPU time vs number of QSS species for IDT HF limits of 1, 2, 3, 4 and 5 %.

6.2.2.1.3 Accuracy of the reduced mechanisms

Figure 6.42 shows the Δ IDT HF (%) vs the number of QSS species. The IDT HF decreases in steps and reaches a value of about -8 % for the most reduced mechanisms for IDT HF limit 1, 2, 3 and 4 %. The curve for IDT HF limit 5 % follows the same path as the others but jumps significantly, due to the high limit, at 70 QSS and reaches a final value of about -13 %.

Figure 6.43 shows the Δ IDT CF (%) vs the number of QSS species. The IDT CF decreases in steps and reaches a value of about -20 % for the most reduced mechanisms for all limits.

Figure 6.44 shows the Δ Max HO_2 CF (%) vs the number of QSS species. The Max HO_2 CF reaches about 10 % for the most reduced mechanisms for all limits.

Figure 6.45 shows the Δ Max OH HF (%) vs the number of QSS species. The Max OH HF remains very low for all QSS for all limits and reaches a maximum of 0,4 % for the 1, 2, 3 and 4 % limit. The curve for 5 % limit follows the same path as the others but jumps significantly, due to the high limit, at 70 QSS and reaches a maximum value of about 0,6 %.

The large jump for Δ IDT HF at 70 QSS species can also be seen in the Δ Max OH HF but cannot be seen for Δ IDT CF or Δ Max HO_2 CF. This means that the species, C_3H_4P, which is responsible for the jump is a typical high temperature species.

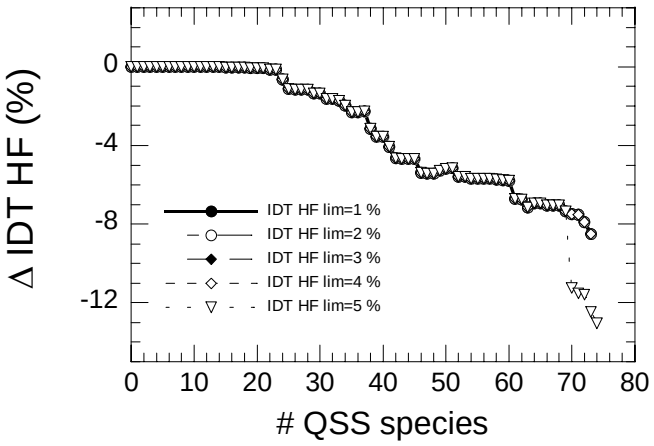


Figure 6.42. Δ IDT HF (%)vs number of QSS species for IDT HF limits of 1, 2, 3, 4 and 5 %.

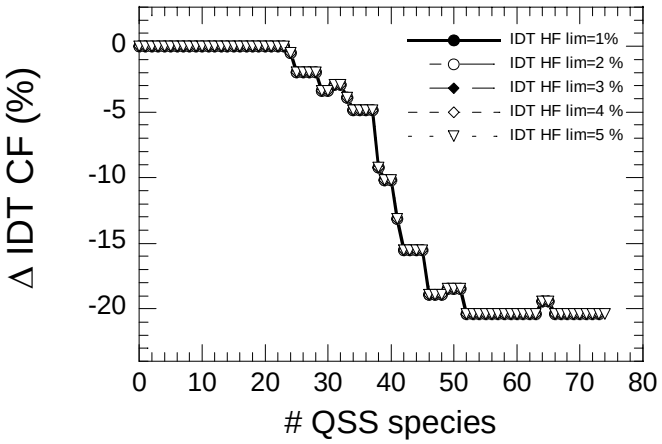


Figure 6.43. Δ IDT CF (%) vs number of QSS species for IDT HF limits of 1, 2, 3, 4 and 5 %.

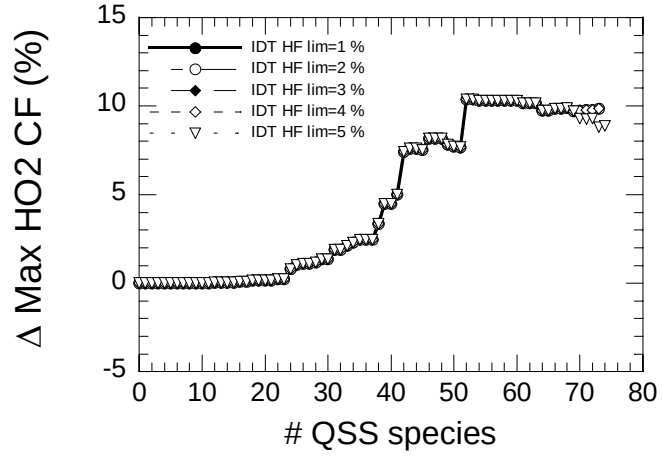


Figure 6.44. Δ Max HO_2 CF (%) vs number of QSS species for IDT HF limits of 1, 2, 3, 4 and 5 %.

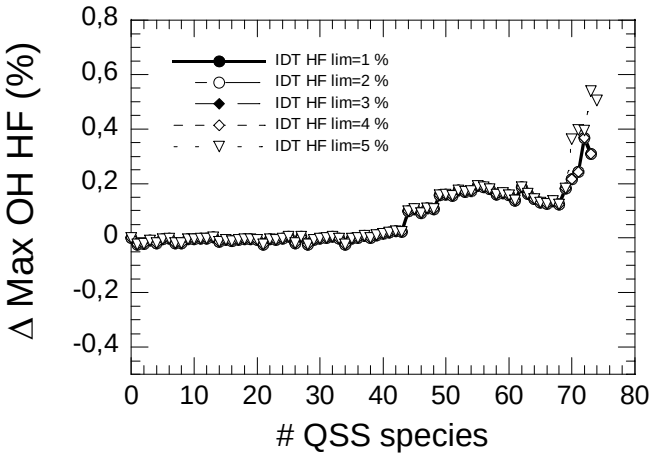


Figure 6.45. Δ Max OH HF (%) vs number of QSS species for IDT HF limits of 1, 2, 3, 4 and 5 %.

Concluding remarks
The IDT HF limit does not affect the final mechanism much when the other ART parameters are set as stated above. If the IDT HF limit is increased to 5 %, a species that affects the accuracy significantly is accepted as QSS species. The accuracy cost is high compared to the reduction gain of one QSS species.

6.2.2.2 Variation of IDT CF limit

Simulations were performed when the IDT CF limit was 1, 3, 5 and 7 %. The IDT HF limit, the Max HO2 CF limit and the Max OH HF limit were fixed at 3 %, 5% and 1 % respectively during the simulations.

6.2.2.2.1 Reduction level of the reduced mechanisms

Figure 6.46 shows that the ART could identify 66, 71, 73 and 74 QSS species for 1, 3, 5 and 7 % respectively. Hence, 40, 35, 34 and 33 % of the original mechanism is still described by ODE for 1, 3, 5 and 7 % respectively. The first species that fails to be accepted occurs at LOI rank 25, 38, 47 and 49 for 1, 3, 5 and 7 % respectively. These simulations show that the IDT CF limit does affect the final mechanism when the other limits were set as stated above.
Table 6.3 shows the species that were accepted and the species that failed to be accepted for the different IDT CF limits.

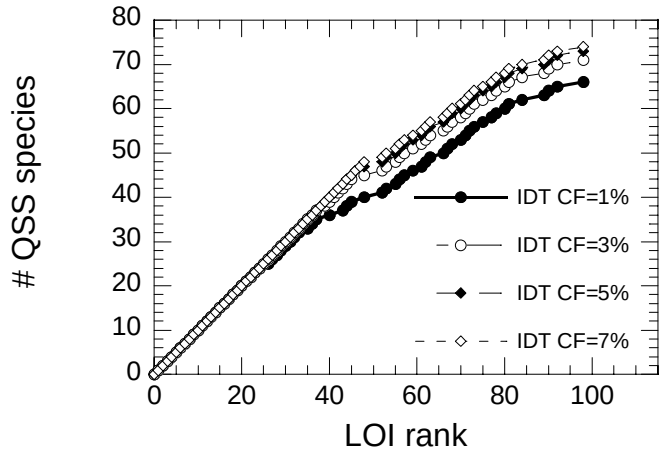


Figure 6.46. Number of QSS species vs LOI rank for IDT CF limits of 1, 3, 5 and 7 %.

Table 6.3. The table shows the LOI rank of the species, the species names and the corresponding LOI value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted for IDT CF limit 1, 3, 5, and 7 %. A blank means that the species was accepted, while the letter F means that the species failed to be accepted.

Nr	Species	LOI rank	1%	3%	5%	7%
1	CH2CH2COCH3	0.57992E-23				
2	C2H5COCH2	0.81611E-23				
3	CH2CH2CHO	0.30747E-22				
4	CH3CHCOCH3	0.67565E-22				
5	N-C3H7CO	0.29803E-21				
6	C2H5O	0.30558E-20				
7	5R-HEOOH-P	0.10122E-19				
8	HOCH2O	0.16819E-19				
9	N-C3H7COC2H4-1	0.10651E-17				
10	C2H5CO	0.11704E-17				
11	7R-HEOOH-P	0.24758E-17				
12	6R-HEOOH-P	0.36896E-17				
13	C2H4O2H	0.39947E-17				
14	5R-HEOOHO2-P	0.86579E-17				
15	7R-HEOOH-S	0.18380E-16				
16	1-C5H11	0.19356E-16				
17	2-C5H11	0.25297E-16				
18	5R-HEOOH-S	0.35434E-16				
19	N-C3H7	0.50926E-16				
20	CH3COCH2	0.58997E-16				
21	1-C4H9	0.66697E-16				
22	6R-HEOOH-S	0.10622E-15				
23	CH	0.18024E-15				
24	C6H10	0.21326E-15				

25	7R-O-HEPOOH-P	0.44094E-15	F			
26	1-C2H4COC2H5	0.71797E-15				
27	I-C3H7	0.99270E-15				
28	CH3O	0.10338E-14				
29	6R-HEOOHO2-P	0.16628E-14				
30	C2H	0.18632E-14				
31	HO2CHO	0.19006E-14				
32	1-CH2	0.25684E-14				
33	L-C7H15	0.26253E-14				
34	7R-HEOOHO2-P	0.27514E-14	F			
35	OCHO	0.35343E-14				
36	2-C4H8	0.36433E-14				
37	C6H5O	0.50284E-14				
38	7R-O-HEPOOH-S	0.71393E-14	F	F		
39	C2H5O2	0.10700E-13	F			
40	C4H7	0.11039E-13				
41	7R-HEOOHO2-S	0.12656E-13	F			
42	CH3O2H	0.14152E-13	F			
43	C7H13	0.16824E-13				
44	A1-	0.18662E-13				
45	C6H8	0.24974E-13				
46	5R-HEOOHO2-S	0.25519E-13	F	F		
47	6R-O-HEPOOH-P	0.33649E-13	F	F	F	
48	C4H612	0.49074E-13				
49	6R-HEOOHO2-S	0.58707E-13	F	F	F	F
50	CH3O2	0.70054E-13	F	F	F	F
51	5R-O-HEPOOH-S	0.71260E-13	F	F	F	F
52	N-C4H3	0.75976E-13				
53	A1	0.86912E-13				
54	O2CHO	0.87279E-13	F	F	F	F
55	C4H10	0.11237E-12				
56	C5H9	0.14757E-12				
57	N-C4H9COCH2	0.17086E-12				
58	6R-O-HEPOOH-S	0.22825E-12	F	F	F	F
59	N-C4H5	0.23363E-12				
60	C2H5O2H	0.27242E-12	F	F	F	F
61	C2H6	0.33900E-12				
62	I-C4H3	0.45295E-12				
63	HCO	0.46041E-12				
64	N-C3H7COCH2	0.65550E-12	F	F	F	F
65	L-C7H15O2	0.66422E-12	F	F	F	F
66	HCCO	0.68469E-12				
67	3-CH2	0.14667E-11				
68	I-C4H5	0.17894E-11				
69	CH3OH	0.34782E-11	F	F	F	F
70	CH2CHO	0.36697E-11				
71	HOCHO	0.49648E-11				
72	C4H4	0.62090E-11				
73	C4H6	0.63781E-11				
74	1-C4H8	0.81147E-11	F	F	F	F
75	C3H8	0.98504E-11				
76	5R-C7H14O	0.98797E-11	F	F	F	F
77	CH2OH	0.16731E-10				

78	C2H5	0.23923E-10				
79	C3H6	0.27350E-10	F	F	F	F
80	C2H3	0.28788E-10				
81	CH3CO	0.40888E-10				
82	1-C5H10	0.42325E-10	F	F	F	F
83	CH3CHO	0.47077E-10	F	F	F	F
84	C3H5	0.53180E-10				
85	N-C3H7CHO	0.60902E-10	F	F	F	F
86	H2O2	0.78482E-10	F	F	F	F
87	C2H5CHO	0.87104E-10	F	F	F	F
88	C3H4P	0.11307E-09	F	F	F	F
89	C3H4	0.13220E-09				
90	C3H3	0.18911E-09				
91	L-C7H14	0.21697E-09	F	F	F	F
92	C2H2	0.23131E-09				
93	HO2	0.25436E-09	F	F	F	F
94	O	0.74893E-09	F	F	F	F
95	CH2CO	0.83539E-09	F	F	F	F
96	CH2O	0.95470E-09	F	F	F	F
97	C2H5COCH3	0.14892E-08	F	F	F	F
98	CH3	0.35709E-08				
99	CH4	0.64018E-08	F	F	F	F
100	C2H4	0.65893E-08	F	F	F	F
101	OH	0.30677E-07	F	F	F	F
102	CO2	0.92212E-07	F	F	F	F
103	H	0.10853E-06	F	F	F	F
104	N-C7H16	0.26023E-06	F	F	F	F
105	H2	0.27517E-06	F	F	F	F
106	CO	0.31018E-06	F	F	F	F
107	H2O	0.49953E-04	F	F	F	F
108	O2	0.62892E-04	F	F	F	F
109	N2	0.10000E+01	F	F	F	F
110	AR	0.10000E+01	F	F	F	F

6.2.2.2.2 CPU time of the reduced mechanisms

Figure 6.47 shows the normalized CPU time vs the number of QSS species. The normalized CPU time decreases as the number of QSS species increases to a minimum point of about 0,5 at about 60 QSS species. Thereafter the normalized CPU time fluctuates between 0,5 and 0,55 for IDT CF limits 3, 5 and 7 %. The normalized CPU time instead fluctuates between 0,55 and 0,6 for the 1 % limit. The slightly higher normalized CPU time for the 1 % limit can be explained by the fact that fewer QSS species are accepted for the 1 % limit before the species with high LOI rank, which affects the convergence negatively, are accepted.
The small differences in CPU time between the different limits for a given number of QSS species are due to computer noise and different trajectories in concentration space for different QSS species combinations.

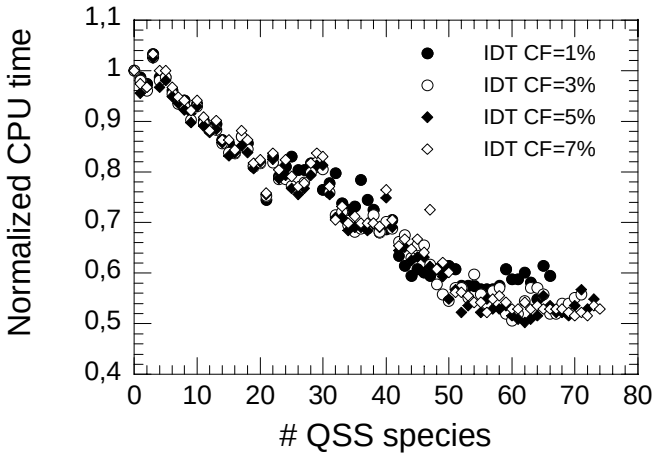


Figure 6.47. Normalized vs number of QSS species for IDT CF limits of 1, 3, 5 and 7 %.

6.2.2.2.3 Accuracy of the reduced mechanisms

Figure 6.48 shows the deviation in Δ IDT HF (%) vs the number of QSS species. The IDT HF deviation for the most reduced mechanism exists between about -5 and -10 % for all IDT CF limits. The deviation is larger for the higher limits and vise versa. Hence, the species that affect the IDT CF do also affect the IDT HF.

Figure 6.49 shows the Δ IDT CF (%) vs the number of QSS species. The IDT CF deviation for the most reduced mechanism exists between about -4 and -27 % for all IDT CF limits. The deviation is larger for the higher limits and vise versa.

Figure 6.50 shows the Δ Max HO2 CF (%) vs the number of QSS species. The Δ Max HO2 CF for the most reduced mechanism exists between about 3 and 12 % for all IDT CF limits. The deviation is larger for the higher limits and vise versa. Hence, the species that affect the IDT CF do also affect the Max HO2 CF.

Figure 6.51 shows the Δ Max OH HF (%) vs the number of QSS species. The Δ Max OH HF for the most reduced mechanism exists between about 0.3 and 0.4 % for all IDT CF limits. The difference in Δ Max OH HF is small between the different limits. Hence, the species that affect the IDT CF do not affect the Max OH HF much.

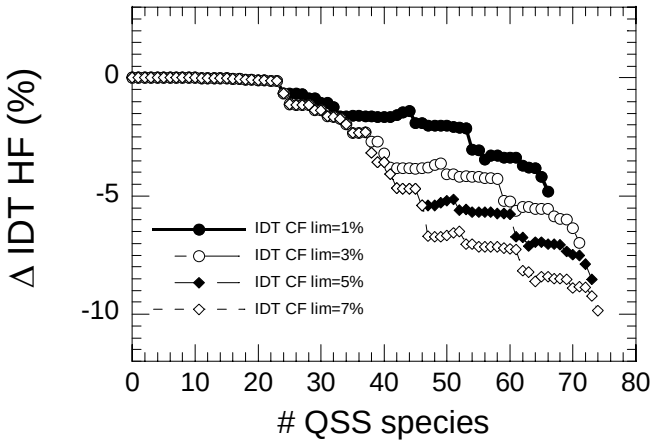


Figure 6.48. Δ IDT HF (%) vs number of QSS species for IDT CF limits of 1, 3, 5 and 7 %.

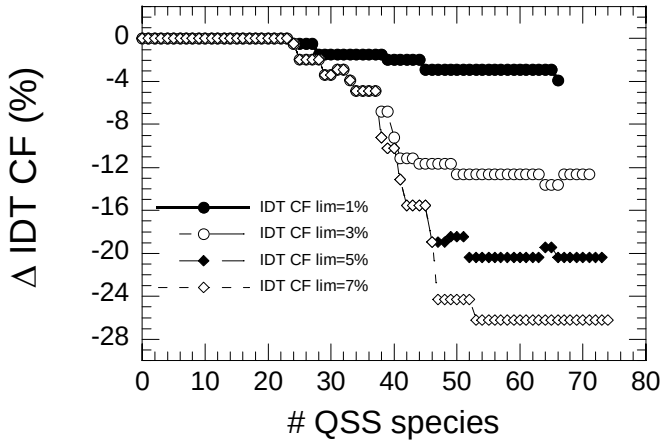


Figure 6.49. Δ IDT CF (%) vs number of QSS species for IDT CF limits of 1, 3, 5 and 7 %.

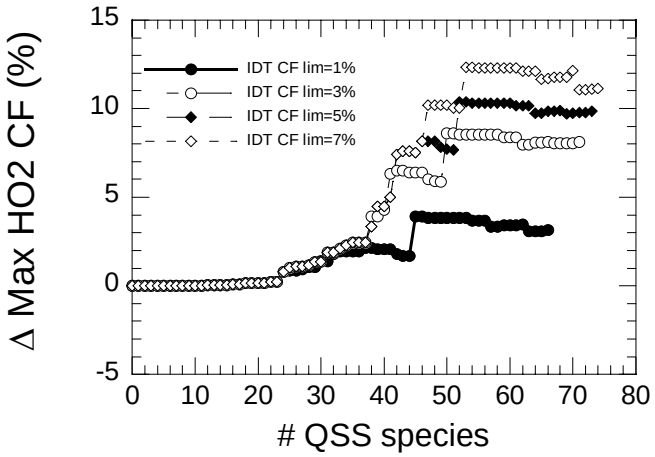


Figure 6.50. Δ Max HO_2 CF (%) vs number of QSS species for IDT CF limits of 1, 3, 5 and 7 %.

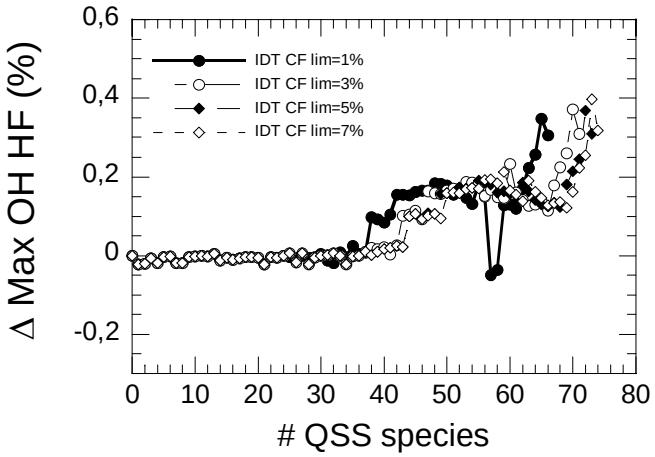


Figure 6.51. Δ Max OH HF (%) vs number of QSS species for IDT CF limits of 1, 3, 5 and 7 %.

Concluding remarks

The IDT CF limit does affect the final mechanism more than the IDT HF limit does when the other ART parameters are set as stated above. If the IDT CF limit is increased, species that affect the accuracy significantly are accepted as QSS species. The accuracy cost is high, especially for IDT CF and Max HO2 CF, compared to the reduction gain of a few QSS species when the IDT CF limit is increased.

6.2.2.3 Variation of Max HO_2 CF limit

Simulations were performed when the Max HO2 CF limit was 1, 2, 3 and 4 %. The IDT HF limit, the IDT CF limit and the Max OH HF limit were fixed at 3 %, 5% and 1 % respectively during the simulations.

6.2.2.3.1 Reduction level of the reduced mechanisms

Figure 6.52 shows that the ART could identify 70, 72, 73 and 73 QSS species for 1, 2, 3 and 4 % respectively. Hence, 36, 35, 34 and 34 % of the original mechanism is still described by ODE for 1, 2, 3 and 4 % respectively. The first species that fails to be accepted occurs at LOI rank 39, 42, 47 and 47 for 1, 2, 3 and 4 % respectively. These simulations show that the Max HO2 CF limit does affect the final mechanism when the other limits were set as stated above.

Table 6.4 shows the species that were accepted and the species that failed to be accepted for the different Max HO2 CF limit.

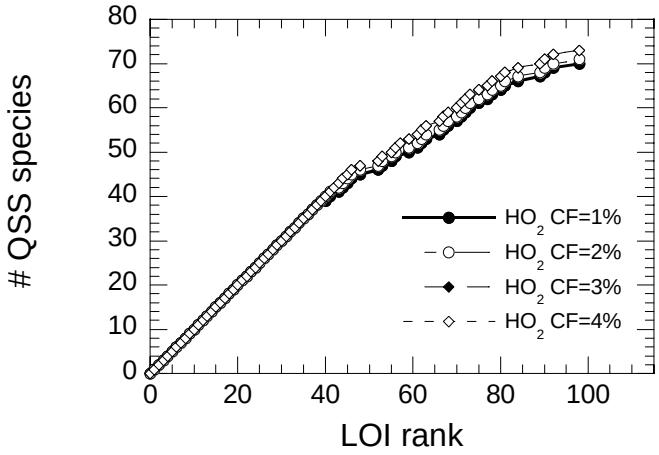


Figure 6.52. Number of QSS species vs LOI rank for HO_2 CF limits of 1, 2, 3 and 4 %.

Table 6.4. The table shows the LOI rank of the species, the species names and the corresponding LOI value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted for Max HO2 CF limit 1, 2, 3, and 4 %. A blank means that the species was accepted, while the letter F means that the species failed to be accepted.

Nr	Species	LOI value	1%	2%	3%	4%
1	CH2CH2COCH3	0.57992E-23				
2	C2H5COCH2	0.81611E-23				
3	CH2CH2CHO	0.30747E-22				
4	CH3CHCOCH3	0.67565E-22				
5	N-C3H7CO	0.29803E-21				
6	C2H5O	0.30558E-20				
7	5R-HEOOH-P	0.10122E-19				
8	HOCH2O	0.16819E-19				
9	N-C3H7COC2H4-1	0.10651E-17				
10	C2H5CO	0.11704E-17				
11	7R-HEOOH-P	0.24758E-17				
12	6R-HEOOH-P	0.36896E-17				
13	C2H4O2H	0.39947E-17				
14	5R-HEOOHO2-P	0.86579E-17				
15	7R-HEOOH-S	0.18380E-16				
16	1-C5H11	0.19356E-16				
17	2-C5H11	0.25297E-16				
18	5R-HEOOH-S	0.35434E-16				
19	N-C3H7	0.50926E-16				
20	CH3COCH2	0.58997E-16				
21	1-C4H9	0.66697E-16				
22	6R-HEOOH-S	0.10622E-15				
23	CH	0.18024E-15				
24	C6H10	0.21326E-15				
25	7R-O-HEPOOH-P	0.44094E-15	F			
26	1-C2H4COC2H5	0.71797E-15				
27	I-C3H7	0.99270E-15				
28	CH3O	0.10338E-14				
29	6R-HEOOHO2-P	0.16628E-14				
30	C2H	0.18632E-14				
31	HO2CHO	0.19006E-14				
32	1-CH2	0.25684E-14				
33	L-C7H15	0.26253E-14				
34	7R-HEOOHO2-P	0.27514E-14				
35	OCHO	0.35343E-14				
36	2-C4H8	0.36433E-14				
37	C6H5O	0.50284E-14				
38	7R-O-HEPOOH-S	0.71393E-14				
39	C2H5O2	0.10700E-13	F			
40	C4H7	0.11039E-13				
41	7R-HEOOHO2-S	0.12656E-13				
42	CH3O2H	0.14152E-13	F	F		
43	C7H13	0.16824E-13				
44	A1-	0.18662E-13				

45	C6H8	0.24974E-13				
46	5R-HEOOHO2-S	0.25519E-13				
47	6R-O-HEPOOH-P	0.33649E-13	F	F	F	F
48	C4H612	0.49074E-13				
49	6R-HEOOHO2-S	0.58707E-13	F	F	F	F
50	CH3O2	0.70054E-13	F	F	F	F
51	5R-O-HEPOOH-S	0.71260E-13	F	F	F	F
52	N-C4H3	0.75976E-13				
53	A1	0.86912E-13				
54	O2CHO	0.87279E-13	F	F	F	F
55	C4H10	0.11237E-12				
56	C5H9	0.14757E-12				
57	N-C4H9COCH2	0.17086E-12	F	F		
58	6R-O-HEPOOH-S	0.22825E-12	F	F	F	F
59	N-C4H5	0.23363E-12				
60	C2H5O2H	0.27242E-12	F	F	F	F
61	C2H6	0.33900E-12				
62	I-C4H3	0.45295E-12				
63	HCO	0.46041E-12				
64	N-C3H7COCH2	0.65550E-12	F	F	F	F
65	L-C7H15O2	0.66422E-12	F	F	F	F
66	HCCO	0.68469E-12				
67	3-CH2	0.14667E-11				
68	I-C4H5	0.17894E-11				
69	CH3OH	0.34782E-11	F	F	F	F
70	CH2CHO	0.36697E-11				
71	HOCHO	0.49648E-11				
72	C4H4	0.62090E-11				
73	C4H6	0.63781E-11				
74	1-C4H8	0.81147E-11	F	F	F	F
75	C3H8	0.98504E-11				
76	5R-C7H14O	0.98797E-11	F	F	F	F
77	CH2OH	0.16731E-10				
78	C2H5	0.23923E-10				
79	C3H6	0.27350E-10	F	F	F	F
80	C2H3	0.28788E-10				
81	CH3CO	0.40888E-10				
82	1-C5H10	0.42325E-10	F	F	F	F
83	CH3CHO	0.47077E-10	F	F	F	F
84	C3H5	0.53180E-10				
85	N-C3H7CHO	0.60902E-10	F	F	F	F
86	H2O2	0.78482E-10	F	F	F	F
87	C2H5CHO	0.87104E-10	F	F	F	F
88	C3H4P	0.11307E-09	F	F	F	F
89	C3H4	0.13220E-09				
90	C3H3	0.18911E-09				
91	L-C7H14	0.21697E-09	F	F	F	F
92	C2H2	0.23131E-09				
93	HO2	0.25436E-09	F	F	F	F
94	O	0.74893E-09	F	F	F	F
95	CH2CO	0.83539E-09	F	F	F	F
96	CH2O	0.95470E-09	F	F	F	F
97	C2H5COCH3	0.14892E-08	F	F	F	F

98	CH3	0.35709E-08				
99	CH4	0.64018E-08	F	F	F	F
100	C2H4	0.65893E-08	F	F	F	F
101	OH	0.30677E-07	F	F	F	F
102	CO2	0.92212E-07	F	F	F	F
103	H	0.10853E-06	F	F	F	F
104	N-C7H16	0.26023E-06	F	F	F	F
105	H2	0.27517E-06	F	F	F	F
106	CO	0.31018E-06	F	F	F	F
107	H2O	0.49953E-04	F	F	F	F
108	O2	0.62892E-04	F	F	F	F
109	N2	0.10000E+01	F	F	F	F
110	AR	0.10000E+01	F	F	F	F

6.2.2.3.2 CPU time of the reduced mechanisms

Figure 6.53 shows the normalized CPU time vs the number of QSS species. The normalized CPU time decreases as the number of QSS species increases to a minimum point of about 0,5 at about 60 QSS species. Thereafter the normalized CPU time fluctuates between 0,5 and 0,55 for all Max HO2 limits. The behavior is very similar for all limits. The small differences in CPU time between the different limits for a given number of QSS species are due to computer noise and different trajectories in concentration space for different QSS species combinations.

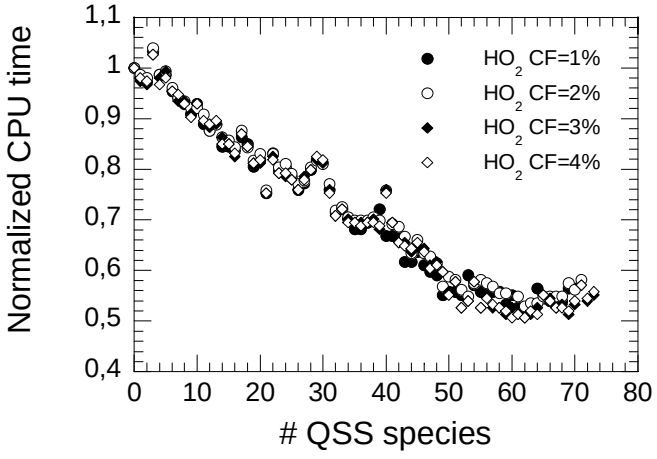


Figure 6.53. Normalized CPU time vs number of QSS species for HO_2 CF limits of 1, 2, 3 and 4 %.

6.2.2.3.3 Accuracy of the reduced mechanisms

Figure 6.54 shows the Δ IDT HF (%) vs the number of QSS species. The Δ IDT HF (%) for the most reduced mechanism exists between about -8 and -9 % for all Max HO2 CF limits. The difference in Δ IDT HF (%) is small between the different limits. Hence, the species that affect the Max HO2 CF do not affect the IDT HF much.

Figure 6.55 shows the Δ IDT CF (%) vs the number of QSS species. The Δ IDT CF (%) for the most reduced mechanism exists between about -14 and -20 % for all Max HO2 CF limits. The deviation is larger for the higher limits and vise versa. Hence, the species that affect the Max HO2 CF do also affect the IDT CF.

Figure 6.56 shows the Δ Max HO2 CF (%) vs the number of QSS species. The Δ Max HO2 CF (%) for the most reduced mechanism exists between about 4 and 10 % for all Max HO2 CF limits. The deviation is larger for the higher limits and vise versa.

Figure 6.57 shows the Δ Max OH HF (%) vs the number of QSS species. The Δ Max OH HF (%) for the most reduced mechanism exists between about 0.3 and 0.4 % for all Max HO2 CF limits. The difference in Δ Max OH HF (%) is small between the different limits. Hence, the species that affect the Max HO2 CF do not affect the Max OH HF much.

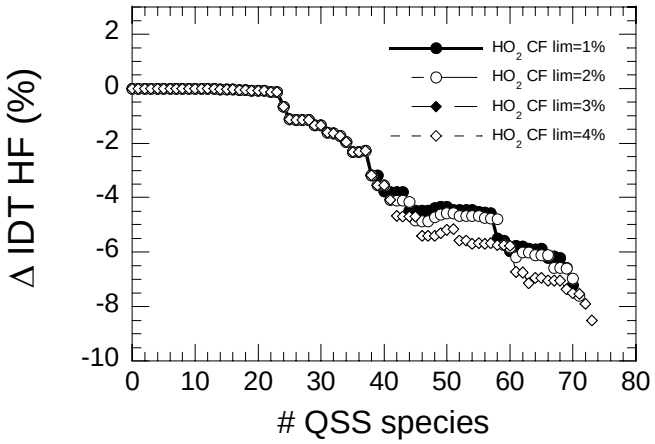


Figure 6.54. Δ IDT HF (%) vs number of QSS species for HO_2 CF limits of 1, 2, 3 and 4 %.

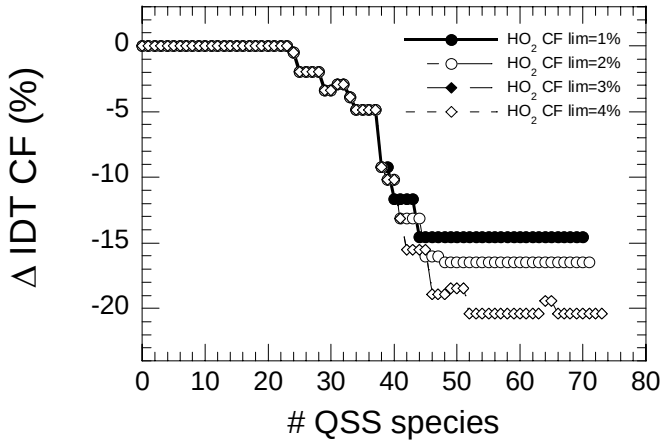


Figure 6.55. Δ IDT CF (%) vs number of QSS species for HO_2 CF limits of 1, 2, 3 and 4 %.

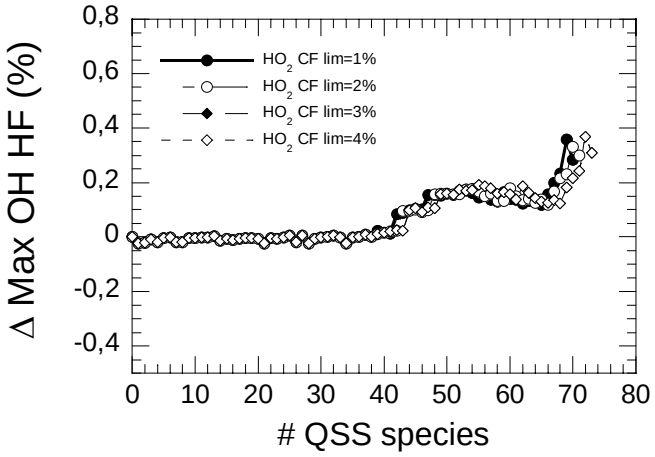


Figure 6.56. Δ Max OH HF (%) vs number of QSS species for HO_2 CF limits of 1, 2, 3 and 4 %.

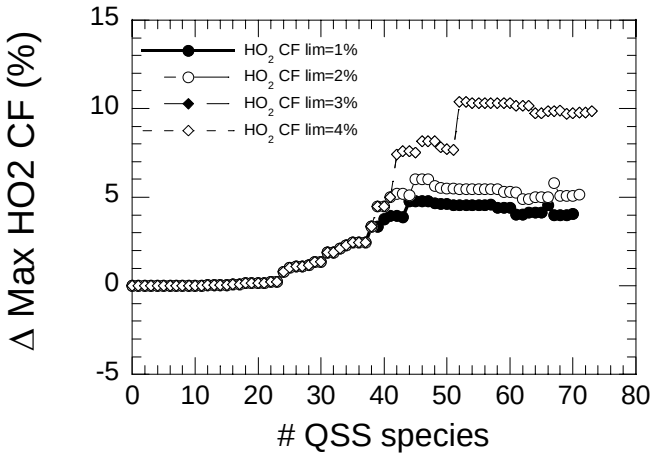


Figure 6.57. Δ Max HO_2 CF (%) vs number of QSS species for HO_2 CF limits of 1, 2, 3 and 4 %.

Concluding remarks
The Max HO2 CF does affect the final mechanism when the other ART parameters are set as stated above. If the Max HO2 CF limit is increased, species that affect the accuracy significantly are accepted as QSS species. The accuracy cost is high, especially for IDT CF and Max HO2 CF, compared to the reduction gain of a few QSS species when the Max HO2 CF limit is increased.

6.2.2.4 Variation of number of ET for chemical species

This section investigates the ART-option of many ET for chemical species, which generally leads to a less reduced mechanism than if one ET for chemical species is used. The 12 ET that are investigated are:

1. OH
2. HO_2
3. H_2
4. CO
5. CH_2O
6. H_2O
7. CH_4
8. C_2H_2
9. O_2
10. CO_2
11. N-C_7H_{16}
12. C_2H_4

Each ET is evaluated according to eq(6.2), where $ART\ ET_{RED}$ and $ART\ ET_{PREVRED}$ represent the maximum concentration of each species. The investigated ET limits for the chemical species are 1, 2 and 3 %. The IDT HF limit, IDT CF limit and the HO2 CF limit were fixed at 3 %, 5% and 5% respectively during the simulations. The ET limits for the chemical species are also referred to as target species limits.

6.2.2.4.1 Reduction level of the reduced mechanisms

Figure 6.58 shows the number of QSS species vs LOI rank. The most reduced mechanism corresponds to 53, 64 and 65 QSS species for 1, 2 and 3 % respectively. Hence, a higher limit corresponds to a more reduced mechanism as expected. The investigation shows that the "second chance" option in the ART is useful, especially when the 1 and 2 % limits are used, since some of the species are just on the border of being accepted the first time for these limits. Hence, some of the species that were on the "wrong side" of the border can be on the "right side" of the border when they are tested for the second time, which happens if X_{RED} and $X_{PREVRED}$ in eq.(6.2) are changed favorably compared to the first attempt. The species given a second chance have a LOI rank larger than 110.
The first species that failed to be accepted is the first to be given a second chance and so on.

It should be noted that the same LOI list that was used in previous investigation was also used in this investigation. That LOI list involved only species sensitivity for OH. The optimum LOI list for this investigation should involve sensitivity for all the ET for chemical species. Such a list would accept a larger number of QSS species before the first species was not accepted. However, the final number of QSS species would be about the same, independent of the LOI list, due to the properties of the ART.

Table 6.5 shows the species that were accepted and the species that failed to be accepted due to ET for chemical species stated above along with the original ETs.
The table shows that some species that fail to be accepted due to a few ET, while others fails for several ET. The species failing for several ET are correlated to higher LOI rank and vise versa. This is expected since species with higher LOI rank generally affects the whole system of DAE more than the species with low LOI rank.
Failure due to ET 2, 5, 8 and 12 are correlated, which means that the species that fails for HO_2 also seems to fail for CH_2O, C_2H_2 and C_2H_4. This is reasonable since those species are closely related and participates in the same set of reactions.
However, few species affect the ET for 6, 9, 10, and 11, which correspond to H_2O, O_2, CO_2 and N-C_7H_{16}. This is expected since the maximum concentration for N-C_7H_{16} and O_2 exists in the beginning of the simulation, before the QSS species has high enough concentration to affect them. The error due to the QSSA is accumulated during the simulation. The highest concentration of the products H_2O and CO_2 exists after the ignition point, which means that they are affected by the QSS species. However, there are

many reaction pathways to the products and the QSS species usually only affects a few of those pathways. Hence, the relative change is small due to most QSS species.

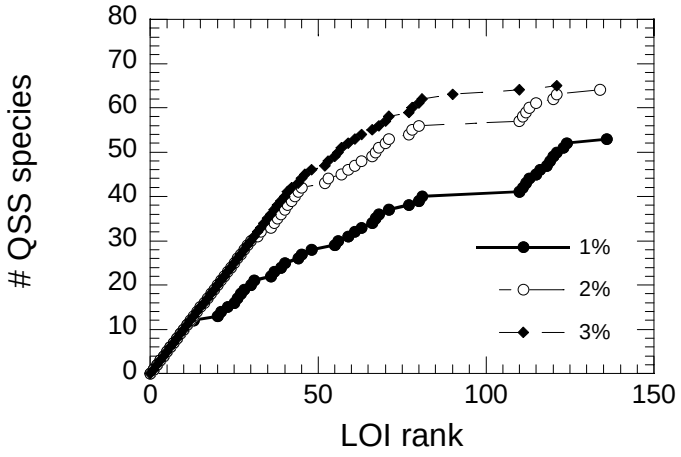


Figure 6.58. Number of QSS species vs LOI rank for the target species limits of 1, 2 and 3 %.

Table 6.5. Shows the species that were accepted and the species that failed to be accepted due to several ET for chemical species. The ET limit for the chemical species was 3%. A blank means that the species was accepted, while an "F" means that the species failed for the particular ET. The ETs are represented by the following numbers; 1) OH, 2) HO_2, 3) H_2, 4) CO, 5) CH_2O, 6) H_2O, 7) CH_4, 8) C_2H_2, 9) O_2, 10) CO_2, 11) N-C_7H_{16}, 12) C_2H_4
The species that failed for all ETs are either forbidden as QSS species beforehand or fail due to convergence problems of the solver.

		IDT HF	IDT CF	Max HO2 CF	1	2	3	4	5	6	7	8	9	10	11	12
1	CH2CH2COCH3															
2	C2H5COCH2															
3	CH2CH2CHO															
4	CH3CHCOCH3															
5	N-C3H7CO															
6	C2H5O															
7	5R-HEOOH-P															
8	HOCH2O															
9	N-C3H7COC2H4-1															
10	C2H5CO															
11	7R-HEOOH-P															
12	6R-HEOOH-P															
13	C2H4O2H															
14	5R-HEOOHO2-P															
15	7R-HEOOH-S															
16	1-C5H11															
17	2-C5H11															
18	5R-HEOOH-S															
19	N-C3H7															

20	CH3COCH2															
21	1-C4H9															
22	6R-HEOOH-S															
23	CH															
24	C6H10															
25	7R-O-HEPOOH-P															
26	1-C2H4COC2H5															
27	I-C3H7															
28	CH3O															
29	6R-HEOOHO2-P															
30	C2H															
31	HO2CHO															
32	1-CH2															
33	L-C7H15															
34	7R-HEOOHO2-P															
35	OCHO															
36	2-C4H8															
37	C6H5O															
38	7R-O-HEPOOH-S															
39	C2H5O2															
40	C4H7															
41	7R-HEOOHO2-S															
42	CH3O2H															
43	C7H13					**F**	**F**		**F**							
44	A1-															
45	C6H8															
46	5R-HEOOHO2-S															
47	6R-O-HEPOOH-P		**F**													
48	C4H612															
49	6R-HEOOHO2-S		**F**													
50	CH3O2	**F**	**F**								**F**					
51	5R-O-HEPOOH-S	**F**	**F**	**F**												
52	N-C4H3															
53	A1															
54	O2CHO	**F**	**F**	**F**												
55	C4H10															
56	C5H9															
57	N-C4H9COCH2															
58	6R-O-HEPOOH-S	**F**	**F**	**F**												
59	N-C4H5															
60	C2H5O2H			**F**												
61	C2H6															
62	I-C4H3											**F**				
63	HCO															
64	N-C3H7COCH2			**F**												
65	L-C7H15O2	**F**	**F**	**F**												
66	HCCO															
67	3-CH2					**F**					**F**					
68	I-C4H5															
69	CH3OH	**F**		**F**												
70	CH2CHO															
71	HOCHO															
72	C4H4											**F**				
73	C4H6					**F**			**F**			**F**				
74	1-C4H8	**F**	**F**	**F**												
75	C3H8															**F**
76	5R-C7H14O	**F**	**F**	**F**												
77	CH2OH															
78	C2H5															
79	C3H6	**F**	**F**	**F**					**F**		**F**	**F**				
80	C2H3															
81	CH3CO															
82	1-C5H10	**F**	**F**	**F**		**F**			**F**			**F**				**F**
83	CH3CHO		**F**	**F**												
84	C3H5										**F**	**F**				
85	N-C3H7CHO	**F**	**F**	**F**												

86	H2O2	F	F						F		F				
87	C2H5CHO	F	F	F											
88	C3H4P	F				F			F		F	F				F
89	C3H4					F					F	F				F
90	C3H3															
91	L-C7H14	F	F	F		F			F		F	F				F
92	C2H2					F						F				
93	HO2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
94	O	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
95	CH2CO	F	F	F		F			F		F	F				
96	CH2O	F	F	F		F			F		F	F				F
97	C2H5COCH3	F	F	F	F	F	F	F	F	F	F	F		F	F	F
98	CH3										F	F				
99	CH4	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
100	C2H4	F	F			F	F		F		F					F
101	OH	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
102	CO2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
103	H	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
104	N-C7H16	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
105	H2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
106	CO	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
107	H2O	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
108	O2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
109	N2	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
110	AR	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F

6.2.2.4.2 Normalized CPU time vs number of QSS species

Figure 6.59 shows that the normalized CPU time behaves just as expected from previous investigations. There is a steady decrease to about 0.6, 0.52 and 0.52 for the 1, 2 and 3 % limits respectively.

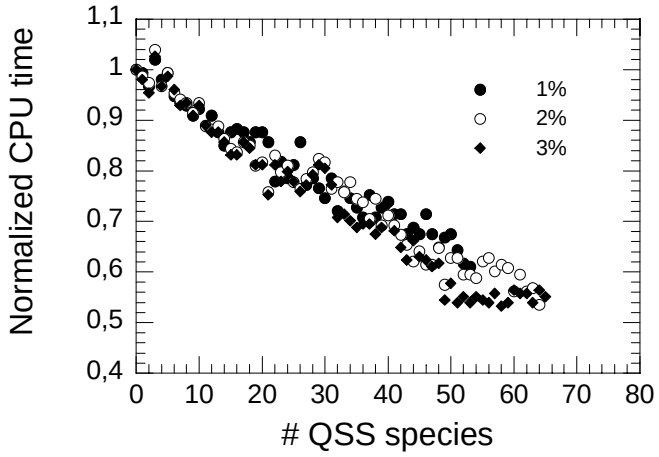


Figure 6.59. Normalized CPU time vs number of QSS species for the target species limits of 1, 2 and 3 %.

6.2.2.4.3 Accuracy of the reduced mechanisms

The Δ IDT HF (%), Δ IDT CF (%) and Δ Max HO_2 CF (%) are shown in Figure 6.60 to 6.62 respectively. The Δ IDT HF (%), Δ IDT CF (%) and Δ Max HO_2 CF (%) are all smaller for the 1 % limit than for the other limits, which is the same behavior as for the deviation in targets species concentrations, which are shown in Figure 6.63 and 6.64. This indicates that the same species that causes deviation in IDT HF, IDT CF and Max HO_2 CF also causes deviation in target species concentrations.

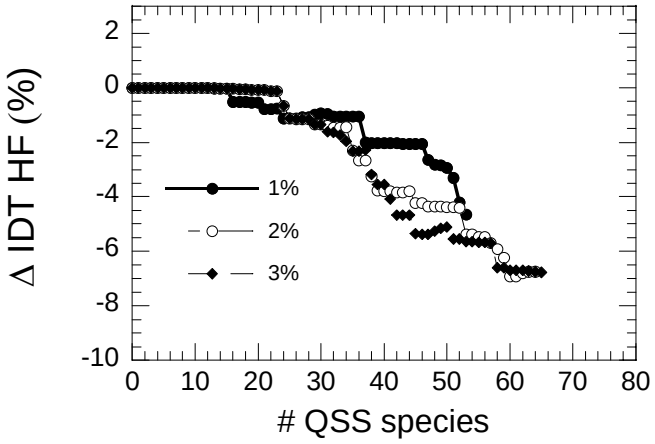


Figure 6.60. Δ IDT HF (%) vs number of QSS species for the target species limits of 1, 2 and 3 %.

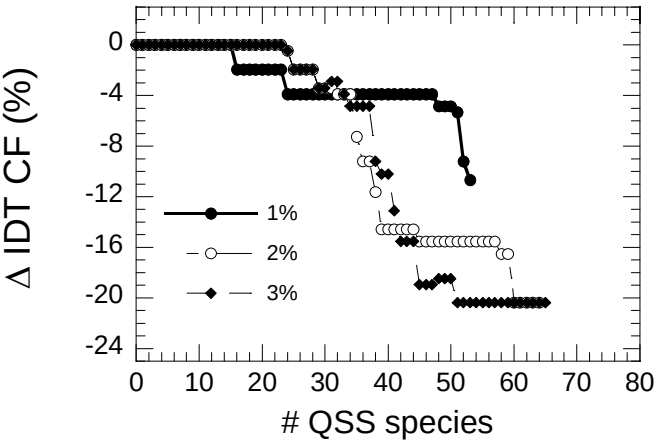


Figure 6.61. Δ IDT CF (%) vs number of QSS species for the target species limits of 1, 2 and 3 %.

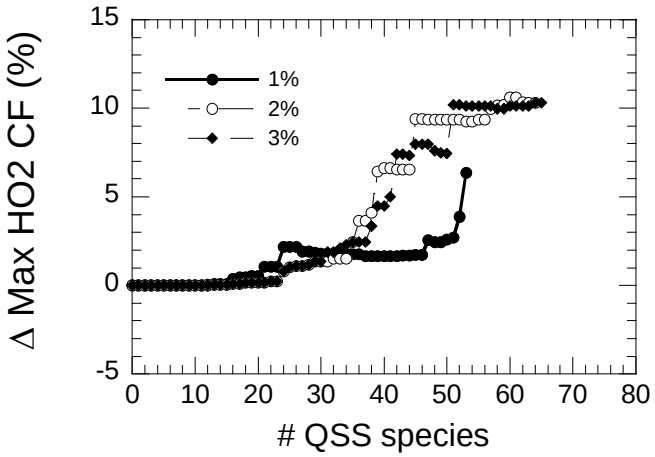


Figure 6.62. Δ Max HO_2 CF (%) vs number of QSS species for the target species limits of 1, 2 and 3 %.

Deviation of ET for chemical species

Figure 6.63 and 6.64 show the deviation (according to eq(6.4)) at the maximum concentration for each of the 12 target species vs number of QSS species for target species limits of 1,2 and 3%.

Figure 6.63a shows the deviation of the species OH and HO_2. The deviation for OH is very low for all limits, while the deviation for HO_2 reaches about 1, 2.5 and 2.5 % for the 1, 2 and 3 % limits respectively.

Figure 6.63b shows the deviation of the species C_2H_2 and C_2H_4. The deviation for C_2H_2 reaches about 4.5, 9 and 12 % for the 1, 2 and 3 % limits respectively, while the deviation for C_2H_4 reaches about 1.5, 2.5 and 2.5 % for the 1, 2 and 3 % limits respectively.

Figure 6.63c shows the deviation of the species and H_2 and CH_2O. The deviation for H_2 reaches about 3, 5 and 6 % for the 1, 2 and 3 % limits respectively, while the deviation for CH_2O reaches about 1.5, 2.5 and 3 % for the 1, 2 and 3 % limits respectively.

Figure 6.64a shows the deviation of the species and CO and CO_2. The deviation for CO reaches about 2, 2.5 and 3 % for the 1, 2 and 3 % limits respectively, while the deviation for CO_2 is very low for all limits.

Figure 6.64b shows the deviation of the species and CH_4 and N-C_7H_{16}. The deviation for CH_4 reaches about 3, 4 and 5 % for the 1, 2 and 3 % limits respectively, while the deviation for N-C_7H_{16} is very low for all limits.

Figure 6.64c shows the deviation of the species and H_2O and O_2. The deviation for H_2O is very low for all limits and reaches about 0.025, 0.036 and 0.036 % for the 1, 2 and 3 % limits respectively, while the deviation for O_2 is very low for all limits.
From these figures it is clear that C_2H_2, H_2 and CH_4 are the target species that are affected most by the chosen QSS species, while OH, CO_2, N-C_7H_{16}, H_2O and O_2 are hardly affected at all.

Figures 6.65 and 6.66 show the mass fraction vs time for all target species for the target species limits 1, 2 and 3 %. The profiles are shifted in time accordance with the IDT HF and IDT CF figures, while profiles are shifted in mass fraction in accordance to Figure 6.63 and 6.64. The profiles for the reduced mechanisms are in good agreement with the profile for the original mechanism.
The species profiles in Figure 6.65 and 6.66, which are a result of 12 target species, should be compared to Figure 6.19 and 6.20, which are a result of one target species. Just as expected, the accuracy of the most reduced mechanism increased for a couple of species profiles when 12 target species was used. The species that improved their accuracy are CH_4, CO, C_2H_4, CH_2O and H_2.

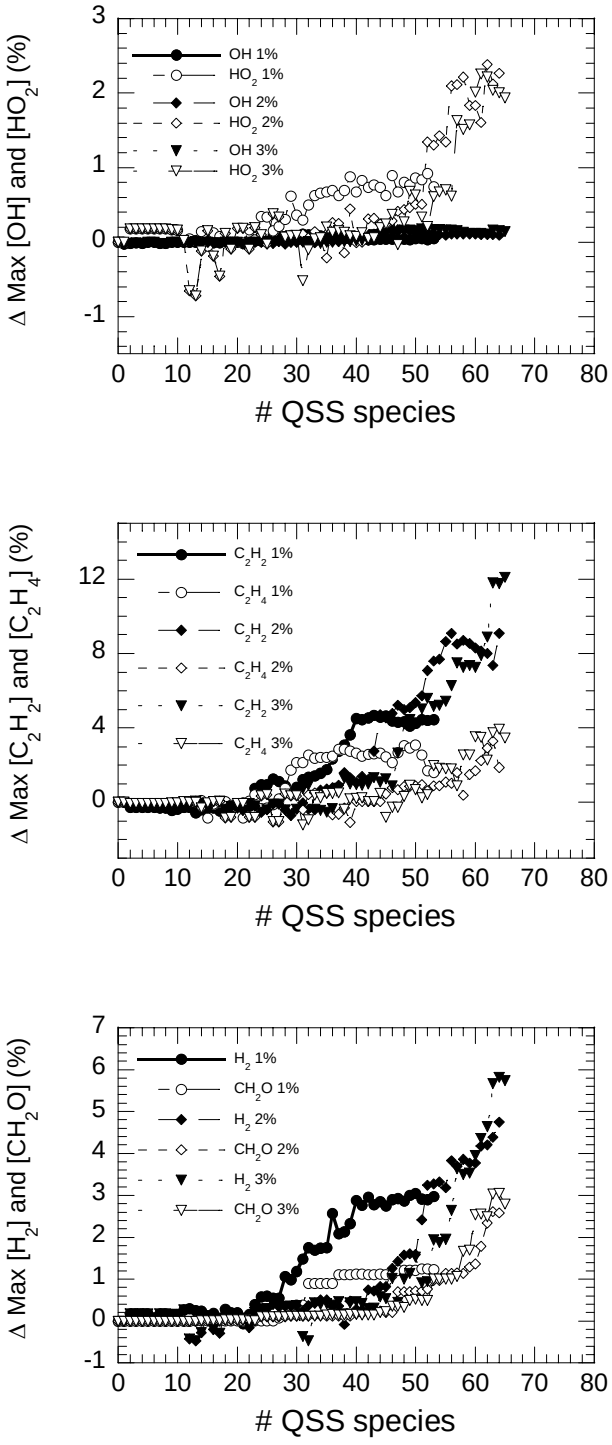


Figure 6.63. Deviation of maximum target species concentration vs number of QSS species for the target species limits of 1, 2 and 3 %. The top sub figure shows target species OH and HO_2, the middle sub figure shows target species C_2H_2 and $C_2H_4$and the lower sub figure shows target species H_2 and CH_2O.

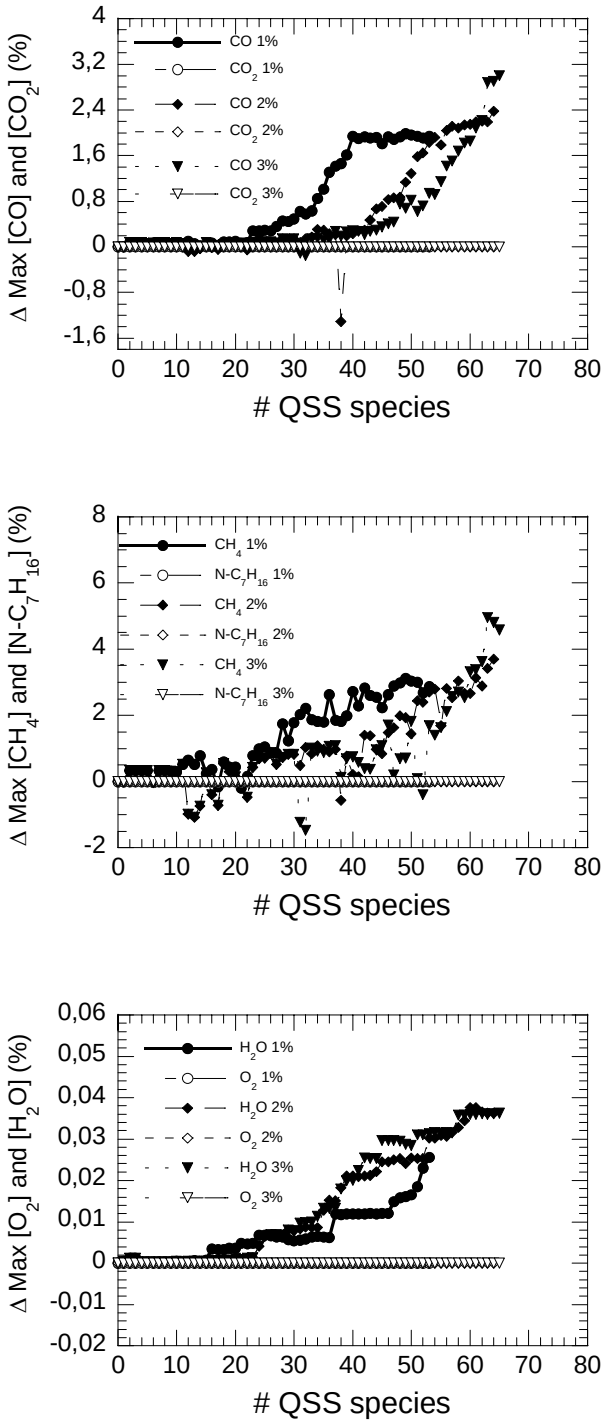


Figure 6.64. Deviation of maximum target species concentration vs number of QSS species for the target species limits of 1, 2 and 3 %. The top sub figure shows target species CO and CO_2, the middle sub figure shows target species CH_4 and N-C_7H_{16} and the lower sub figure shows target species O_2 and H_2O.

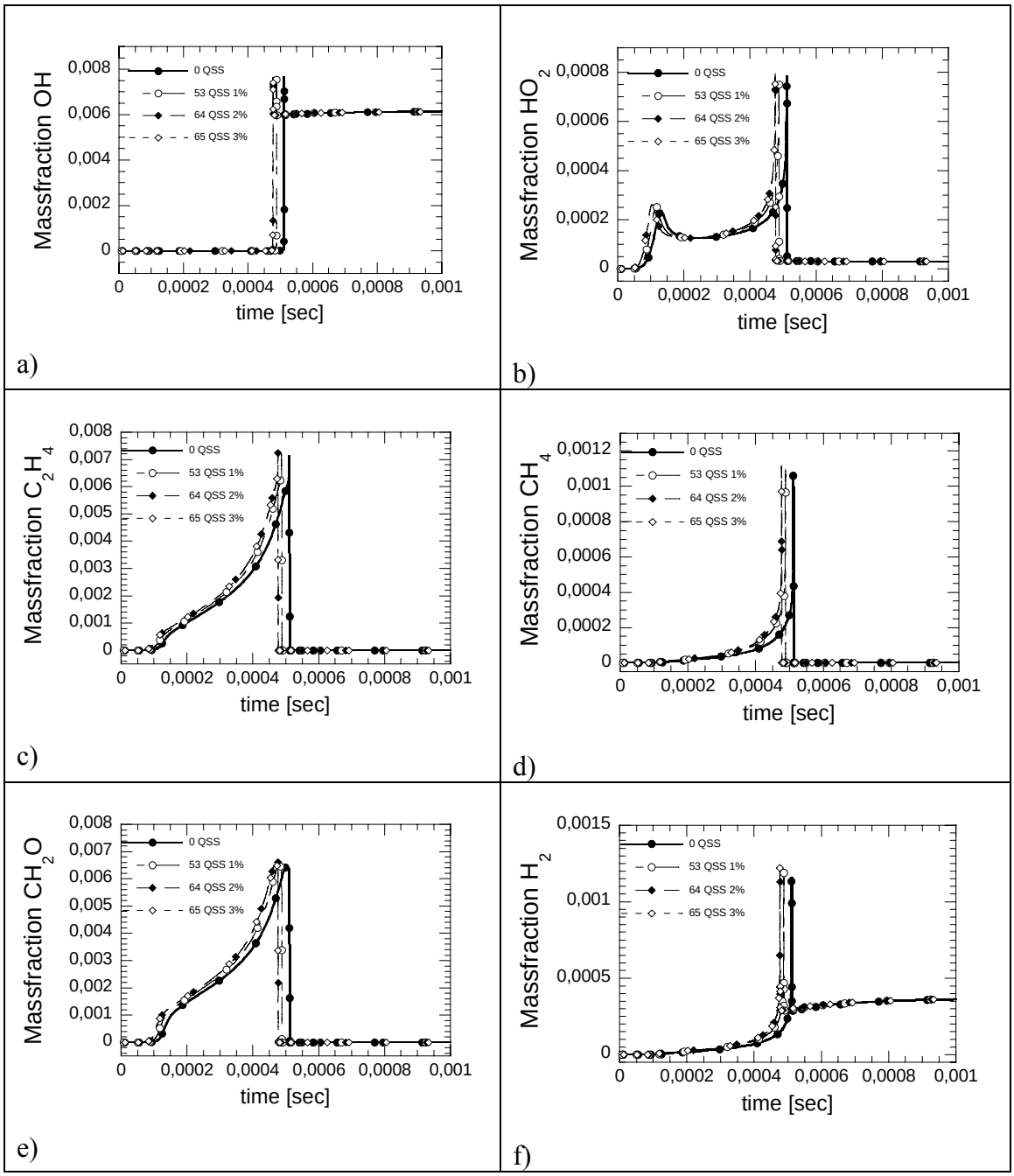


Figure 6.65. Mass fraction of important species vs time for 0, 53, 64 and 65 QSS. The reduced mechanisms correspond to ET limits of 1, 2 and 3% respectively. The sub figures corresponds to the species;
a) OH b) HO_2 c) C_2H_4 d) CH_4 e) CH_2O f) H_2

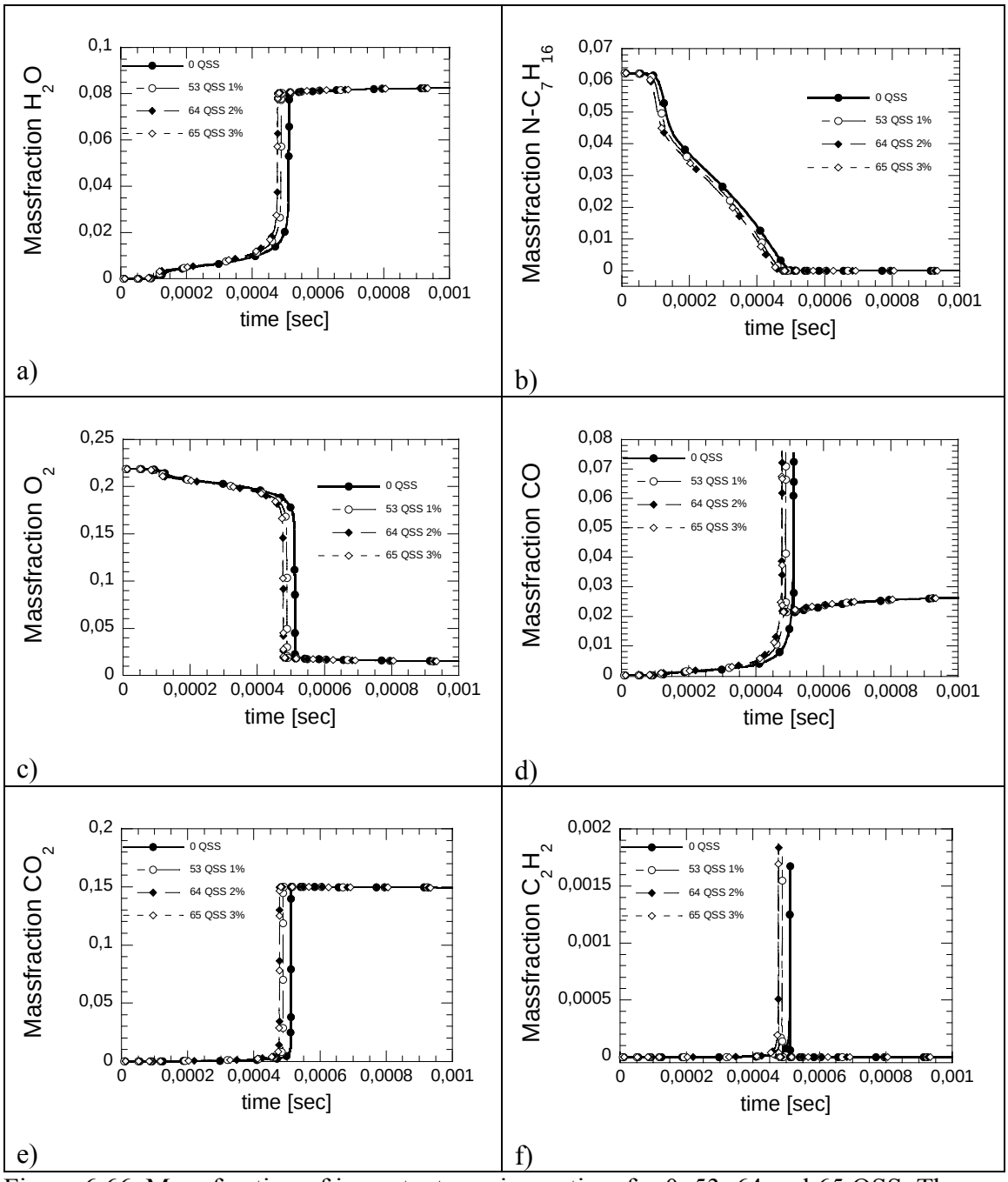


Figure 6.66. Mass fraction of important species vs time for 0, 53, 64 and 65 QSS. The reduced mechanisms correspond to ET limits of 1, 2 and 3% respectively. The sub figures corresponds to the species;
a) H_2O b) N-C_7H_{16} c) O_2 d) CO e) CO_2 f) C_2H_2

Concluding remarks
Fewer species are accepted as QSS species if the number of ET for chemical species is increased from 1 to 12. This is expected since the different ET for chemical species are affected differently by the QSS species. The number of accepted QSS species increases with increasing limits, which also is expected.
The accuracy cost is low compared to the reduction gain of 11 QSS species when the ET for chemical species limit is increased from 1 to 2 %. However, the accuracy cost is high compared to the reduction gain of a single QSS species when the ET for chemical species limit is increased from 2 to 3 %. Hence, an ET for chemical species limit of 2 % is preferable for this species case.

6.2.2.5 Variation of QSS species ranking list

This section investigates the ranking performance of different QSS species ranking lists. QSS species ranking lists can be based on different ranking measures like LT, sensitivity and species concentration. These ranking measures are explained more in detail below. The ranking measures can be combined into other ranking measures like LOI and LTC. The computational cost for the different ranking measures varies. It is low for LT and concentration and high for sensitivity. Hence, a QSS species ranking lists based on LOI is computationally demanding, while a QSS species ranking lists based solely on LT, solely on concentration and LTC is not. It is interesting to investigate the ranking performance of a QSS species ranking list and compare it to the computational cost for the ranking measure the QSS species ranking list is based on. This comparison is found in Table 6.6.

This investigation compares the ranking performance of QSS species ranking lists based on LT, Concentration, LOI and LTC. The ranking measures can be evaluated at different points in time during the combustion process. This thesis investigates the points in time for the CF, the HF and MaxS, where MaxS corresponds to the point in time when the species has its maximum concentration.

The following QSS species ranking lists have been investigated;

1. **LT MaxHO2 CF**: LT at the time when the HO_2 concentration has its first maximum.
2. **LOI MaxHO2 CF Sens$_{HO2}$**: LOI where the sensitivity is taken towards HO_2 at the time when the HO_2 concentration has its first maximum.
3. **LTC MaxHO2 CF**: LTC at the time when the HO_2 concentration has its first maximum.
4. **Conc. MaxHO2 CF**: Concentration at the time when the HO_2 concentration has its first maximum.
5. **LT MaxS**: LT at the time when the concentration of the species has its maximum.
6. **LOI MaxS Sens$_{OH}$**: LOI where the sensitivity is taken towards OH at the time when the concentration of the species has its maximum.
7. **LOI MaxS Sens$_{HO2}$**: LOI where the sensitivity is taken towards HO_2 at the time when the concentration of the species has its maximum.
8. **LTC MaxS**: LTC at the time when the concentration of the species has its maximum.
9. **LOI MaxHO2 HF**: LOI where the sensitivity is taken towards HO_2 at the time when the concentration of HO_2 has its maximum.
10. **Conc. MaxHO2 HF**: Concentration at the time when the concentration of HO_2 has its maximum.
11. **Max (LOI MaxS Sens$_{OH}$, LOI MaxHO2 CF Sens$_{HO2}$)**: The maximum value of "LOI MaxS Sens$_{OH}$" and "LOI MaxHO2 CF Sens$_{HO2}$" for each species.
12. **Max (LOI MaxS Sens$_{HO2}$, LOI MaxHO2 CF Sens$_{HO2}$)**: The maximum value of "LOI MaxS Sens$_{HO2}$" and "LOI MaxHO2 CF Sens$_{HO2}$" for each species.

All the QSS species ranking lists that have been tested in this thesis consist of combinations of LT, sensitivity and concentration. How they affect the QSS species rankings and which ranking targets they produce is described below. Explanations of "Ranking measure", "Ranking target" and "Evaluation target" are needed for the following discussion.

Ranking target (RT)
A ranking target is a quantity that the ranking measure is supposed to rank well. If the ranking target is the same as the evaluation target, a direct comparison can be made and the ranking performance can be evaluated. Otherwise a direct evaluation of the ranking performance cannot be made. However, the ranking measure can still have a good ranking performance even though the ranking targets and evaluation targets do not coincide.

Evaluation target (ET)
The ranking performance is evaluated by evaluation targets. The evaluation targets in this thesis are IDT HF, IDT CF, Max HO2 CF and Max OH HF. However, none of the ranking measures used IDT HF or IDT CF as a ranking target. This means that the

ranking measure cannot be expected to perform good ranking on IDT HF and IDT CF. However, the ranking performance on IDT HF and IDT CF can be optimized indirectly, since everything is connected. For example, the IDT CF is connected to Max HO2 CF through the ODE for HO_2, while IDT HF and Max OH HF are connected through the ODE for OH.

Ranking measures

LT
If a species is set in QSS, an error will be introduced in the QSS species profile which is proportional to the LT itself [4-5]. Hence, a large LT gives a large error in the QSS species profile and vise versa. However, this does not correspond to a ranking target that can be evaluated by any of the evaluation targets.
The LT is calculated from the diagonal elements in the Jacobian for the system of ODE, which is an approximation of the total LT. This is discussed further in section 5.4.2.1.1.1.

Sensitivity
Sensitivity taken towards species *X* makes the species *X* a ranking target. If the species *X* is OH or HO_2, the ranking target can be evaluated by the evaluation targets Max OH HF and Max HO2 CF respectively.
Hence, a LOI measure will also have the species *X* as a ranking target.

Concentration
A reason for the concentration to be a good measure for QSS species ranking is that the source term of the ODEs is a function of the species concentrations according to eq.(2.53). The larger a specific species concentration is and the more reactions the species is involved in, the more it affects the system of ODE.
Hence, if the species concentrations are observed at a specific point in time, the concentrations will be proportional to and thereby rank the impact the species have on the system of ODE at that point in time. This means that species with low concentrations will be appropriate QSS species, since they will have low impact on the system of ODE. The opposite is true for species with high concentrations. Hence, if the concentrations are measured at the time for Max HO2 CF, Max HO2 CF will be the ranking target.
A LTC measure will have the same ranking target as concentration does.

Other reasons for the concentration to be a good measure for QSS species ranking is discussed in section 5.4.2.1.4.

Investigation and ranking performance
The investigation is performed in the following way for each QSS species ranking list: The species are tested as QSS as species one by one, starting with the species with the lowest ranking and finishing with the species with the highest ranking. The accuracy of each reduced mechanism, based on a single QSS species, is observed. The accuracy of each reduced mechanism is evaluated by the evaluation targets. The deviations of all the evaluation targets are calculated according to eq.(6.4).

The ranking performance is based on two criteria;

1) For each evaluation target, the accumulated deviation, which is the sum of the deviations from each single QSS species, is used for evaluation of the ranking performance. The theoretical number of QSS species a reduced mechanism would contain before the accumulated deviation becomes unacceptable determines the ranking performance of the specific evaluation target. The limit where the deviation is unacceptable is subjective, but an accumulated deviation above about 10-15 % is considered unacceptable in this thesis.

2) A good ranking performance is also characterized by low deviation for a specific evaluation target for low QSS species ranking and high deviation for high QSS species ranking. A bad ranking performance is characterized by a randomly distributed deviation for all QSS species rankings. The ranking performance is divided into four groups, which are called are “Ordered”, “Semi Ordered”, “Semi Random” and “Random” and have decreasing ranking performance.

The performance of each QSS species ranking list is shown below. All figures shows absolute values of the deviation of single QSS species calculated from eq.(6.4). The accumulated deviation of the QSS species is theoretical and does not consider additive effects among the QSS species. The limit where the accumulated deviation is unacceptable is subjective. The reduced mechanisms that theoretically can be generated before the accumulated deviation is unacceptable **do not** take into account the included species possible effect on the convergence rate of the solver. The ART must be used in order to investigate the effect on the convergence rate.
Some of the figures show deviation vs LOI rank. These figures are used to determine how many QSS species that a reduced mechanism before the accumulated deviation becomes unacceptable. Some figures show deviation vs Log LOI value. These figures are used to determine how ordered the ranking performance is. The error bars in some of the sub figures correspond to the LT of the species.

The ranking performance for all QSS species ranking lists is summarized in Table 6.6, which shows the number of QSS species that can be accepted for each QSS species ranking list before the accuracy of the solution (assuming no additive effects) becomes unacceptable. The accuracy of IDT HF, IDT CF, Max HO2 CF and Max OH HF is used

for evaluation of the solution accuracy. The ranking performance for each ET is also shown.
The table also shows a rough estimate of the achieved reduction level and CPU cost for obtaining each QSS species ranking list.

Table 6.6 The table shows the number of QSS species that can be accepted for each QSS species ranking list before the accuracy of the solution (assuming no additive effects) becomes unacceptable. The accuracy of IDT HF, IDT CF, MaxHO2 CF and Max OH HF is used for evaluation of the solution accuracy. The table also shows a rough estimate of the achieved reduction level and CPU cost for each QSS species ranking list. The two best QSS species ranking list have bold text.

QSS species ranking list	IDT HF	IDT CF	Max HO2 CF	Max OH HF	Reduction level	CPU cost
1. LT MaxHO2 CF	45-60 Semi Ordered	35 Semi Ordered	65-70 Semi Random Good Prediction	90 Semi Random	Low	Low
2. LOI MaxHO2 CF $Sens_{HO2}$	**80-85 Ordered**	**80 Ordered**	**80-85 Ordered** **Ranking Target: Very Good Prediction**	**95 Semi Ordered**	**High**	**High**
3 LTC MaxHO2 CF	65-75 Semi Ordered	60-65 Semi Ordered	80 Semi Ordered Very Good Prediction	90 Semi Ordered	Average	Low
4. Conc. MaxHO2 CF	**80-85 Ordered**	**85 Ordered**	**80-85 Ordered** **Ranking Target: Very Good Prediction**	**95 Semi Ordered**	**High**	**Low**
5. LT MaxS	35 Semi Ordered	35 Semi Ordered	65-70 Semi Ordered	35 Random	Low	Low
6. LOI MaxS $Sens_{OH}$	40-50 Semi Ordered	30-40 Semi Ordered	40-50 Semi Ordered	95 Semi Ordered Ranking Target: Very Good Prediction	Low	High
7. LOI MaxS $Sens_{HO2}$	50-55 Semi Ordered	55-60 Semi Ordered	65-70 Semi Ordered	85 Semi Ordered	Average	High
8 LTC MaxS	55-65 Semi Ordered	45-55 Semi Ordered	75-80 Semi Ordered	65 Semi Ordered	Low	Low
9. LOI MaxHO2 HF	45-65 Random	10 Random	30-40 Random	95 Semi Ordered	Low	High
10. Conc. MaxHO2 HF	40 Random	10 Random	30-40 Random	90 Semi Ordered	Low	Low
11. Max(LOI MaxS $Sens_{OH}$, LOI MaxHO2 CF $Sens_{HO2}$)	55 Semi Ordered	50 Semi Ordered	55 Semi Ordered Ranking Target: Average Prediction	95 Semi Ordered Ranking Target: Very Good Prediction	Average	High
12. Max(LOI MaxS $Sens_{HO2}$, LOI MaxHO2 CF $Sens_{HO2}$)	60-70 Semi Ordered	55 Semi Ordered	60-70 Semi Ordered Ranking Target: Good Prediction	85 Semi Ordered	Average	High

The results for all lists are presented below. The same figures are shown for all lists. The first figure consists of four sub figures. Sub figure a) shows the deviation for IDT HF and IDT CF vs X rank, while sub figure b) shows a zoom of sub figure a). Sub figure c) shows the deviation for MaxHO2 CF and Max OH HF vs X rank, while sub figure d) shows a zoom of sub figure c).
X can either be LT, LOI, LTC or concentration.

The second figure also consists of four sub figures. Sub figure a), b), c) and d) shows IDT HF, IDT CF, Max HO2 CF and Max OH HF vs Log X respectively. X can be either LT, LOI, LTC or concentration.

QSS species ranking list 1: LT MaxHO2 CF
Since this list is based on LT only, the ranking target of this list does not correspond to any of the ETs. Hence, the ranking performance cannot be evaluated directly.

Figure 6.67a) shows the deviation for IDT HF and IDT CF vs LT rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 45-60 QSS species and at about 35 QSS species for IDT HF and IDT CF respectively.

Figure 6.67c) shows the deviation for MaxHO2 CF and Max OH HF vs LT rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable between 65-70 QSS species and at about 90 QSS species for MaxHO2 CF and Max OH HF respectively.

Hence, a reduced mechanism of maximum 35 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of about 45-60 QSS species can be generated.

Figure 6.68a) shows the deviation for IDT HF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.68b) shows the deviation for IDT CF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa.
However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.68c) shows the deviation for MaxHO2 CF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa.
However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Random”.

Figure 6.68d) shows the deviation for MaxOH HF vs Log LT. The figure shows that the highest values of Log LT correspond to highest deviations. Otherwise it hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Random”.

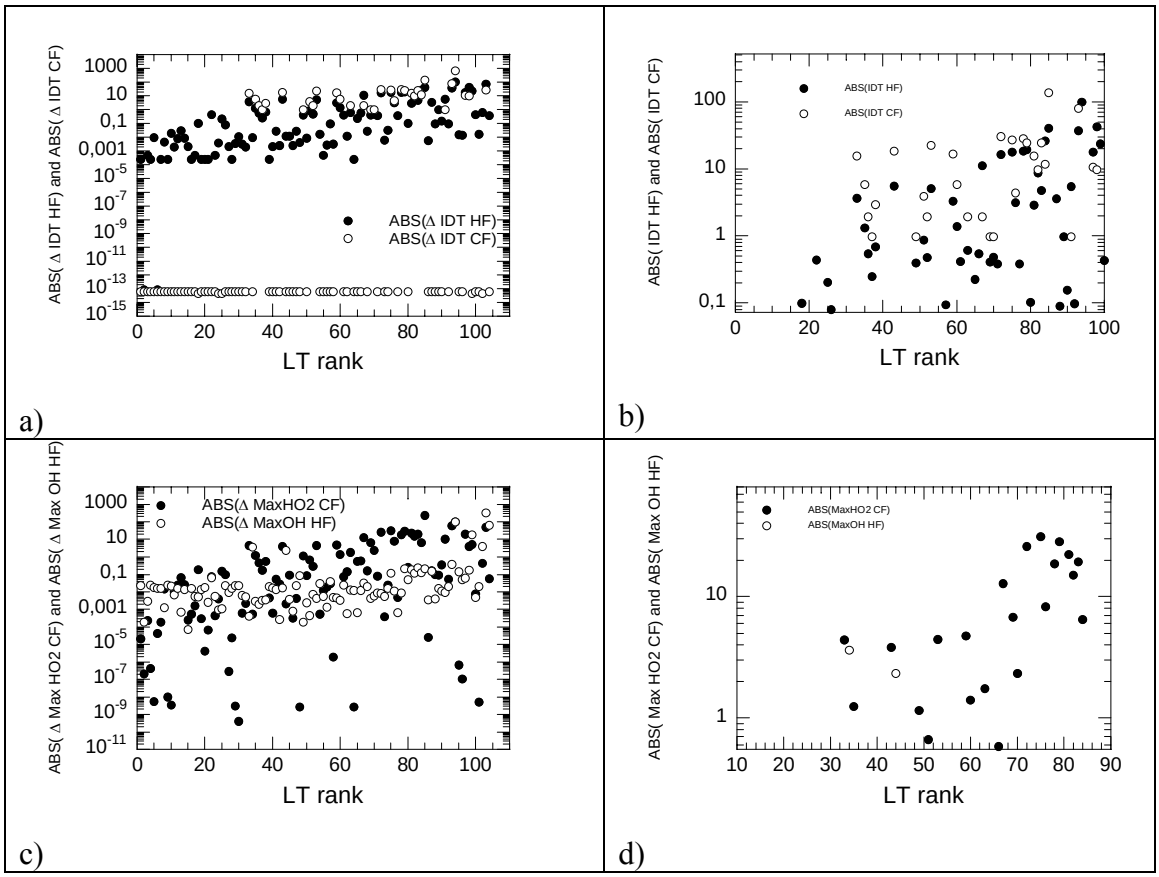


Figure 6.67.a) IDT HF and IDT CF vs LT rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LT rank
d) zoom of c).

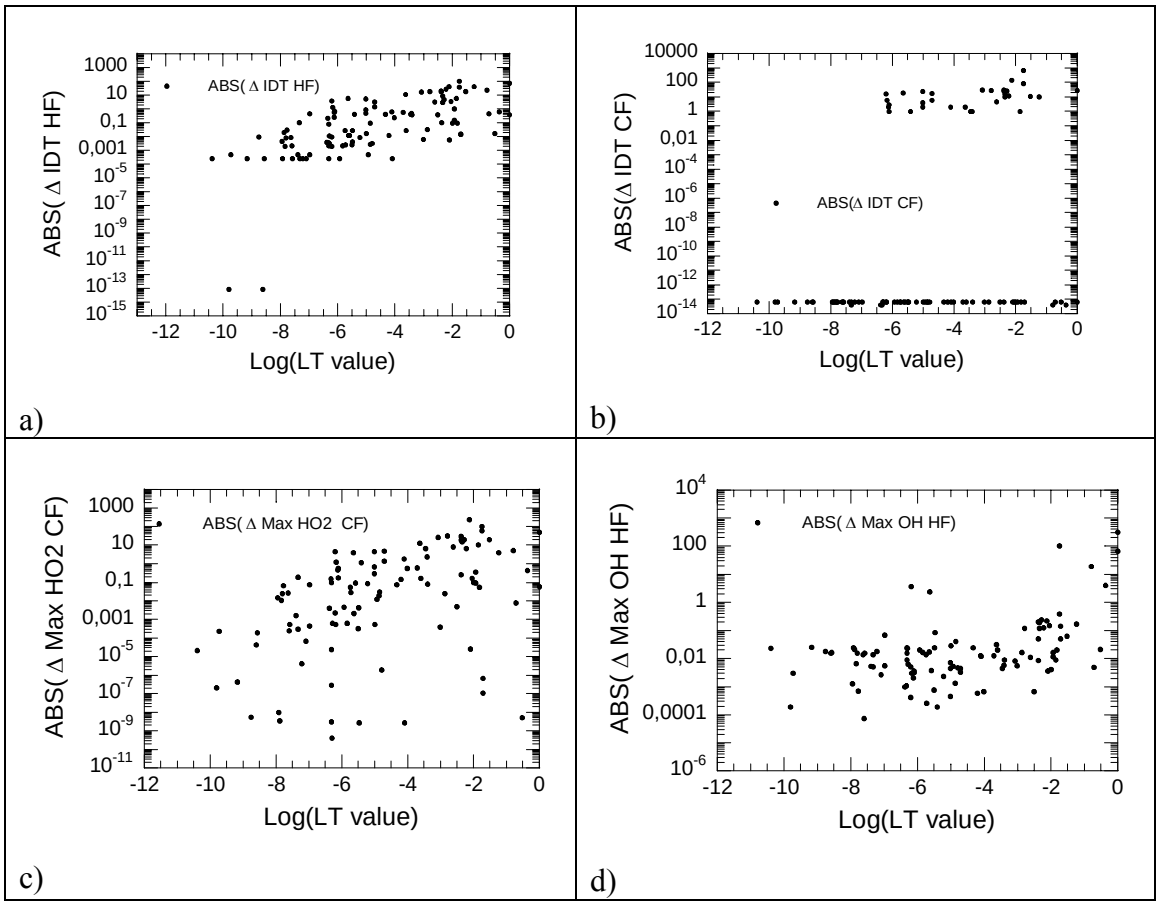


Figure 6.68. a) IDT HF vs Log LT.
b) IDT CF vs Log LT
c) MaxHO2 CF vs Log LT
d) MaxOH HF vs Log LT

QSS species ranking list 2: LOI Max HO2 CF $Sens_{HO2}$

The ranking target for this QSS species ranking list is Max HO2 CF, since the LOI is taken at the time for Max HO2 CF and the sensitivity is towards HO_2.
The ranking performance is considered as "Ordered" for the evaluation target Max HO2 CF as seen below. And since 80-85 QSS species are accepted before the accuracy of becomes unacceptable, the prediction of the number of QSS species is very good for evaluation target Max HO2 CF. The ranking does also predict the evaluation targets IDT HF, IDT CF and Max OH HF very well. The reason for this is that some species that affect the CF also affects the HF.

Figure 6.69a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 80-85 QSS species and at about 80 QSS species for IDT HF and IDT CF respectively.

Figure 6.69c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable between 80-85 QSS species and at about 30 QSS species for MaxHO2 CF and Max OH HF respectively. However, the deviation in Max OH HF at LOI rank 30, 32 and 44 are caused by CO_2, O and H respectively. These species are well known and can be eliminated from the ranking beforehand. This means that the MaxOH CF deviation becomes unacceptable at 95 QSS species.
Hence, a reduced mechanism of maximum about 80 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance the reduced mechanism cannot be reduced any further.

Figure 6.70a) shows the deviation for IDT HF vs Log LOI. The figure shows a strong trend that high values of Log LOI correspond to high deviations and vise versa. However, the pattern shows some randomness for low LOI values. Hence, the ranking performance is considered "Ordered".

Figure 6.70b) shows the deviation for IDT CF vs Log LOI. The figure shows a strong trend that high values of Log LOI correspond to high deviations and vise versa. Hence, the ranking performance is considered "Ordered".

Figure 6.70c) shows the deviation for Max HO2 CF vs Log LOI. The figure shows a strong trend that high values of Log LOI correspond to high deviations and vise versa. Hence, the ranking performance is considered "Ordered".

Figure 6.70d) shows the deviation for Max OH HF vs Log LOI. The figure shows that the highest values of Log LOI correspond to highest deviations. Otherwise it hard to notice

any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

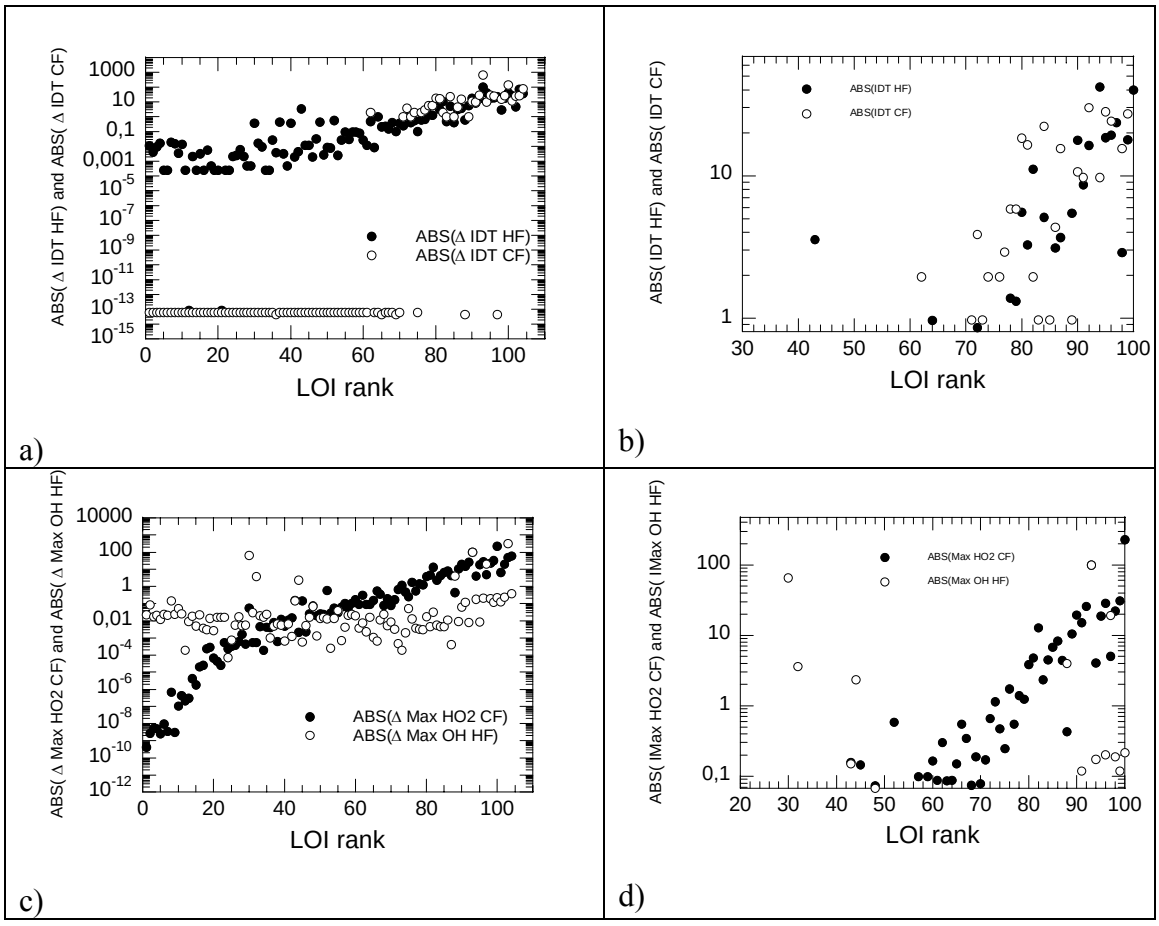


Figure 6.69.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).

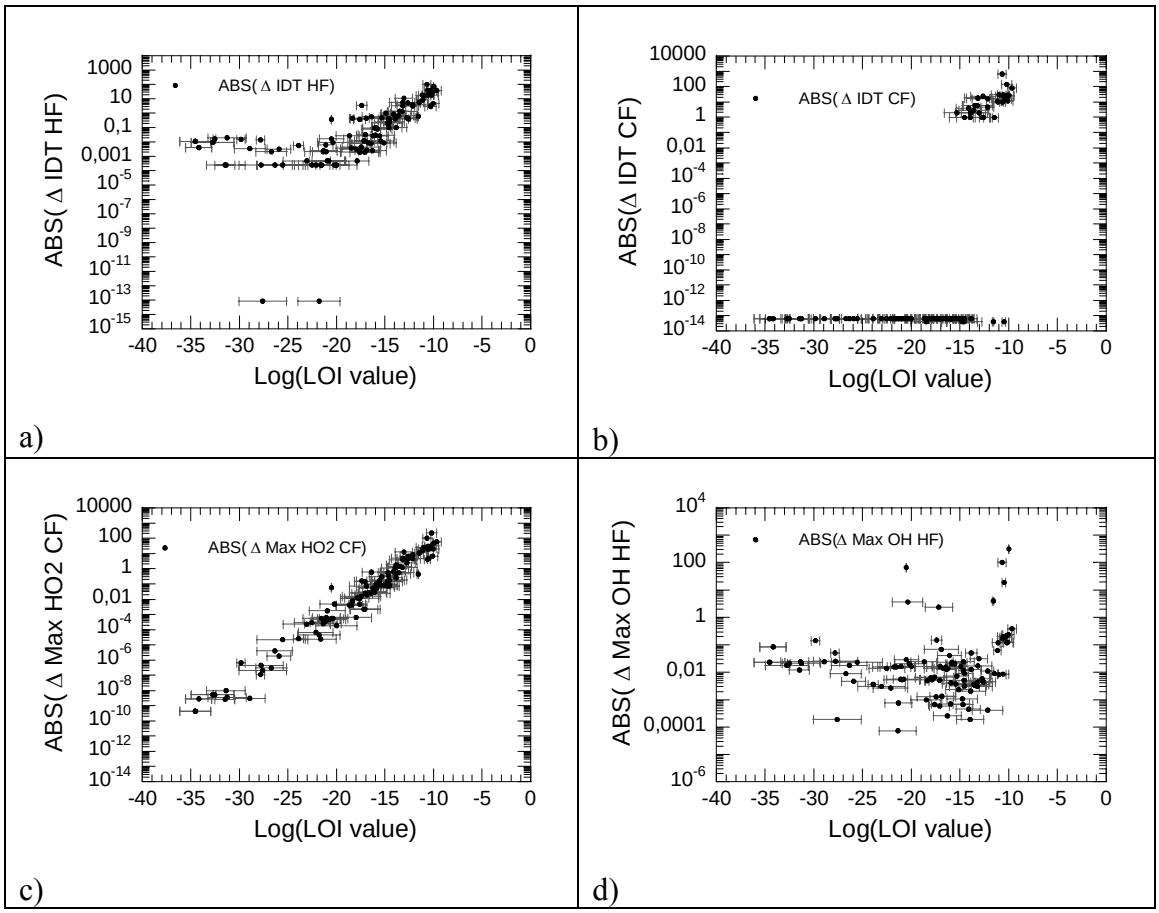


Figure 6.70. a) IDT HF vs Log LOI.
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI
The error bars correspond to the LT in all sub figures.

QSS species ranking list 3: LTC MaxHO2 CF

This QSS species ranking list is supposed to rank the Max HO2 CF well, since the LTC is taken at the time for Max HO2 CF. The ranking performance is considered as "Semi Ordered" as seen below. And since 80 QSS species are accepted before the accuracy of becomes unacceptable, the prediction of the number of QSS species is very good if Max HO2 CF is used as a measure. The ranking does also predict the IDT HF, IDT CF and max OH HF well. The reason for this is that some species that affect the CF also affects the HF.

Figure 6.71a) shows the deviation for IDT HF and IDT CF vs LTC rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 65-75 QSS species and between 60-65 QSS species for IDT HF and IDT CF respectively.

Figure 6.71c) shows the deviation for MaxHO2 CF and Max OH HF vs LTC rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at about 80 QSS species and at about 90 QSS species for MaxHO2 CF and Max OH HF respectively.
Hence, a reduced mechanism of maximum about 60-65 QSS species can be achieved with LTC MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 65-75 QSS species can be generated.

Figure 6.72a) shows the deviation for IDT HF vs Log LTC. The figure shows a strong trend that high values of Log LTC correspond to high deviations and vise versa. However, pattern shows some randomness. Hence, the ranking performance is considered "Semi Ordered".

Figure 6.72b) shows the deviation for IDT CF vs Log LTC. The figure shows a strong trend that high values of Log LTC correspond to high deviations and vise versa. However, pattern shows some randomness. Hence, the ranking performance is considered "Semi Ordered".

Figure 6.72c) shows the deviation for Max HO2 CF vs Log LTC. The figure shows a strong trend that high values of Log LTC correspond to high deviations and vise versa. However, pattern shows some randomness. Hence, the ranking performance is considered "Semi Ordered".

Figure 6.72d) shows the deviation for Max OH HF vs Log LTC. The figure shows that the highest values of Log LTC correspond to highest deviations. Otherwise it is hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered "Semi Ordered".

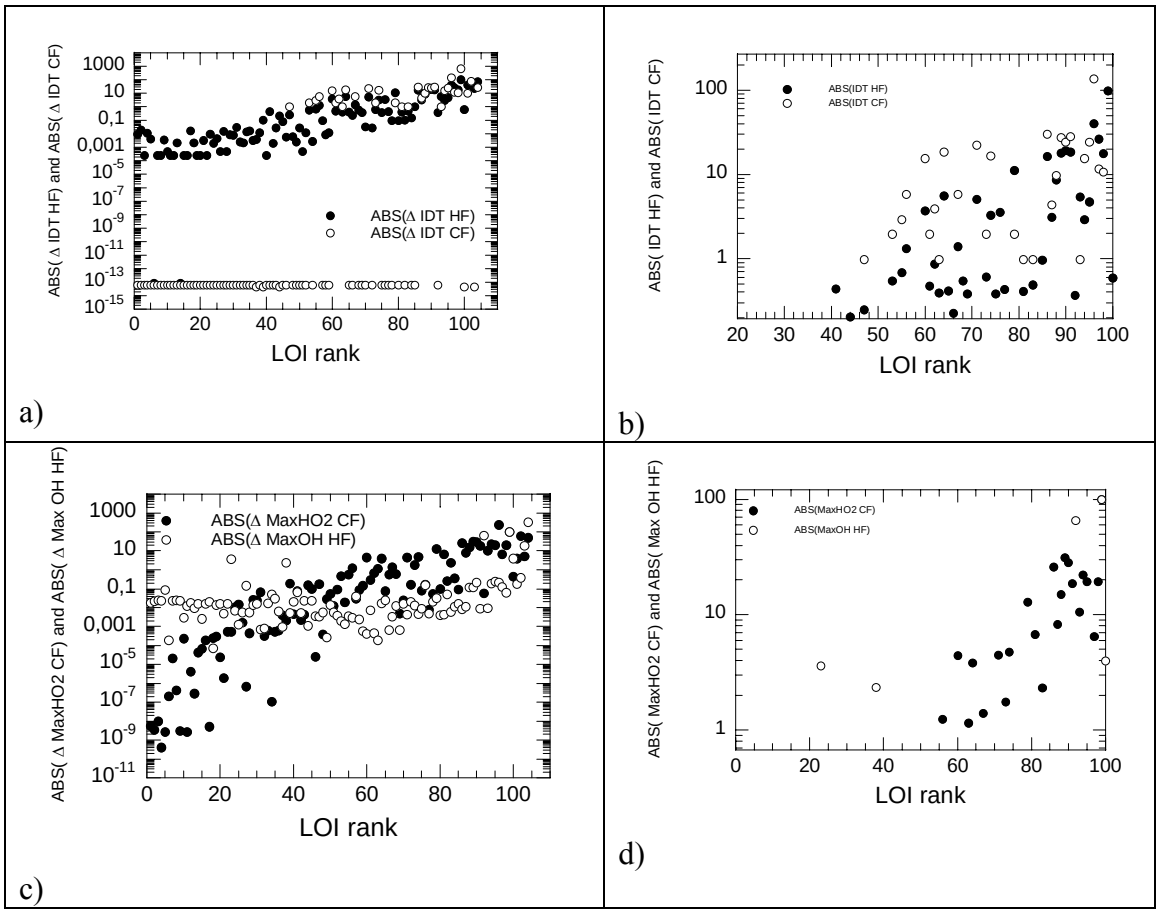


Figure 6.71.a) IDT HF and IDT CF vs LTC rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LTC rank
d) zoom of c).

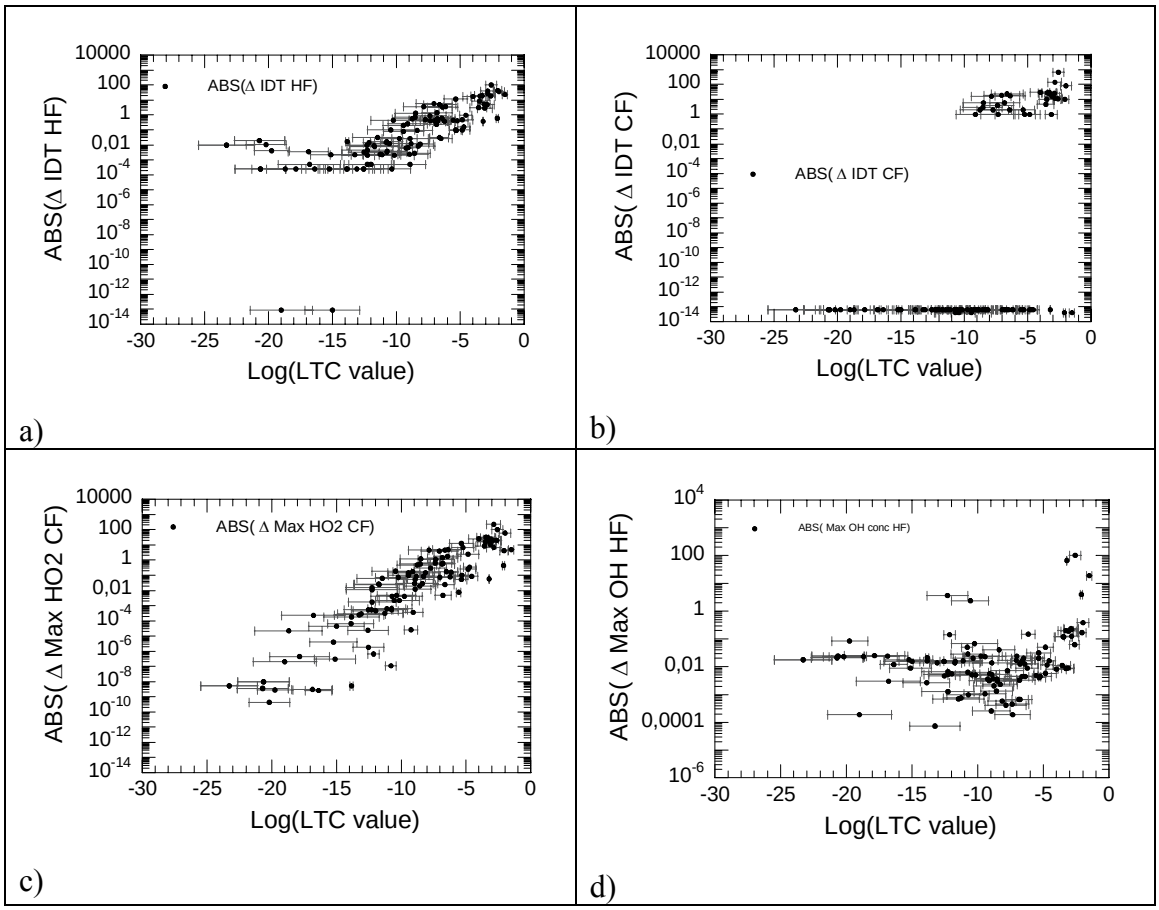


Figure 6.72. a) IDT HF vs Log LTC
b) IDT CF vs Log LTC
c) MaxHO2 CF vs Log LTC
d) MaxOH HF vs Log LTC

QSS species ranking list 4: Conc. MaxHO2 CF

This QSS species ranking list is supposed to rank the Max HO2 CF well, since the concentration is taken at the time for Max HO2 CF. The ranking performance is considered as "Ordered" as seen below. And since 80-85 QSS species are accepted before the accuracy of becomes unacceptable, the prediction of the number of QSS species is very good if Concentration Max HO2 CF is used as a measure. The ranking does also predict the IDT HF, IDT CF and Max OH HF very well. The reason for this is that some species that affect the CF also affects the HF.

Figure 6.73a) shows the deviation for IDT HF and IDT CF vs Concentration rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 80-85 QSS species and at about 85 QSS species for IDT HF and IDT CF respectively.

Figure 6.73c) shows the deviation for MaxHO2 CF and Max OH HF vs Concentration rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at between 80-85 QSS species and at about 57 QSS species for MaxHO2 CF and Max OH HF respectively. However, the deviation in Max OH HF at LOI rank 22, 31 and 57 are caused by O, H and CO_2 respectively. These species are well known are can be eliminated from the ranking beforehand. This means that the MaxOH CF deviation becomes unacceptable at 95 QSS species.
Hence, a reduced mechanism of maximum about 80-85 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance the reduced mechanism cannot be reduced any further.

Figure 6.74a) shows the deviation for IDT HF vs Log Concentration. The figure shows a strong trend that high values of Log Concentration correspond to high deviations and vise versa. Hence, the ranking performance is considered “Ordered”.

Figure 6.74b) shows the deviation for IDT CF vs Log Concentration. The figure shows a strong trend that high values of Log Concentration correspond to high deviations and vise versa. Hence, the ranking performance is considered “Ordered”.

Figure 6.74c) shows the deviation for Max HO2 CF vs Log Concentration. The figure shows a strong trend that high values of Log Concentration correspond to high deviations and vise versa. Hence, the ranking performance is considered “Ordered”.

Figure 6.74d) shows the deviation for Max OH HF vs Log Concentration. The figure shows that the highest values of Log Concentration correspond to highest deviations. Otherwise it hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

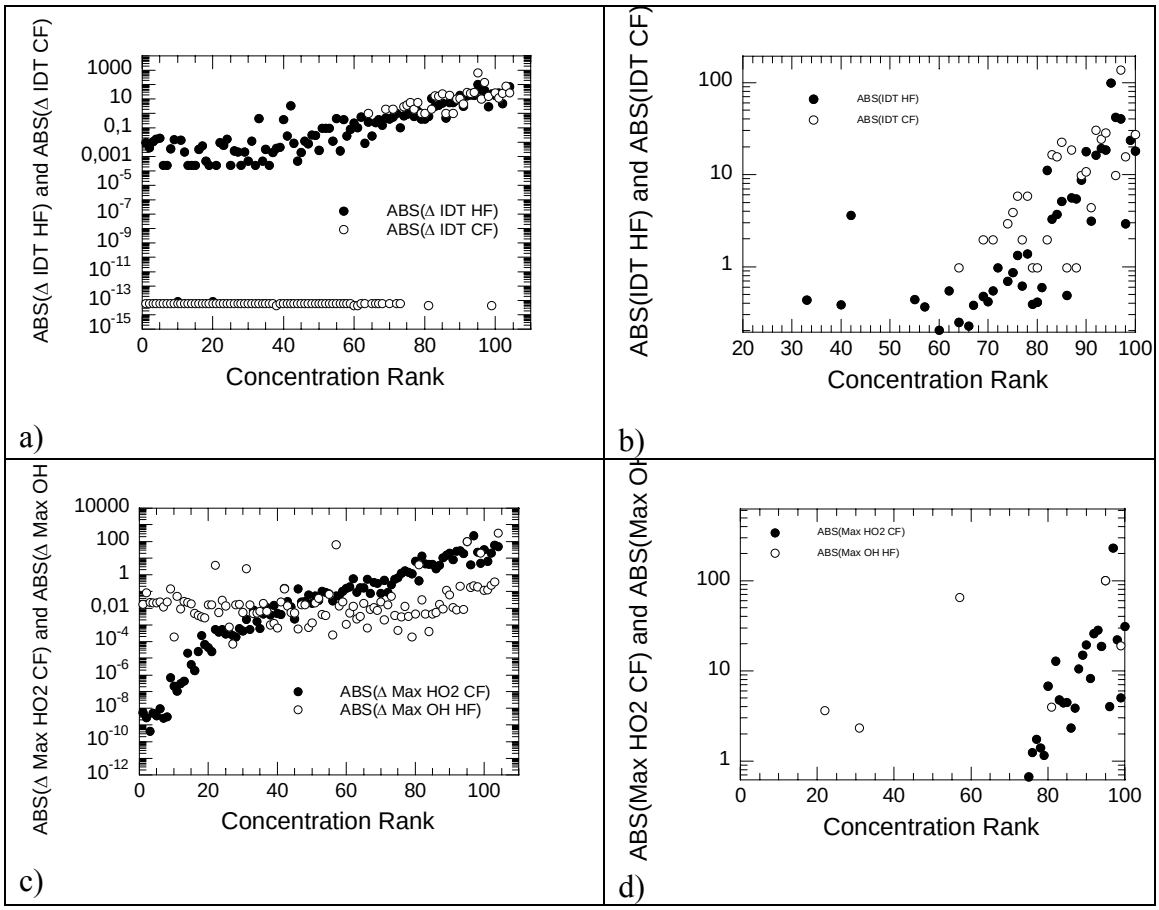


Figure 6.73.a) IDT HF and IDT CF vs Concentration rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs Concentration rank
d) zoom of c).

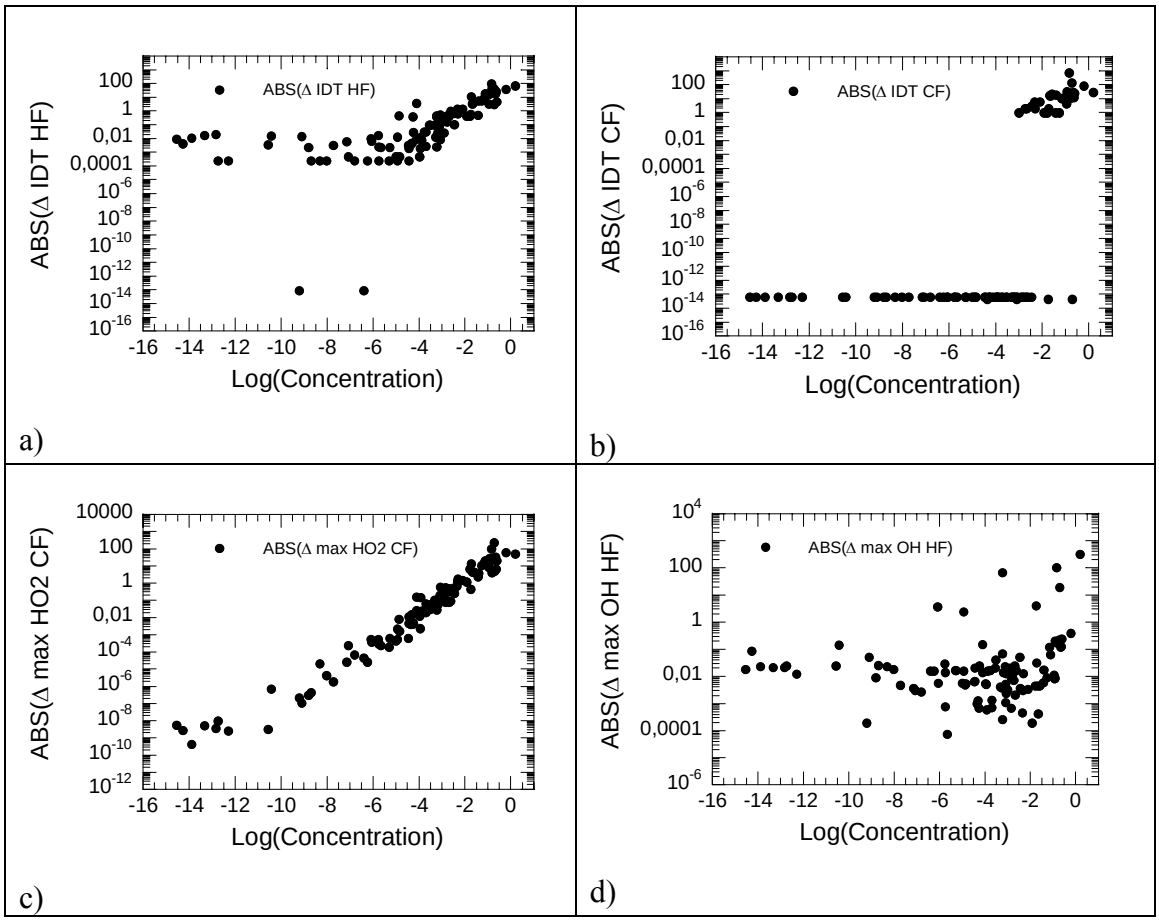


Figure 6.74. a) IDT HF vs Log Concentration
b) IDT CF vs Log Concentration
c) MaxHO2 CF vs Log Concentration
d) MaxOH HF vs Log Concentration

QSS species ranking list 5: LT MaxS

Since this list is based on LT only, the ranking target of this list does not correspond to any of the evaluation targets. Hence, the ranking performance cannot be evaluated directly.

Figure 6.75a) shows the deviation for IDT HF and IDT CF vs LT rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable at about 35 QSS species and at about 35 QSS species for IDT HF and IDT CF respectively.
Figure 6.75c) shows the deviation for MaxHO2 CF and Max OH HF vs LT rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at between 65-70 QSS species and at about 35 QSS species for MaxHO2 CF and Max OH HF respectively. Hence, a reduced mechanism of maximum about 35 QSS species can be achieved with LT MaxHO2 CF if

both HF and CF are important for the user. However, if only the HF is of importance the reduced mechanism cannot be reduced any further.

Figure 6.76a) shows the deviation for IDT HF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.76b) shows the deviation for IDT CF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.76c) shows the deviation for Max HO2 CF vs Log LT. The figure shows a trend that high values of Log LT correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.76d) shows the deviation for Max OH HF vs Log LT. The figure shows that average high values of Log LT correspond to highest deviations. Otherwise it is hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Random”.

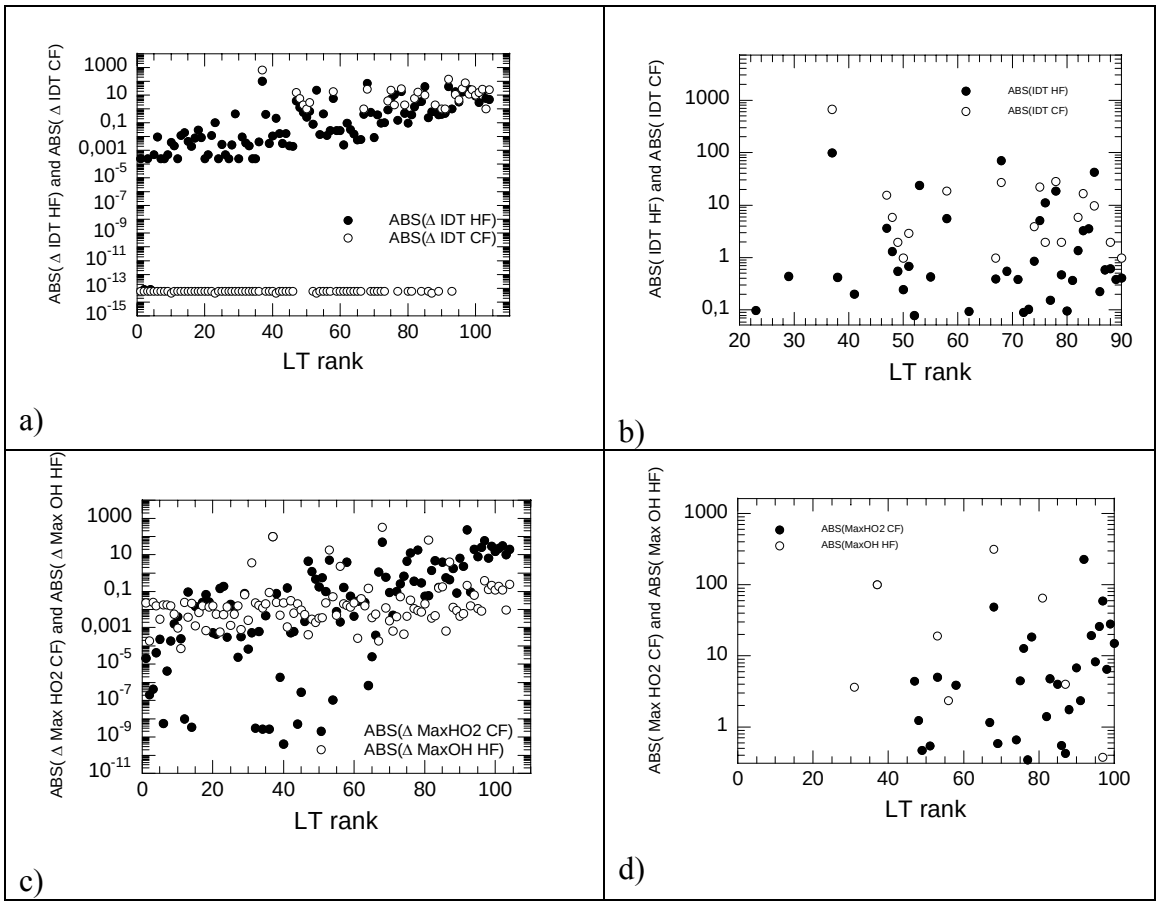


Figure 6.75.a) IDT HF and IDT CF vs LT rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LT rank
d) zoom of c).

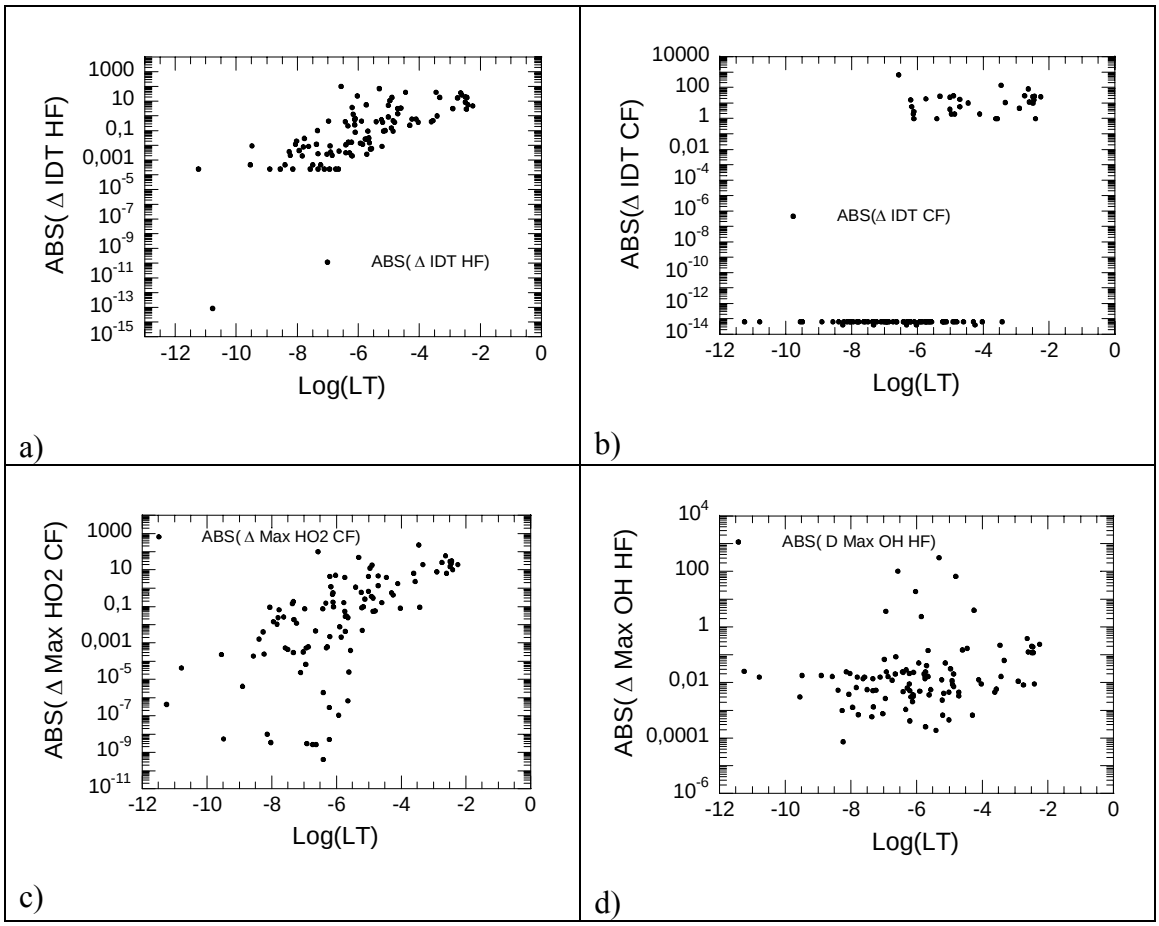


Figure 6.76. a) IDT HF vs Log LT
b) IDT CF vs Log LT
c) MaxHO2 CF vs Log LT
d) MaxOH HF vs Log LT

QSS species ranking list 6: LOI MaxS Sens$_{OH}$

Since this list is based on LT at MaxS and the sensitivity is taken towards OH, the ranking target of this list is Max OH HF. This means that the ranking target can be evaluated by an evaluated target.
The ranking performance is considered as “Semi Ordered” as seen below. And since 95 QSS species are accepted before the accuracy of becomes unacceptable, the prediction of the number of QSS species is very good if Max OH HF is used as a measure. The ranking does not predict the IDT HF, IDT CF and Max HO2 CF very well. The reason for this is that some species that affect the CF also affects the HF and that the CF is not considered for many species when MaxS in used, since most species have their maximum concentration at a later point in time than the CF.

Figure 6.77a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 40-50 QSS species and between 30-40 QSS species for IDT HF and IDT CF respectively.

Figure 6.77c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at between 40-50 QSS species and at about 95 QSS species for MaxHO2 CF and Max OH HF respectively. Hence, a reduced mechanism of maximum about 30-40 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 40-50 QSS species can be generated.

Figure 6.78a) shows the deviation for IDT HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.78b) shows the deviation for IDT CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.78c) shows the deviation for MaxHO2 CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.78d) shows the deviation for MaxOH HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

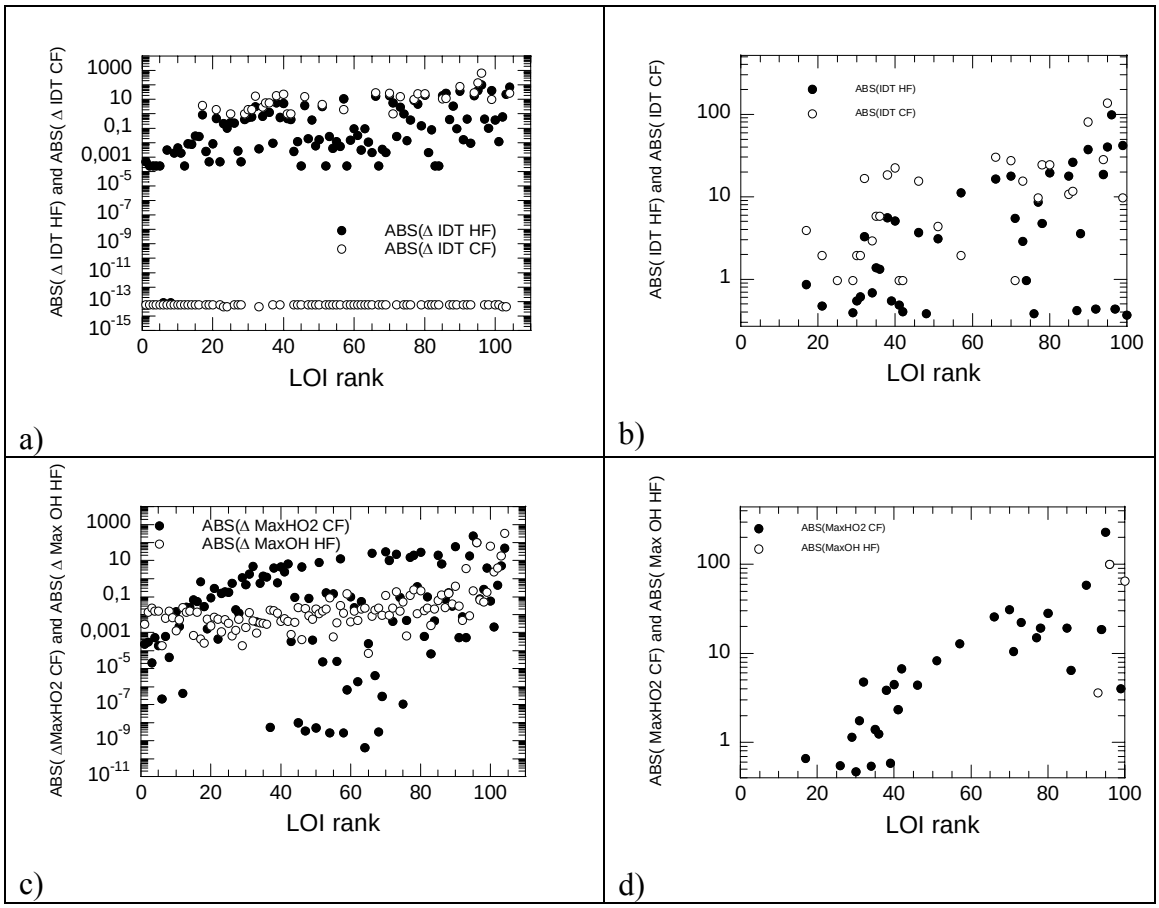


Figure 6.77.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).
The sensitivity was taken towards OH.

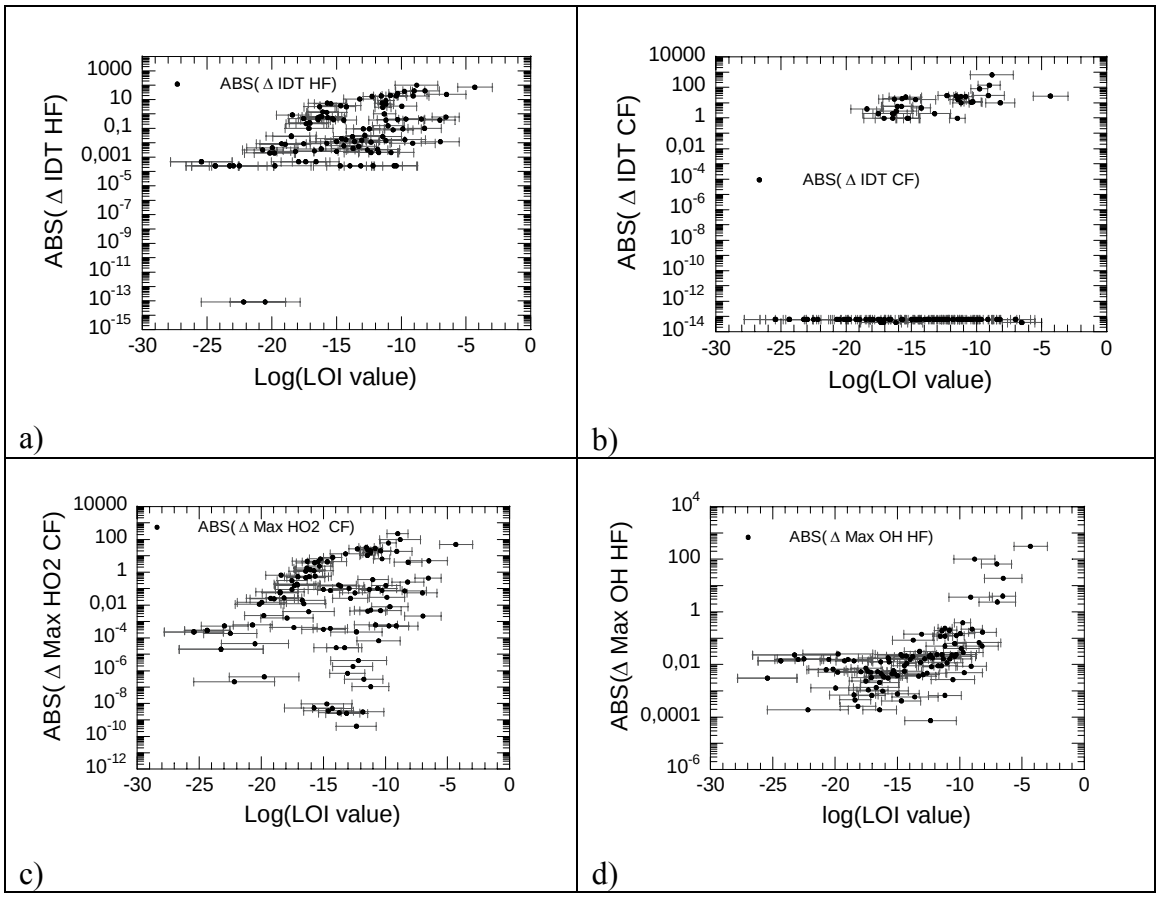


Figure 6.78. a) IDT HF vs Log LOI
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI
The sensitivity was taken towards OH.

QSS species ranking list 7: LOI MaxS Sens$_{HO2}$

Since this list is based on LT at MaxS and the sensitivity is taken towards HO_2, the ranking target of this list does not correspond to any of the evaluation targets. Hence, the ranking performance cannot be evaluated directly.

Figure 6.79a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 50-55 QSS species and between 55-60 QSS species for IDT HF and IDT CF respectively.

Figure 6.79c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).

The accumulated deviation becomes unacceptable at between 65-70 QSS species and at about 85 QSS species for MaxHO2 CF and Max OH HF respectively. Hence, a reduced mechanism of maximum about 50-55 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance the reduced mechanism cannot be reduced any further.

Figure 6.80a) shows the deviation for IDT HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.80b) shows the deviation for IDT CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.80c) shows the deviation for MaxHO2 CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.80d) shows the deviation for MaxOH HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

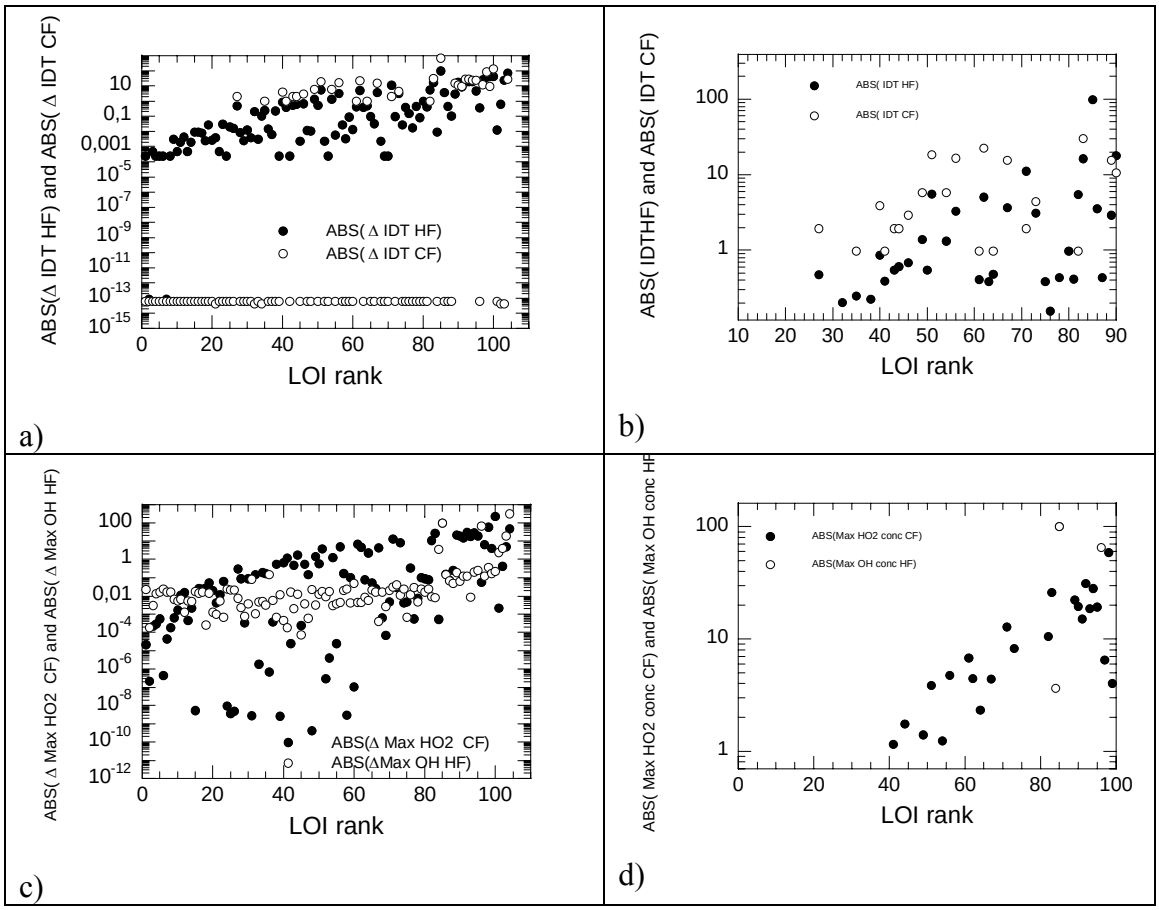


Figure 6.79.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).
The sensitivity was taken towards HO_2.

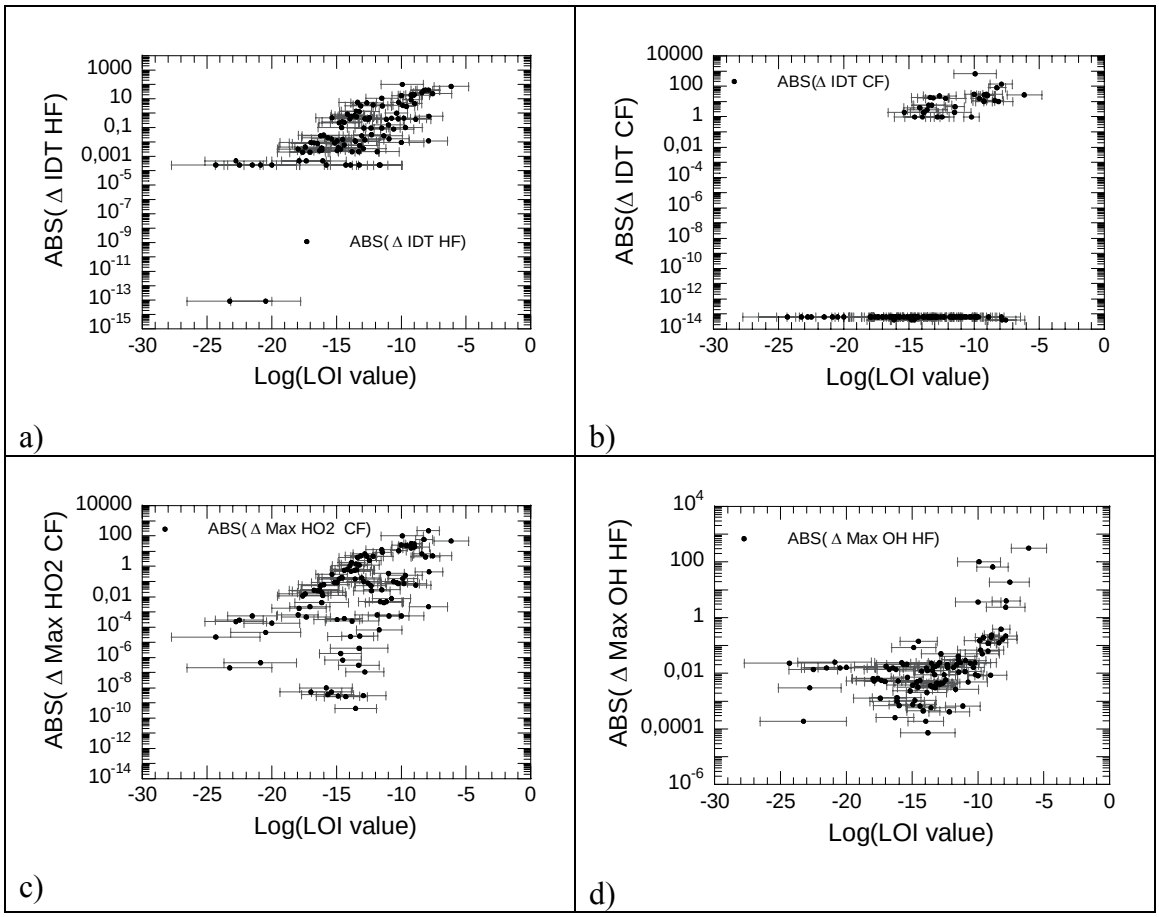


Figure 6.80. a) IDT HF vs Log LOI
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI
The sensitivity was taken towards HO_2.

QSS species ranking list 8: LTC MaxS

Since this list is based on LT at MaxS and concentration, the ranking target of this list does not correspond to any of the evaluation targets. Hence, the ranking performance cannot be evaluated directly.

Figure 6.81a) shows the deviation for IDT HF and IDT CF vs LTC rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 55-65 QSS species and between 45-55 QSS species for IDT HF and IDT CF respectively.

Figure 6.81c) shows the deviation for MaxHO2 CF and Max OH HF vs LTC rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at about 75-80 QSS species and at about 65 QSS species for MaxHO2 CF and Max OH HF respectively.
Hence, a reduced mechanism of maximum about 45-55 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 55-65 QSS species can be generated.

Figure 6.82a) shows the deviation for IDT HF vs Log LTC. The figure shows a strong trend that high values of Log LTC correspond to high deviations and vise versa. However, the pattern shows some randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.82b) shows the deviation for IDT CF vs Log LTC. The figure shows a trend that high values of Log LTC correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.82c) shows the deviation for Max HO2 CF vs Log LTC. The figure shows a trend that high values of Log LTC correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.82d) shows the deviation for Max OH HF vs Log LTC. The figure shows that the highest values of Log LTC correspond to highest deviations. Otherwise it hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

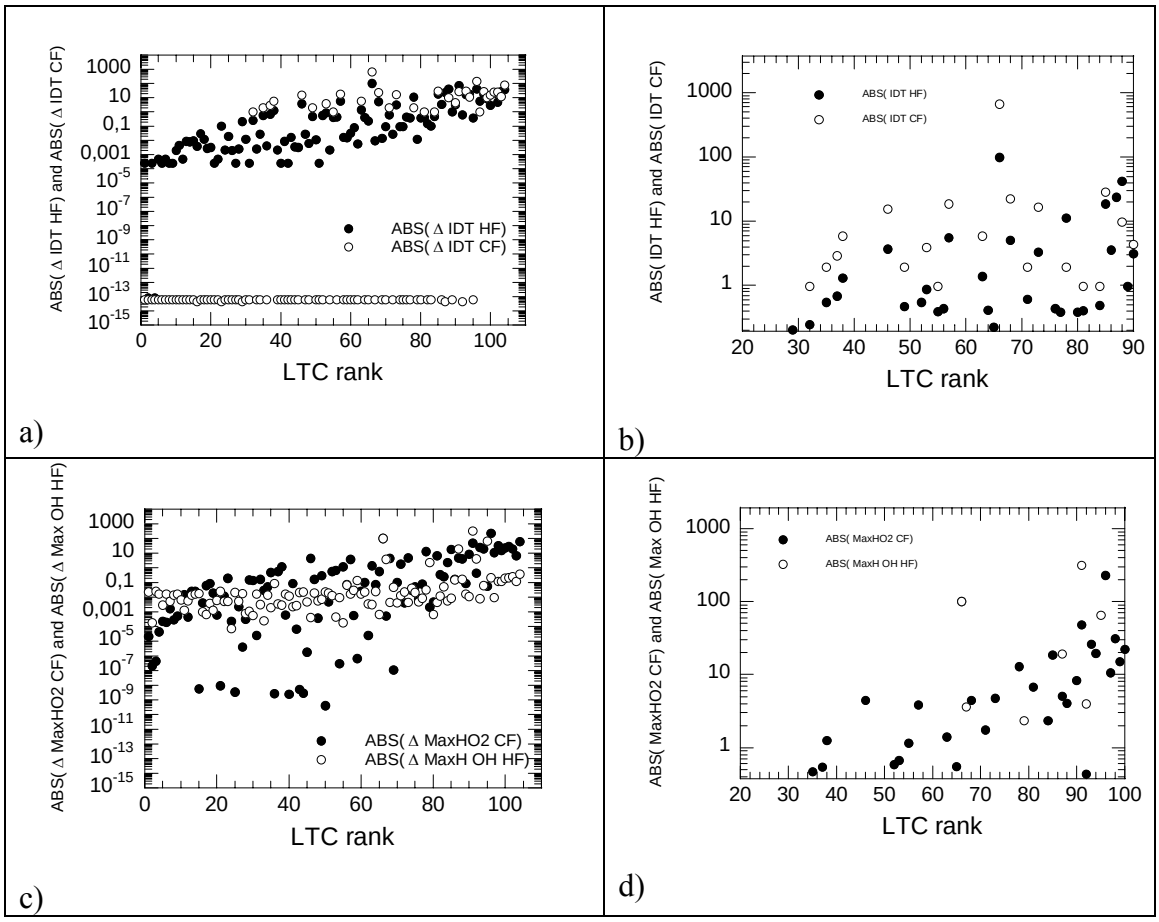


Figure 6.81.a) IDT HF and IDT CF vs LTC rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LTC rank
d) zoom of c).

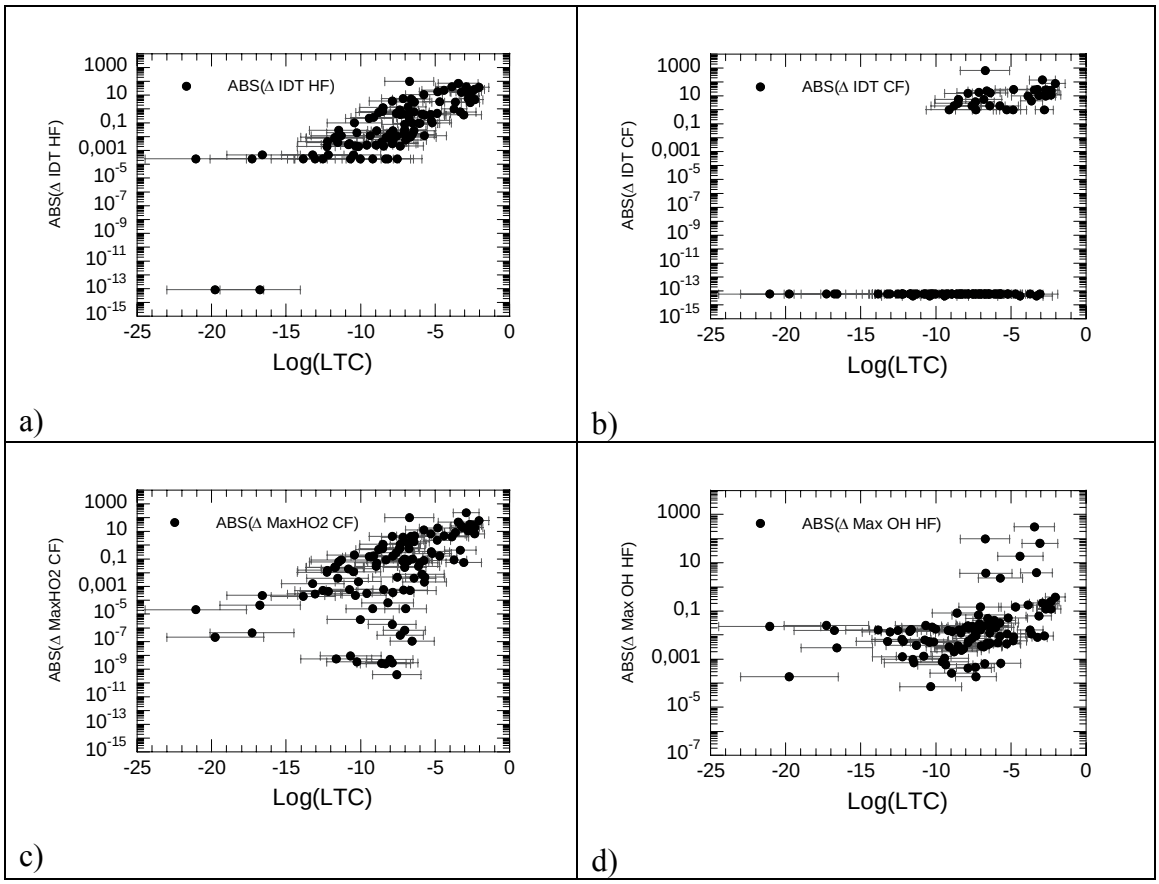


Figure 6.82. a) IDT HF vs Log LTC
b) IDT CF vs Log LTC
c) MaxHO2 CF vs Log LTC
d) MaxOH HF vs Log LTC

QSS species ranking list 9: LOI Max HO2 HF

Since this list is based on LT at Max HO2 HF and the sensitivity is taken towards HO_2, the ranking target of this list is Max HO2 HF. However, this ranking target does not correspond to any of the evaluation targets. Hence, the ranking performance cannot be evaluated directly.

Figure 6.83a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 45-65 QSS species and at about 10 QSS species for IDT HF and IDT CF respectively.

Figure 6.83c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at about 30-40 QSS species and at about 95 QSS species for MaxHO2 CF and Max OH HF respectively.
Hence, a reduced mechanism of maximum about 10 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 45-65 QSS species can be generated.

Figure 6.84a) shows the deviation for IDT HF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.

Figure 6.84b) shows the deviation for IDT CF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.
Figure 6.84c) shows the deviation for MaxHO2 CF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.

Figure 6.84d) shows the deviation for Max OH HF vs Log LTC. The figure shows that the highest values of Log LTC correspond to highest deviations. Otherwise it hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

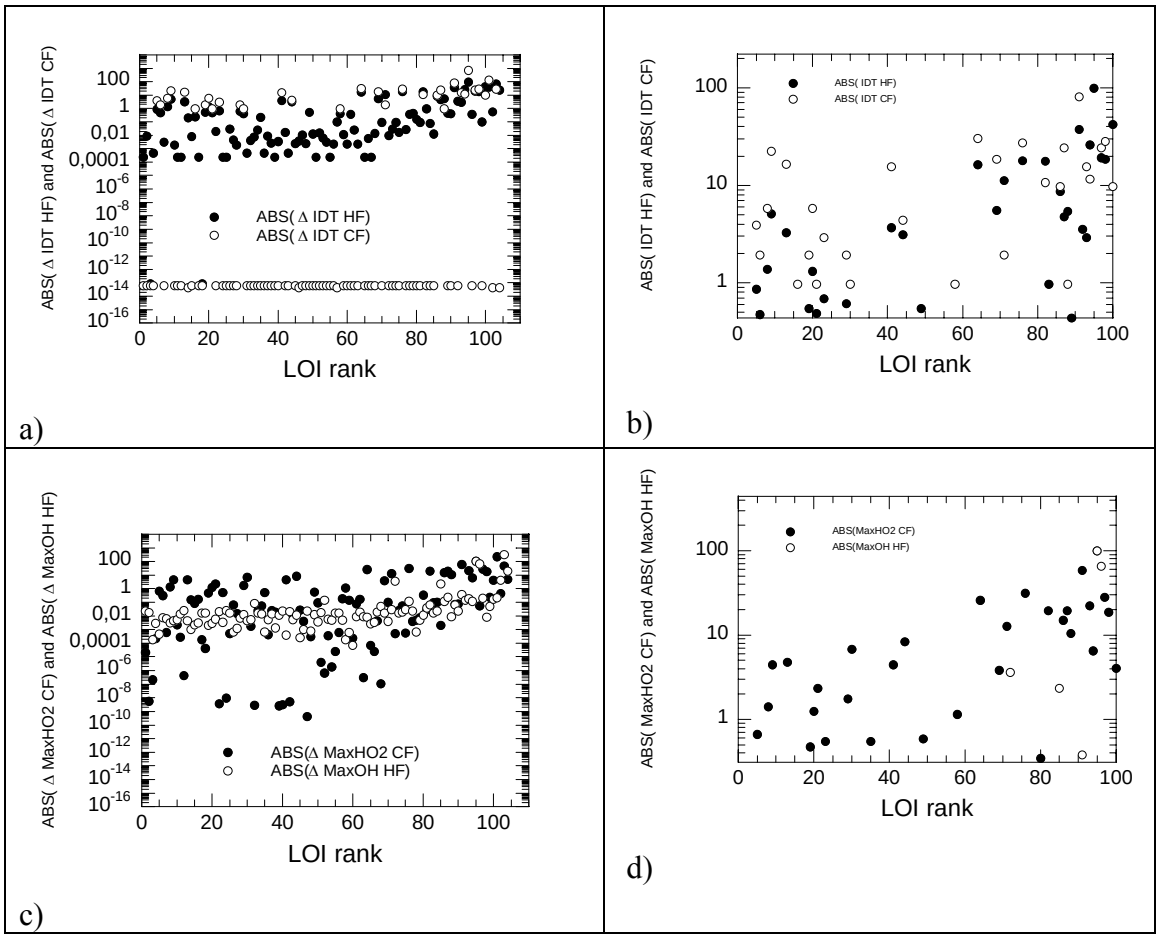


Figure 6.83.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).

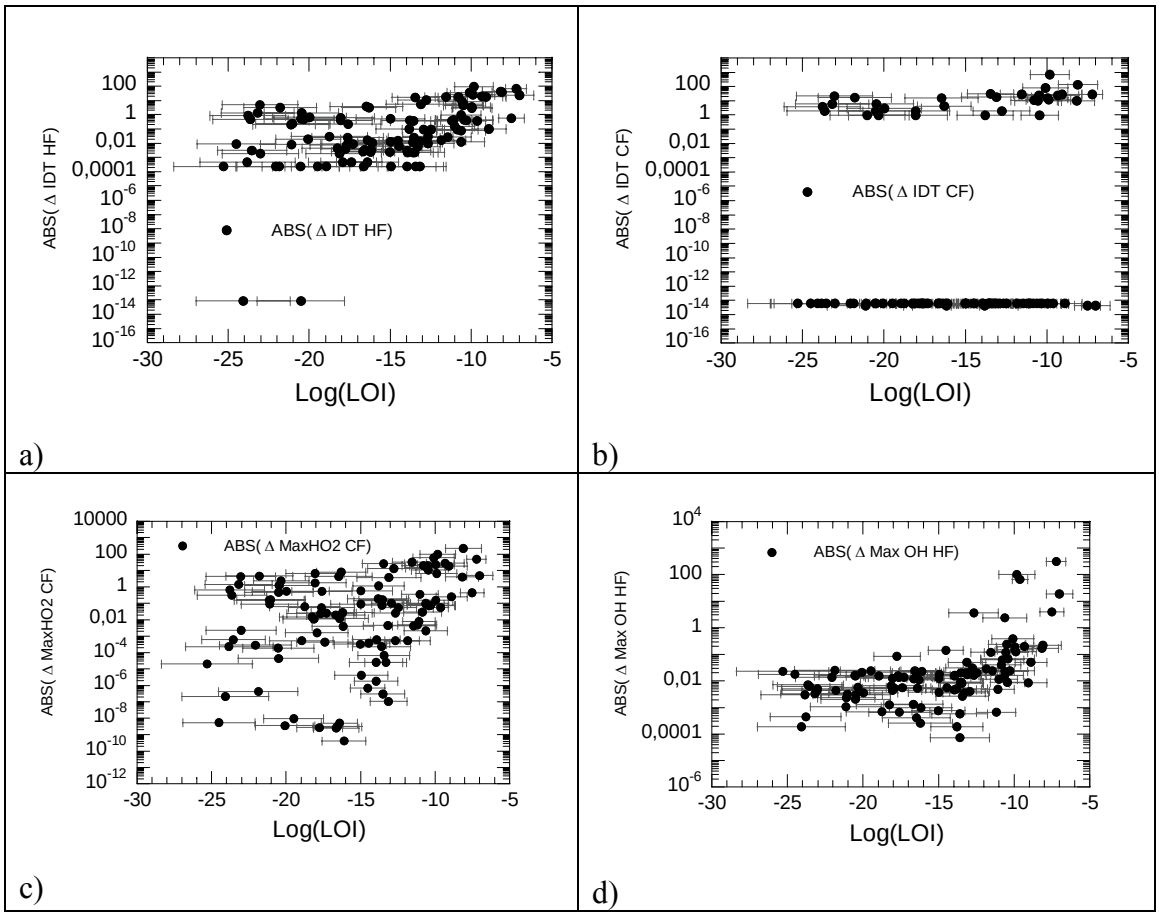


Figure 6.84. a) IDT HF vs Log LOI
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI

QSS species ranking list 10: Concentration Max HO2 HF

Since this list is based on concentration at Max HO2 HF the ranking target of this list does not correspond to any of the evaluation targets. Hence, the ranking performance cannot be evaluated directly.

Figure 6.85a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable at about 40 QSS species and at about 10 QSS species for IDT HF and IDT CF respectively.

Figure 6.85c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at about 30-40 QSS species and at about 90 QSS species for MaxHO2 CF and Max OH HF respectively.

Hence, a reduced mechanism of maximum about 10 QSS species can be achieved with LT MaxHO2 CF if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 45-65 QSS species can be generated.

Figure 6.86a) shows the deviation for IDT HF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.

Figure 6.86b) shows the deviation for IDT CF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.

Figure 6.86c) shows the deviation for MaxHO2 CF vs Log LTC. The figure shows a very random pattern. Hence, the ranking performance is considered “Random”.

Figure 6.86d) shows the deviation for MaxOH HF vs Log LTC. The figure shows that the highest values of Log LTC correspond to highest deviations. Otherwise it hard to notice any trend and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

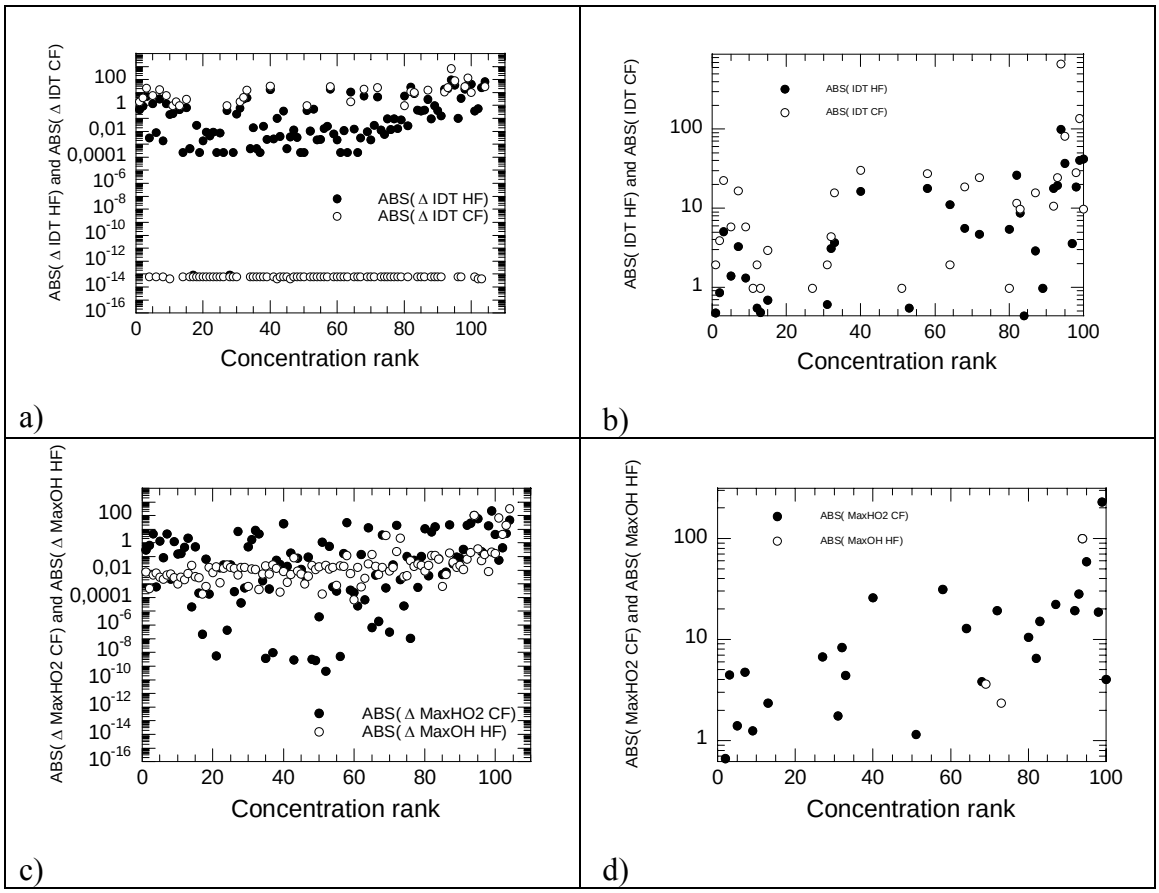


Figure 6.85.a) IDT HF and IDT CF vs Concentration rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs Concentration rank
d) zoom of c).

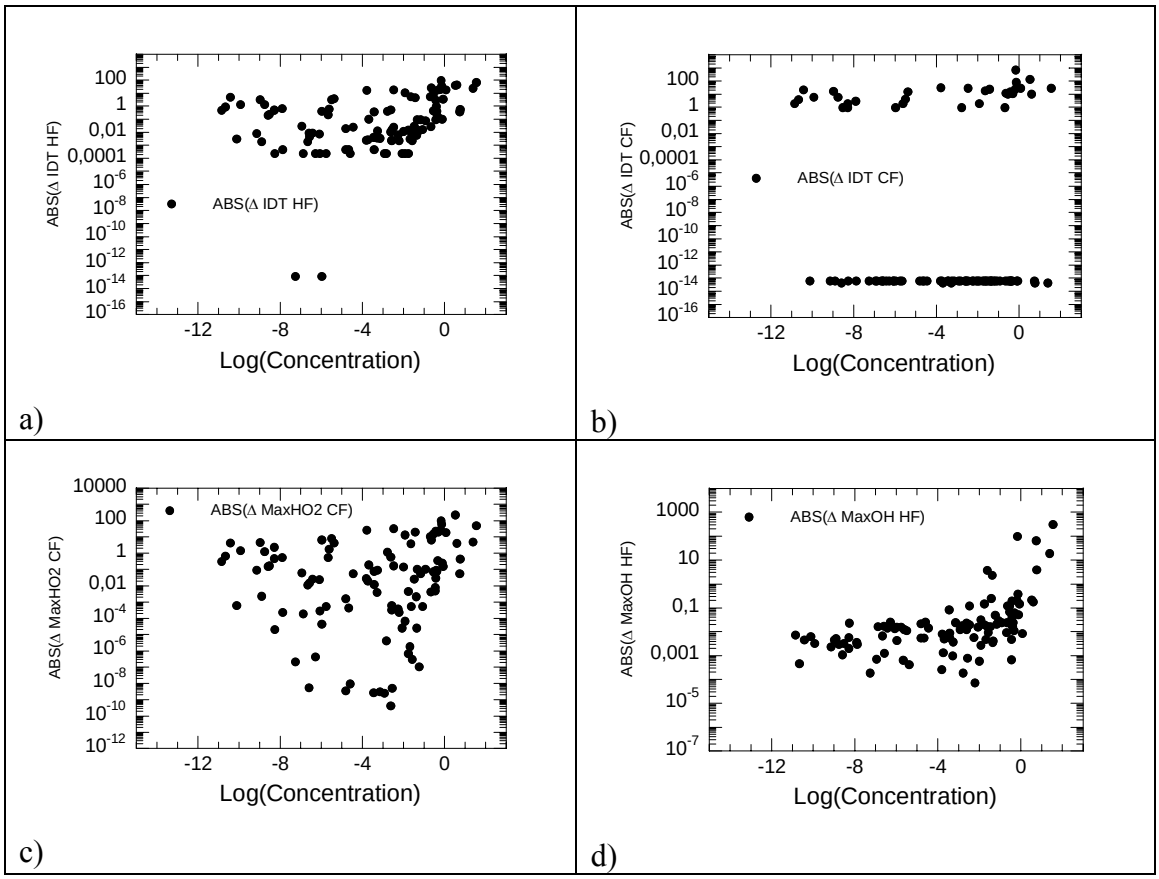


Figure 6.86. a) IDT HF vs Log Concentration
b) IDT CF vs Log Concentration
c) MaxHO2 CF vs Log Concentration
d) MaxOH HF vs Log Concentration

QSS species ranking list 11: LOI Max (MaxS Sens$_{OH}$, Max HO2 CF Sens$_{HO2}$)

The first list has Max OH HF as ranking target and the other list has Max HO2 CF as ranking target, which means that both ranking targets can be evaluated by evaluation targets.
The ranking performance is considered as "Semi Ordered" for both Max OH HF and Max HO2 CF as seen below. And since 95 QSS species are accepted before the accuracy becomes unacceptable, the prediction of the number of QSS species is very good if Max OH HF is used as a measure. However, since 55 QSS species are accepted before the accuracy becomes unacceptable, the prediction of the number of QSS species is average if Max HO2 CF is used as a measure. The ranking does not predict the IDT HF and IDT CF very well. The reason for this is that some species that the MaxS Sens$_{OH}$ list generally has higher values than the Max HO2 CF SensHO2 list. Hence, the values from the MaxS Sens$_{OH}$ are chosen for most species.

Figure 6.87a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable at about 55 QSS species and at about 50 QSS species for IDT HF and IDT CF respectively.

Figure 6.87c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable at about 55 QSS species and at about 95 QSS species for Max HO2 CF and Max OH HF respectively.
Hence, a reduced mechanism of maximum about 50 QSS species can be achieved if both HF and CF are important for the user. However, if only the HF is of importance further reduction can not be generated.

Figure 6.88a) shows the deviation for IDT HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.88b) shows the deviation for IDT CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.88c) shows the deviation for Max HO2 CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.88d) shows the deviation for Max OH HF vs Log LOI. The figure shows that the highest values of Log LOI correspond to highest deviations. Hence, the ranking performance is considered “Semi Ordered”.

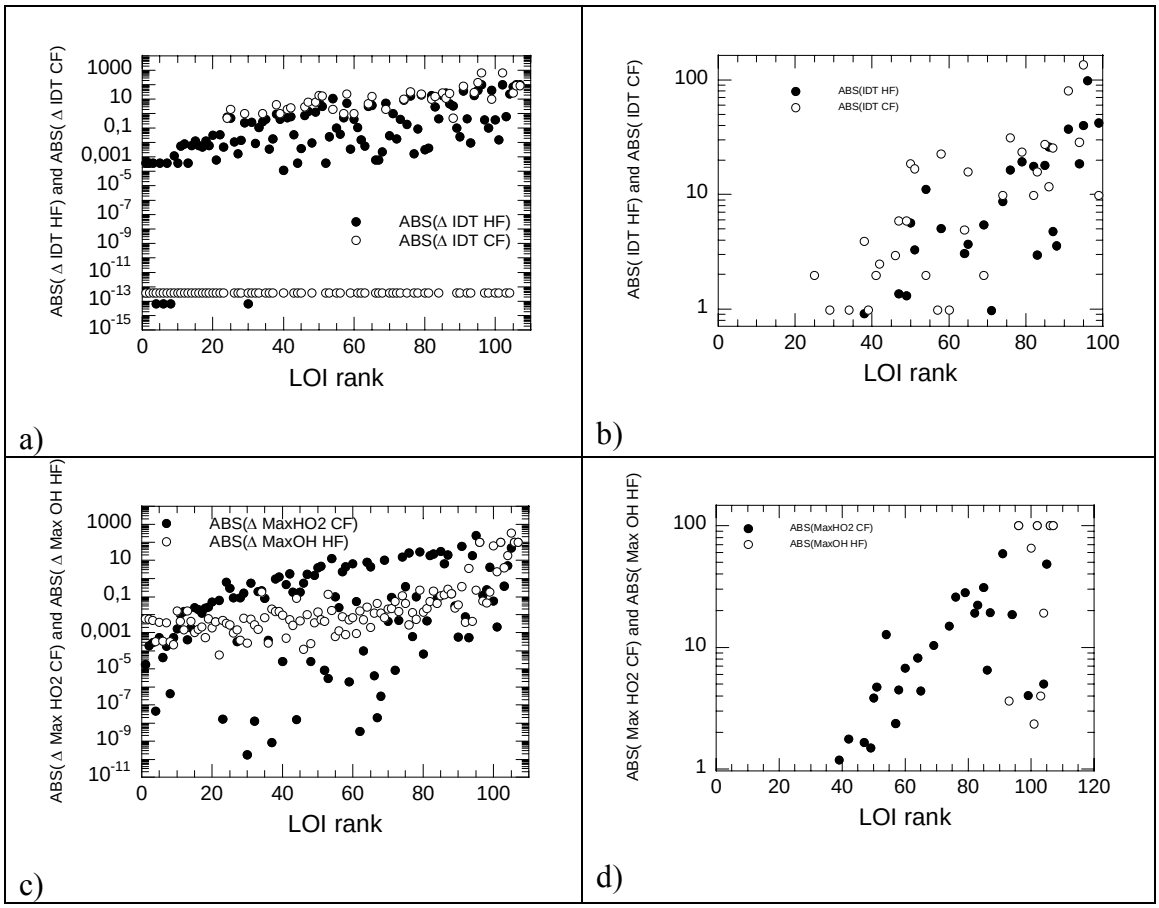


Figure 6.87.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).

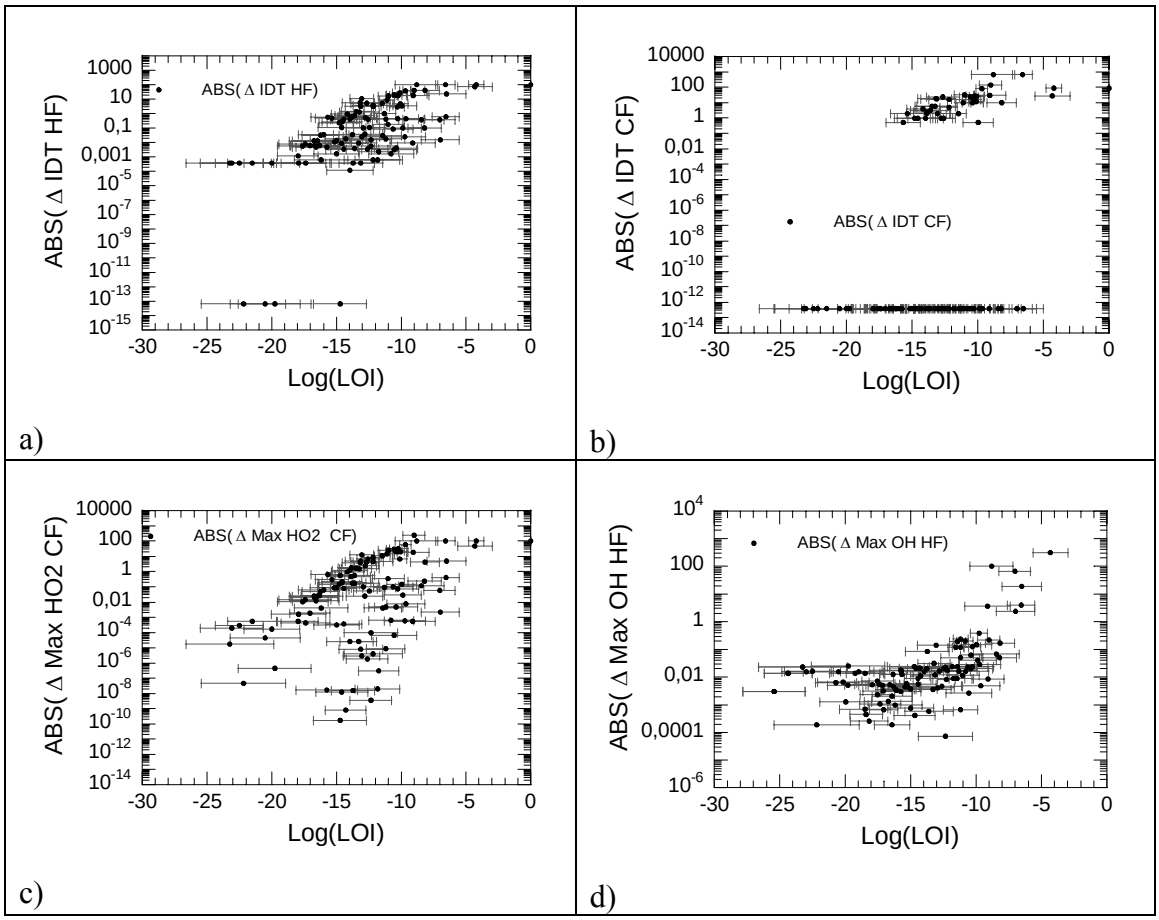


Figure 6.88. a) IDT HF vs Log LOI
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI

QSS species ranking list 12: LOI Max (MaxS $Sens_{HO2}$, Max HO2 CF $Sens_{HO2}$)

The second list has Max HO2 as a ranking target, which means that it can be evaluated by an evaluated target. The ranking performance is considered as "Semi Ordered" for Max HO2 CF as seen below. However, since 60-70 QSS species are accepted before the accuracy becomes unacceptable, the prediction of the number of QSS species is good if Max HO2 CF is used as a measure. The ranking does not predict the IDT HF and IDT CF very well. The reason for this is that some species that the MaxS $Sens_{HO2}$ list generally has higher values than the Max HO2 CF SensHO2 list. Hence, the values from the MaxS $Sens_{HO2}$ are chosen for most species.

Figure 6.89a) shows the deviation for IDT HF and IDT CF vs LOI rank, while figure b) shows a zoom of a). The accumulated deviation becomes unacceptable between 60-70 QSS species and at about 55 QSS species for IDT HF and IDT CF respectively.

Figure 6.89c) shows the deviation for MaxHO2 CF and Max OH HF vs LOI rank, while figure d) shows a zoom of c).
The accumulated deviation becomes unacceptable between 60-70 QSS species and at about 85 QSS species for Max HO2 CF and Max OH HF respectively.
Hence, a reduced mechanism of maximum about 55 QSS species can be achieved if both HF and CF are important for the user. However, if only the HF is of importance a reduced mechanism of 60-70 QSS species can be generated.

Figure 6.90a) shows the deviation for IDT HF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.90b) shows the deviation for IDT CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.90c) shows the deviation for Max HO2 CF vs Log LOI. The figure shows a trend that high values of Log LOI correspond to high deviations and vise versa. However, the trend is not very strong and the pattern shows randomness. Hence, the ranking performance is considered “Semi Ordered”.

Figure 6.90d) shows the deviation for Max OH HF vs Log LOI. The figure shows that the highest values of Log LOI correspond to highest deviations. Hence, the ranking performance is considered “Semi Ordered”.

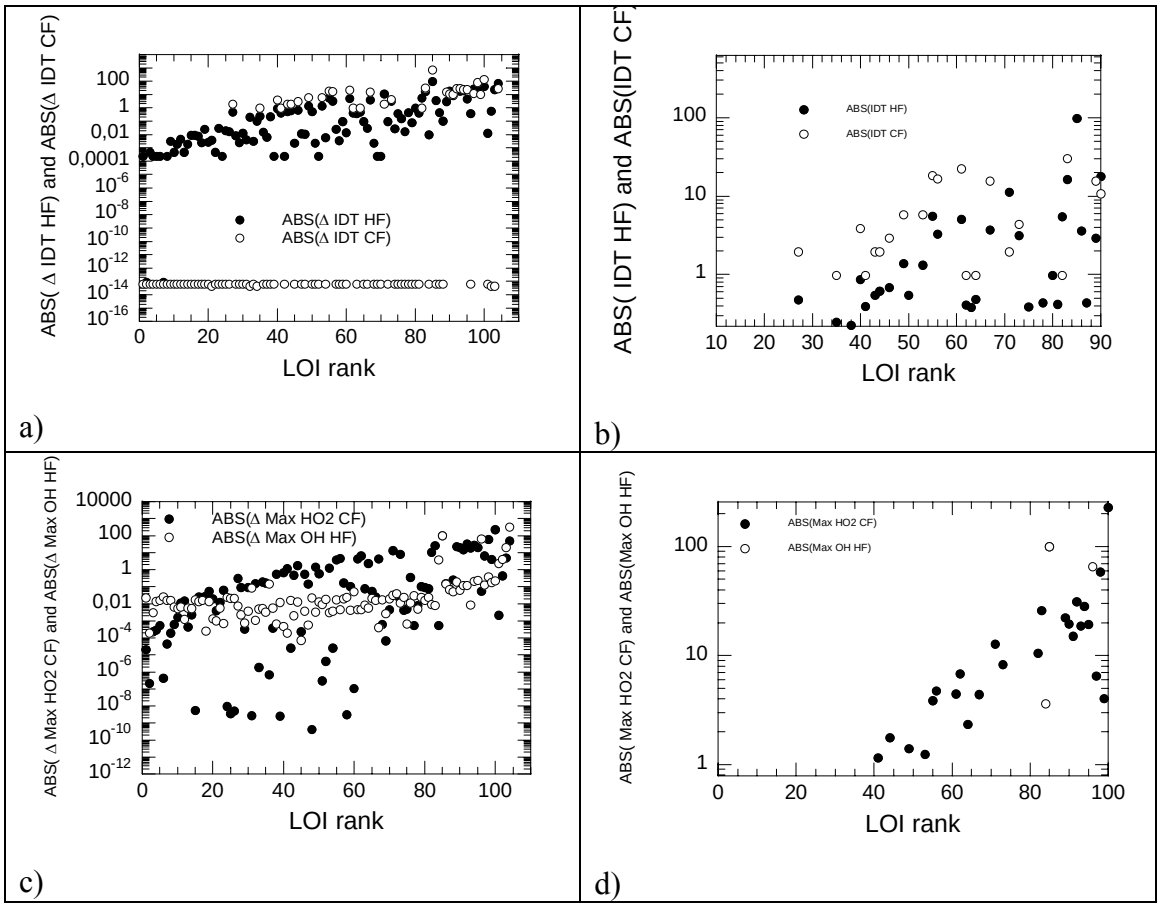


Figure 6.89.a) IDT HF and IDT CF vs LOI rank.
b) zoom of a).
c) MaxHO2 CF and Max OH HF vs LOI rank
d) zoom of c).

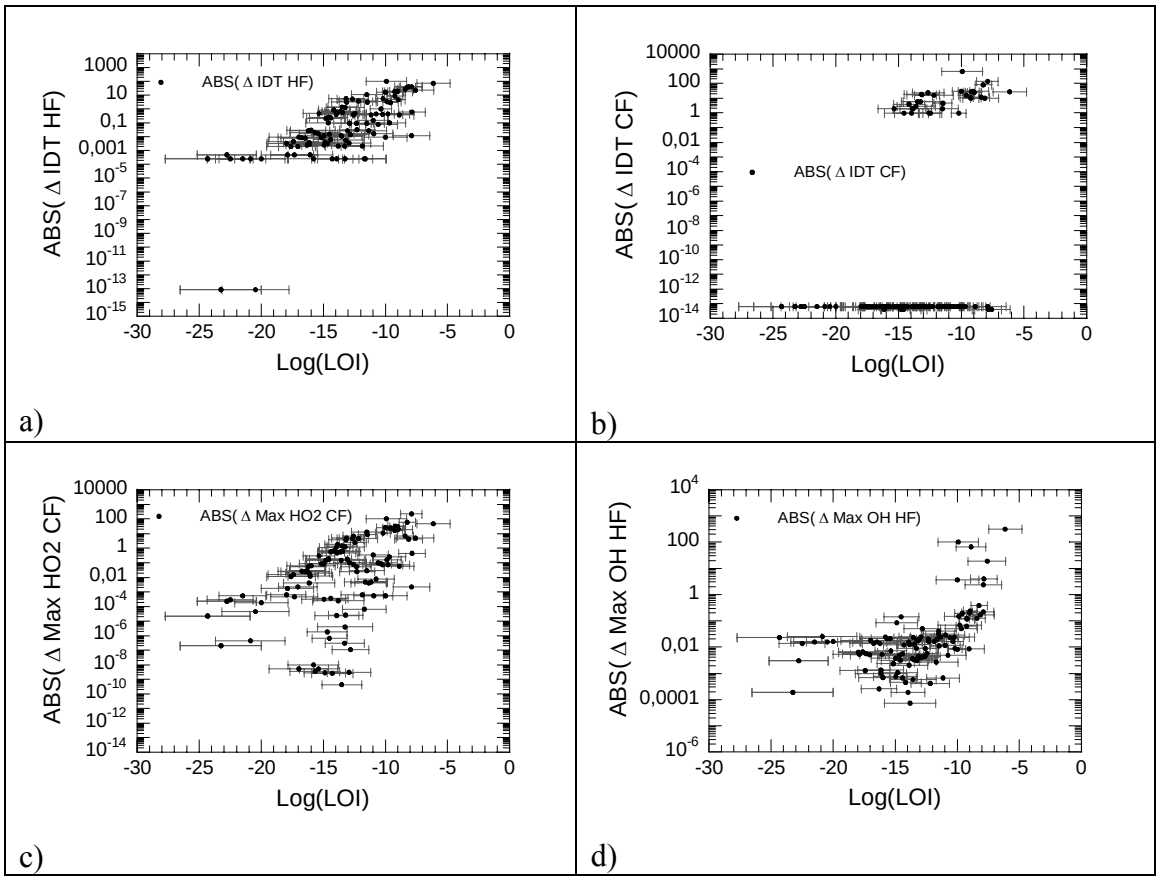


Figure 6.90. a) IDT HF vs Log LOI
b) IDT CF vs Log LOI
c) MaxHO2 CF vs Log LOI
d) MaxOH HF vs Log LOI

Summary and concluding remarks

Work done by previous members of the Division of Combustion Physics at Lund University has shown that LOI evaluated at MaxS and sensitivity toward OH or HO_2 gives good results. However, these investigations were performed on methane, which did not exhibit any two stage auto-ignition process.
The investigation in this thesis was performed on N-Heptane, which did exhibit a two stage auto-ignition process.
. For this reason the CF plays an important role when ranking the species for N-Heptane. This investigation has shown that the LOI measure ranks the QSS species for the chosen ranking target very well. When LT is taken at CF and sensitivity is taken towards HO_2, the ranking target is Max HO2 CF. The performance for the evaluation target Max HO2 CF is expected to be and also is very good. Similarly, when LT is taken at MaxS and sensitivity is taken towards OH, the ranking target is Max OH HF. The performance for the evaluation target Max OH HF is expected to be and also is very good.

However, the question is which ranking target to choose in order to get good ranking performance on the other evaluation targets as well. This investigation has shown that Max HO2 CF is a very good ranking target, since the species that affect the CF also affects the HF.

However, this investigation shows that the Conc. Max HO2 CF, alongside LOI Max HO2 CF, generates the most reduced mechanism and performs the most ordered ranking. The concentration can be a good measure for QSS species ranking, since the all ODEs, including the ODE for HO_2, is a function of the concentration of the species (see eq.(2.53)). The larger the concentration is and the more reactions the species is involved in the more it affects the time derivative of HO_2. Hence, both the IDT CF and Max HO2 CF are affected by the concentration via the time derivative of HO_2. And since the species that affect the CF also affect the HF, the IDT HF is also affected.

The investigation also shows that LOI Max HO2 HF and Conc. Max HO2 HF do not generate as highly reduced mechanisms and do not perform as ordered ranking as LOI Max HO2 CF and Conc. Max HO2 CF. The reason for this is that the species that affect the CF, and thereby indirectly the HF, gets very low ranking for LOI Max HO2 HF and Conc. Max HO2 HF. Hence, those species are set as QSS early, which contributes to an early unacceptable accumulated deviation.

The results show that LOI generates more reduced mechanisms and performs more ordered ranking than LT at both Max HO2 HF and MaxS. The reason for this is that LT does not take the species sensitivity toward important features in the combustion process into account. Species that are sensitive toward the evaluation targets can therefore get a low rank.
Also noticeable is that sensitivity toward HO_2 generates more reduced mechanisms than sensitivity towards OH for MaxS.
And finally, the combinations of two LOI lists do not perform better than the list based solely on LOI Max HO2 CF. The reason for this is that the MaxS list generally has larger values than the LOI Max HO2 CF list and therefore dominates.

6.2.2.6 Additional effects in groups of QSS species

It is of importance to know whether or not additional effects occur in groups of QSS species for the ET. This occurrence is investigated by comparing two different sampling methods for different levels of reduction for the chosen ET. In the first method the chosen ET is sampled for **groups of QSS species**.
In the second method the chosen ET is calculated from sum of the **individual QSS species** (included in the group of QSS species) contribution to the chosen ET. The latter method tests each species (included in the group) at the time as QSS and then sums up the contribution of the individual QSS species according to eq (6.11). If there are no additional effects in groups of QSS species for the chosen ET, the two methods should be equal.

Figures 6.91 and 6.92 show the ΔX_{ORG}, caused by single QSS species according to eq(6.9), vs LOI rank. X represents IDT HF, IDT CF, Max OH HF and Max HO2 CF. The ΔX_{ORG} values increases, even though fluctuating, with LOI rank as expected. It is clear that the IDT CF is not affected at all by some species, since they have a value of 10^{-13}.
The QSS species ranking list used for this investigation is **Max (MaxS Sens$_{OH}$, Max HO2 CF Sens$_{HO2}$).** The meaning of this list is explained in section 6.2.2.5.

Figures 6.93 to 6.96 show the additional effects of IDT HF, IDT CF, Max HO_2 CF and Max OH HF respectively. Each figure shows the curves of X caused by groups of QSS species from the ART selection process and $X_{SUM\ 1QSS}$ for the same groups of QSS species. The difference between the two curves increases with the number of QSS species in all figures.
Figure 6.97 shows ΔX_{RED}, which is the difference between the two curves, according to eq (6.10), in Figure 6.93 to 6.96.

$$\Delta X_{ORG} = \left| \frac{X_{i,1QSS} - X_{ORG}}{X_{ORG}} \cdot 100 \right| \tag{6.9}$$

$$\Delta X_{RED} = \frac{\left(X_{SUM\ 1QSS} - X_{RED}\right)}{X_{RED}} \cdot 100 \tag{6.10}$$

$$X_{SUM\ 1QSS} = \sum_{i=1}^{N_{QSS}} \left(X_{i,1QSS} - X_{ORG}\right) \tag{6.11}$$

where X_{RED} and X_{ORG} are the values of X for the reduced and original mechanism respectively. $X_{i,1QSS}$ is the value of X when only the particular QSS species, i, is set a QSS species. N_{QSS} is the number of QSS species at the given reduction level. Hence, $X_{SUM\ 1QSS}$ is the sum of the difference between $X_{i,1QSS}$ and the original mechanism for the given reduction level.

X represents either IDT HF, IDT CF, Max HO_2 CF or Max OH HF.

It is clear that the additional effects exist for all the parameters but that they are small and within 3 % for all *X* except IDT CF, which is within 8 %. This opens up an opportunity for reduced mechanism to be **tailor made** by the user. The user chooses the species wanted as QSS species in the reduced mechanism and the ART responds by estimating the deviation in the chosen parameter compared to the detailed mechanism within a few percent error in the estimation.

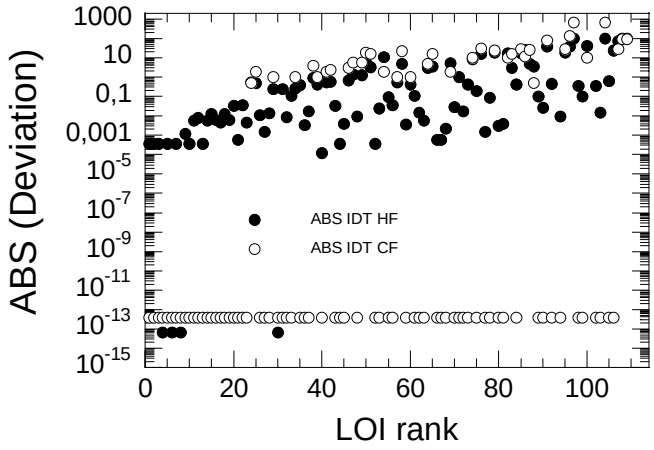


Figure 6.91. Shows the ΔX_{ORG}, caused by single QSS species according to eq(6.9),vs LOI rank. *X* is IDT HF and IDT CF.

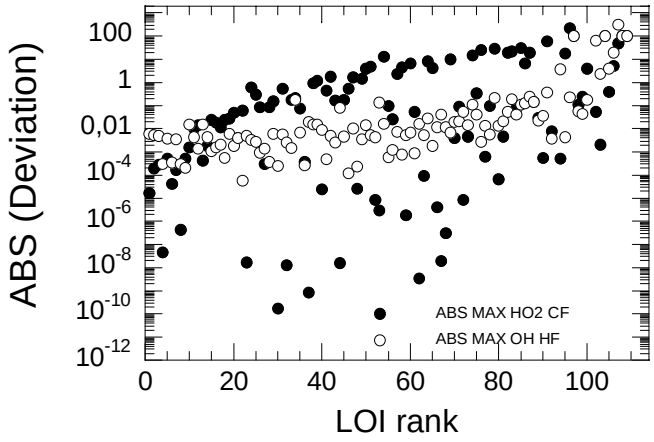


Figure 6.92. Shows the ΔX_{ORG}, caused by single QSS species according to eq(6.9),vs LOI rank. *X* is Max OH HF and Max HO2 CF.

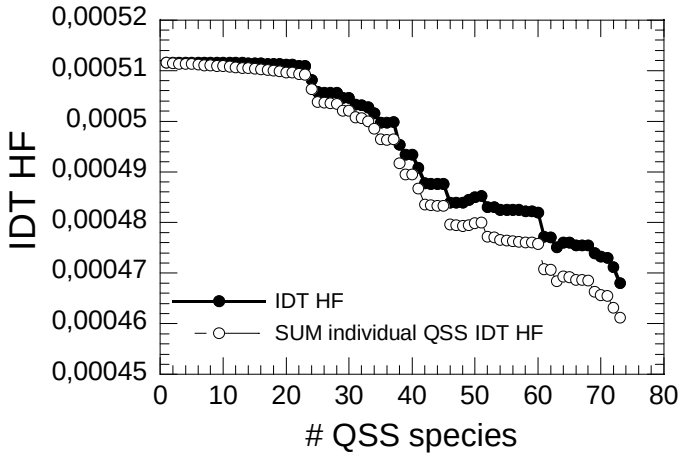


Figure 6.93. IDT HF and the sum of IDT HF contributions from individual QSS species vs number of QSS species. The additional effects exist but are small.

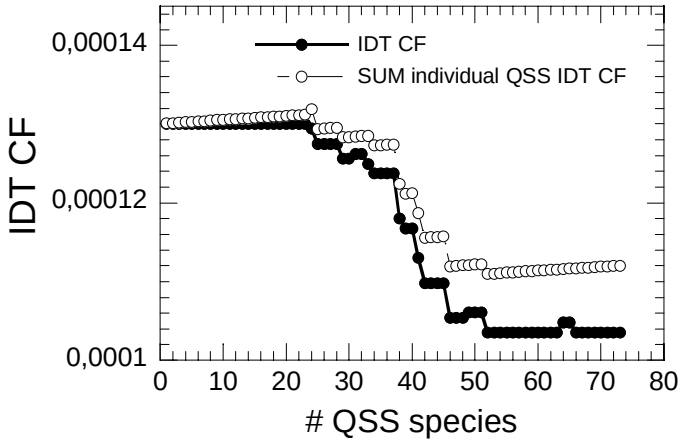


Figure 6.94. IDT CF and the sum of IDT CF contributions from individual QSS species vs number of QSS species. The additional effects exist but are quite small.

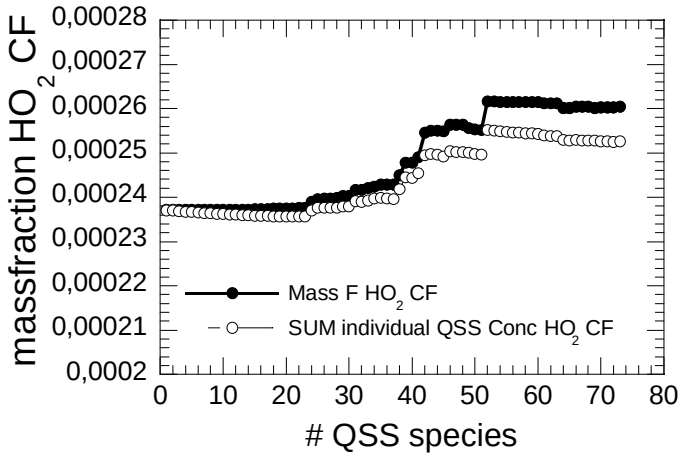


Figure 6.95. Mass fraction HO_2 CF and the sum of Mass fraction HO_2 CF contributions from individual QSS species vs number of QSS species. The additional effects exist but are quite small.

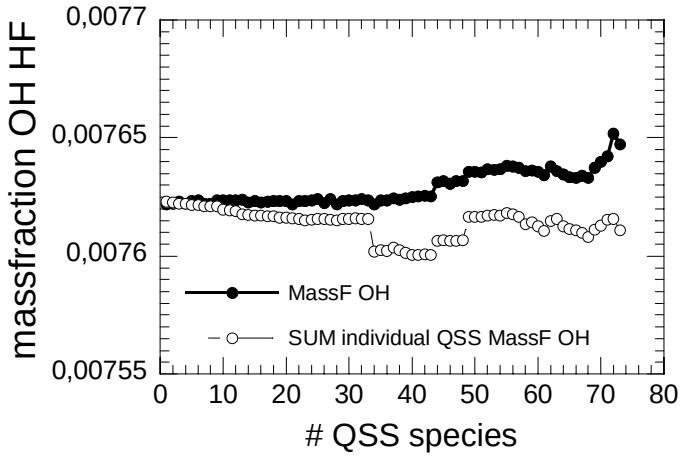


Figure 6.96. Mass fraction OH HF and the sum of Mass fraction OH HF contributions from individual QSS species vs number of QSS species. The additional effects exist but are very small.

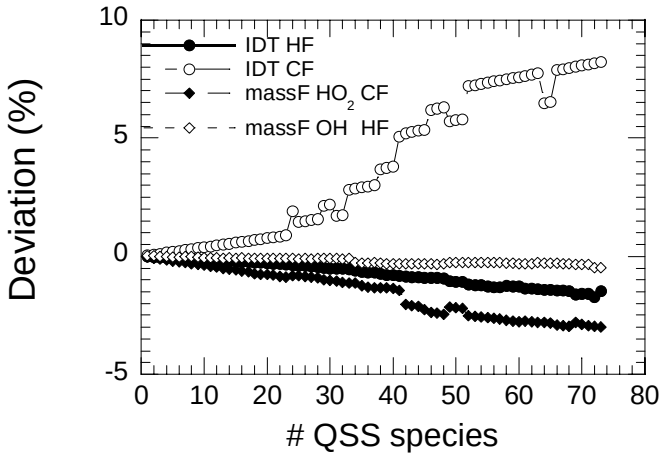


Figure 6.97. ΔX_{RED}, according to eq(6.10), for IDT HF, IDT CF, Mass fraction HO_2 CF and Mass fraction OH HF vs number of QSS species. The additional effects are largest for IDT CF.

6.2.3 Variation of physical ranges

This section investigates the variation of physical ranges like temperature, fuel/air ratio and pressure. The investigation involves three different cases, called “Case 2”, “Case 3” and “Case 4”, which are presented below. The physical ranges of the cases are presented in Table 6.1.

It should be noted that the same LOI list that was optimized for and used for Case I also was used for Case 2, 3 and 4. Hence, some species that fail as QSS species for Case 2, 3 and 4 have low LOI rank, since the LOI list is optimized for Case I.

6.2.3.1 Case 2: Temperature range

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature range: 18 temperature points between 625 and 1300 K
- Pressure point: 40 bar
- Fuel/air ratio point: 1.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %

- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 1 %

6.2.3.1.1 Reduction level of the reduced mechanisms

The reduced mechanism, containing 60 QSS species, is identical and valid for all initial temperatures, which is shown in Figure 6.98. The figure also shows that the number of QSS species is less than for a single temperature, even though it is in the NTC region, when an entire temperature range is considered. This is expected since some species are very important for low temperature chemistry and therefore cannot be used as a QSS species, while other species are very important for high temperature chemistry and therefore cannot be used as a QSS species.

Table 6.7 shows the species that failed to be accepted as QSS species, as well as the reason for failure, for the initial temperatures T=625, 900, 950 and 1300 K. It is clear that the first 72 species are not important for high temperature chemistry, since the species fail to be accepted only for the lower and middle high initial temperatures. From species 73 and higher the high temperature chemistry is important. Some species fail for one temperature only, while others fail for many temperatures.
Also noteworthy is that same species fail due to one reason only, while others fail for two three or all four reasons. The species with higher LOI rank have a tendency to fail for many reasons, which is natural since they are expected to affect the system more.

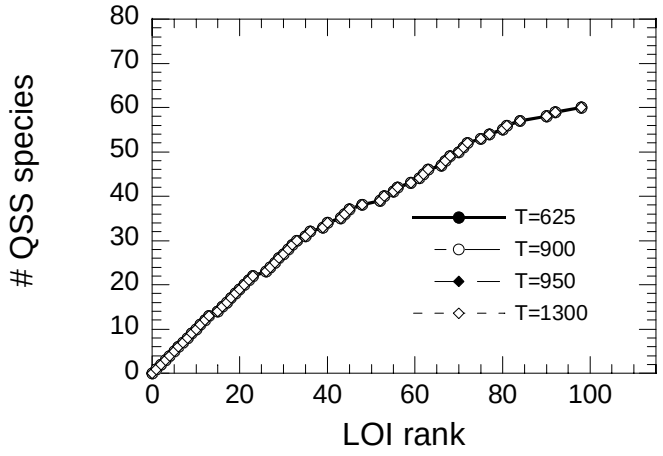


Figure 6.98. Number of QSS species vs LOI rank for initial temperatures of 625, 900, 950 and 1300 K. The curves are identical, since the same mechanism must be valid for all physical conditions.

Table 6.7. The table shows the species names and the corresponding LOI value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted and the reason for the failure for the initial temperatures 625, 900, 950 and 1300 K. A blank means that the species was accepted, while a letter and number combination means that the species failed to be accepted. The numbers represent the deviation according to eq(6.2) for each ET in percent. The letters mean failure due to; A) IDT HF limit B) IDT CF limit C) Max HO2 CF limit D) Max OH HF limit E) No convergence in the inner solver F) The species is forbidden as a QSS species beforehand G) There was no ignition H) Temperature limit I) CPU time limit

Nr	**Species**	T=625	T=900	T=950	T=1300
1	CH2CH2COCH3				
2	C2H5COCH2				
3	CH2CH2CHO				
4	CH3CHCOCH3				
5	N-C3H7CO				
6	C2H5O				
7	5R-HEOOH-P				
8	HOCH2O				
9	N-C3H7COC2H4-1				
10	C2H5CO				
11	7R-HEOOH-P				
12	6R-HEOOH-P				
13	C2H4O2H				
14	5R-HEOOHO2-P	A= -3.2			
15	7R-HEOOH-S				
16	1-C5H11				
17	2-C5H11				
18	5R-HEOOH-S				
19	N-C3H7				
20	CH3COCH2				
21	1-C4H9				
22	6R-HEOOH-S				
23	CH				
24	C6H10	C= 6.1			
25	7R-O-HEPOOH-P	A= -77.5 B= -77.6			
26	1-C2H4COC2H5				
27	I-C3H7				
28	CH3O				
29	6R-HEOOHO2-P				
30	C2H				
31	HO2CHO				
32	1-CH2				
33	L-C7H15				
34	7R-HEOOHO2-P	A= -3.3			
35	OCHO				
36	2-C4H8				
37	C6H5O	E			
38	7R-O-HEPOOH-S	A= -81.1 B= -81.1			
39	C2H5O2				
40	C4H7				
41	7R-HEOOHO2-S	A= -3.2			
42	CH3O2H	A= -4.5			
43	C7H13				
44	A1-				
45	C6H8				
46	5R-HEOOHO2-S	A= -7.9 B= -7.9			
47	6R-O-HEPOOH-P	A= -77.2 B= -77.3	B= -6.5		

48	C4H612				
49	6R-HEOOHO2-S	A= -6.7 B= -6.7	B= -6.5		
50	CH3O2		A= -5.6 B= -20.0	A= -6.8 B= -6.8	
51	5R-O-HEPOOH-S	A= -57.9 B= -57.9 C= -9.0	A= -3.3 B= -17.5 C= 5.2		
52	N-C4H3				
53	A1				
54	O2CHO	A= -3.6 C= 109.2	A= -11.2 C= 17.2	A= -5.0	
55	C4H10				
56	C5H9				
57	N-C4H9COCH2	C= 8.5			
58	6R-O-HEPOOH-S	A= -92.2 B= -92.2 C= -8.3 D= 1.1	A= -5.1 B= -24.0		
59	N-C4H5				
60	C2H5O2H	C= 7.9	C= 7.5		
61	C2H6				
62	I-C4H3				
63	HCO				
64	N-C3H7COCH2		A= -3.0 C= 8.8	A= -3.9 B= -74.4 C= -87.8	
65	L-C7H15O2	A= -20.7 B= -20.7 C= 8.4	A= -3.8 B= -17.0		
66	HCCO				
67	3-CH2				
68	I-C4H5				
69	CH3OH		A= 5.3 C= -10.6		
70	CH2CHO				
71	HOCHO				
72	C4H4				
73	C4H6				A= -3.4
74	1-C4H8		A= 8.4 B= 9.0 C= -15.0	A= 4.9	A= -11.2 B= -11.4
75	C3H8				
76	5R-C7H14O	A= 14.0 B= 14.0	A= 16.1 B= 28.0 C= -25.9	A= 12.4 B= 12.4	
77	CH2OH				
78	C2H5				A= -4.6
79	C3H6		A= 19.6 B= 24.0 C= -29.3	A= 13.7 B= 13.8	A= -3.9
80	C2H3				
81	CH3CO				
82	1-C5H10		A= 20.2 B= 7.0 C= -23.3	A= 9.7 B= 9.7	A= -48.6 B= -49.1
83	CH3CHO	A= 6.8 B= 6.8 D= 1.1	B= 16.0 C= -23.0	A= -11.6 B= -11.6	
84	C3H5				
85	N-C3H7CHO	E			
86	H2O2	A= -9.5 B= -9.5 C= 24.6	A= -26.7 B= -12.0 C= 5.2	A= -56.4 B= -86.9 C= -70.0	A= -18.1 B= -18.4
87	C2H5CHO	A= 5.1 B= 5.1 C= 36.3 D= 1.1	A= -5.8 B= 20.0 C= -18.9	A= -11.5 B= -11.5	

88	C3H4P		A= -3.9		
89	C3H4				A= -3.9
90	C3H3				
91	L-C7H14	A= 7.5 B= 7.5 C= -8.2	A= -7.6 B= 263.5 C= 230.2	A= -34.7 B= -34.7	A= -28.2 B= -28.5
92	C2H2				
93	HO2	F			
94	O	E	E		
95	CH2CO		A= -20.1 B= 22.0 C= -16.4	A= -19.0 B= -19.0	
96	CH2O	C= 72.3 D= 1.1	A= -41.8 B= 127.9 C= 246.2	A= -43.6 B= -43.6	A= -27.2 B= -27.3
97	C2H5COCH3	NO IGNITION			
98	CH3				
99	CH4	F			
100	C2H4	A= -10.9 B= -10.9 C= 27.6 D= 1.1	A= -43.8 B= -13.0 C= 5.4	A= -45.3 B= -45.4	A= -61.8 B= -62.4
101	OH	F	F	F	F
102	CO2	F	F	F	F
103	H	E	E	E	E
104	N-C7H16	F	F	F	F
105	H2	E	E	E	E
106	CO	E	E	E	E
107	H2O	F	F	F	F
108	O2	F	F	F	F
109	N2	F	F	F	F
110	AR	F	F	F	F

6.2.3.1.2 CPU time of the reduced mechanisms

The normalized CPU time vs number of QSS species is shown in Figure 6.99. The normalized CPU time decreases until about 50 QSS for all initial temperatures. Thereafter the curves corresponding to 900, 950 and 1300 K fluctuates around the same value, while the curve corresponding to 625 K continues to decrease.
The normalized CPU time of the reduced mechanisms varies somewhat for the different physical condition, but the trend is similar for all initial temperatures and fuel/air ratios. The variation can partly be explained by CPU noise due to other processes in the computer. However, the general explanation for the variation is that each physical condition corresponds to a unique trajectory in species concentration space and that the convergence of the solver combination, and thereby CPU time, is dependent on the trajectory. The amount of CPU time that can be gained by applying the QSSA to a particular species is also trajectory dependent. Hence, a variation in CPU time exists among the physical conditions for different reduction levels.

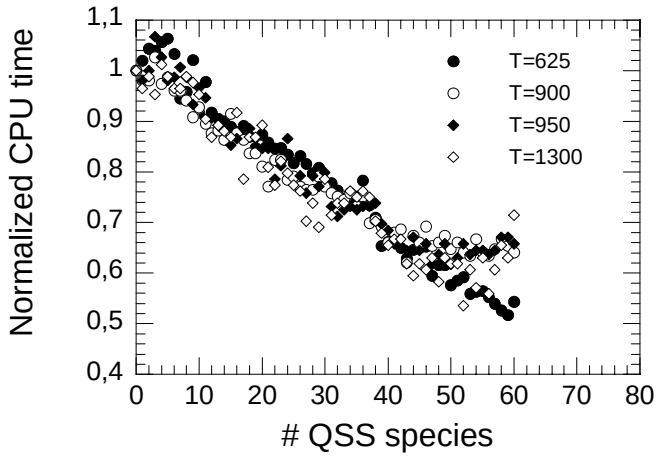


Figure 6.99. Normalized CPU time vs number of QSS species for initial temperatures of 625, 900, 950 and 1300 K.

6.2.3.1.3 Accuracy of the reduced mechanisms

Figure 6.100 shows IDT HF and IDT CF vs 1000/T for experiments [2] and simulation with 0 QSS species (original mechanism). The curves for the experiment and simulation have good agreement for IDT CF for most initial temperatures, although the agreement decreases a bit for the lowest experimental temperatures.
The curves for the experiment and simulation have good agreement for IDT HF for most initial temperatures, although the agreement decreases a bit for the highest experimental temperatures.

Figure 6.101 shows the IDT HF vs 1000/T for the original mechanism and the most reduced mechanism containing 60 QSS species. The figure shows that the agreement between the two curves is very good for all initial temperatures.
However, the deviation, according to eq (6.4), between the two mechanisms, which is shown in Figure 6.102, is larger than one may think from looking at Figure 6.101. Hence, a figure like Figure 6.101, which is standard within the field, can be quite deceiving about the accuracy of the reduced mechanism.

Figure 1.03 shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism containing 60 QSS species. The agreement between the two curves corresponding to the 1:st ignition is very good. The figure also shows that the 1:st and 2:nd ignition coincide for low temperatures.

The agreement between the original and reduced mechanism is better than the agreement between the original mechanism and the experimental data if all points are considered for both IDT HF and IDT CF. This gives an argument for the use of a less accurate reduced mechanism, so that the deviation between the original and reduced mechanism is of the

same order as the deviation between the original mechanism and the experimental results. The accuracy of the reduced mechanism can be decreased by using more generous ART ET limits.

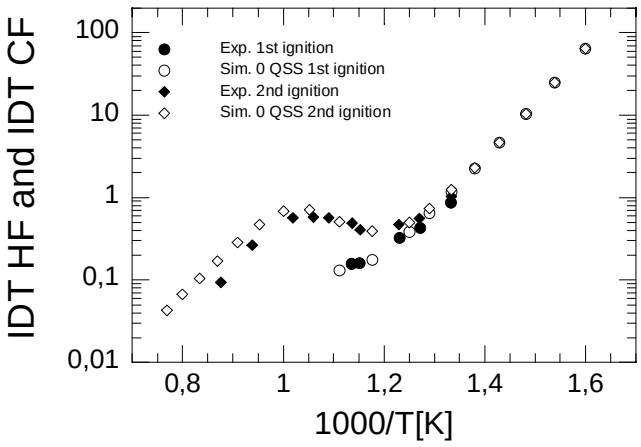


Figure 6.100. Shows IDT HF and IDT CF vs 1000/T for experiments and simulation with 0 QSS species. 1:st and 2:nd ignition corresponds to IDT CF and IDT HF respectively. The fuel/air ratio is 1.0 and the initial pressure is 40 bar.

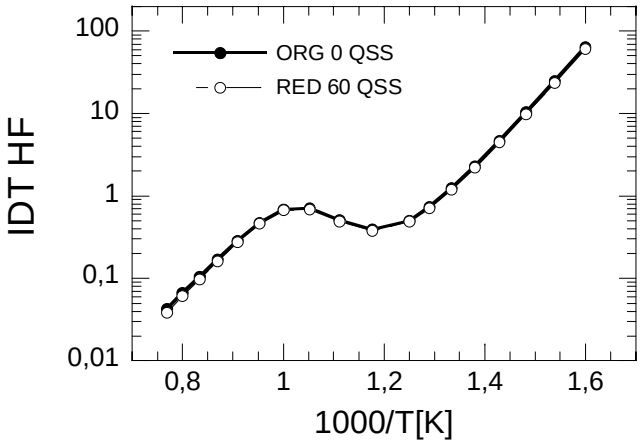


Figure 6.101. IDT HF vs 1000/T for the original and most reduced mechanism for the temperature range 625-1300 K.

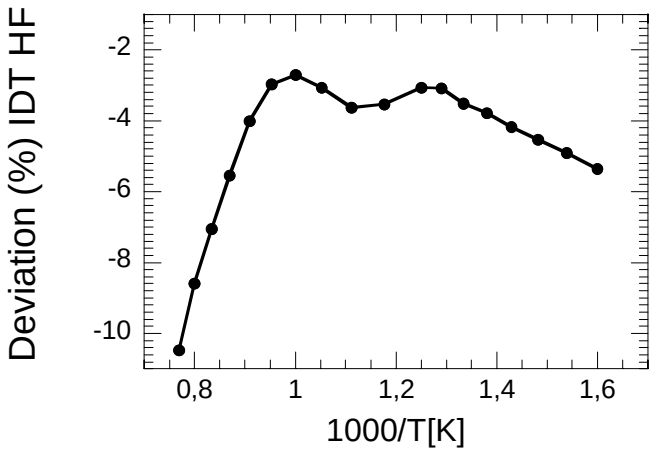


Figure 6.102. Difference for IDT HF between the original and reduced mechanism vs 1000/T for the original and most reduced mechanism for the temperature range 625-1300 K.

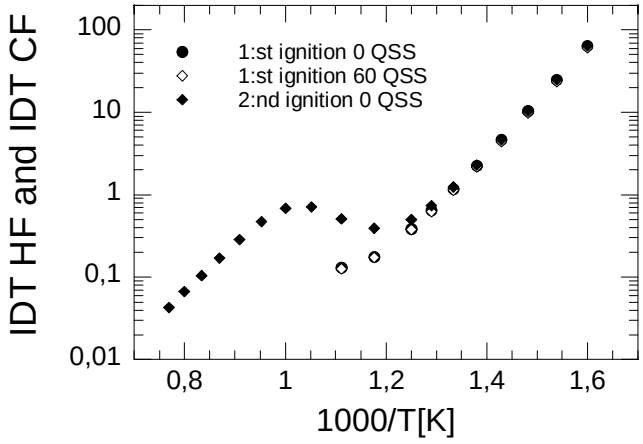


Figure 6.103. IDT HF and IDT CF vs 1000/T[K]. The figure shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism, which contains 0 and 60 QSS species respectively. The temperature range is 625-1300 K, the pressure is 40 bar and the fuel/air 1.0. The 1:st and 2:nd ignition coincide for low temperatures.

6.2.3.2 Case 3: Temperature and Fuel/air ratio range

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature range: 18 temperature points between 625 and 1300 K
- Pressure point: 40 bar
- Fuel/air ratio range: 0.5, 1.0 and 2.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 2 %

6.2.3.2.1 Reduction level of the reduced mechanisms

The reduced mechanism, containing 51 QSS species, is identical and valid for all initial temperatures and fuel/air ratios, which is shown in Figure 6.104. The figure also shows that the number of QSS species of the most reduced mechanism of case 3 is less than for case 2. This is expected since some species are very important for low fuel/air ratios and therefore cannot be used as a QSS species, while other species are very important for high fuel/air ratios and therefore cannot be used as a QSS species. The figure also shows the importance of the “Second Chance” option in the ART, since one QSS species was accepted the second time it was tested. Hence, the “Second chance” option increases the reduction from 50 to 51 QSS species.

Table 6.8 shows the species that failed to be accepted as QSS species, as well as the reason for failure, for the initial temperatures T=625, 900, 950 and 1300 K. Some species fail for one reason only, while others fail for many reasons. The species with higher LOI rank have a tendency to fail for many reasons, which is natural since they affect the system more.

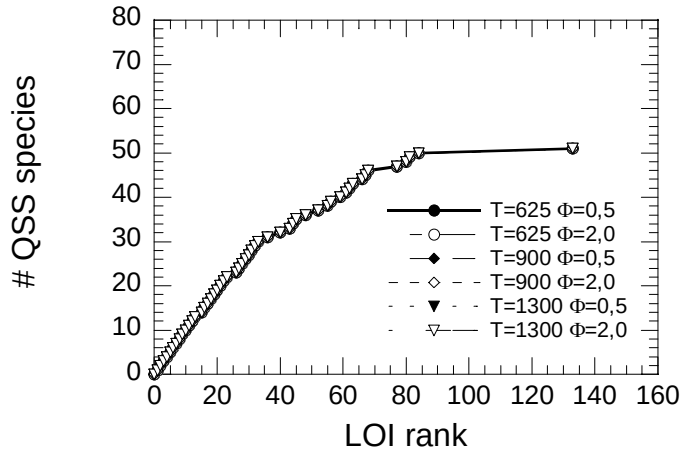


Figure 6.104. Number of QSS species vs LOI rank when the entire temperature range T=625-1300 K and the fuel/air ratios; 0.5, 1.0 and 2.0 is considered. Only six combinations of physical conditions are shown in the figure.

Table 6.8. The table shows the species names and the corresponding LOI value. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted and the reason for the failure for the initial temperatures 625, 900, 950 and 1300 K. A blank means that the species was accepted, while a letter and number combination means that the species failed to be accepted. The first number corresponds to the fuel/air ratio. The second number represents the deviation according to eq(6.2) for each ET in percent. The letters mean failure due to; A) IDT HF limit B) IDT CF limit C) Max HO2 CF limit D) Max OH HF limit E) No convergence in the inner solver F) The species is forbidden as a QSS species beforehand G) There was no ignition H) Temperature limit I) CPU time limit

Nr	**Species**	T=625	T=900	T=950	T=1300
1	CH2CH2COCH3				
2	C2H5COCH2				
3	CH2CH2CHO				
4	CH3CHCOCH3				
5	N-C3H7CO				
6	C2H5O				
7	5R-HEOOH-P				
8	HOCH2O				
9	N-C3H7COC2H4-1				
10	C2H5CO				
11	7R-HEOOH-P				
12	6R-HEOOH-P				
13	C2H4O2H				
14	5R-HEOOHO2-P	1.0 A= -3.2 2.0 A= -4.0			
15	7R-HEOOH-S				
16	1-C5H11				
17	2-C5H11				
18	5R-HEOOH-S				
19	N-C3H7				
20	CH3COCH2				
21	1-C4H9				

22	6R-HEOOH-S				
23	CH				
24	C6H10	1.0 C= 6.1 0.5 C= -50.5			
25	7R-O-HEPOOH-P	1.0 A= -77.5 1.0 B= -77.6 0.5 A= -76.5 0.5 B= -76.6 2.0 A= -78.3 2.0 B= -78.3			
26	1-C2H4COC2H5				
27	I-C3H7				
28	CH3O				
29	6R-HEOOHO2-P				
30	C2H				
31	HO2CHO				
32	1-CH2				
33	L-C7H15				
34	7R-HEOOHO2-P	1.0 A= -3.3 0.5 A= -3.2 2.0 A= -3.5			
35	OCHO				
36	2-C4H8				
37	C6H5O				
38	7R-O-HEPOOH-S	1.0 A= -81.1 1.0 B= -81.2 0.5 A= -81.3 0.5 B= -81.9 0.5 C= -50.6 2.0 A= -80.7 2.0 B= -80.7 2.0 C= 5.1			
39	C2H5O2				
40	C4H7				
41	7R-HEOOHO2-S	1.0 A= -3.2 0.5 A= -3.1 2.0 A= -3.3			
42	CH3O2H	1.0 A= -4.7 0.5 A= -3.2 2.0 A= -6.2 2.0 B= -6.2	2.0 D= 40.2		
43	C7H13				
44	A1-				
45	C6H8				
46	5R-HEOOHO2-S	1.0 A= -7.8 1.0 B= -7.8 0.5 A= -7.5 0.5 B= -7.5 2.0 A= -8.1 2.0 B= -8.1			
47	6R-O-HEPOOH-P	1.0 A= -77.4 1.0 B= -77.5 0.5 A= -78.0 0.5 B= -78.2 2.0 A= -76.5 2.0 B= -76.5	1.0 B= -5.9 0.5 B= -5.7 2.0 B= -5.4		
48	C4H612				
49	6R-HEOOHO2-S	1.0 A= -6.7 1.0 B= -6.7 0.5 A= -6.4 0.5 B= -6.4 2.0 A= -7.0 2.0 B= -7.0	1.0 B= -5.9 2.0 B= -5.4		
50	CH3O2	0.5 A= -3.5	1.0 A= -5.6 1.0 B= -19.8 0.5 A= -3.2 0.5 B= -20.5 2.0 A= -7.4	1.0 A= -6.8 1.0 B= -6.8 0.5 A= -5.3 0.5 B= -5.4 2.0 A= -8.0	

			2.0 B= -17.5	2.0 B= -25.0	
51	5R-O-HEPOOH-S	1.0 A= -57.1 1.0 B= -57.2 1.0 C= -8.4 0.5 A= -56.5 0.5 B= -56.6 2.0 A= -57.7 2.0 B= -57.7	1.0 A= -3.3 1.0 B= -16.8 0.5 B= -16.4 0.5 C= 5.3 2.0 A= -5.0 2.0 B= -16.3	2.0 B= -7.7	
52	N-C4H3				
53	A1	2.0 D= -9.4	2.0 D= -11.5	2.0 D= -10.0	2.0 D= -2.0
54	O2CHO	1.0 A= -3.2 1.0 C= 104.4 0.5 A= -5.1 0.5 B= -5.2 0.5 C= 48.1 2.0 C= 37.2	1.0 A= -11.3 1.0 C= 15.1 0.5 A= -11.9 0.5 C= 18.2 2.0 A= -8.5 2.0 C= 11.9	1.0 A= -4.8 0.5 A= -4.7 2.0 A= -4.3	
55	C4H10				
56	C5H9				
57	N-C4H9COCH2	1.0 C= 10.0 0.5 C= -46.9 2.0 C= 11.1			
58	6R-O-HEPOOH-S	1.0 A= -92.2 1.0 B= -98.5 1.0 C=1000.0 0.5 A= -91.9 0.5 B= -98.1 0.5 C=1000.0 2.0 A= -92.0 2.0 B= -98.7 2.0 C=1000.0	1.0 A= -5.1 1.0 B= -23.8 0.5 B= -24.6 0.5 C= 5.6 2.0 A= -7.6 2.0 B= -22.9	2.0 B= -5.8	
59	N-C4H5				
60	C2H5O2H	1.0 C= 8.4 0.5 C= -40.5	1.0 C= 6.8 0.5 C= 6.3 2.0 C= 7.3		
61	C2H6				
62	I-C4H3				
63	HCO				
64	N-C3H7COCH2	1.0 C= 6.5 0.5 C= 8.0 2.0 A= -3.6 2.0 C= 9.3	1.0 A= -3.0 1.0 C= 8.1 0.5 C= 7.1 2.0 A= -3.7 2.0 B= -5.4 2.0 C= 9.3	1.0 A= -3.9 1.0 B= -74.2 1.0 C= -88.0 2.0 A= -4.9 2.0 B= -13.5 2.0 C= 10.4	
65	L-C7H15O2	1.0 A= -20.7 1.0 B= -20.7 0.5 A= -19.7 0.5 B= -19.8 2.0 A= -21.6 2.0 B= -21.6 2.0 C= -8.2	1.0 A= -3.7 1.0 B= -15.3 0.5 B= -14.8 2.0 A= -6.0 2.0 B= -16.3	2.0 B= -7.7	
66	HCCO				
67	3-CH2				
68	I-C4H5				
69	CH3OH	1.0 C= 10.5 0.5 C= 6.5 2.0 C= 21.0	1.0 A= 5.5 1.0 C= -10.8 0.5 A= 3.0 0.5 C= -10.1 2.0 A= 6.7 2.0 C= -11.0	2.0 A= 3.5 2.0 B= 9.6 2.0 C= -6.9	
70	CH2CHO				
71	HOCHO				
72	C4H4	2.0 D= 3.3	2.0 D= 3.9	2.0 D= 3.6	
73	C4H6				1.0 A= -3.5
74	1-C4H8	0.5 C= 5.5	1.0 A= 8.8 1.0 B= 9.9 1.0 C= -15.3 0.5 A= 5.0 0.5 B= 13.1 0.5 C= -16.3	1.0 A= 5.0 1.0 B= 5.0 2.0 A= 6.6 2.0 B= 197.7 2.0 C= 355.8	1.0 A= -11.5 1.0 B= -11.6 0.5 A= -10.3 0.5 B= -10.5 2.0 A= -11.5 2.0 B= -11.7

			2.0 A= 10.8 2.0 B= 7.2 2.0 C= -14.7		
75	C3H8				
76	5R-C7H14O	1.0 A= 14.1 1.0 B= 14.1 0.5 A= 14.8 0.5 B= 14.9	1.0 A= 16.4 1.0 B= 30.7 1.0 C= -26.0 0.5 A= 11.1 0.5 B= 39.3 0.5 C= -28.0	1.0 A= 12.4 1.0 B= 12.4 0.5 A= 9.5 0.5 B= 9.5	
77	CH2OH				
78	C2H5				1.0 A= -4.6 0.5 A= -3.7 2.0 A= -4.6
79	C3H6	0.5 A= 3.7 0.5 C= 51.6	1.0 A= 19.7 1.0 B= 24.8 1.0 C= -29.2 0.5 A= 15.2 0.5 B= 32.8 0.5 C= -32.0 2.0 A= 21.2 2.0 B= 19.3 2.0 C= -26.7	1.0 A= 13.5 1.0 B= 13.5 0.5 A= 9.5 0.5 B= 9.5 2.0 A= 16.6 2.0 B= 225.8 2.0 C= 352.5	1.0 A= -4.2 2.0 A= -8.5 2.0 B= -8.4
80	C2H3				
81	CH3CO				
82	1-C5H10		1.0 A= 20.9 1.0 B= 6.9 1.0 C= -23.4 0.5 A= 9.7 0.5 B= 9.0 0.5 C= -21.3 2.0 A= 31.9 2.0 B= 6.0 2.0 C= -25.2	1.0 A= 9.7 1.0 B= 9.7 2.0 A= 23.5 2.0 B= 245.1 2.0 C= 359.2	1.0 A= -50.0 1.0 B= -50.1 0.5 A= -50.0 0.5 B= -50.3 2.0 A= -47.3 2.0 B= -86.8 2.0 C= -71.0
83	CH3CHO	1.0 A= 6.0 1.0 B= 6.0 0.5 A= 8.5 0.5 B= 8.5 2.0 A= 3.9 2.0 C= 5.7	1.0 B= 15.8 1.0 C= -22.6 0.5 A= 3.7 0.5 B= 27.9 0.5 C= -28.8 2.0 B= 7.2 2.0 C= -16.5	1.0 A= -11.0 1.0 B= -11.0 0.5 A= -8.3 0.5 B= -8.3 2.0 A= -13.7 2.0 B= -7.7	
84	C3H5				
85	N-C3H7CHO	1.0 A= 5.0 0.5 A= 7.6 0.5 B= 7.2 0.5 C= -55.6	1.0 A= 18.0 1.0 B= 26.7 1.0 C= -31.2 0.5 A= 16.4 0.5 B= 44.3 0.5 C= -37.9		
86	H2O2	1.0 A= -10.0 1.0 B= -10.1 1.0 C= 22.1 0.5 A= -6.4 0.5 B= -6.4 0.5 C= 16.9 2.0 A= -13.6 2.0 B= -13.6 2.0 C= 17.1	1.0 A= -26.2 1.0 B= -12.9 1.0 C= 5.7 0.5 A= -24.9 0.5 B= -9.0 2.0 A= -24.2 2.0 B= -15.7 2.0 C= 10.8	1.0 A= -56.0 1.0 B= -86.9 1.0 C= -69.8 0.5 A= -50.5 0.5 B= -90.7 0.5 C= -74.0 2.0 A= -54.8 2.0 B= -50.0 2.0 C= 147.6	1.0 A= -17.9 1.0 B= -18.1 0.5 A= -19.9 0.5 B= -20.1 2.0 A= -15.3 2.0 B= -15.6
87	C2H5CHO	1.0 A= 6.3 1.0 B= 6.4 1.0 C= 52.6 0.5 A= 8.3 0.5 B= 8.3 0.5 C= 11.1 2.0 E	1.0 A= -4.7 1.0 B= 24.8 1.0 C= -19.7 0.5 B= 41.0 0.5 C= -27.8	1.0 A= -11.0 1.0 B= -11.0 0.5 A= -6.9 0.5 B= -6.9	
88	C3H4P	0.5 C= 15.8 2.0 D= -2.5	1.0 A= -3.9 0.5 A= -6.1	0.5 A= -3.9	2.0 A= -4.9
89	C3H4	0.5 E			1.0 A= -4.3
90	C3H3			2.0 D= -2.0	2.0 D= -2.2
91	L-C7H14	1.0 A= 7.8	1.0 A= -6.3	1.0 A= -34.0	1.0 A= -28.7

		1.0 B= 7.8 1.0 C= -7.8 0.5 A= 10.1 0.5 B= 10.1 2.0 A= 6.2 2.0 B= 6.2 2.0 C= -6.6	1.0 B= 271.1 1.0 C= 230.2 0.5 A= -17.9 0.5 B= 525.6 0.5 C= 253.8 2.0 B= 161.7 2.0 C= 87.1	1.0 B= -34.1 0.5 A= -37.9 0.5 B= -38.0 2.0 A= -28.0 2.0 B= 100.2 2.0 C= 343.2	1.0 B= -29.0 0.5 A= -28.5 0.5 B= -28.7 2.0 A= -27.0 2.0 B= -27.5
92	C2H2	2.0 D= -5.7	2.0 D= -34.0	2.0 D= -36.4	2.0 D= -42.2
93	HO2	F			
94	O	1.0 E			
95	CH2CO	1.0 C= 76.6 0.5 C= 25.8 2.0 D= -2.4	1.0 A= -19.5 1.0 B= 24.8 1.0 C= -16.9 0.5 A= -20.5 0.5 B= 37.7 0.5 C= -24.9 2.0 A= -16.2 2.0 B= 15.7 2.0 C= -8.7	1.0 A= -17.4 1.0 B= -17.4 0.5 A= -14.4 0.5 B= -14.4 2.0 A= -17.8 2.0 B= 128.8 2.0 C= 359.0	2.0 D= 3.9
96	CH2O	1.0 C= 92.5 0.5 C= 55.7 2.0 A= -3.1 2.0 C= 28.6 2.0 D= -4.8	1.0 A= -41.4 1.0 B= 132.0 1.0 C= 236.2 0.5 A= -39.0 0.5 B= 68.9 0.5 C= -17.3 2.0 A= -39.4 2.0 B= 44.9 2.0 C= 27.7	1.0 A= -41.8 1.0 B= -41.8 0.5 A= -35.4 0.5 B= -35.4 2.0 A= -45.5 2.0 B= 51.4 2.0 C= 369.8	1.0 A= -26.0 1.0 B= -26.3 0.5 A= -23.1 0.5 B= -23.4 2.0 A= -27.9 2.0 B= -28.3
97	C2H5COCH3	NO IGNITION			
98	CH3				0.5 A= -3.1
99	CH4	F	F	F	F
100	C2H4	1.0 A= -10.2 1.0 B= -10.2 1.0 C= 30.3 0.5 A= -7.9 0.5 B= -7.8 0.5 C= 62.6 2.0 A= -12.3 2.0 B= -12.3 2.0 C= 27.8 2.0 D= -15.5	1.0 A= -42.6 1.0 B= -10.9 1.0 C= 5.1 0.5 A= -43.3 0.5 B= -8.2 2.0 A= -39.7 2.0 B= -13.3 2.0 C= 10.2 2.0 D= -5.1	1.0 A= -43.1 1.0 B= -43.1 0.5 A= -40.7 0.5 B= -40.7 0.5 C= 6.4 2.0 A= -43.6 2.0 B= 56.7 2.0 C= 396.4 2.0 D= -3.9	1.0 A= -51.7 1.0 B= -52.3 0.5 A= -45.4 0.5 B= -45.9 2.0 A= -59.1 2.0 B= -60.3 2.0 D= 3.8
101	OH	F	F	F	F
102	CO2	F	F	F	F
103	H	1.0 E			
104	N-C7H16	F	F	F	F
105	H2	1.0 E			
106	CO	1.0 E			
107	H2O	F	F	F	F
108	O2	F	F	F	F
109	N2	F	F	F	F
110	AR	F	F	F	F

6.2.3.2.2 CPU time of the reduced mechanisms

The normalized CPU time vs number of QSS species for some key combinations of physical conditions is shown in Figure 6.105. The normalized CPU time of the reduced mechanisms varies somewhat for the different physical condition, but the trend is similar for all initial temperatures and fuel/air ratios. The variation can partly be explained by CPU noise due to other processes in the computer. However, the general explanation for the variation is that each physical condition corresponds to a unique trajectory in species concentration space and that the convergence of the solver combination, and thereby CPU time, is dependent on the trajectory. The amount of CPU time that can be gained by applying the QSSA to a particular species is also trajectory dependent. Hence, a variation in CPU time exists among the physical conditions for different reduction levels.

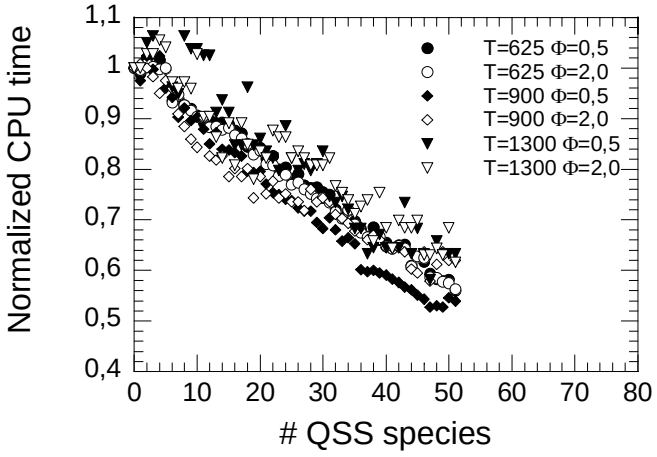


Figure 6.105. Normalized CPU time vs number of QSS species when the entire temperature range T=625-1300 K and the fuel/air ratios; 0.5, 1.0 and 2.0 is considered. Only six combinations of physical conditions are shown in the figure.

6.2.3.2.3 Accuracy of the reduced mechanisms

Figure 6.106 shows IDT HF and IDT CF vs 1000/T for experiments [2] and simulation with 0 QSS species (original mechanism). The curves for the experiment and simulation have good agreement for IDT CF for most initial temperatures, although the agreement decreases a bit for the lowest experimental temperatures.
The curves for the experiment and simulation have good agreement for IDT HF for most initial temperatures, although the agreement decreases a bit for the highest experimental temperatures.

Figure 6.107 shows the IDT HF vs 1000/T for the original mechanism and the most reduced mechanism containing 51 QSS species for the fuel/air ratio 0.5, 1.0 and 2.0. The figure shows that the agreement between the two curves is very good for all initial temperatures and all fuel/air ratios.

Figure 6.108 shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism for the fuel/air ratio 0.5, 1.0 and 2.0. The agreement between the two curves corresponding to the 1:st ignition is very good for all fuel/air ratios. The figure also shows that the 1:st and 2:nd ignition coincide for low temperatures.
The agreement between the original and reduced mechanism is better than the agreement between the original mechanism and the experimental data if all points are considered for both IDT HF and IDT CF. This gives an argument for the use of a less accurate reduced mechanism, so that the deviation between the original and reduced mechanism is of the same order as the deviation between the original mechanism and the experimental results. The accuracy of the reduced mechanism can be decreased by using more generous ART ET limits.

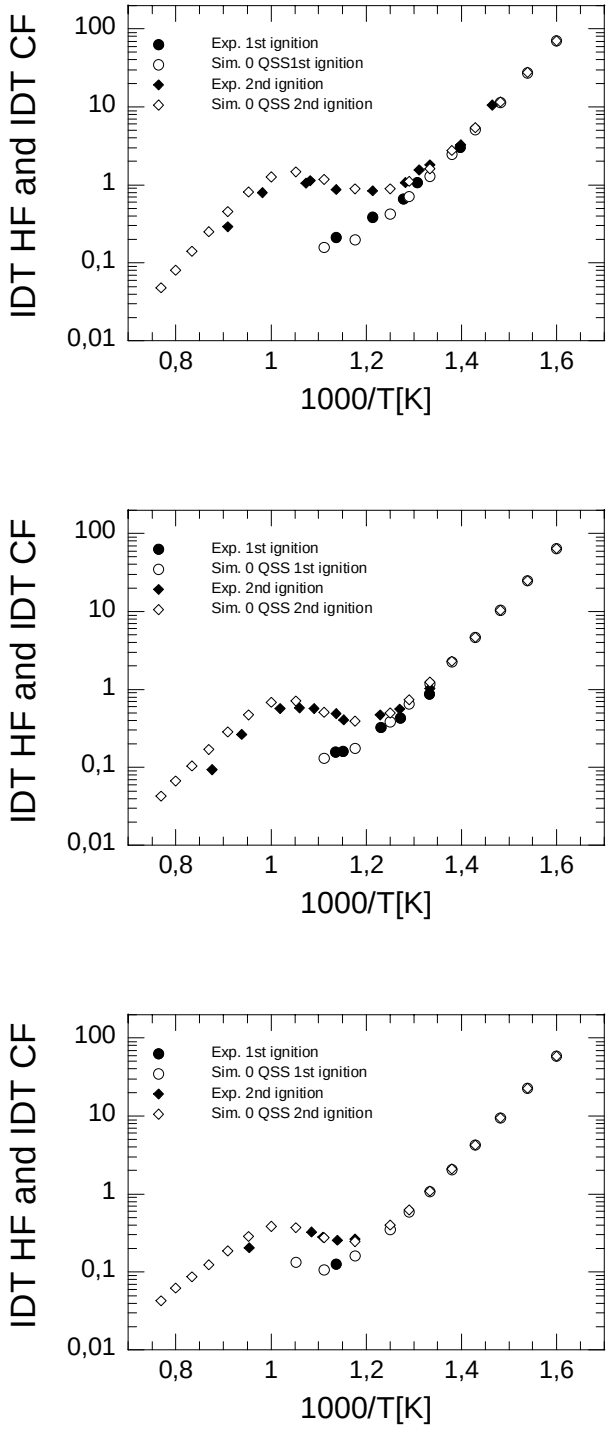


Figure 6.106. Shows IDT HF and IDT CF vs 1000/T for experiments and simulation with 0 QSS species. 1:st and 2:nd ignition correspond to IDT CF and IDT HF respectively. The top, middle and lower sub figure corresponds to fuel/air ratio 0.5, 1.0 and 2.0 respectively. The initial pressure is 40 bar for all subfigures.

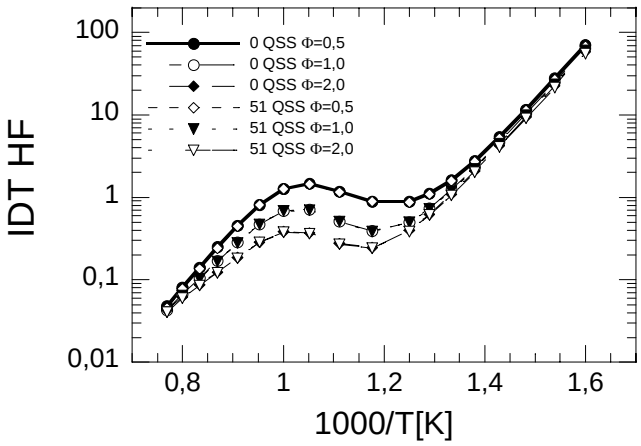


Figure 6.107. IDT HF vs 1000/T[K] for the original and most reduced mechanism containing 0 and 47 QSS species respectively. The temperature range is 625-1300 K, the fuel/air ratio is 0.5, 1.0 and 2.0 and the pressure is 40 bar.

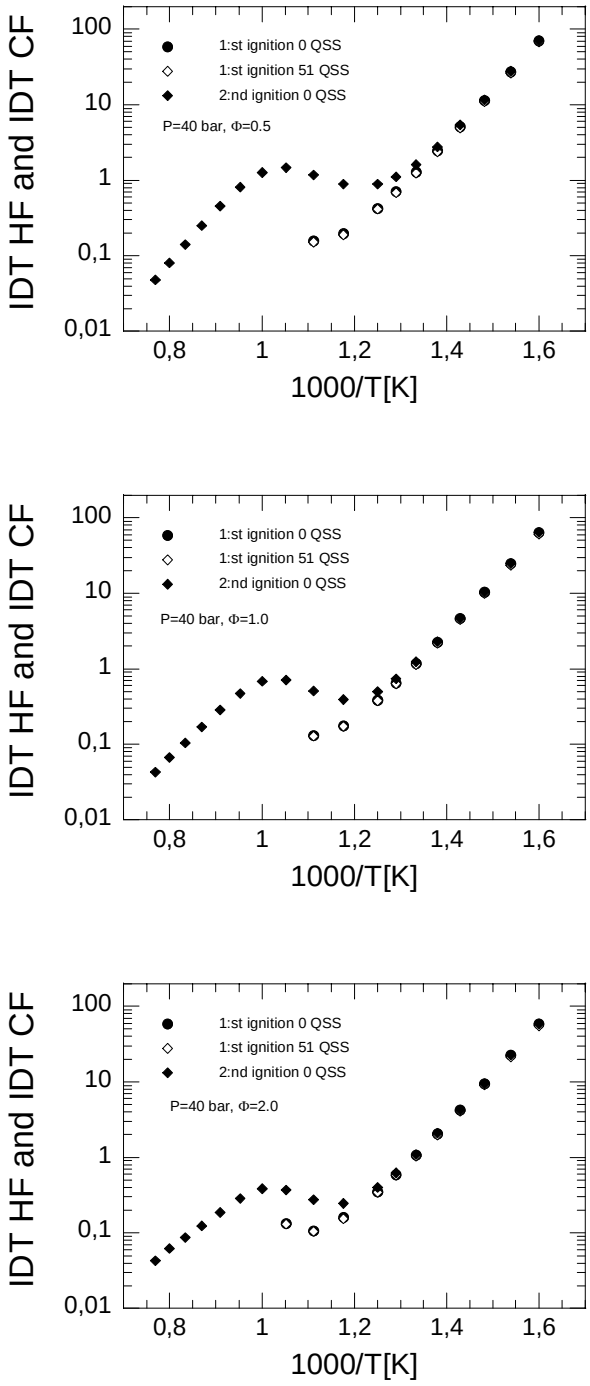


Figure 6.108. IDT HF and IDT CF vs 1000/T[K]. The figure shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism, which contains 0 and 51 QSS species respectively. The temperature range is 625-1300 K, the pressure is 40 bar and the fuel/air ratio is 0.5, 1.0 and 2.0 for the upper, middle and lower subfigure respectively. The 1:st and 2:nd ignition coincide for low temperatures.

6.2.3.3 Case 4: Temperature, Fuel/air ratio and pressure range

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature range: 18 temperature points between 625 and 1300 K
- Pressure range: 13,5 and 40 bar
- Fuel/air ratio range: 0.5, 1.0 and 2.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 2 %

6.2.3.3.1 Reduction level of the reduced mechanisms

The reduced mechanism, which is shown in Figure 6.109, contains 47 QSS species and is identical and valid for all initial temperatures, all initial pressures and all fuel/air ratios. The figure also shows the importance of the “Second Chance” option in the ART, since four QSS species were accepted the second time they were tested. Hence, the “Second chance” option increases the reduction from 43 to 47 QSS species. The most reduced mechanism of case 4 contains less species than the most reduced mechanism case 3. This is expected since some species are very important for low pressures and therefore cannot be used as a QSS species, while other species are very important for high pressures and therefore cannot be used as a QSS species.

Table 6.9 shows the species that were accepted and those that failed to be accepted as QSS species for all four cases. Hence, the table shows the species that are important for different physical conditions.

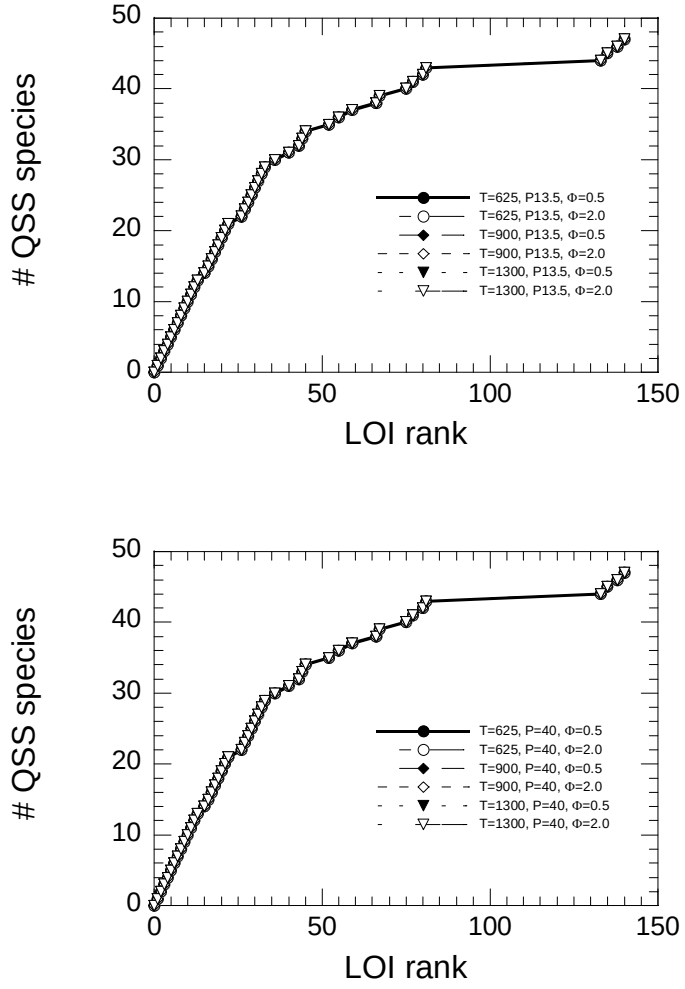


Figure 6.109. Number of QSS species vs LOI rank when the entire temperature range T=625-1300 K and the fuel/air ratios; 0.5, 1.0 and 2.0 are considered. The upper and lower sub figure corresponds to the initial pressure of 13,5 and 40 bar respectively. Six key combinations of physical conditions are shown in each sub figure. The two sub figures are identical, since the same mechanism must be valid for all physical conditions.

Table 6.9. The table shows the species that were accepted and the species that failed to be accepted for the four different cases. “F” means that the species failed, a blank means the species was accepted.

Nr	Species	LOI rank	Case 1	Case 2	Case 3	Case 4
1	CH2CH2COCH3	0.57992E-23				
2	C2H5COCH2	0.81611E-23				
3	CH2CH2CHO	0.30747E-22				
4	CH3CHCOCH3	0.67565E-22				
5	N-C3H7CO	0.29803E-21				
6	C2H5O	0.30558E-20				
7	5R-HEOOH-P	0.10122E-19				
8	HOCH2O	0.16819E-19				
9	N-C3H7COC2H4-1	0.10651E-17				
10	C2H5CO	0.11704E-17				
11	7R-HEOOH-P	0.24758E-17				
12	6R-HEOOH-P	0.36896E-17				
13	C2H4O2H	0.39947E-17				
14	5R-HEOOHO2-P	0.86579E-17		F	F	F
15	7R-HEOOH-S	0.18380E-16				
16	1-C5H11	0.19356E-16				
17	2-C5H11	0.25297E-16				
18	5R-HEOOH-S	0.35434E-16				
19	N-C3H7	0.50926E-16				
20	CH3COCH2	0.58997E-16				
21	1-C4H9	0.66697E-16				
22	6R-HEOOH-S	0.10622E-15				
23	CH	0.18024E-15				F
24	C6H10	0.21326E-15		F	F	F
25	7R-O-HEPOOH-P	0.44094E-15		F	F	F
26	1-C2H4COC2H5	0.71797E-15				
27	I-C3H7	0.99270E-15				
28	CH3O	0.10338E-14				
29	6R-HEOOHO2-P	0.16628E-14				
30	C2H	0.18632E-14				
31	HO2CHO	0.19006E-14				
32	1-CH2	0.25684E-14				
33	L-C7H15	0.26253E-14				
34	7R-HEOOHO2-P	0.27514E-14		F	F	F
35	OCHO	0.35343E-14			F	F
36	2-C4H8	0.36433E-14				
37	C6H5O	0.50284E-14		F	F	F
38	7R-O-HEPOOH-S	0.71393E-14		F	F	F
39	C2H5O2	0.10700E-13			F	F
40	C4H7	0.11039E-13				
41	7R-HEOOHO2-S	0.12656E-13		F	F	F
42	CH3O2H	0.14152E-13		F	F	F
43	C7H13	0.16824E-13				
44	A1-	0.18662E-13				
45	C6H8	0.24974E-13				
46	5R-HEOOHO2-S	0.25519E-13		F	F	F
47	6R-O-HEPOOH-P	0.33649E-13	F	F	F	F
48	C4H612	0.49074E-13				F
49	6R-HEOOHO2-S	0.58707E-13	F	F	F	F
50	CH3O2	0.70054E-13	F	F	F	F
51	5R-O-HEPOOH-S	0.71260E-13	F	F	F	F
52	N-C4H3	0.75976E-13				
53	A1	0.86912E-13			F	F
54	O2CHO	0.87279E-13	F	F	F	F
55	C4H10	0.11237E-12				
56	C5H9	0.14757E-12				F
57	N-C4H9COCH2	0.17086E-12		F	F	F
58	6R-O-HEPOOH-S	0.22825E-12	F	F	F	F
59	N-C4H5	0.23363E-12				

60	C2H5O2H	0.27242E-12	F	F	F	F
61	C2H6	0.33900E-12				F
62	I-C4H3	0.45295E-12				
63	HCO	0.46041E-12				F
64	N-C3H7COCH2	0.65550E-12	F	F	F	F
65	L-C7H15O2	0.66422E-12	F	F	F	F
66	HCCO	0.68469E-12				
67	3-CH2	0.14667E-11				
68	I-C4H5	0.17894E-11				
69	CH3OH	0.34782E-11	F	F	F	F
70	CH2CHO	0.36697E-11			F	F
71	HOCHO	0.49648E-11			F	F
72	C4H4	0.62090E-11			F	F
73	C4H6	0.63781E-11		F	F	F
74	1-C4H8	0.81147E-11	F	F	F	F
75	C3H8	0.98504E-11			F	
76	5R-C7H14O	0.98797E-11	F	F	F	F
77	CH2OH	0.16731E-10				
78	C2H5	0.23923E-10		F	F	F
79	C3H6	0.27350E-10	F	F	F	F
80	C2H3	0.28788E-10				
81	CH3CO	0.40888E-10				
82	1-C5H10	0.42325E-10	F	F	F	F
83	CH3CHO	0.47077E-10	F	F	F	F
84	C3H5	0.53180E-10				F
85	N-C3H7CHO	0.60902E-10	F	F	F	F
86	H2O2	0.78482E-10	F	F	F	F
87	C2H5CHO	0.87104E-10	F	F	F	F
88	C3H4P	0.11307E-09	F	F	F	F
89	C3H4	0.13220E-09		F	F	F
90	C3H3	0.18911E-09			F	F
91	L-C7H14	0.21697E-09	F	F	F	F
92	C2H2	0.23131E-09			F	F
93	HO2	0.25436E-09	F	F	F	F
94	O	0.74893E-09	F	F	F	F
95	CH2CO	0.83539E-09	F	F	F	F
96	CH2O	0.95470E-09	F	F	F	F
97	C2H5COCH3	0.14892E-08	F	F	F	F
98	CH3	0.35709E-08			F	F
99	CH4	0.64018E-08	F	F	F	F
100	C2H4	0.65893E-08	F	F	F	F
101	OH	0.30677E-07	F	F	F	F
102	CO2	0.92212E-07	F	F	F	F
103	H	0.10853E-06	F	F	F	F
104	N-C7H16	0.26023E-06	F	F	F	F
105	H2	0.27517E-06	F	F	F	F
106	CO	0.31018E-06	F	F	F	F
107	H2O	0.49953E-04	F	F	F	F
108	O2	0.62892E-04	F	F	F	F
109	N2	0.10000E+01	F	F	F	F
110	AR	0.10000E+01	F	F	F	F

6.2.3.3.2 CPU time of the reduced mechanisms

The normalized CPU time vs number of QSS species for some key combinations of physical conditions is shown in Figure 6.110. The normalized CPU time of the reduced mechanisms varies somewhat for different physical conditions, but the trend is similar for all initial temperatures, fuel/air ratios and initial pressures. The variation can partly be explained by CPU noise due to other processes in the computer. However, the general explanation for the variation is that each physical condition corresponds to a unique trajectory in species concentration space and that the convergence of the solver combination, and thereby CPU time, is dependent on the trajectory. The amount of CPU time that can be gained by applying the QSSA to a particular species is also trajectory dependent. Hence, a variation in CPU time exists among the physical conditions for different reduction levels.

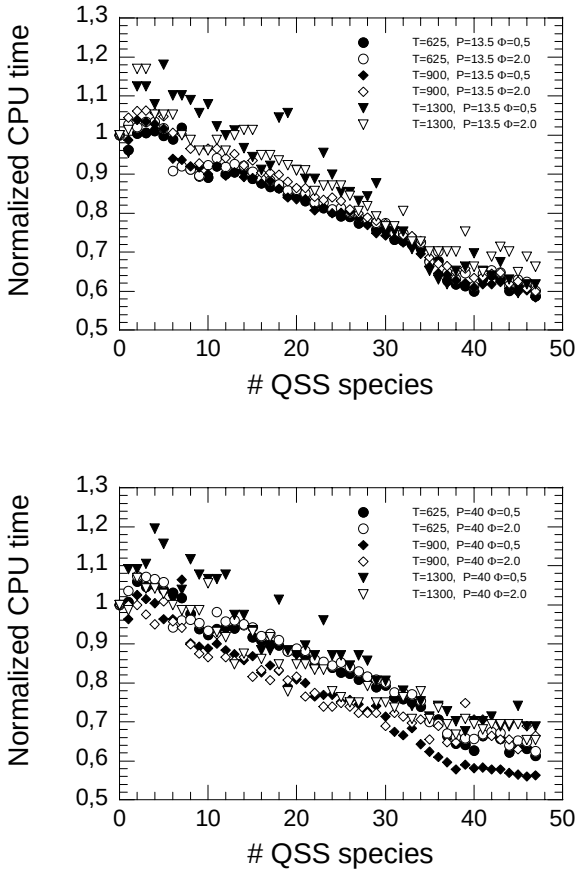


Figure 6.110. Normalized CPU time vs number of QSS species when the entire temperature range T=625-1300 K and the fuel/air ratios; 0.5, 1.0 and 2.0 are considered. The upper and lower sub figure corresponds to the initial pressure of 13,5 and 40 bar respectively. Six key combinations of physical conditions are shown in each sub figure.

6.2.3.3.3 Accuracy of the reduced mechanisms

Figure 6.111 shows IDT HF and IDT CF vs 1000/T for experiments [2] and simulation with 0 QSS species (original mechanism) for the fuel/air ratio 0.5, 1.0 and 2.0 and the pressure 13.5 bar (the experimental results for IDT CF for fuel/air ratio 0,5 was not accessible). The corresponding figure for 40 bar is shown for Case 3. The curves for the experiment and simulation have good agreement for IDT CF and IDT HF for all fuel/air ratios most initial temperatures.

Figure 6.112 shows the IDT HF vs 1000/T for the original mechanism and the most reduced mechanism containing 47 QSS species for the fuel/air ratio 0.5, 1.0 and 2.0 and the pressures 40 and 13.5 bar . The figure shows that the agreement between the two curves is very good for all initial temperatures, all fuel/air ratios and both pressures.

Figure 6.113 and 6.114 show the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism for the fuel/air ratio 0.5, 1.0 and 2.0 and the pressures 40 and 13.5 bar. The agreement between the two curves corresponding to the 1:st ignition is very good for all fuel/air ratios and both pressures. The figures also shows that the 1:st and 2:nd ignition coincide for very low and high temperatures.

The agreement between the original and reduced mechanism is better than the agreement between the original mechanism and the experimental data if all points are considered for both IDT HF and IDT CF. This gives an argument for the use of a less accurate reduced mechanism, so that the deviation between the original and reduced mechanism is of the same order as the deviation between the original mechanism and the experimental results. The accuracy of the reduced mechanism can be decreased by using more generous ART ET limits.

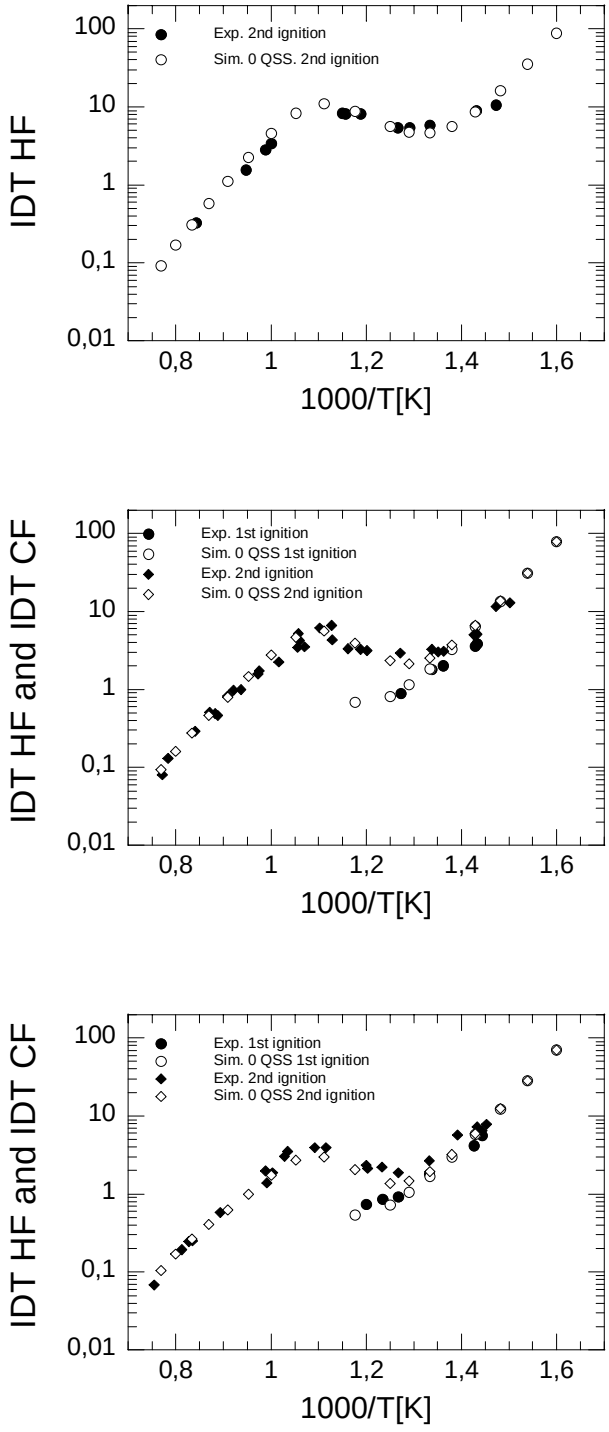


Figure 6.111. Shows IDT HF and IDT CF vs 1000/T for experiments and simulation with 0 QSS species. 1:st and 2:nd ignition corresponds to IDT CF and IDT HF respectively. The top, middle and lower sub figure corresponds to fuel/air ratio 0.5, 1.0 and 2.0 respectively. The initial pressure is 13,5 bar for all subfigures.

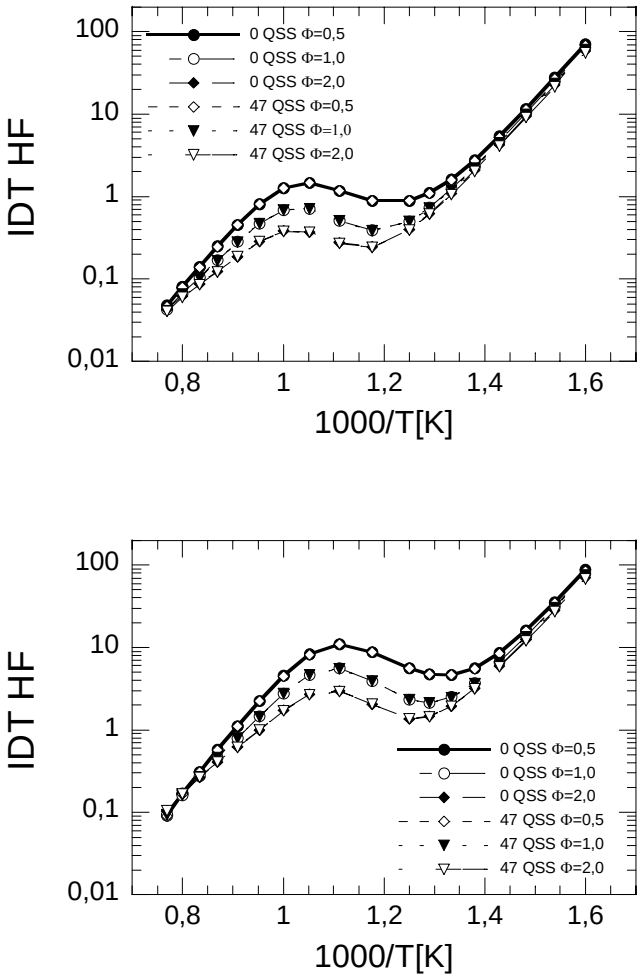


Figure 6.112. IDT HF vs 1000/T[K] for the original and most reduced mechanism containing 0 and 47 QSS species respectively. The temperature range is 625-1300 K, the fuel/air ratio is 0.5, 1.0 and 2.0 for both figures and the pressure is 13.5 and 40 bar for the upper and lower subfigure respectively.

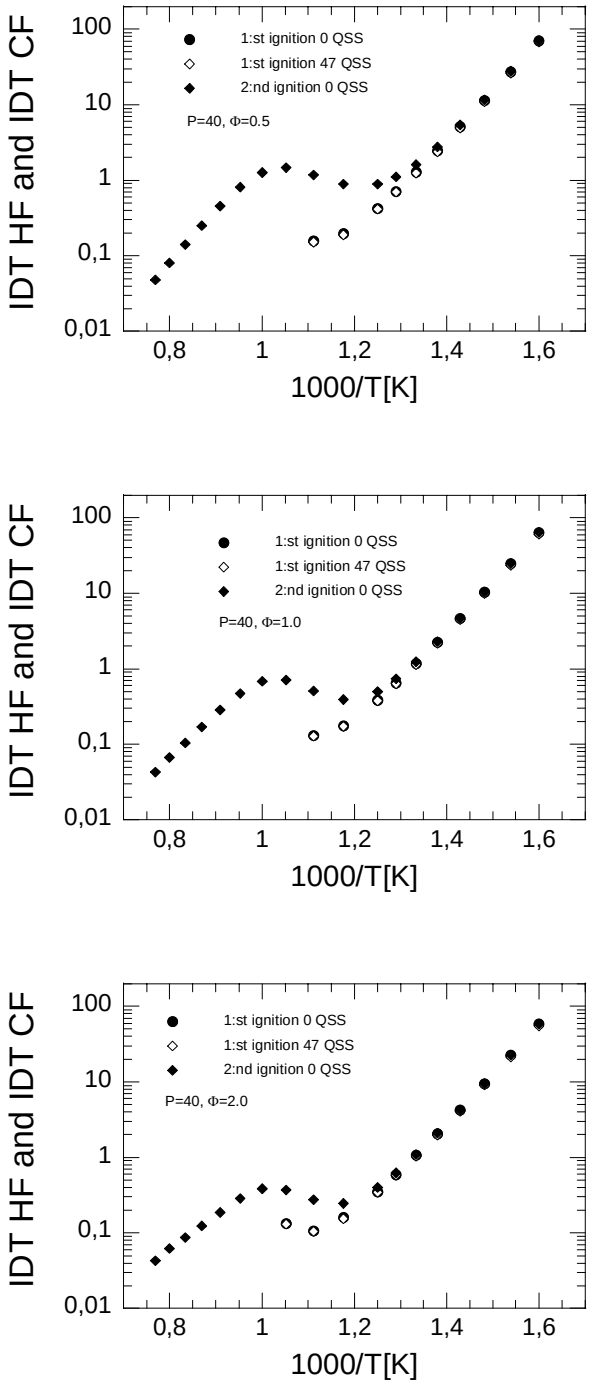


Figure 6.113. IDT HF and IDT CF vs 1000/T[K]. The figure shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism, which contains 0 and 47 QSS species respectively. The temperature range is 625-1300 K, the pressure is 40 bar and the fuel/air ratio is 0.5, 1.0 and 2.0 for the upper, middle and lower subfigure respectively. The 1:st and 2:nd ignition coincide for low temperatures.

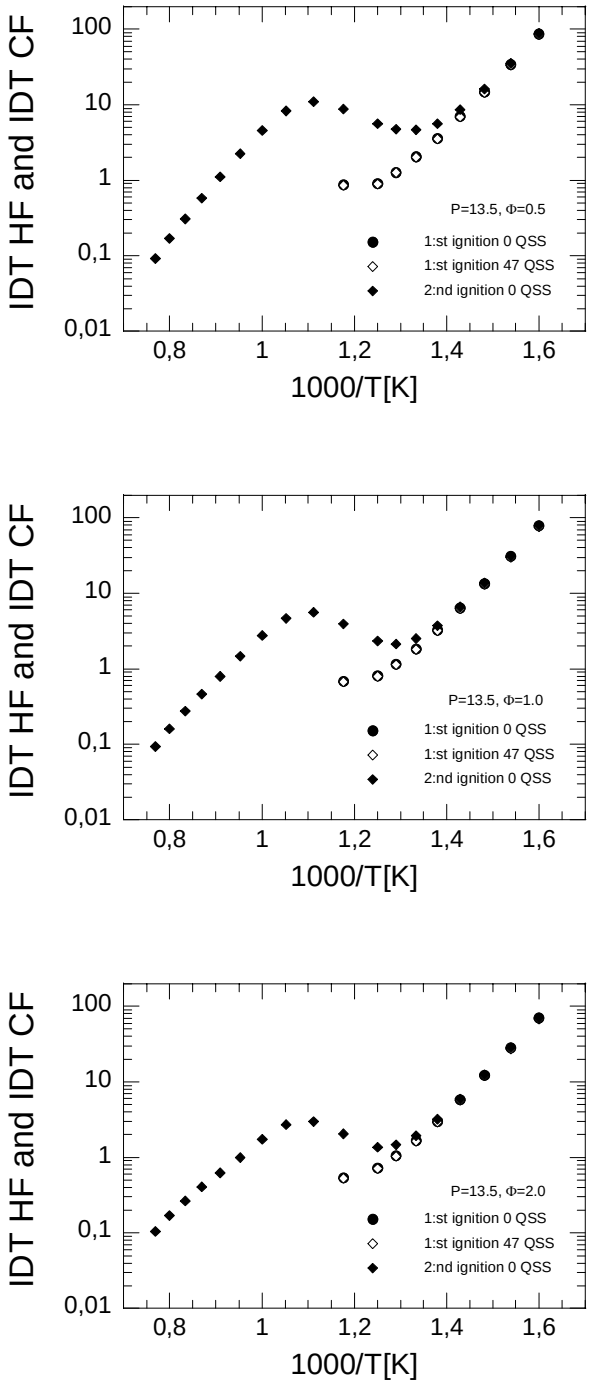


Figure 6.114. IDT HF and IDT CF vs 1000/T[K]. The figure shows the 2:nd ignition for the original mechanism and the 1:st ignition for the original and the most reduced mechanism, which contains 0 and 47 QSS species respectively. The temperature range is 625-1300 K, the pressure is 13.5 bar and the fuel/air ratio is 0.5, 1.0 and 2.0 for the upper, middle and lower subfigure respectively. The 1:st and 2:nd ignition coincide for low temperatures.

6.2.4. CFD application

The performances of the Newton-Newton and Newton-FP solver combinations have been tested in interaction with a commercial STAR-CD CFD code.
The solver combinations were tested on a "real" test case, i.e. a N-heptane injection into a constant volume vessel. However, each grid point in the CFD code interacted with a CPR, since STAR-CD useds enthalpy form of the energy equation.
A Newton solver with the full original mechanism, with 0 QSS species, was used as a reference simulation. Thereafter a reduced mechanism of 50 QSS was tested with both the Newton-Newton and the Newton-FP solver combinations. The reduced mechanism is the same as the optimized mechanism for case 3 in section 6.2.3.2.

Simulation parameters

- **CFD model**
  - The simulation was a RANS simulation with a standard k-ε turbulence model with default STAR-CD parameters.
  - A chemico-thermal enthalpy form of the energy equation was used.
  - The built in spray models in STAR-CD was used.
    - Droplet Break up was modeled with the Reitz model.
    - Atomization was modeled with the Huh model.
  - Ideal gas was used for calculation of the density.
  - The fuel was injected though a 0,14 mm diameter nozzle hole in the upper left corner of the grid.
  - The fuel was liquid form of N-Heptane at temperature 333 K.
  - The injection rate was constant 3 grams per second.
  - The total simulation time was 2 msec and the injection duration was 1.5 msec.
  - A 6 degree sector with 4000 cells was used. The mesh size was 80*50 mm with an edge size of 0.7 mm close to the nozzle. The cell size increased with a growth factor of 1.01 away from the nozzle. This is illustrated in Figure 6.115.
- **Chemistry model**
  - Each cell used a CPR reactor for calculation of the chemistry.
  - Simulations were performed with either the Newton-Newton solver or the Newton-FP solver in each CPR.
  - Simulations were performed with either a detailed chemistry or a reduced chemistry with 50 QSS species.
- **CFD/chemistry coupling model**
  - An operator splitting method was used.
    - The CFD solver transported the scalars in the flow field and the chemistry solver was called in the beginning at each time step to solve the chemistry
- **Physical conditions**
  - The initial pressure was 40 bar.
  - The initial temperature was 1000 K.

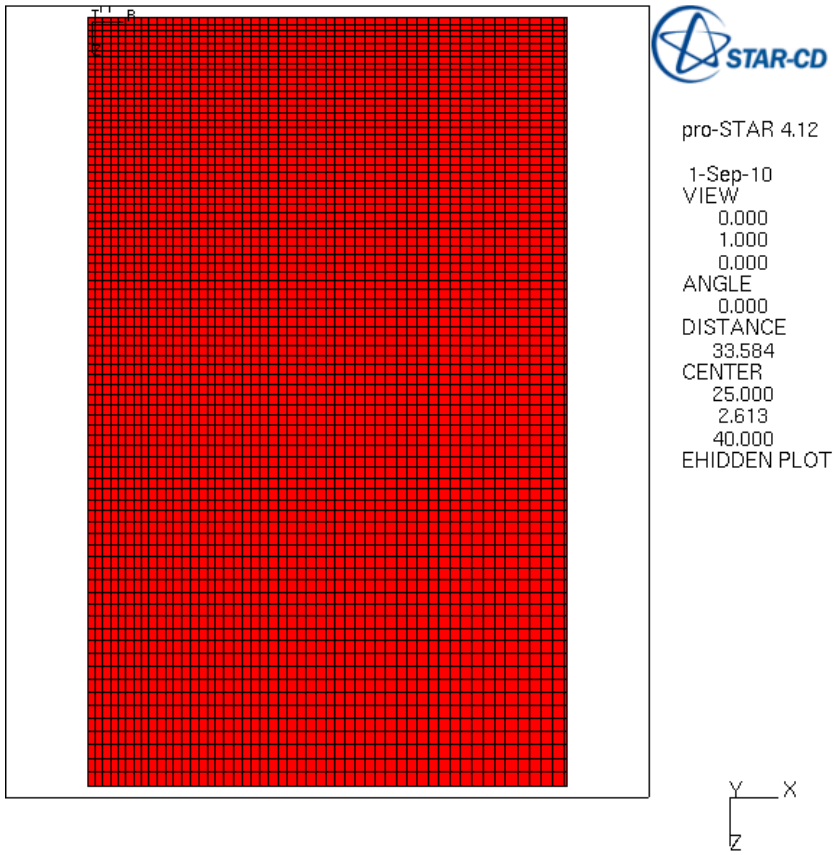


Figure 6.115. An illustration by STAR-CD of the grid used in the simulations. The cell size increases from the nozzle in the upper left corner.

6.2.4.1 CPU time of the simulation

The top sub figure in Figure 6.116 shows $CPU_{CHEM+CFD}$, CPU_{CHEM} and CPU_{CFD} for the CFD simulation. Solver combination 1, 2 and 3 corresponds to Newton solver with 0 QSS, Newton-Newton solver with 50 QSS and Newton-FP solver with 50 QSS respectively. The total CPU time for the Newton solver with 0 QSS, the Newton-Newton solver with 50 QSS and the Newton-FP solver with 50 QSS is about 30000, 20000 and 1020000 seconds respectively, which shows the time scale for the CFD application.
It is clear that CPU_{CHEM} represents the majority of the $CPU_{CHEM+CFD}$ for all solver combinations. This shows the importance of a fast solver combination for the system of chemical DAE in the CPR.
The lower sub figure shows normalized version of the top sub figure, where the normalization is for the total time for 0 QSS. This shows that the normalized CPU time is about 0,66 and 34 for the Newton-Newton and Newton-FP solver respectively. Hence, the Newton-Newton solver is much faster Newton-FP solver when used in the CFD application. This is similar to the results for a 0-D CVR in section 6.2.1.1.

The top sub figure in Figure 6.117 shows the ratio between the Newton-Newton solver with 50 QSS and the Newton solver with 0 QSS for $CPU_{CHEM+CFD}$, CPU_{CHEM} and CPU_{CFD} for the CFD simulation. The ratio of the $CPU_{CHEM+CFD}$, CPU_{CHEM} and CPU_{CFD} is 0.66, 0.68 and 0.52 respectively. This means that the CPU_{CFD} decreased linearly as expected.
The CPU_{CHEM} was a little higher than expected from the CVR in section 6.2.1.1. The probable explanation for this is that the reduced mechanism was validated only for fuel/air ratios 0.5 to 2.0, while the CFD application has much wider range of fuel/air ratios. Hence, an optimization for a wider set of fuel/air ratios could be done in order to remove some QSS species from the reduced mechanism and thereby reach better convergence and optimum CPU gain.

If the inner solver was infinitely fast the ratio between the reduced and original mechanism for CPU_{CHEM} would be about 0,3 for 50 QSS species (using the same reasoning as in section 6.2.1.3. And since CPU_{CHEM} is the largest part of the $CPU_{CHEM+CFD}$, one would expect the ratio of $CPU_{CHEM+CFD}$ to be about the same. This shows an upper limit of CPU time that can be gained by using a reduced mechanism of 50 QSS species in the particular CFD application. If the performance of the Newton-Newton solver is seen from this perspective it can be considered as quite a good performance, since only about a factor two can be won at maximum.

The lower sub figure of Figure 6.117 shows the ratio between the Newton-FP solver with 50 QSS and the Newton-Newton solver with 50 QSS for $CPU_{CHEM+CFD}$, CPU_{CHEM} and CPU_{CFD} for the CFD simulation. The ratio of the $CPU_{CHEM+CFD}$, CPU_{CHEM} and CPU_{CFD} is 51, 57 and 0.97 respectively. This means that the Newton-Newton solver performs about 50 times better than the Newton-FP solver for the CFD application. The slightly faster CPU_{CFD} for the Newton-FP solver is probably due to noise.

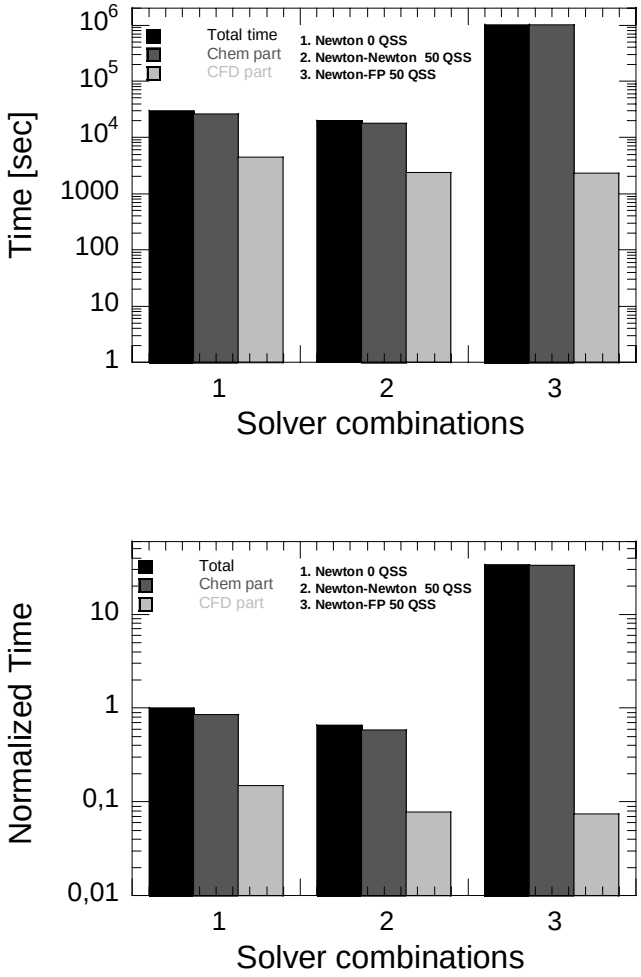


Figure 6.116. The top sub figure shows $CPU_{CHEM+CFD}$ (black bar), CPU_{CHEM} (dark grey bar) and CPU_{CFD} (light grey bar) for the CFD simulation. Solver combination 1, 2 and 3 corresponds to Newton solver with 0 QSS, Newton-Newton solver with 50 QSS and Newton-FP solver with 50 QSS respectively. The lower sub figure shows normalized version of the top sub figure, where the normalization is for the total time for 0 QSS.

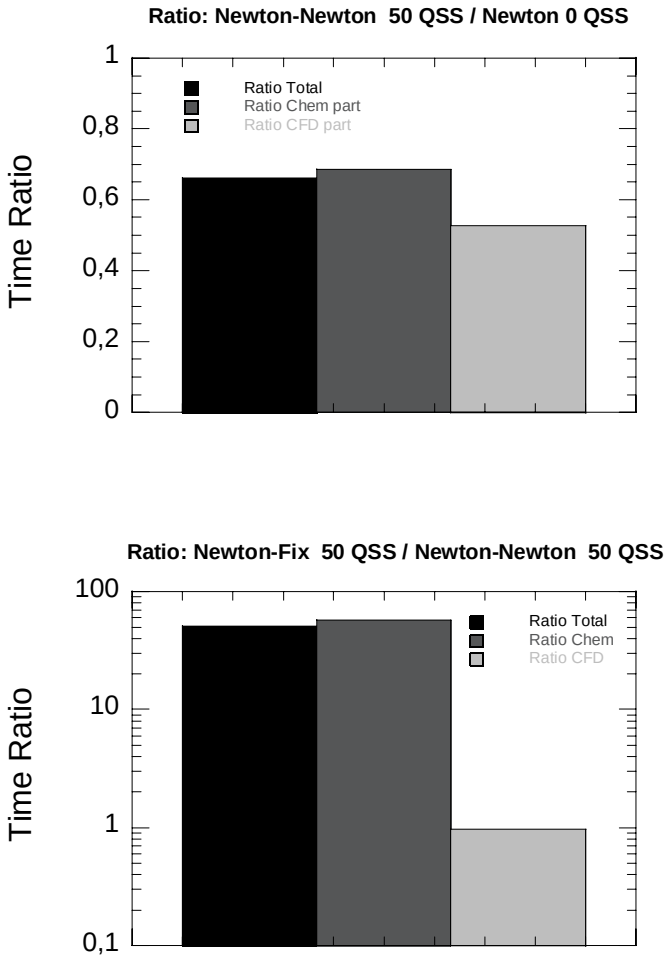


Figure 6.117. The top sub figure shows the ratio between the Newton-Newton solver with 50 QSS and the Newton solver with 0 QSS for $CPU_{CHEM+CFD}$ (black bar), CPU_{CHEM} (dark grey bar) and CPU_{CFD} (light grey bar) for the CFD simulation. The lower sub figure shows the ratio between the Newton-FP solver with 50 QSS and the Newton-Newton solver with 50 QSS for $CPU_{CHEM+CFD}$ (black bar), CPU_{CHEM} (dark grey bar) and CPU_{CFD} (light grey bar) for the CFD simulation.

6.2.4.2 Accuracy of the simulation

The accuracy of the simulation with 50 QSS must not deviate much compared to the reference simulation with 0 QSS if the CPU gain from using the reduced mechanism should be considered as valid. The accuracy is validated by observing the temperature and important species concentrations over time.
The temperature, OH, HO_2, CH_2O, H_2, C_2H_2, C_2H_4, CH_4, N-C_7H_{16}, O_2, H_2O, CO, and CO_2 have been investigated for Newton-Newton solver with 50 QSS and 0 QSS at different points in time. The main observation from the investigation is that the original and reduced mechanisms are almost identical for all investigated species and all time points. Hence, the reduced mechanism has very high accuracy.

However, only a part of the investigated species is shown in this thesis. Figure 6.118 to 6.121 shows temperature, OH, HO_2 and CH_2O at six chosen points in time.

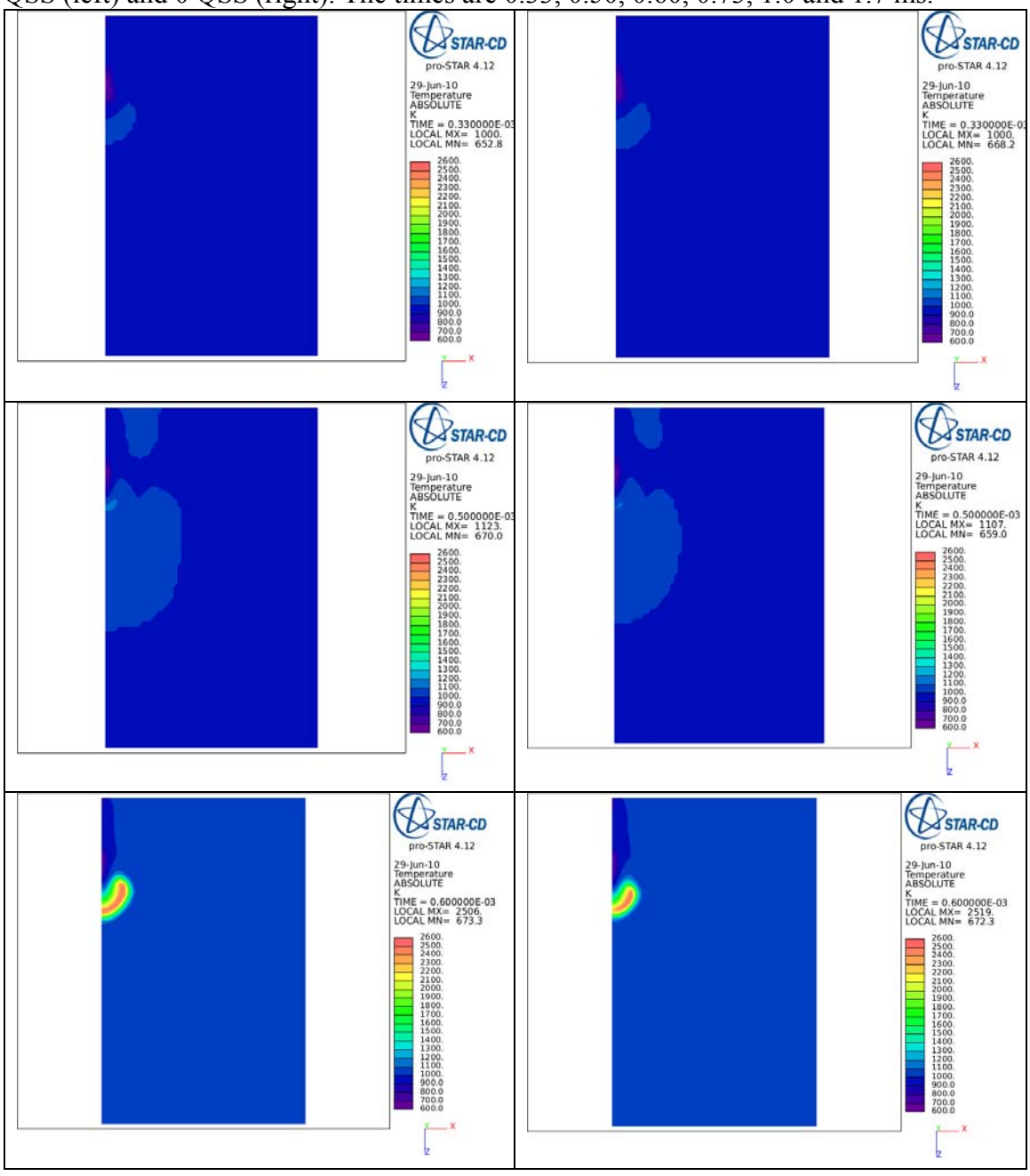
The OH concentration and temperature follow each other from 0.60 ms, which is expected since OH is used as an indicator for the HF. The small difference between the original and reduced mechanism at 0.60 ms is due to a slight difference in IDT HF.

The HO_2 concentration is large and the temperature shows a small increase at early time points, which is an indicator for the CF. At later time points (0.60, 0.73 and 1.0 ms) the HO_2 concentration is large just outside the HF region. This is expected since the region outside of the HF region has a lower temperature and therefore contains lower temperature reactions.

The CH_2O concentration is also large at early time points and surrounds the HF region at later time points. This is also expected since CH_2O participates in lower temperature chemistry.

The other investigated species, which are shown in the chapter appendix, show similar behavior for the original and reduced mechanism for all time points.

Figure 6.118. Temperature at four different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



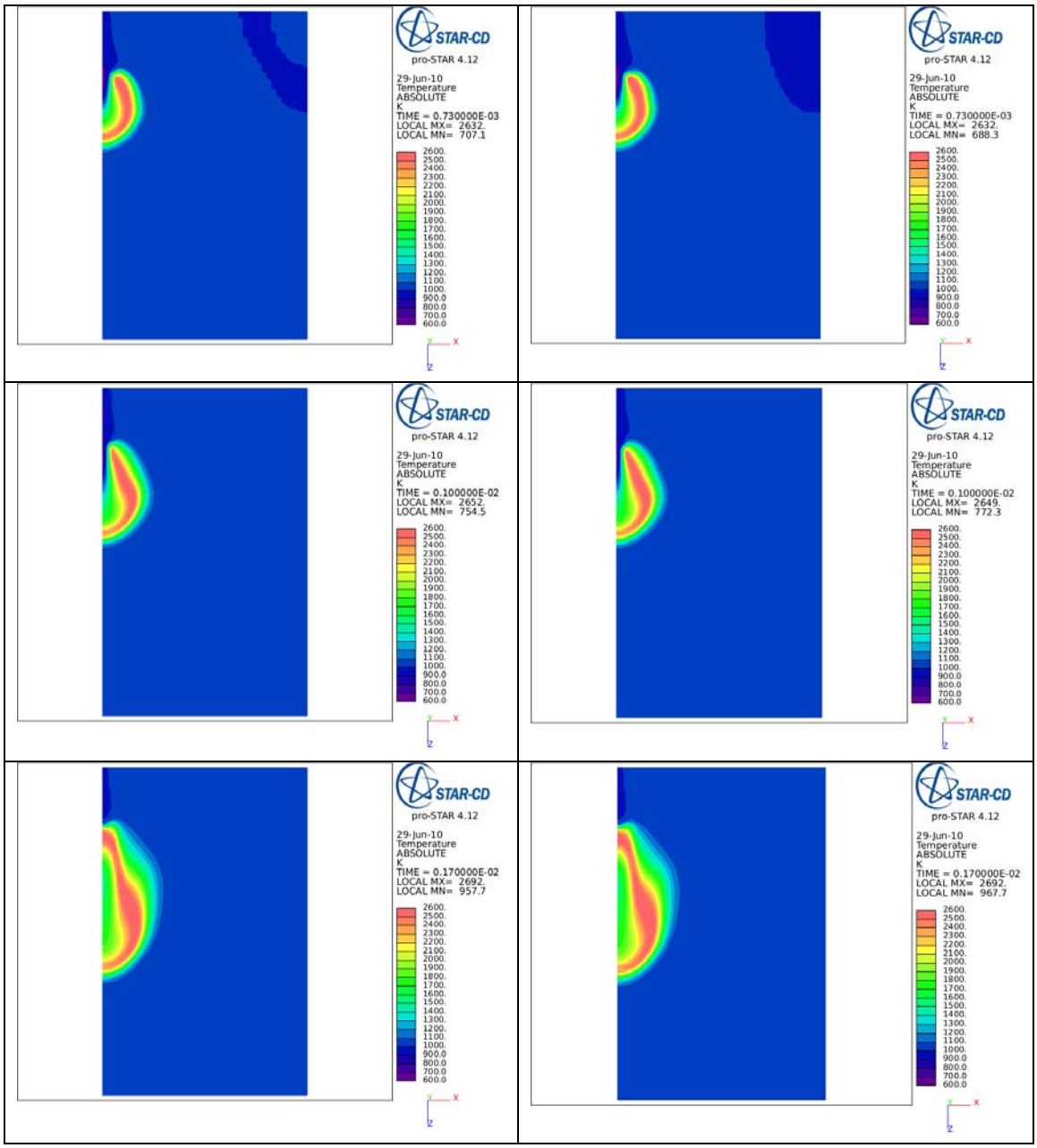
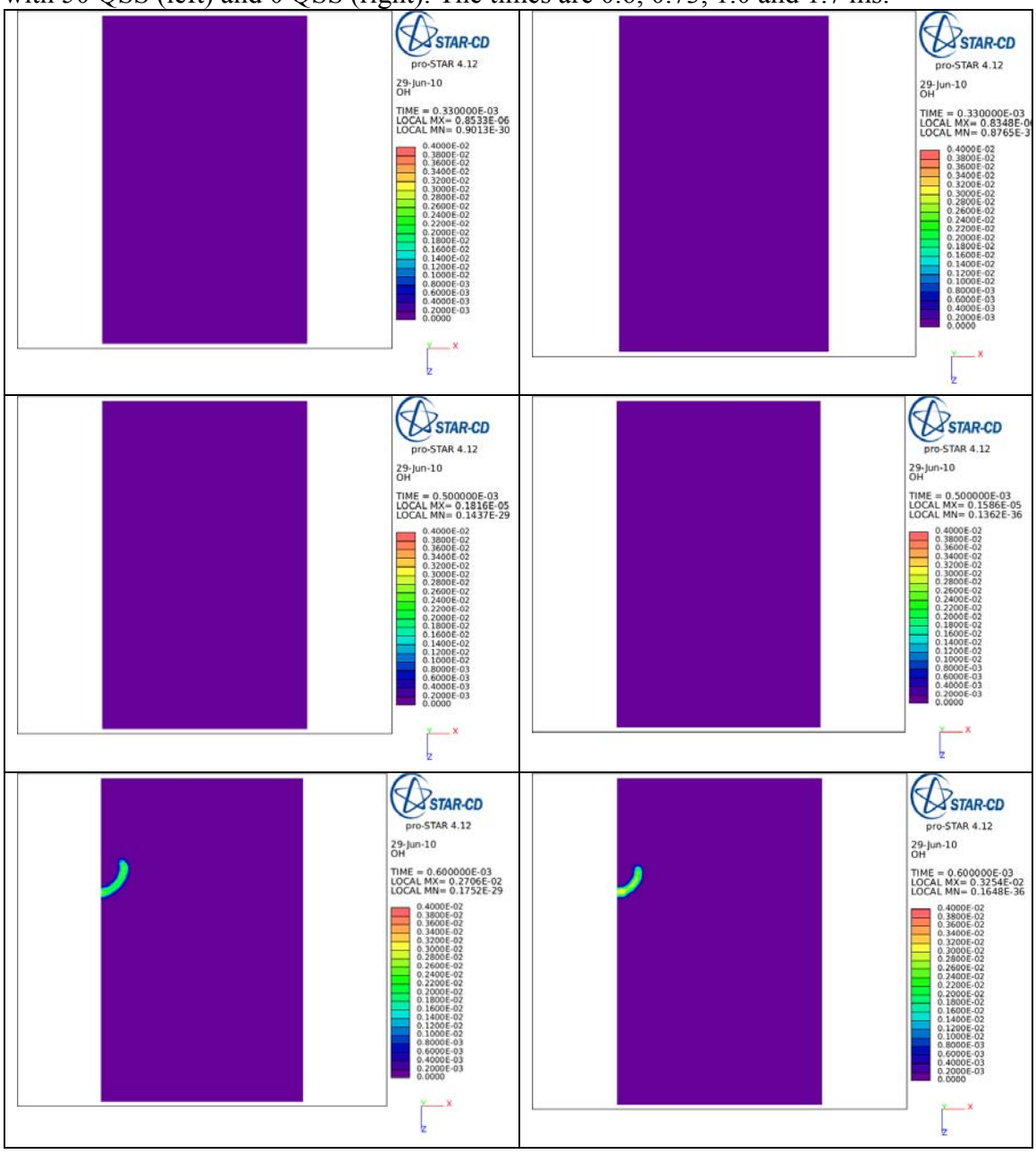

STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.730000E-03
LOCAL MX= 2632.
LOCAL MN= 707.1
STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.730000E-03
LOCAL MX= 2632.
LOCAL MN= 688.3
STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.100000E-02
LOCAL MX= 2652.
LOCAL MN= 754.5
STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.100000E-02
LOCAL MX= 2649.
LOCAL MN= 772.3
STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.170000E-02
LOCAL MX= 2692.
LOCAL MN= 957.7
STAR-CD
pro-STAR 4.12
29-Jun-10
Temperature
ABSOLUTE
K
TIME = 0.170000E-02
LOCAL MX= 2692.
LOCAL MN= 967.7

Figure 6.119. Concentration for OH at four different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.6, 0.73, 1.0 and 1.7 ms.



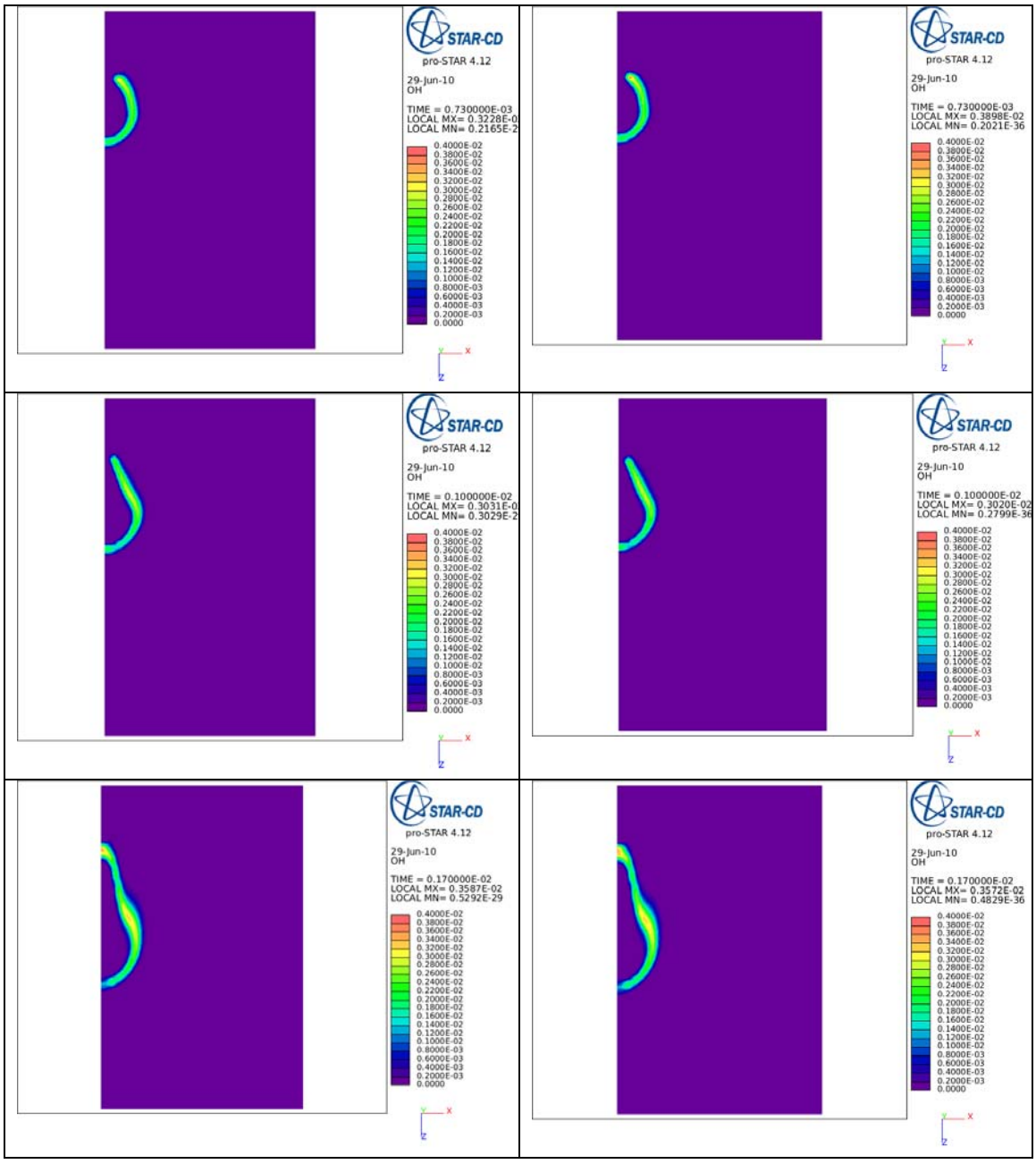
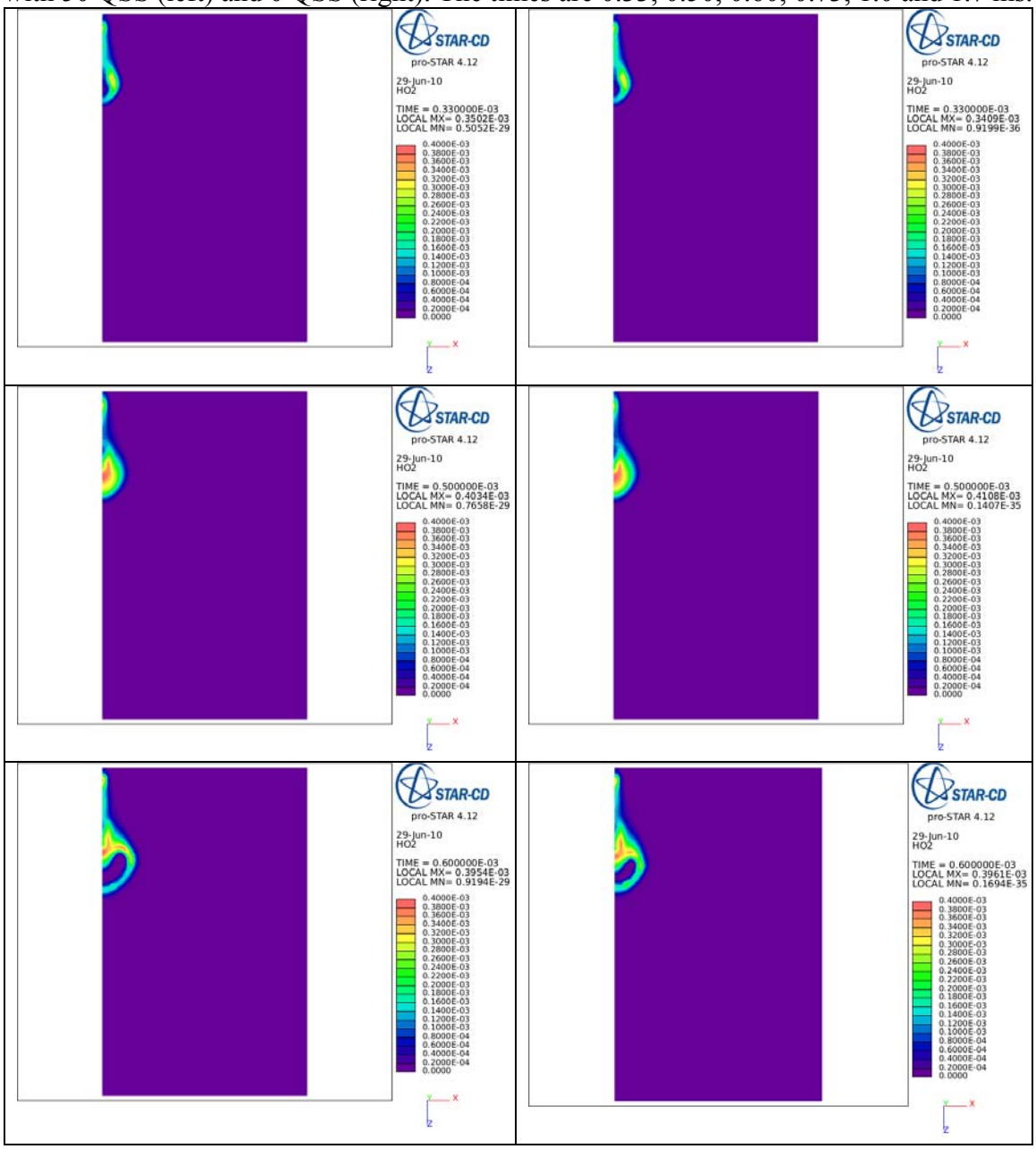

STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.730000E-03
LOCAL MX= 0.3228E-0
LOCAL MN= 0.2165E-2
STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.730000E-03
LOCAL MX= 0.3898E-02
LOCAL MN= 0.2021E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.100000E-02
LOCAL MX= 0.3031E-0
LOCAL MN= 0.3029E-2
STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.100000E-02
LOCAL MX= 0.3020E-02
LOCAL MN= 0.2799E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.170000E-02
LOCAL MX= 0.3587E-02
LOCAL MN= 0.5292E-29
STAR-CD
pro-STAR 4.12
29-Jun-10
OH
TIME = 0.170000E-02
LOCAL MX= 0.3572E-02
LOCAL MN= 0.4829E-36

Figure 6.120. Concentration of HO_2 at four different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



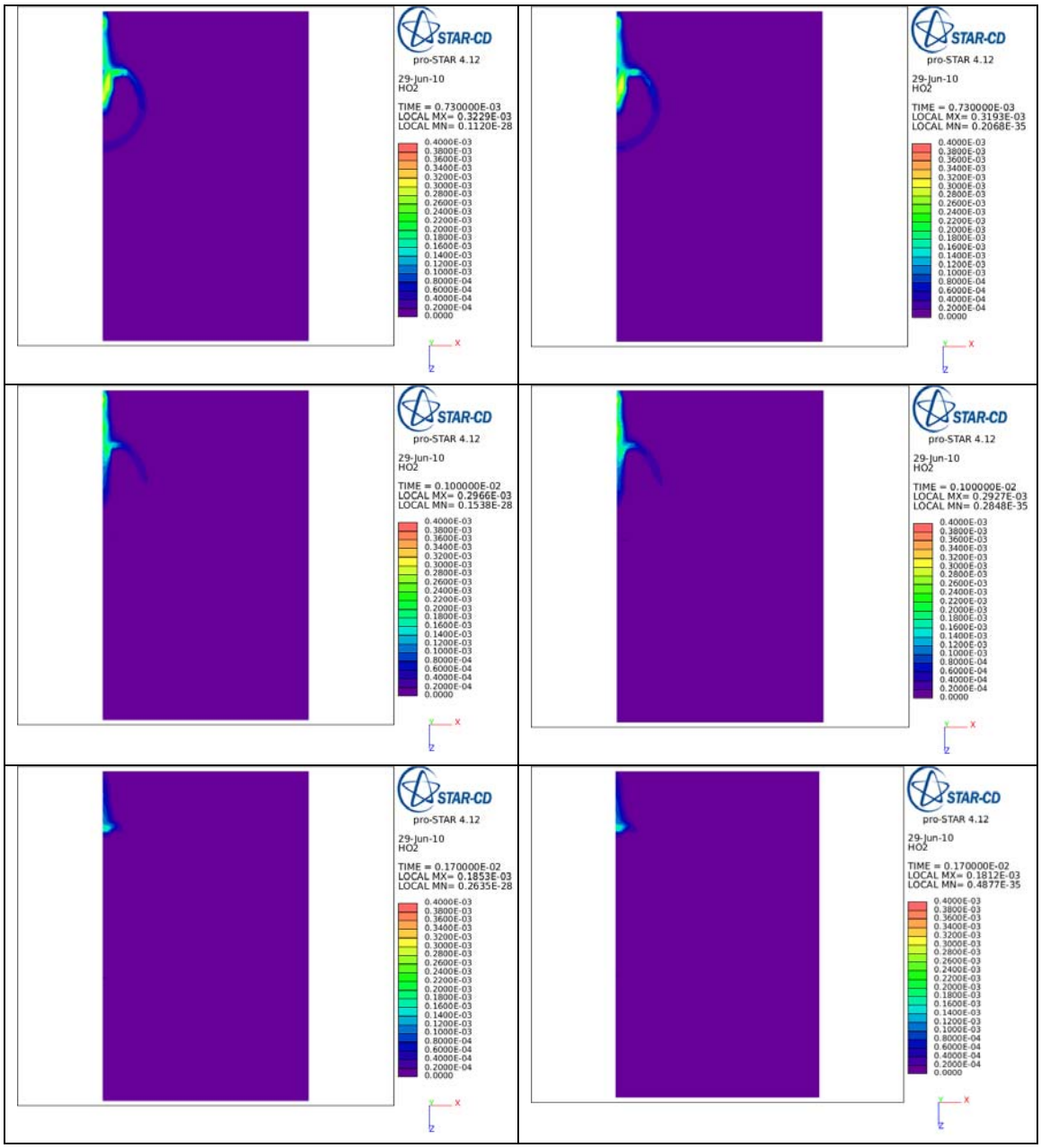
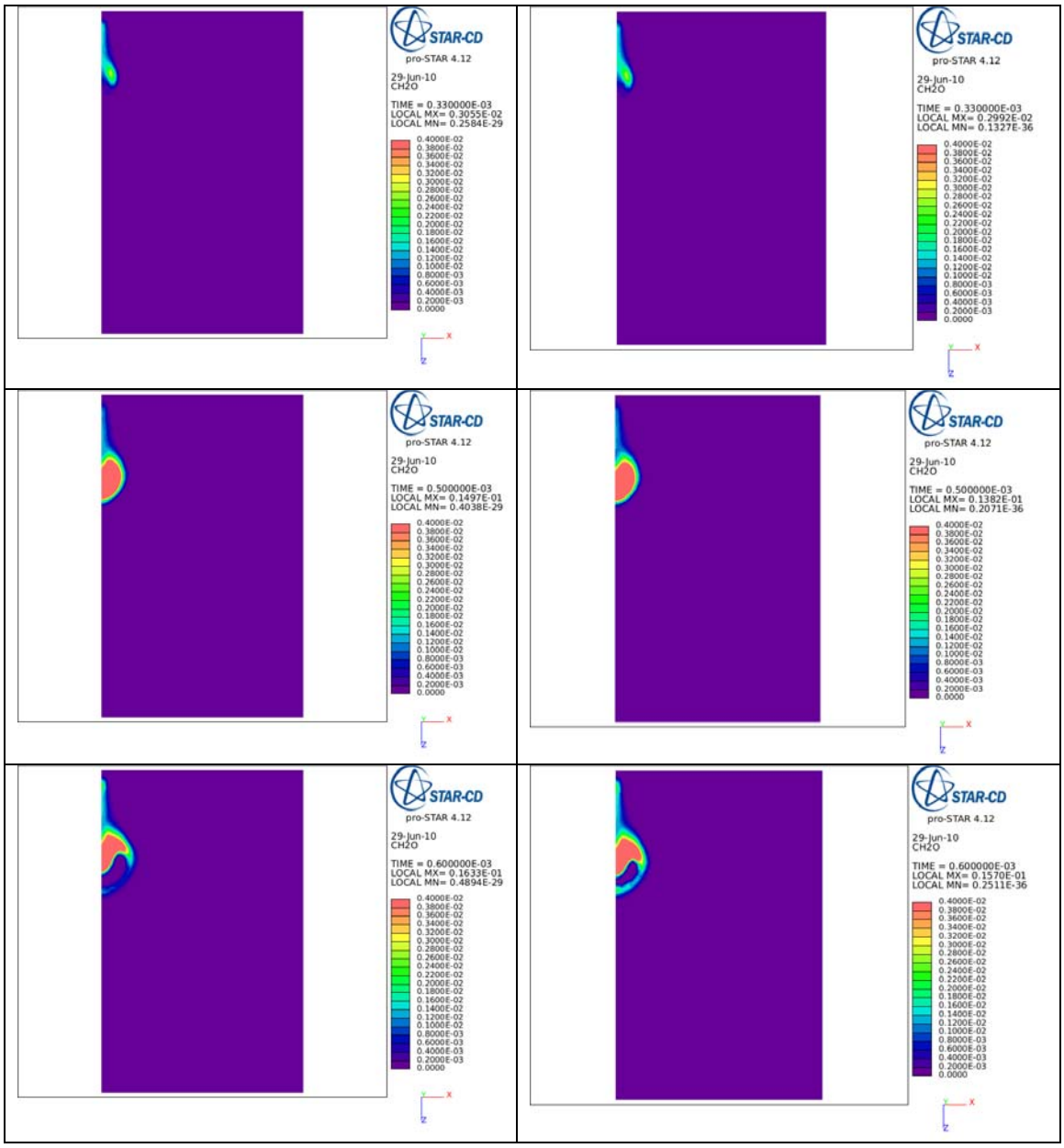
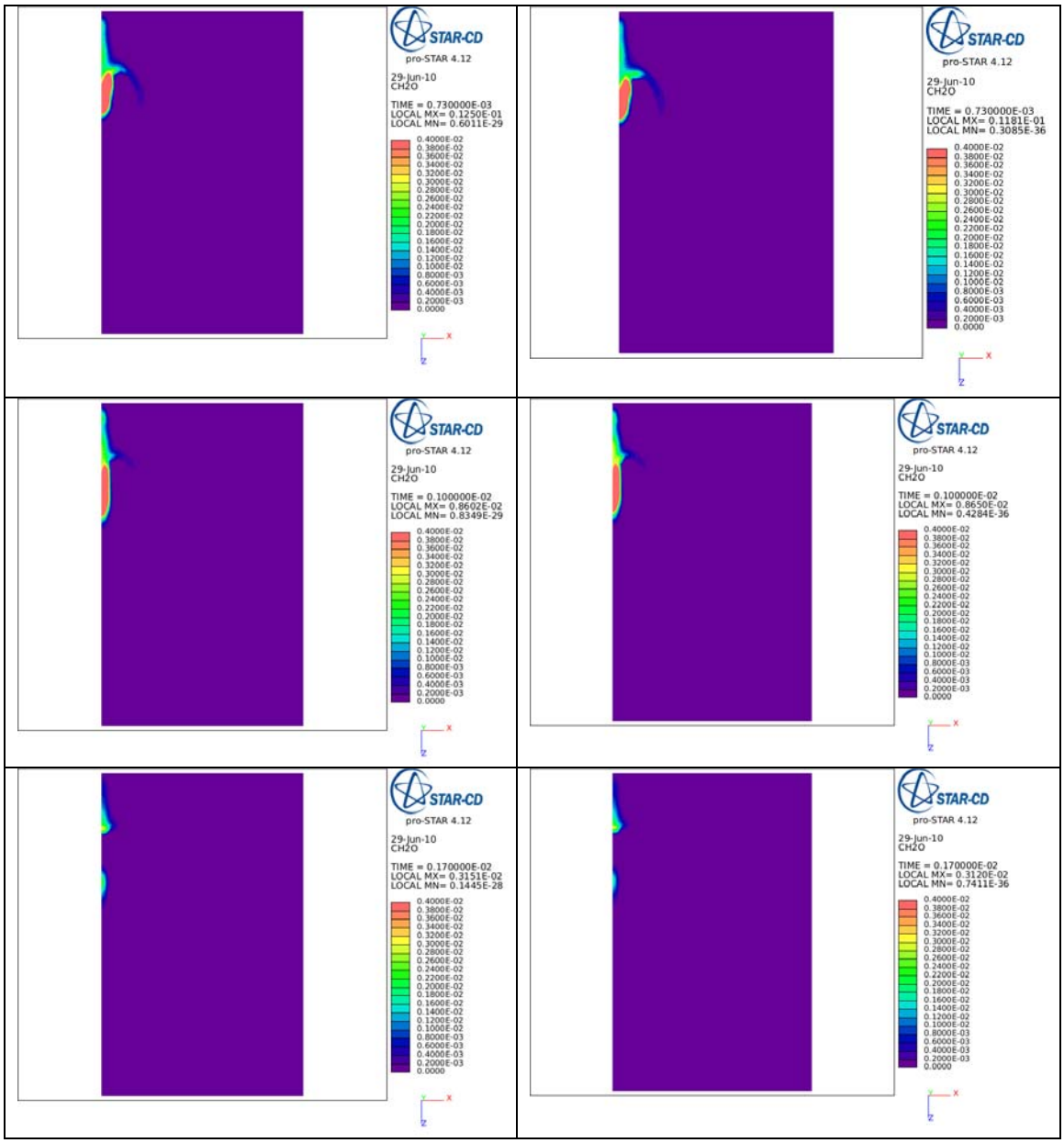

STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.730000E-03
LOCAL MX= 0.3229E-03
LOCAL MN= 0.1120E-28
STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.730000E-03
LOCAL MX= 0.3193E-03
LOCAL MN= 0.2068E-35
STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.100000E-02
LOCAL MX= 0.2966E-03
LOCAL MN= 0.1538E-28
STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.100000E-02
LOCAL MX= 0.2927E-03
LOCAL MN= 0.2848E-35
STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.170000E-02
LOCAL MX= 0.1853E-03
LOCAL MN= 0.2635E-28
STAR-CD
pro-STAR 4.12
29-Jun-10
HO2
TIME = 0.170000E-02
LOCAL MX= 0.1812E-03
LOCAL MN= 0.4877E-35

Figure 6.121. Concentration of CH_2O at four different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.




STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.730000E-03
LOCAL MX= 0.1250E-01
LOCAL MN= 0.6011E-29
STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.730000E-03
LOCAL MX= 0.1181E-01
LOCAL MN= 0.3085E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.100000E-02
LOCAL MX= 0.8602E-02
LOCAL MN= 0.8349E-29
STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.100000E-02
LOCAL MX= 0.8650E-02
LOCAL MN= 0.4284E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.170000E-02
LOCAL MX= 0.3151E-02
LOCAL MN= 0.1445E-28
STAR-CD
pro-STAR 4.12
29-Jun-10
CH2O
TIME = 0.170000E-02
LOCAL MX= 0.3120E-02
LOCAL MN= 0.7411E-36

Concluding remarks

The Newton-Newton solver and Newton-FP solver were used in a CPR in each grid point in a CFD simulation. The performance of both solver combinations, which used a reduced mechanism of 50 QSS species, was compared to the performance of a reference mechanism with 0 QSS species. The CPU time for the CFD solver decreased linearly with the number of QSS species for both solver combinations as expected.

The total CPU time of the Newton-Newton solver and Newton-FP was about a factor 0,66 and 34 respectively of the total CPU time of the reference solution, while the accuracy of the solution was within acceptable limits. Hence, it is advantageous to use the Newton-Newton solver with 50 QSS species instead of a Newton solver with 0 QSS species from a CPU time point of view. The total CPU time ratio between the Newton-FP and Newton-Newton solver is about 51, which clearly shows that the Newton-Newton solver is the fastest of the two solvers and rules out any use of the Newton-FP solver in this CFD application.

6.3. Methane/Propane Mechanism

The Methane/Propane mechanism [3] is a detailed mechanism that does contain low temperature chemistry. Hence, the QSSA is applied directly to the detailed mechanism in this section. However, only one HO_2 peak is observed for the physical conditions presented below. Hence, the Methane/Propane mechanism did not exhibit any two stage auto-ignition process. For this reason the IDT CF instead represents the time of the maximum HO_2 peak for all physical conditions. Hence, the Max HO2 CF represents the Max HO2 HF.
The IDT CF and IDT HF are almost but not exactly identical, since some QSS species affects HO_2 and OH profiles differently.

The ART was applied to different physical conditions, which all contain an initial temperature range, an initial pressure range and a fuel/air equivalence ratio range. This gives a set of cases which are shown in Table 6.10

Table 6.10. The table shows the physical conditions for three different cases.

Case:	Temperature [K]	Pressure [bar]	Fuel/air ratio [-]
Case I	1100	8,1	1,0
Case II	1100-1500	8,1	1,0
Case III	1100-1500	8,1-23,5	1,0

The ART used the same LOI list for all cases when the reduced mechanisms were generated. The LOI list used was based on sensitivity towards OH and generated in a pre-processing step for Case I.

The outer time step size is 10^{-5} for all simulations for the Methane/Propane mixture.

6.3.1 Case 1: One physical point

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature point: 1100 K
- Pressure point: 8,1 bar
- Fuel/air ratio point: 1.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 1 %

The fuel mixture was:

- 90 % Methane (CH_4) and 10 % Propane (C_3H_8)

6.3.1.1 Reduction level of the reduced mechanisms

Figure 6.122 shows the number of QSS species vs LOI rank. The most reduced mechanism contains 95 QSS species out of 118 species, which gives a reduction of about 81 % of the species. The number of QSS species that are accepted before the first species fails to be accepted is 78, which indicates that the LOI list was chosen properly for the range of physical conditions. Table 6.11 shows the species that failed to be accepted and the reason for failure.
The Methane/Propane mechanism could be reduced further than the N-Heptane mechanism. A reason for this is that the N-Heptane mechanism is a skeletal mechanism, while the Methane/Propane mechanism is a detailed mechanism, which is easier to reduce since it has not undergone any reduction procedure. Another reason is that the Methane/Propane mechanism was reduced for simpler physical conditions.

The species CH_2O, with LOI rank 98, affects that the deviation of IDT HF and IDT CF differently when it is set as a QSS species. The reason for this is that CH_2O is involved in reactions with both HO_2 and OH but affects them differently.

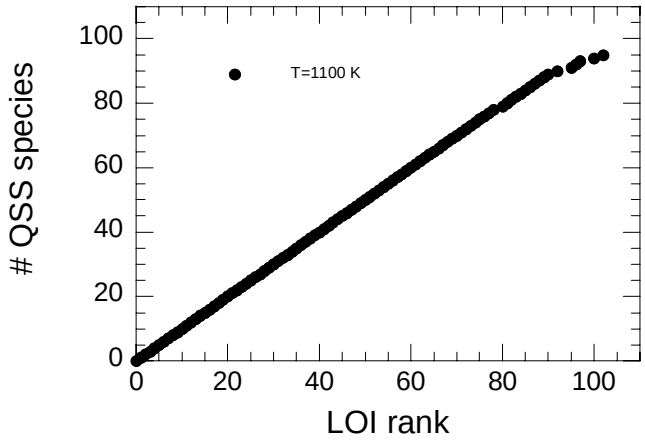


Figure 6.122. Number of QSS species vs LOI rank.

Table 6.11. The table shows the LOI rank of the species, the species names and the corresponding LOI value for the Methane/Propane mechanism. The table is sorted by LOI value, with the lowest first and the highest last. The table also shows the species that failed to be accepted and the reason for the failure for the two solver combinations. A blank means that the species was accepted, while a letter and number combination means that the species failed to be accepted. The numbers represents the deviation according to eq(6.2) for each ET in percent. The letters mean failure due to; A) IDT HF limit B) IDT CF limit C) Max HO2 CF limit D) Max OH HF limit E) No convergence in the inner solver F) The species is forbidden as a QSS species in advance G) There was no ignition H) Temperature limit I) CPU time limit

LOI rank	Species name	LOI value	Reason for failure
1	HO2CH2OCHO	7.0663e-35	
2	C3H6OOH2-2	9.9517e-35	
3	CH3OCH2O2H	1.6541e-34	
4	CH3COCH2O2H	1.2348e-33	
5	O2CH2OCH2O2H	1.2683e-33	
6	CH3CO3H	1.2172e-31	
7	CH3COCH2O2	2.5189e-31	
8	C3H52-1_3OOH	3.8204e-31	
9	C3H51-2_3OOH	6.6499e-31	
10	CH3COCH2O	8.4993e-31	
11	NC3H7O2H	1.9815e-30	
12	OCH2OCHO	3.0308e-30	
13	C3KET12	7.0799e-30	
14	C2H3OOH	7.5606e-30	
15	IC3H7O2H	8.6487e-30	
16	CH2OCH2O2H	1.0390e-29	
17	C3H6OOH1-2O2	1.4259e-29	
18	C3KET13	2.4241e-29	
19	C3KET21	4.2646e-29	
20	CH3OCH2O2	4.4811e-29	
21	C3H6OOH2-1O2	6.0921e-29	
22	C3H6OOH1-3O2	6.9326e-29	
23	CH3OCH2O	1.1897e-28	
24	O2C2H4OH	1.0914e-27	
25	C2H5O2H	1.3083e-27	
26	NC3H7O	5.7778e-27	
27	AC3H5OOH	8.1115e-27	
28	HOC3H6O2	9.7077e-27	
29	C3H6OOH1-3	1.0614e-26	
30	HOCH2O2	1.3876e-26	
31	IC3H7O	1.4960e-26	
32	CH3OCO	2.1962e-26	
33	HOCH2O2H	2.2662e-26	
34	CH3CO2	2.6732e-26	
35	C3H6OOH1-2	3.1375e-26	
36	CH2OCHO	9.6776e-26	
37	C3H6OOH2-1	1.6086e-25	
38	CH3CO3	3.0046e-25	
39	C2H5CO	5.1128e-25	
40	HO2CHO	9.0013e-25	

41	C2H3O1-2	3.5273e-24	
42	CH3COCH2	3.5958e-24	
43	OCH2O2H	5.0832e-24	
44	NC3H7O2	9.3842e-24	
45	C3H5O	4.9403e-23	
46	IC3H7O2	8.8734e-23	
47	CH3O2H	2.4621e-22	
48	CH3OCH2	3.6594e-22	
49	C2H5O	8.2031e-22	
50	HOCH2OCO	1.3692e-21	
51	C2H4O2H	2.6782e-21	
52	C2H5O2	3.4578e-21	
53	IC3H7	6.3811e-21	
54	HOCHO	1.2734e-20	
55	HOCH2O	1.3076e-20	
56	CH3OCHO	3.3327e-20	
57	OCHO	9.0805e-20	
58	C3H6OH	1.2570e-19	
59	O2CHO	3.6683e-19	
60	C2H3CO	5.6714e-19	
61	NC3H7	7.6395e-19	
62	CH3O2	1.5608e-18	
63	C3H6O1-3	1.2766e-17	
64	CH3COCH3	1.7841e-17	
65	C2H5CHO	2.9926e-17	
66	C	4.9105e-17	
67	C3H5-S	6.5104e-17	
68	C3H5-T	9.7927e-17	
69	PC2H4OH	1.4793e-16	
70	CH3CO	2.1659e-16	
71	CH3OCH3	2.9759e-16	
72	CH2S	6.9617e-16	
73	C3H2	1.4539e-15	
74	C2H5OH	1.8675e-15	
75	C2H4O1-2	1.9586e-15	
76	CH2CHO	3.7607e-15	
77	C3H6O1-2	1.0021e-14	
78	CH	1.0679e-14	
79	H2O2	2.4657e-14	A= -4.2
80	SC2H4OH	2.6019e-14	
81	HCCOH	3.0725e-14	
82	C2H	4.5705e-14	
83	CH2	1.5248e-13	
84	C3H5-A	1.5331e-13	
85	C2H5	3.6958e-13	
86	CH3O	4.5815e-13	
87	C3H4-P	5.1900e-13	
88	CH3CHCO	9.2222e-13	
89	CH3CHO	1.3322e-12	
90	C3H4-A	1.4248e-12	
91	C3H3	2.1188e-12	E
92	CH2OH	1.4810e-11	
93	C3H6	1.8155e-11	E

94	C2H3CHO	2.2336e-11	A= -14.8 B= -14.8
95	CH3OH	2.3531e-11	
96	CH2CO	1.0483e-10	
97	C2H3	2.5279e-10	
98	CH2O	4.4050e-10	A= -52.6 B= -75.8 C= -87.4
99	C2H6	1.1391e-09	A= -23.9 B= -23.9 C= -5.4
100	HCO	2.2225e-09	
101	C2H2	3.8229e-09	A= -3.8 C= 6.5
102	HCCO	4.3783e-09	
103	HO2	4.4328e-09	F
104	C2H4	3.4129e-08	E
105	CO2	5.8494e-08	F
106	CH3	9.0586e-08	E
107	O	1.0389e-07	E
108	OH	2.4458e-07	F
109	H2O	1.0411e-06	F
110	CO	7.8922e-06	E
111	H	1.6453e-05	E
112	H2	2.2235e-05	E
113	C3H8	1.0000	F
114	O2	1.0000	F
115	CH4	1.0000	F
116	HE	1.0000	F
117	AR	1.0000	F
118	N2	1.0000	F

6.3.1.2 CPU time of the reduced mechanisms

Figure 6.123 shows the normalized CPU time vs number of QSS species for the Newton-Newton solver and the Newton-FP solver. It is clear that the Newton-Newton solver is much faster than the Newton-FP solver, which is the same result obtained from the N-Heptane mechanism.
The lower sub figure shows the normalized CPU time ratio between the Newton-FP solver and the Newton-Newton solver. The ratio reaches about a factor 23 for the most reduced mechanism. The same ratio for the N-Heptane mechanism is about a factor 40. The analysis of both mechanisms shows that the Newton-Newton solver is much faster than the Newton-FP solver. For this reason, further analysis of the Newton-FP solver is considered uninteresting. Hence, only the results from the Newton-Newton solver are presented below.
The difference in ratios between the two mechanisms can possibly be explained with that fact that the N-Heptane mechanism is a skeletal mechanism and that the physical point is

within the Negative Temperature Coefficient (NTC) region, while the Methane/Propane mechanism is a detailed mechanism and does not show any NTC behavior.
Hence, the N-Heptane mechanism contains more important species at the specific physical condition than the Methane/Propane mechanism does. Important species causes more convergence problems when set in QSS. This means that the Newton-Newton solver has larger impact on the convergence compared to the Newton-FP on the N-Heptane mechanism.

Figure 6.124 shows the normalized CPU time vs number of QSS species for a simulation with the Newton-Newton solver and for a hypothetical solver combination with an infinitely fast inner solver. The Newton-Newton solver reaches a minimum of about 0,27 and the hypothetical solver combination has a value of 0,05 for the most reduced mechanism. The normalized CPU time suddenly jumps when some QSS species are added to the reduced mechanism and then suddenly jumps back again when other QSS species are added to the reduced mechanism. The same behavior can be seen in the solver information, which is shown in Figure 6.129.
The number of Jacobians and BS for the system of ODE as well as the number of BS for the system of NAE follows the jumps in the CPU time, while the jumps in the average time step size are opposite to the CPU time. This solver information shows that the Newton-Newton solver has convergence problems for some reduced mechanism and therefore must decrease the time step size and increase the number of Jacobians and BS in order to converge.
The reason for convergence problems and large jumps for some reduced mechanism can possibly be explained be pairs of QSS species.

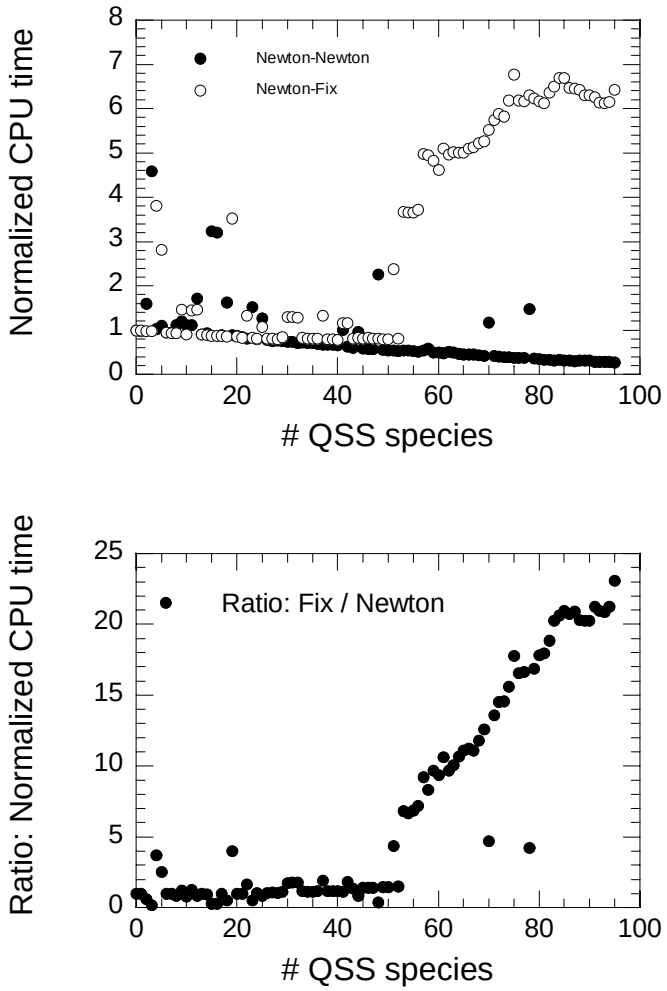


Figure 6.123. The upper sub figure shows the normalized CPU time vs number of QSS species for the Newton-Newton solver and the Newton-FP solver. The lower sub figure shows the ratio difference between the Newton-FP solver and the Newton-Newton solver.

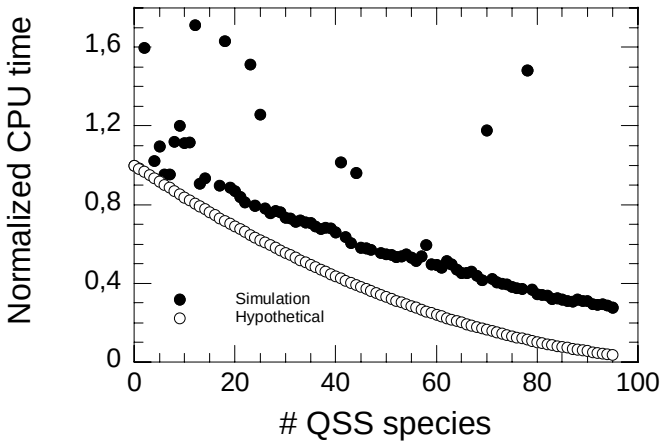


Figure 6.124. The figure shows the normalized CPU time vs number of QSS species for a simulation with the Newton-Newton solver and for a hypothetical solver combination with an infinitely fast inner solver.

6.3.1.3 Accuracy of the reduced mechanisms

Figure 6.125 shows the IDT HF, IDT CF, Max HO2 CF and Max OH HF deviation vs number of QSS species. The IDT HF and IDT CF start to deviate at about 60 QSS and reach a maximum deviation of about -7 % for the most reduced mechanism. This is within acceptable limits. The Max HO2 CF and Max OH HF jumps up and down but are within 4 and 0.2 % respectively, which is well within acceptable limits.

Figures 6.126 to 6.128 show the mass fraction for important species, temperature and pressure vs time for the original and most reduced mechanism. The figures show small deviations well within acceptable limits.

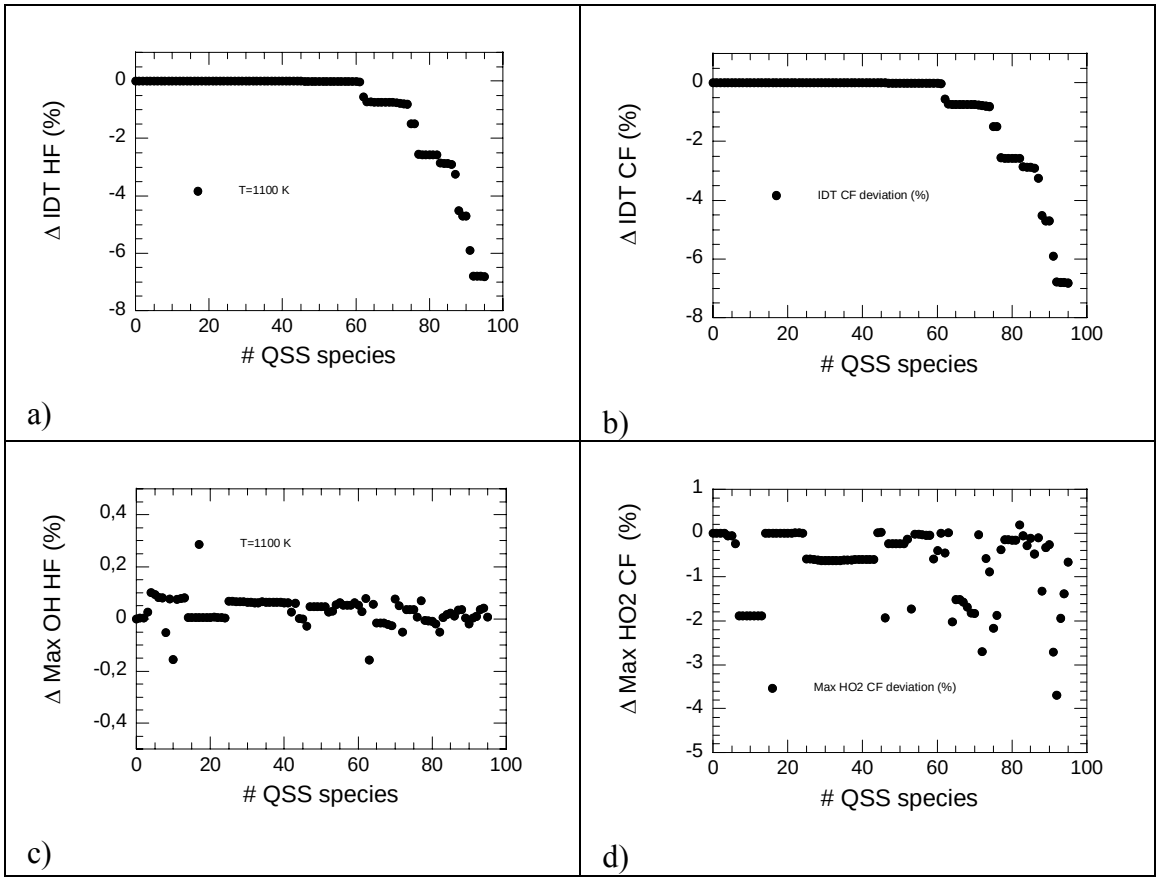


Figure 6.125. Deviation vs number of QSS species.

a) IDT HF vs number of QSS species.
b) IDT CF vs number of QSS species.
c) Max OH HF vs number of QSS species.
d) Max HO2 CF vs number of QSS species.

Since the CF does not exist for the physical conditions the IDT CF represents the time of the HO_2 peak and the Max HO2 CF represents the Max HO2 HF.

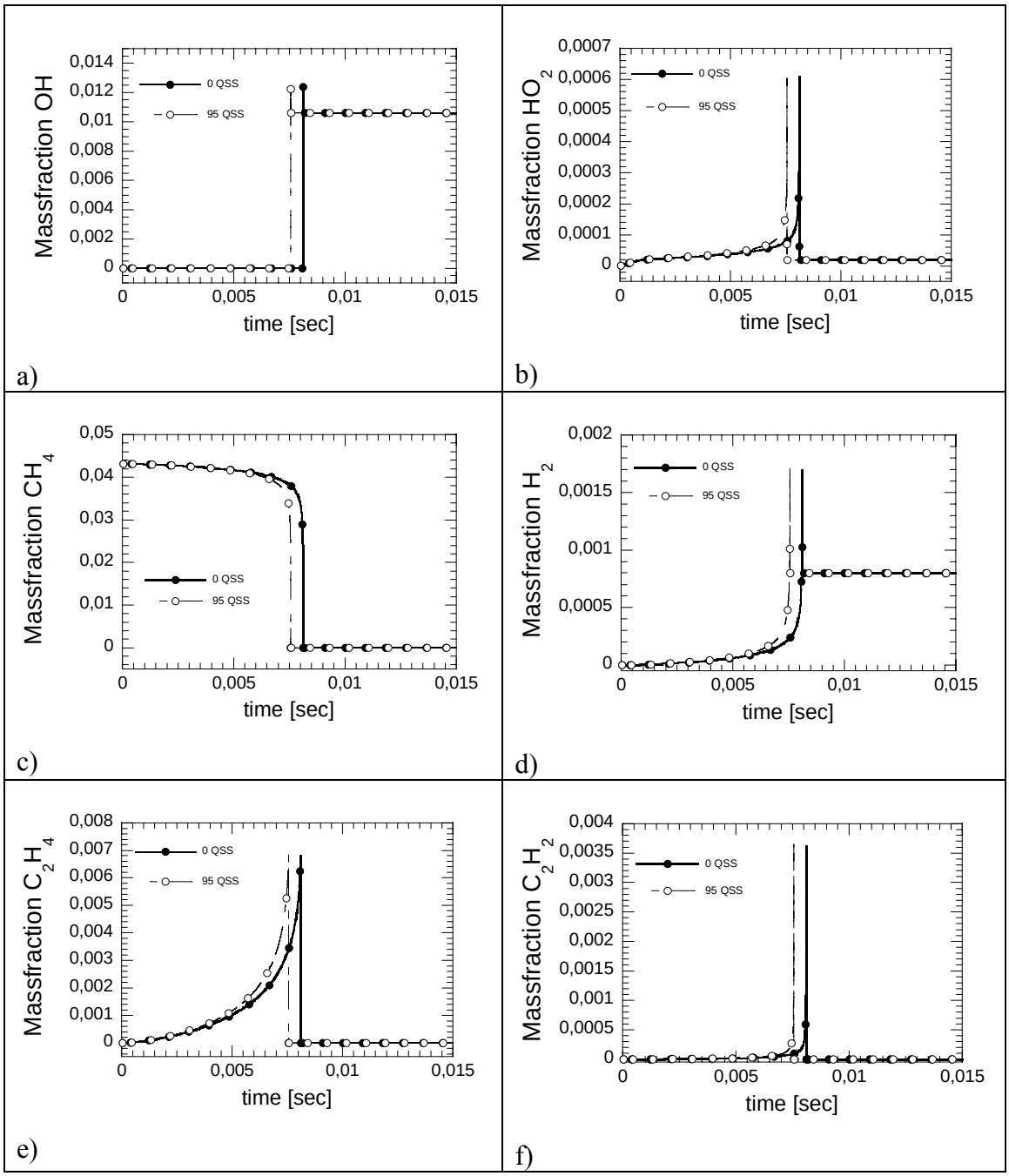


Figure 6.126. Mass fraction of important species vs time for the original and most reduced mechanism with 95 QSS species.

a) OH b) HO_2 c) CH_4 d) H_2 e) C_2H_4 f) C_2H_2

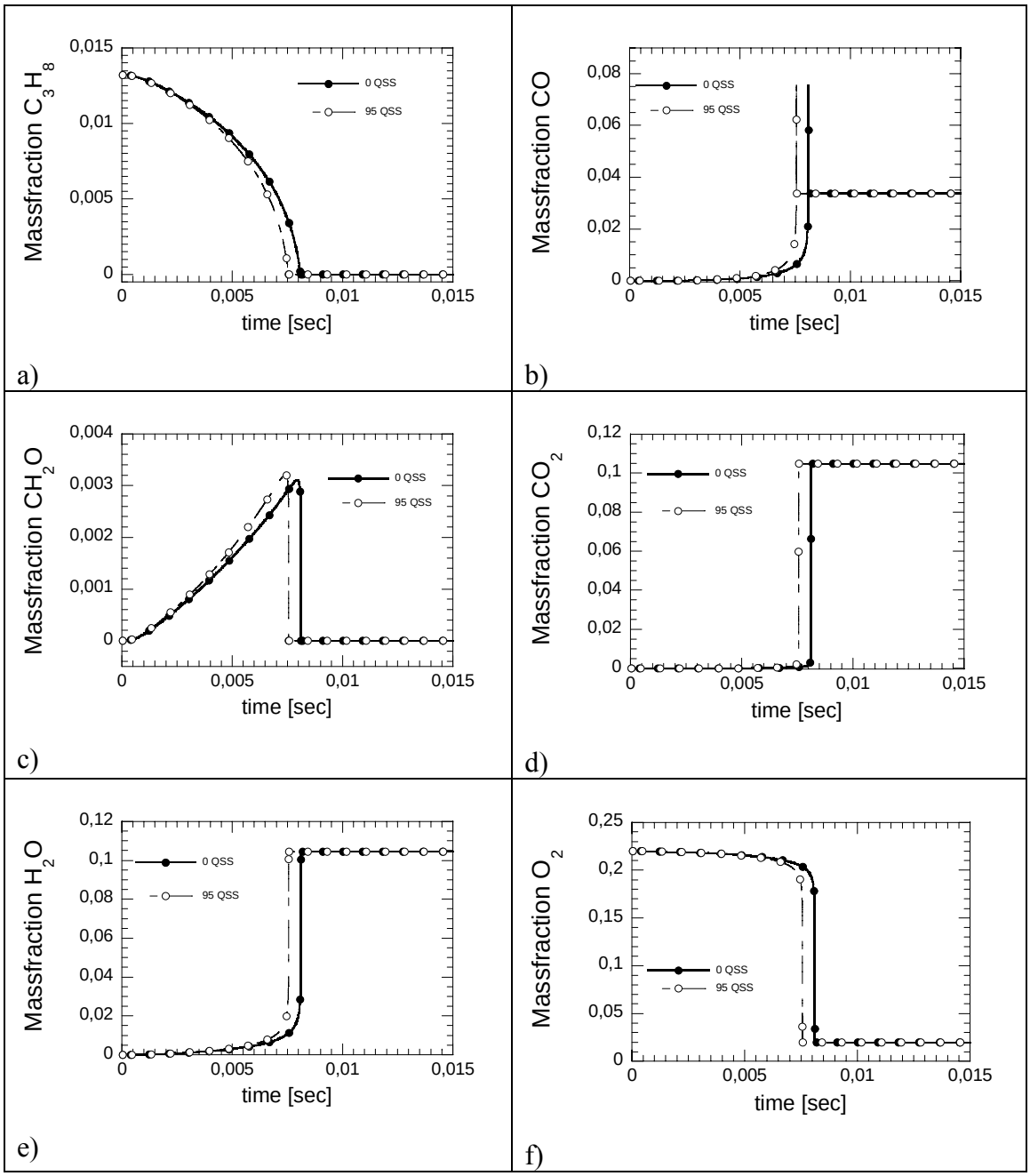


Figure 6.127. Mass fraction of important species vs time for the original and most reduced mechanism with 95 QSS species.
a) C_3H_8 b) CO c) CH_2O d) CO_2 e) H_2O f) O_2

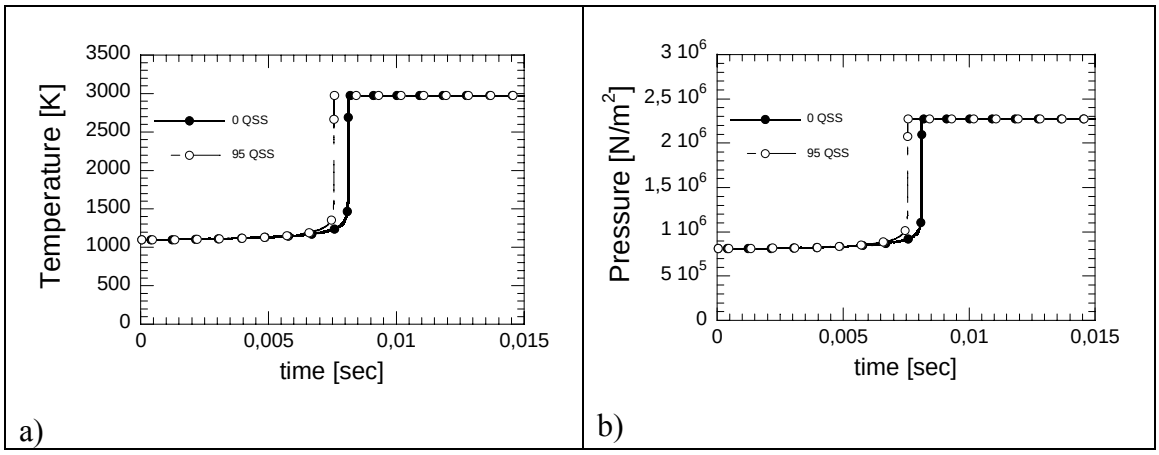


Figure 6.128. Temperature and Pressure vs time for the original and most reduced mechanism with 95 QSS species.
a) Temperature b) Pressure

6.3.1.4 Solver information

Figure 6.129a) shows the normalized average time step size vs number of QSS species. The number is constant with the number of QSS species.

Figure 6.129b) shows the normalized number of Jacobians and BS for the ODE system vs number of QSS species. Both numbers are constant with the number of QSS species.

Figure 6.129c) shows the normalized number of Jacobians and BS for the NAE system vs number of QSS species. The number of Jacobians increases while the number of BS decreases with the number of QSS species. The reason for the increasing number of Jacobians with increasing number of QSS species is due to convergence problems of the system of NAE. The reason for the decrease for the number of BS is that the number of Jacobians already compensated for the convergence problems of the system of NAE with increasing number of QSS species.

The number of Jacobians and BS for the system of ODE as well as the number of BS for the system of NAE follow the jumps in the CPU time, while the jumps in the average time step size are opposite to the CPU time. This solver information shows that the Newton-Newton solver has convergence problems for some reduced mechanism and therefore must decrease the time step size and increase the number of Jacobians and BS in order to converge.
The reason for convergence problems and large jumps for some reduced mechanism can possibly be explained by pairs of QSS species.

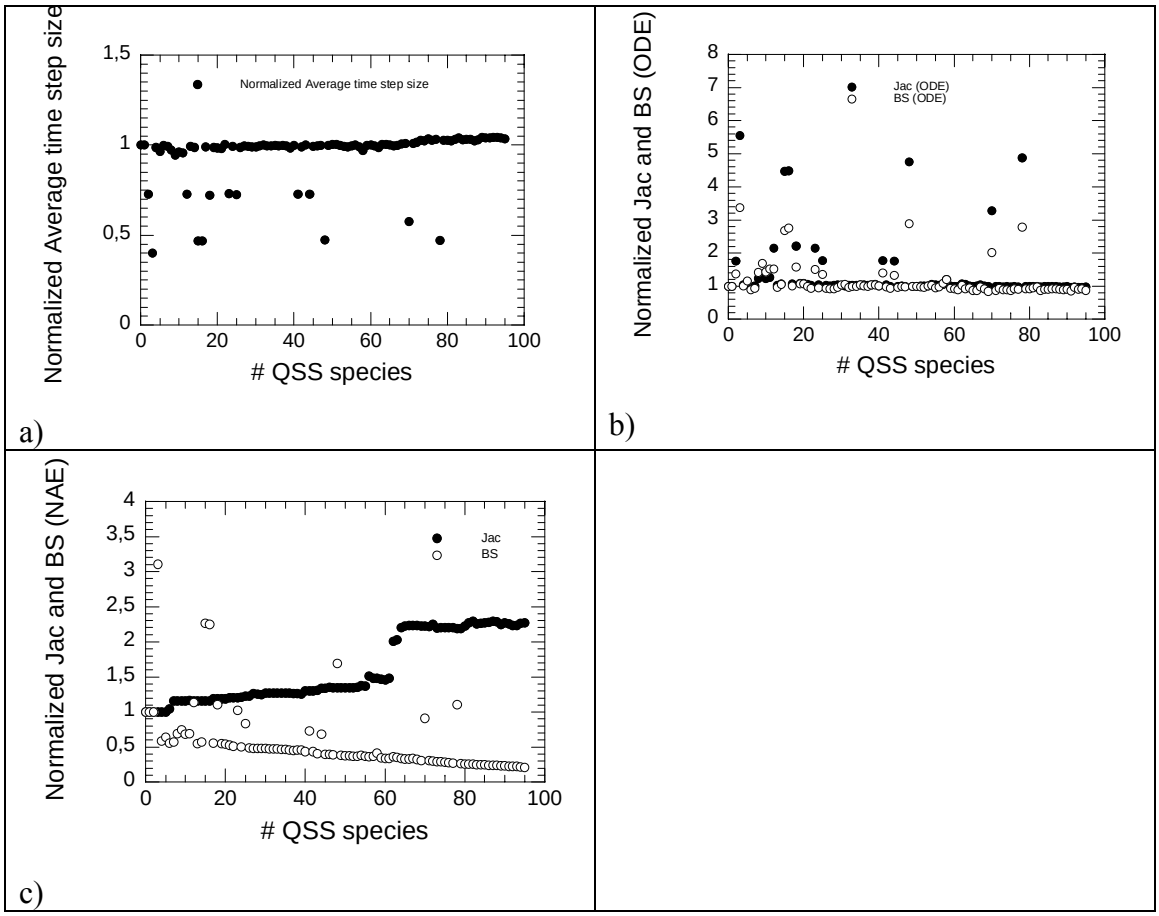


Figure 6.129. Normalized solver information vs number of QSS species.

a) Normalized average time step size vs number of QSS species.
b) Normalized number of Jacobians and BS (ODE) vs number of QSS species.
c) Normalized number of Jacobians and BS (NAE) vs number of QSS species.

6.3.1.4.1 Sparseness pattern of the NAE Jacobian

Figure 6.130 shows the number of operations in the GE and BS vs number of QSS species. The most reduced mechanism with 95 QSS species has about 1500 operations in the GE and BS, which is about 15 % of the theoretical maximum number of operations. This shows the efficiency of the MBSA algorithm. The theoretical maximum corresponds to the number of operations in an algorithm that does not use any sparseness advantages and instead just eliminates the entire sub diagonal part of the Jacobian during the GE and uses every element in the upper triangular matrix, resulting from the GE, during the BS. The number of operations in the GE and BS increases with the number of QSS species. This means that the inner solver uses more operations and thereby more CPU time with increasing number of QSS species. This in turn indicates that the difference between the CPU time from the simulation and the CPU time for the hypothetical solver should increase with the number of QSS species. This can be seen in Figure 6.124.

Figure 6.131 shows the number of reactions per QSS species vs number of QSS species. There is a trend that species with higher LOI rank are involved in a larger number of reactions and vise versa. The sparseness pattern of the Jacobian becomes more dense when species that are involved in many reactions are included in the reduced mechanism. This explains the steeper increase in the number of operations in the in the GE and BS for high number of QSS species.

Figure 6.132 to 6.134 shows the minimized sparseness pattern of the Jacobian before and after GE and the GE matrix for 90 QSS species. The minimized sparseness pattern of the Jacobian before GE seems to be very random at a first glance. However, a closer look shows that the pattern is more dens in the lower right corner of the Jacobian. This is the same result as obtained for N-Heptane in section 6.2.1.3.1. The pattern of the Jacobian after GE and of the GE matrix is very sparse as expected from Figure 6.130.

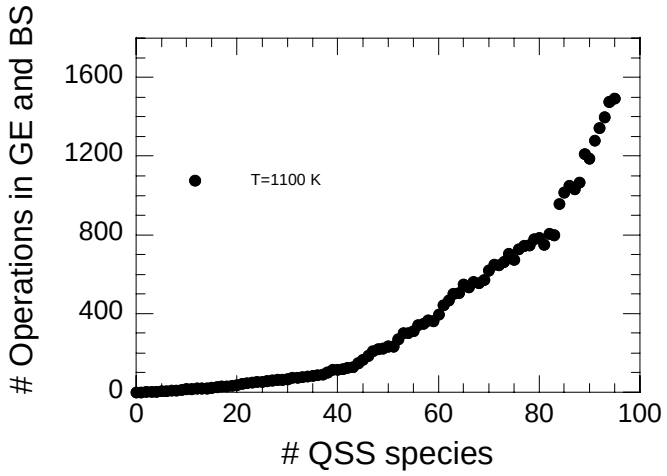


Figure 6.130. Number of operations in the GE and BS vs number of QSS species.

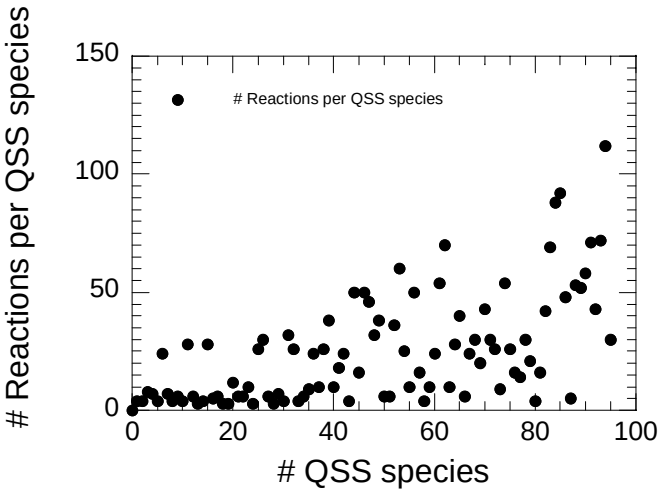


Figure 6.131. Number of reactions per QSS species vs number of QSS species.

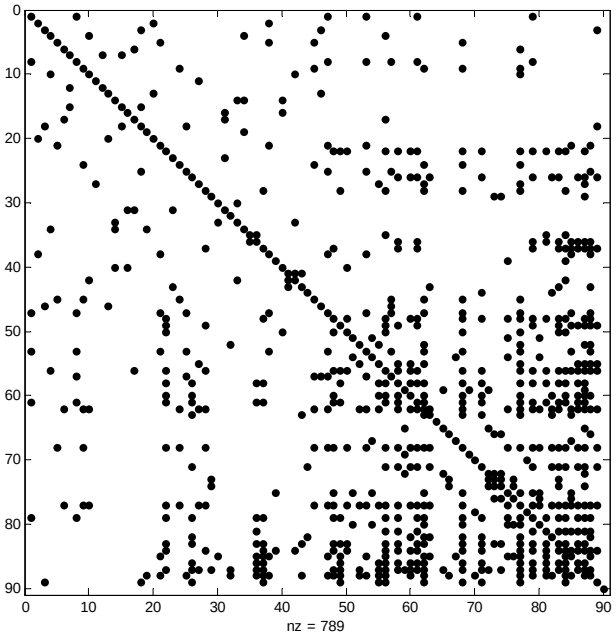


Figure 6.132. Sparseness pattern of the Jacobian with 90 QSS species before the GE.

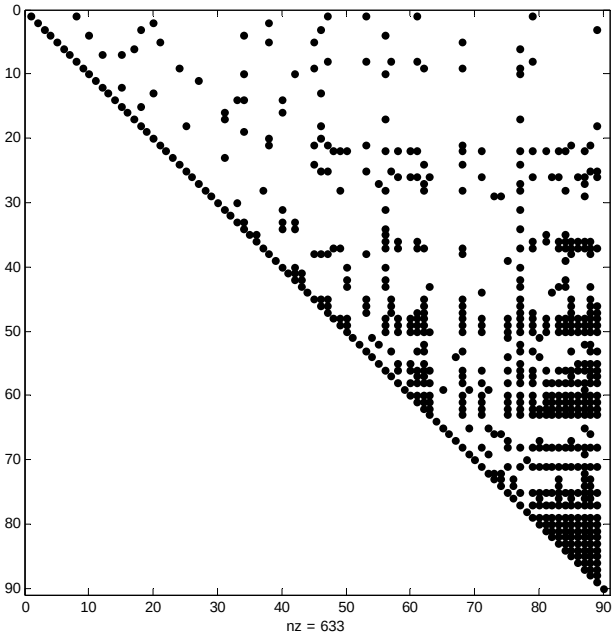


Figure 6.133. Sparseness pattern of the Jacobian with 90 QSS species after the GE.

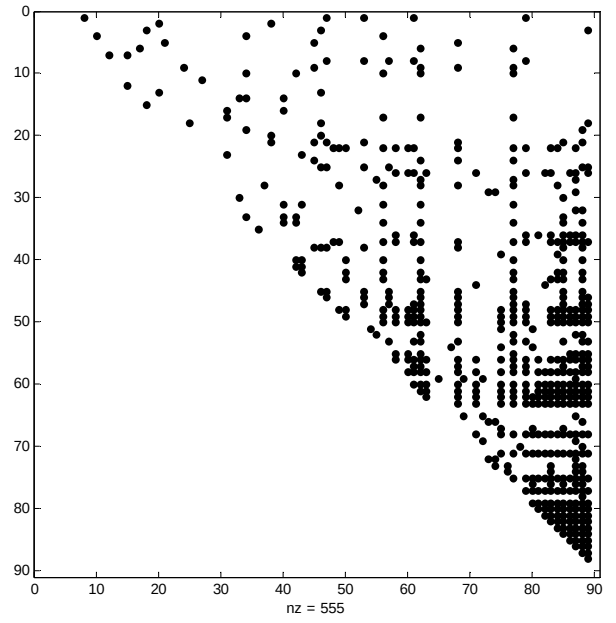


Figure 6.134. Sparseness pattern of the GE matrix with 90 QSS species.

6.3.2 Case 2: Temperature range

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature range: 1100-1500 K
- Pressure point: 8,1 bar
- Fuel/air ratio point: 1.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 1 %

The fuel mixture was:

- 90 % Methane (CH_4) and 10 % Propane (C_3H_8)

6.3.2.1 Reduction level of the reduced mechanisms

The reduced mechanism, which is shown in Figure 6.135, is identical and valid for all initial temperatures. The most reduced mechanism contains 91 QSS species out of 118 species, which gives a reduction of about 77 % of the species. The number of QSS species accepted before the first species fails to be accepted is 78, which indicates that the LOI list was chosen properly for the range of physical conditions.
The number of accepted QSS species is less than for a single temperature when an entire temperature range is considered. This is expected since some species are very important for lower temperature chemistry and therefore cannot be used as a QSS species, while other species are very important for high temperature chemistry and therefore cannot be used as a QSS species.

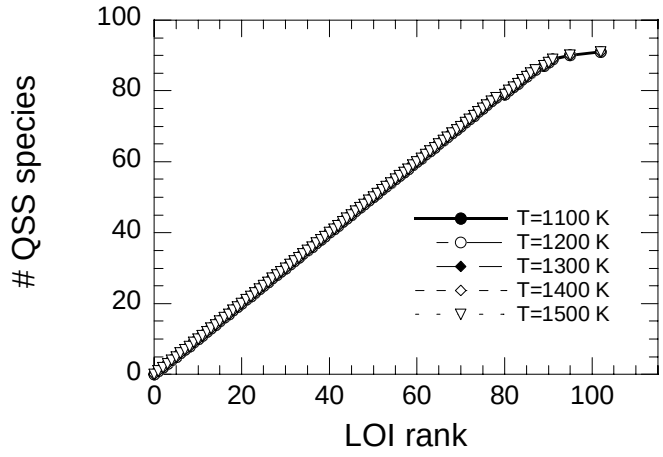


Figure 6.135. Number of QSS species vs LOI rank. The initial temperature range is 1100-1500 K, the initial pressure is 8.1 bar and the fuel/air ratio is 1.0.

6.3.2.2 CPU time of the reduced mechanisms

The normalized CPU time vs number of QSS species is shown in Figure 6.136. The normalized CPU time decreases until the most reduced mechanism for all initial temperatures. The normalized CPU time of the reduced mechanisms varies somewhat for the different physical condition, but the decrease is similar for all initial temperatures and fuel/air ratios. The variation can partly be explained by CPU noise due to other processes in the computer. However, the general explanation for the variation is that each physical condition corresponds to a unique trajectory in the species concentration space and that the convergence of the solver combination, and thereby CPU time, is dependent on the trajectory. The amount of CPU time that can be gained by applying the QSSA to a particular species is also trajectory dependent. Hence, a variation in CPU time exists among the physical conditions for different reduction levels.

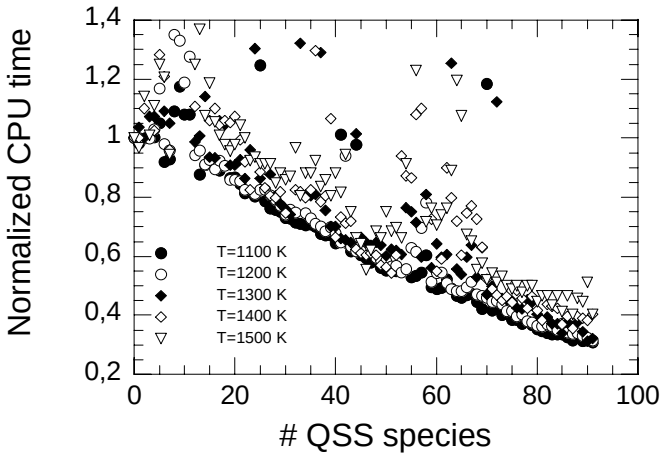


Figure 6.136. Normalized CPU time vs number of QSS species. The initial temperature range is 1100-1500 K, the initial pressure is 8.1 bar and the fuel/air ratio is 1.0.

6.3.2.3 Accuracy of the reduced mechanisms

Figure 6.137 shows the IDT HF vs 1000/T for the original mechanism and the most reduced mechanism. The figure shows that the agreement between the two curves is very good for all initial temperatures.

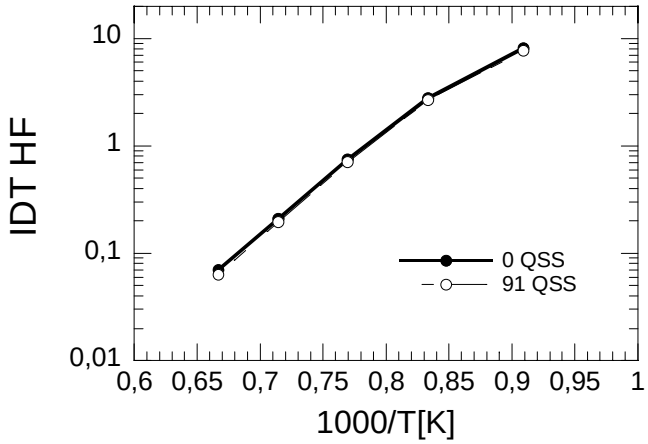


Figure 6.137. IDT HF vs 1000/T for the original and most reduced mechanism for the temperature range 1100-1500 K, the initial pressure 8.1 bar and the fuel/air ratio1.0.

6.3.3 Case 3: Temperature and pressure range

This section investigates the reduced mechanisms that the ART produces for the following physical conditions;

- Temperature range: 1100-1500 K
- Pressure points: 8.1, 16.6 and 23.5 bar
- Fuel/air ratio point: 1.0

The following ART limits were used during the simulations;

- IDT HF limit: 3 %
- IDT CF limit: 5 %
- HO2 CF limit: 5 %
- OH HF limit: 1 %

The fuel mixture was:

- 90 % Methane (CH_4) and 10 % Propane (C_3H_8)

6.3.3.1 Reduction level of the reduced mechanisms

The reduced mechanism, which is shown in Figure 6.138, is identical and valid for all initial temperatures. The most reduced mechanism contains 88 QSS species out of 118 species, which gives a reduction of about 75 % of the species. The number of QSS species accepted before the first species fails to be accepted is 54, which indicates that the LOI list was not chosen properly for the range of physical conditions.
The number of accepted the number of QSS species is less than for a single pressure when an entire pressure range is considered. This is expected since some species are very important for lower pressures and therefore cannot be used as a QSS species, while other species are very important for high pressures and therefore cannot be used as a QSS species.
Table 6.12 shows the species that were accepted and failed to be accepted respectively for all three cases. This shows the importance of the different species for various physical conditions. More species are accepted for narrow ranges of physical conditions as expected.

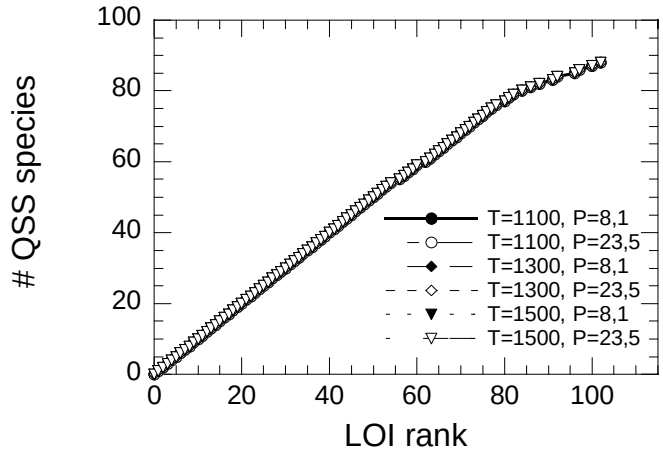


Figure 6.138. Number of QSS species vs LOI rank. The initial temperature range is 1100-1500 K, the initial pressures are 8.1, 16.6 and 23.5 bar and the fuel/air ratio is 1.0. The figure shows only a subgroup of the possible combinations.

Table 6.12. The table shows the LOI rank, species name and LOI value. The table also shows the species that were accepted and failed to be accepted by the ART for the three cases. A blank means that the species was accepted, while an "F" means that the species failed to be accepted.

Nr	Species name	LOI	Case 1	Case 2	Case 3
1	HO2CH2OCHO	7.0663e-35			
2	C3H6OOH2-2	9.9517e-35			
3	CH3OCH2O2H	1.6541e-34			
4	CH3COCH2O2H	1.2348e-33			
5	O2CH2OCH2O2H	1.2683e-33			
6	CH3CO3H	1.2172e-31			
7	CH3COCH2O2	2.5189e-31			
8	C3H52-1_3OOH	3.8204e-31			
9	C3H51-2_3OOH	6.6499e-31			
10	CH3COCH2O	8.4993e-31			
11	NC3H7O2H	1.9815e-30			
12	OCH2OCHO	3.0308e-30			
13	C3KET12	7.0799e-30			
14	C2H3OOH	7.5606e-30			
15	IC3H7O2H	8.6487e-30			
16	CH2OCH2O2H	1.0390e-29			
17	C3H6OOH1-2O2	1.4259e-29			
18	C3KET13	2.4241e-29			
19	C3KET21	4.2646e-29			
20	CH3OCH2O2	4.4811e-29			
21	C3H6OOH2-1O2	6.0921e-29			
22	C3H6OOH1-3O2	6.9326e-29			
23	CH3OCH2O	1.1897e-28			

24	O2C2H4OH	1.0914e-27			
25	C2H5O2H	1.3083e-27			
26	NC3H7O	5.7778e-27			
27	AC3H5OOH	8.1115e-27			
28	HOC3H6O2	9.7077e-27			
29	C3H6OOH1-3	1.0614e-26			
30	HOCH2O2	1.3876e-26			
31	IC3H7O	1.4960e-26			
32	CH3OCO	2.1962e-26			
33	HOCH2O2H	2.2662e-26			
34	CH3CO2	2.6732e-26			
35	C3H6OOH1-2	3.1375e-26			
36	CH2OCHO	9.6776e-26			
37	C3H6OOH2-1	1.6086e-25			
38	CH3CO3	3.0046e-25			
39	C2H5CO	5.1128e-25			
40	HO2CHO	9.0013e-25			
41	C2H3O1-2	3.5273e-24			
42	CH3COCH2	3.5958e-24			
43	OCH2O2H	5.0832e-24			
44	NC3H7O2	9.3842e-24			
45	C3H5O	4.9403e-23			
46	IC3H7O2	8.8734e-23			
47	CH3O2H	2.4621e-22			
48	CH3OCH2	3.6594e-22			
49	C2H5O	8.2031e-22			
50	HOCH2OCO	1.3692e-21			
51	C2H4O2H	2.6782e-21			
52	C2H5O2	3.4578e-21			
53	IC3H7	6.3811e-21			
54	HOCHO	1.2734e-20			
55	HOCH2O	1.3076e-20			F
56	CH3OCHO	3.3327e-20			
57	OCHO	9.0805e-20			
58	C3H6OH	1.2570e-19			
59	O2CHO	3.6683e-19			
60	C2H3CO	5.6714e-19			
61	NC3H7	7.6395e-19			F
62	CH3O2	1.5608e-18			
63	C3H6O1-3	1.2766e-17			
64	CH3COCH3	1.7841e-17			
65	C2H5CHO	2.9926e-17			
66	C	4.9105e-17			
67	C3H5-S	6.5104e-17			
68	C3H5-T	9.7927e-17			
69	PC2H4OH	1.4793e-16			
70	CH3CO	2.1659e-16			
71	CH3OCH3	2.9759e-16			
72	CH2S	6.9617e-16			
73	C3H2	1.4539e-15			
74	C2H5OH	1.8675e-15			
75	C2H4O1-2	1.9586e-15			
76	CH2CHO	3.7607e-15			

77	C3H6O1-2	1.0021e-14			
78	CH	1.0679e-14			
79	H2O2	2.4657e-14	F	F	F
80	SC2H4OH	2.6019e-14			
81	HCCOH	3.0725e-14			
82	C2H	4.5705e-14			
83	CH2	1.5248e-13			F
84	C3H5-A	1.5331e-13			
85	C2H5	3.6958e-13			F
86	CH3O	4.5815e-13			
87	C3H4-P	5.1900e-13			F
88	CH3CHCO	9.2222e-13		F	
89	CH3CHO	1.3322e-12			F
90	C3H4-A	1.4248e-12			F
91	C3H3	2.1188e-12	F		
92	CH2OH	1.4810e-11		F	
93	C3H6	1.8155e-11	F	F	F
94	C2H3CHO	2.2336e-11	F	F	F
95	CH3OH	2.3531e-11			F
96	CH2CO	1.0483e-10		F	
97	C2H3	2.5279e-10		F	
98	CH2O	4.4050e-10	F	F	F
99	C2H6	1.1391e-09	F	F	F
100	HCO	2.2225e-09		F	
101	C2H2	3.8229e-09	F	F	F
102	HCCO	4.3783e-09			
103	HO2	4.4328e-09	F	F	F
104	C2H4	3.4129e-08	F	F	F
105	CO2	5.8494e-08	F	F	F
106	CH3	9.0586e-08	F	F	F
107	O	1.0389e-07	F	F	F
108	OH	2.4458e-07	F	F	F
109	H2O	1.0411e-06	F	F	F
110	CO	7.8922e-06	F	F	F
111	H	1.6453e-05	F	F	F
112	H2	2.2235e-05	F	F	F
113	C3H8	1.0000	F	F	F
114	O2	1.0000	F	F	F
115	CH4	1.0000	F	F	F
116	HE	1.0000	F	F	F
117	AR	1.0000	F	F	F
118	N2	1.0000	F	F	F

6.3.3.2 CPU time of the reduced mechanisms

The normalized CPU time vs number of QSS species is shown in Figure 6.139. The normalized CPU time decreases until the most reduced mechanism for all initial temperatures. The normalized CPU time of the reduced mechanisms varies somewhat for the different physical condition, but the decrease is similar for all initial temperatures and fuel/air ratios. The variation can partly be explained by CPU noise due to other processes in the computer. However, the general explanation for the variation is that each physical condition corresponds to a unique trajectory in species concentration space and that the convergence of the solver combination, and thereby CPU time, is dependent on the trajectory. The amount of CPU time that can be gained by applying the QSSA to a particular species is also trajectory dependent. Hence, a variation in CPU time exists among the physical conditions for different reduction levels.

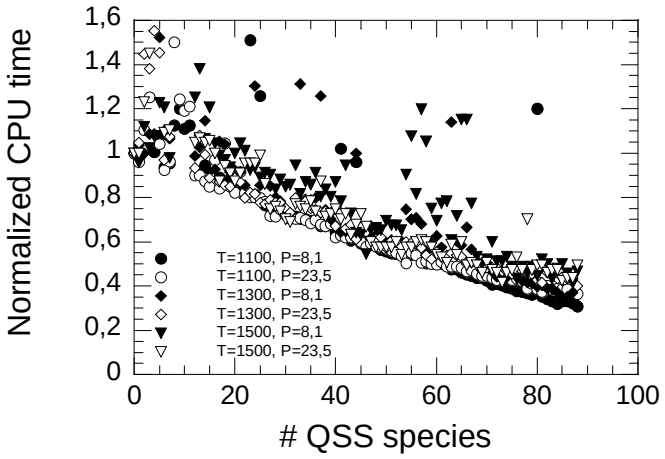


Figure 6.139. Normalized CPU time vs Number of QSS species. The initial temperature range is 1100-1500 K, the initial pressures are 8.1, 16.6 and 23.5 bar and the fuel/air ratio is 1.0. The figure shows only a subgroup of the possible combinations.

6.3.3.3 Accuracy of the reduced mechanisms

Figure 6.140 shows the IDT HF vs 1000/T for the original mechanism and the most reduced mechanism. The figure shows that the agreement between the two curves is very good for all initial temperatures and all initial pressures.

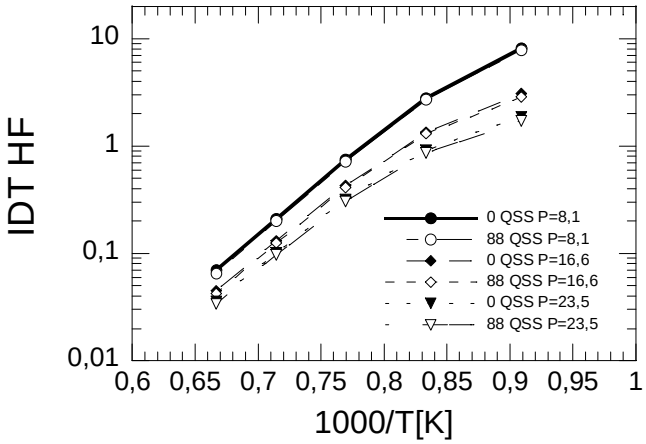


Figure 6.140. IDT HF vs 1000/T[K] for the original and most reduced mechanism containing 0 and 88 QSS species respectively. The initial temperature range is 1100-1500 K, the initial pressures are 8.1, 16.6 and 23.5 bar and the fuel/air ratio is 1.0.

Concluding remarks

The Methane/Propane mechanism reached a higher level of reduction and lower CPU values than the N-Heptane mechanism, while the accuracy of both the reduced mechanisms was within similar acceptable limits. A reason for this is that the N-Heptane mechanism is a skeletal mechanism, while the Methane/Propane mechanism is a detailed mechanism, which is easier to reduce since it has not undergone any reduction procedure such as lumping and "skeletization". Another reason is that the CF does not exist for the Methane/Propane mechanism and that the Methane/Propane mechanism was reduced for simpler physical conditions.

Reduction of the Methane/Propane mechanism for the entire initial temperature range and entire initial pressure range resulted only in seven less QSS species than for a single temperature and pressure point. This also indicates that the Methane/Propane mechanism was reduced for simpler physical conditions than N-Heptane.

The normalized CPU time reached 0,25 and 0,5 for the most reduced Methane/Propane and N-Heptane mechanism respectively. One reason for this is that the Methane/Propane

mechanism was reduced further. Another reason is that the Methane/Propane mechanism was reduced for simpler physical conditions, which simplifies the convergence.

The jumps in CPU time were not observed for the N-Heptane mechanism, which suggests that the two mechanisms are constructed in different ways.

The jumps in CPU time can also be seen in the solver data, which can be seen as interpreted as convergence problems for some reduced mechanisms. The convergence problems are caused by groups of QSS species.

The MBSA algorithm managed to reduce the number of operations in the GE and BS to about 15 % of the maximum, which is similar to the results from the N-Heptane mechanism.

6.4. Chapter references

[1] S.S. Ahmed, F. Mauss, G. Moreac, T. Zeuch, Combustion and Flame, 155 (2008) 651-674
[2] K. Fieweger, R. Blumenthal, G, Adomeit, *Combust. Flame* 109 (1997) 599-619
[3] E.L. Petersen, D.M. Kalitan, S. Simmons, G. Bourque, H.J. Curran, and J.M. Simmie, (2006). Proc. Comb. Inst., vol. 31, Elsevier.
[4] T. Lövås, Automatic *Reduction procedures for Chemical mechanisms in Reactive Systems*, (2002), Doctoral Thesis, Combustion Physics, Lund University, Sweden, Lund reports on Physics, LRCP-78, ISRN LUTFD2/TFCP—78—SE
[5] T. Turanyi, A.S. Tomlin, M.J. Pilling, *On the error of the Quasi-Steady-State Approximation J. Phys. Chem*. 1993, 97, 163-172

6.5. Chapter Appendix

A.6.1. Absolute and Relative tolerances in the inner solver

The convergence of the inner solver is decided from eq(6.6) and eq(6.7). If the species concentration is lower than the absolute tolerance, the absolute tolerance will dominate the denominator according to eq(6.7). Hence, when eq(6.6) is fulfilled and convergence is reached, the species with concentrations lower than the absolute tolerance will have lower accuracy than the species with concentrations higher than the absolute tolerance. This can be understood by the following example.

Table A.6.1 shows the relative change in species concentrations needed in order to reach convergence for AbsoluteTolerance=10^{-15} and RelativeTolerance=10^{-7} for different species concentrations. Table A.6.2 shows the same thing for AbsoluteTolerance=10^{-8} and RelativeTolerance=10^{-6}.

If the absolute tolerance equals 10^{-15} and the relative tolerance equals 10^{-7}, convergence will be reached if $\Delta x_i < 10^{-22}$ for all concentrations that are smaller than the absolute tolerance. Similarly, if the absolute tolerance equals 10^{-8} and the relative tolerance equals 10^{-6}, convergence will be reached if $\Delta x_i < 10^{-14}$ for all concentrations that are smaller than the absolute tolerance.
Hence, the examples in Table A.6.1 and A.6.2 show that the QSS species with concentrations higher than the absolute tolerance have a relative change in concentrations that is lower than the relative tolerance when they fulfill the inequality. These species have reliable accuracy in the concentrations compared to the relative tolerance.

However, the species with lower concentrations than the absolute tolerance fulfill the inequality for higher relative change in the concentration, which means less accuracy in the concentration for these species. Hence, the absolute tolerance of 10^{-15} allows more species to be accurate than the absolute tolerance of 10^{-8}.

Table A.6.1. The table shows the relative change in species concentrations needed in order to reach convergence for AbsoluteTolerance=10^{-15} and RelativeTolerance=10^{-7} for different species concentrations. Column 1, 2, 3, 4 and 5 shows the species concentration, the inequality, the concentration change needed for convergence, the approximate concentration change needed for convergence and the approximate relative change in concentration needed for convergence respectively.

X_i	The inequality	Concentration change needed for convergence	Approximate concentration change needed for convergence	Relative change in concentration needed for convergence
10^{-14}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}1 \cdot 10^{-14}\right)$	$\Delta x_i < 10^{-21}$	$\frac{\Delta x_i}{x_i} < 10^{-7}$
10^{-15}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(2 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-7}$
10^{-16}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}1 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-6}$
10^{-17}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}01 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-5}$
10^{-18}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}001 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-4}$
10^{-19}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}0001 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-3}$
10^{-20}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}00001 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-2}$
10^{-21}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}000001 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{-1}$
10^{-22}	$\frac{\Delta x_i}{x_i + 10^{-15}} < 10^{-7}$	$\Delta x_i < 10^{-7} \cdot \left(1{,}0000001 \cdot 10^{-15}\right)$	$\Delta x_i < 10^{-22}$	$\frac{\Delta x_i}{x_i} < 10^{0}$

Table A.6.2. The table shows the relative change in species concentrations needed in order to reach convergence for AbsoluteTolerance=10^{-8} and RelativeTolerance=10^{-6} for different species concentrations. Column 1, 2, 3, 4 and 5 shows the species concentration, the inequality, the concentration change needed for convergence, the approximate concentration change needed for convergence and the approximate relative change in concentration needed for convergence respectively.

X_i	The inequality	Concentration change needed for convergence	Approximate concentration change needed for convergence	Relative change in concentration needed for convergence
10^{-7}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}1 \cdot 10^{-7}\right)$	$\Delta x_i < 10^{-13}$	$\frac{\Delta x_i}{x_i} < 10^{-6}$
10^{-8}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(2 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-6}$
10^{-9}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}1 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-5}$
10^{-10}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}01 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-4}$
10^{-11}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}001 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-3}$
10^{-12}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}0001 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-2}$
10^{-13}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}00001 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{-1}$
10^{-14}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}000001 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{0}$
10^{-15}	$\frac{\Delta x_i}{x_i + 10^{-8}} < 10^{-6}$	$\Delta x_i < 10^{-6} \cdot \left(1{,}0000001 \cdot 10^{-8}\right)$	$\Delta x_i < 10^{-14}$	$\frac{\Delta x_i}{x_i} < 10^{1}$

A.6.2. CFD simulation results

Figure A.6.1 to A.6.9 shows the species concentrations for H_2, C_2H_4, C_2H_2, CH_4, N-C_7H_{16}, O_2, H_2O, CO and CO_2. Each figure shows the species concentration at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.

The H_2 concentration slowly builds up at early time points and is large throughout the entire HF region at later time points. This is expected from CVR simulation (see section 6.2.1.1.5)

The C_2H_4 concentration is also large at early time points and surrounds the HF region at later time points. This is also expected since C_2H_4 participates in lower temperature chemistry.

The C_2H_2 concentration and temperature follow each other from 0.60 ms, which is expected since C_2H_2 is a high temperature species.

The CH_4 concentration is visible for all times except the first, which is expected since it participates in many reactions.

The N-C_7H_{16} concentration is visible for all times except the last, which is expected since all fuel is supposed to be burnt then.

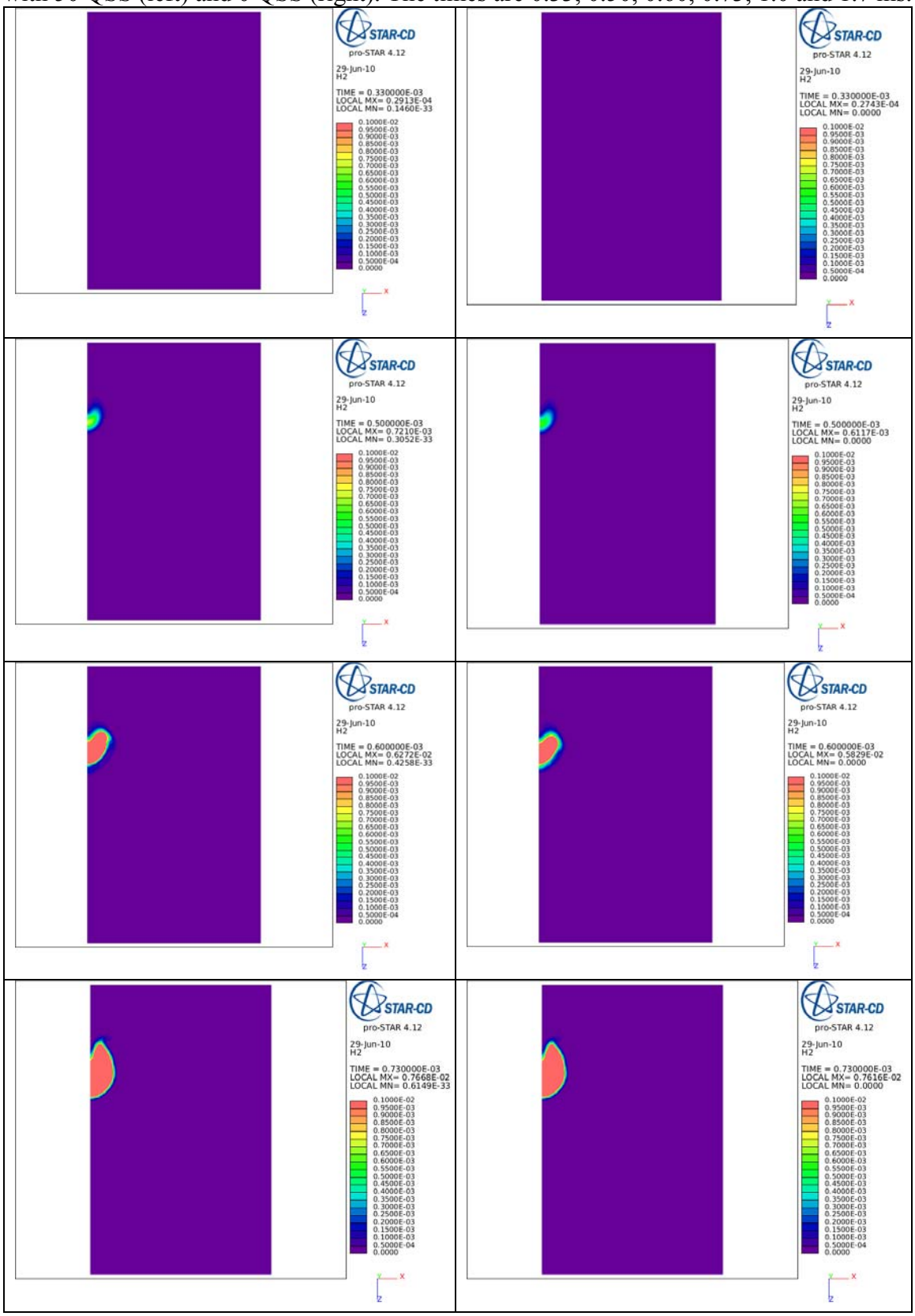
The O_2 concentration is high everywhere except where the flame is, which is expected since the O_2 is consumed at the flame.

The H_2O concentration is visible for all times, which is expected since it is a product.

The CO concentration is visible for all times except the first, which is expected since it is a product.

The CO_2 concentration is visible for all times except the first, which is expected since it is a product.

Figure A.6.1. Concentration of H_2 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



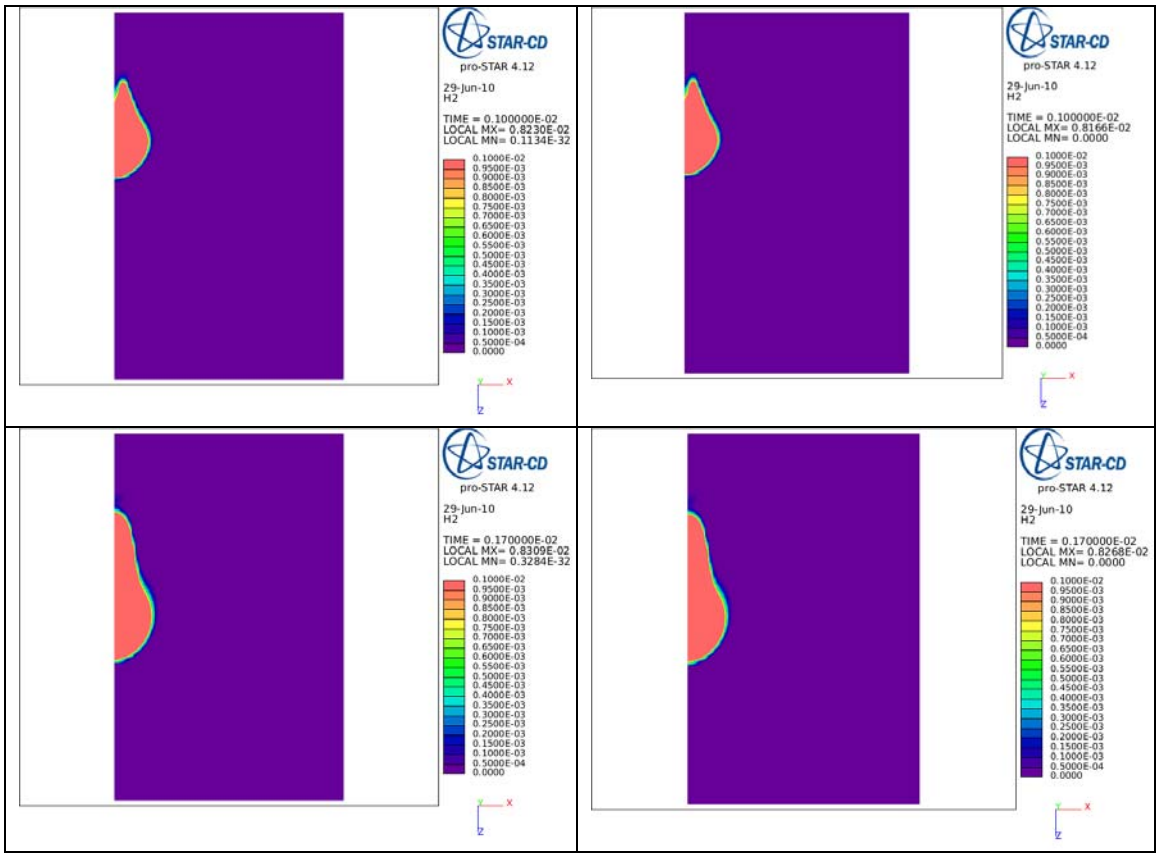
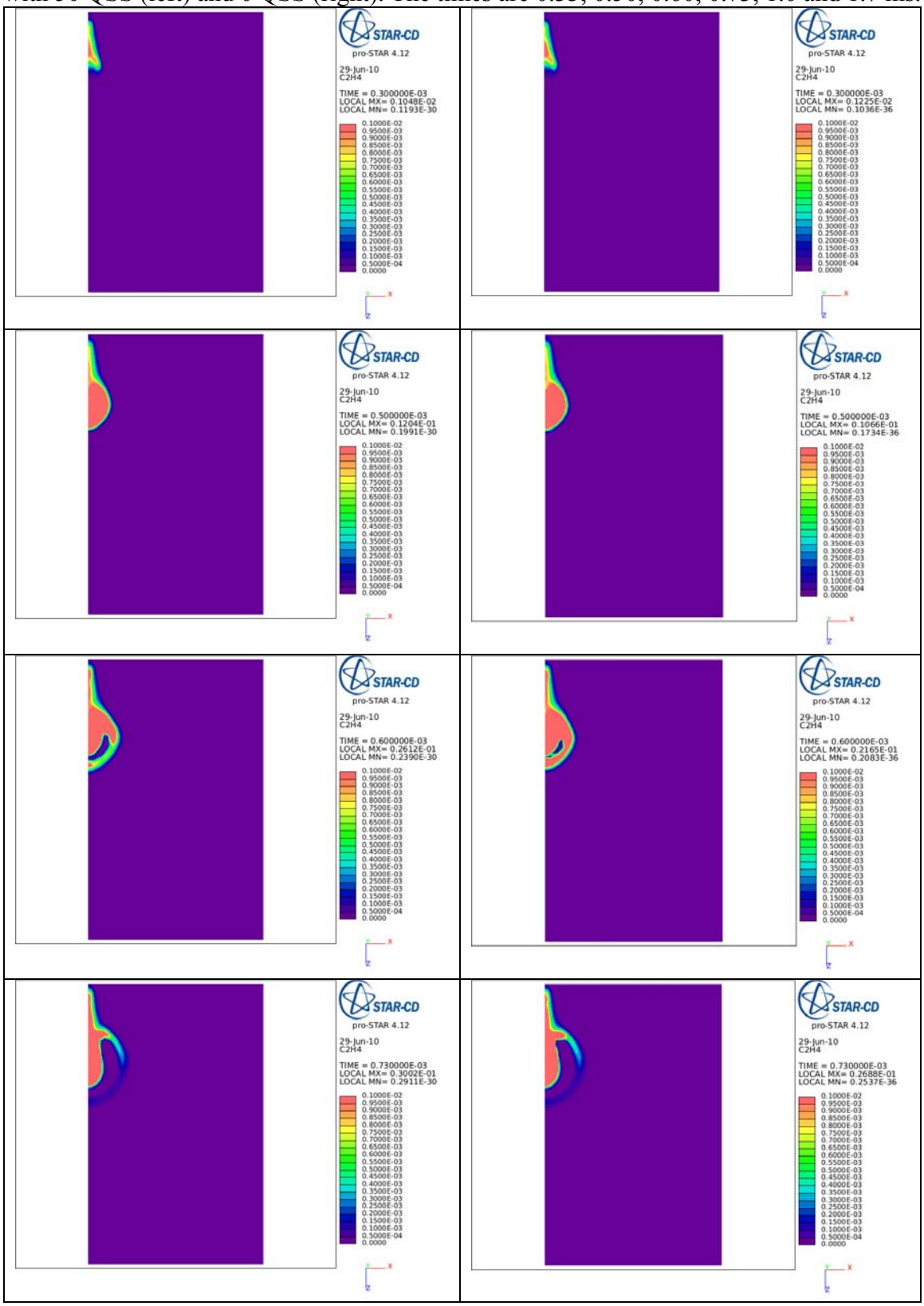

STAR-CD
pro-STAR 4.12
29-Jun-10
H2
TIME = 0.100000E-02
LOCAL MX= 0.8230E-02
LOCAL MN= 0.1134E-32
0.1000E-02
0.9500E-03
0.9000E-03
0.8500E-03
0.8000E-03
0.7500E-03
0.7000E-03
0.6500E-03
0.6000E-03
0.5500E-03
0.5000E-03
0.4500E-03
0.4000E-03
0.3500E-03
0.3000E-03
0.2500E-03
0.2000E-03
0.1500E-03
0.1000E-03
0.5000E-04
0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
H2
TIME = 0.100000E-02
LOCAL MX= 0.8166E-02
LOCAL MN= 0.0000
0.1000E-02
0.9500E-03
0.9000E-03
0.8500E-03
0.8000E-03
0.7500E-03
0.7000E-03
0.6500E-03
0.6000E-03
0.5500E-03
0.5000E-03
0.4500E-03
0.4000E-03
0.3500E-03
0.3000E-03
0.2500E-03
0.2000E-03
0.1500E-03
0.1000E-03
0.5000E-04
0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
H2
TIME = 0.170000E-02
LOCAL MX= 0.8309E-02
LOCAL MN= 0.3284E-32
0.1000E-02
0.9500E-03
0.9000E-03
0.8500E-03
0.8000E-03
0.7500E-03
0.7000E-03
0.6500E-03
0.6000E-03
0.5500E-03
0.5000E-03
0.4500E-03
0.4000E-03
0.3500E-03
0.3000E-03
0.2500E-03
0.2000E-03
0.1500E-03
0.1000E-03
0.5000E-04
0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
H2
TIME = 0.170000E-02
LOCAL MX= 0.8268E-02
LOCAL MN= 0.0000
0.1000E-02
0.9500E-03
0.9000E-03
0.8500E-03
0.8000E-03
0.7500E-03
0.7000E-03
0.6500E-03
0.6000E-03
0.5500E-03
0.5000E-03
0.4500E-03
0.4000E-03
0.3500E-03
0.3000E-03
0.2500E-03
0.2000E-03
0.1500E-03
0.1000E-03
0.5000E-04
0.0000

Figure A.6.3. Concentration of C_2H_4 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



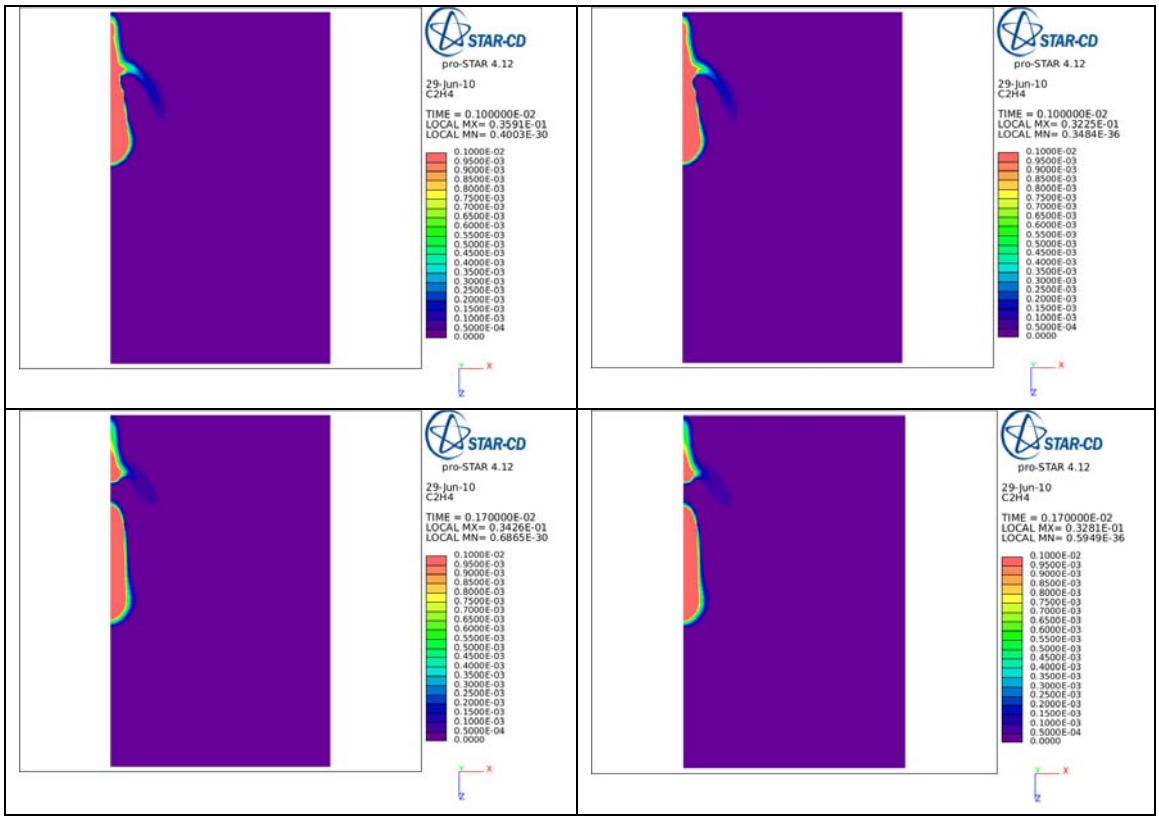
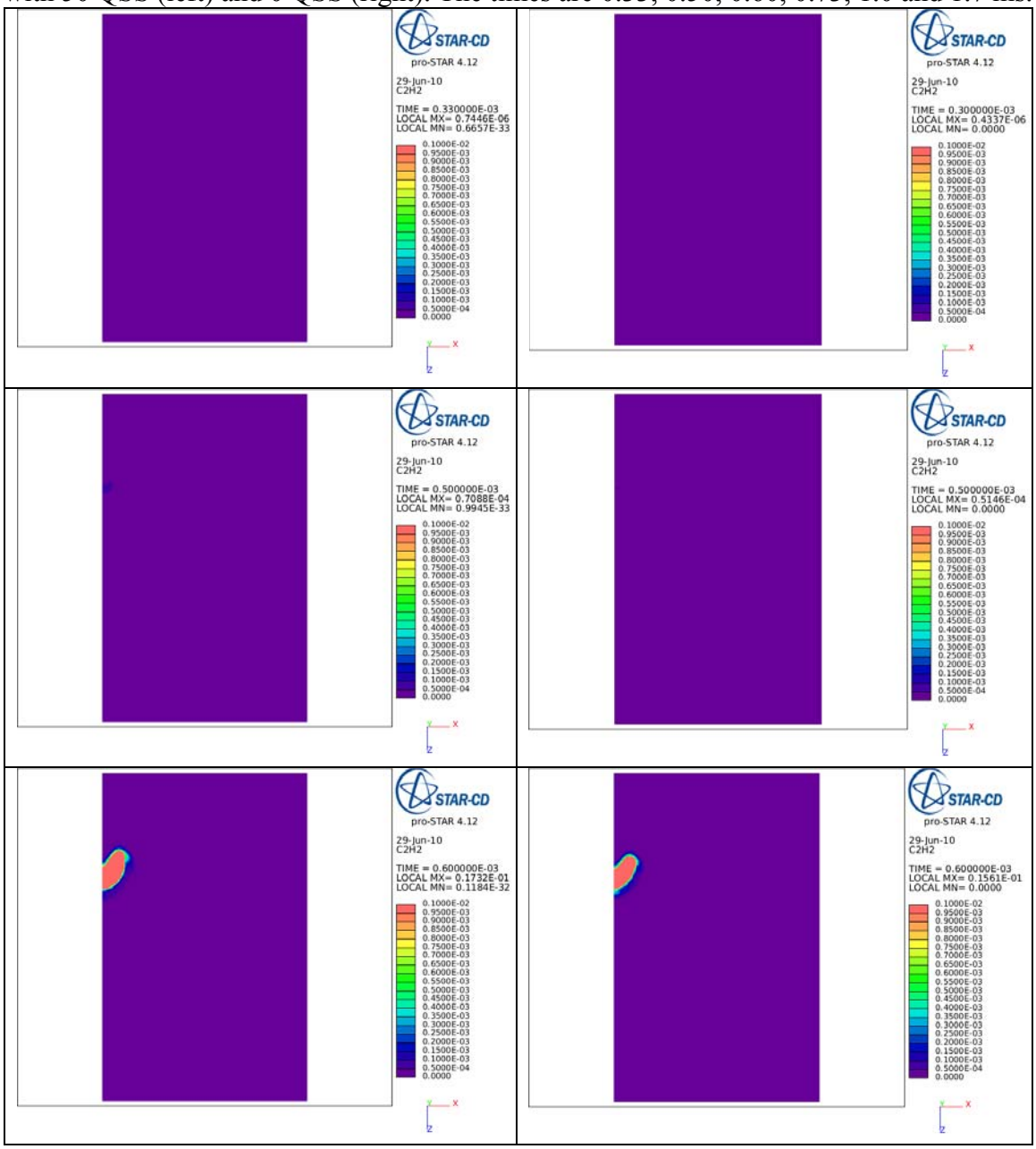

STAR-CD
pro-STAR 4.12
29-Jun-10
C2H4
TIME = 0.100000E-02
LOCAL MX= 0.3591E-01
LOCAL MN= 0.4003E-30
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H4
TIME = 0.100000E-02
LOCAL MX= 0.3225E-01
LOCAL MN= 0.3484E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H4
TIME = 0.170000E-02
LOCAL MX= 0.3426E-01
LOCAL MN= 0.6865E-30
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H4
TIME = 0.170000E-02
LOCAL MX= 0.3281E-01
LOCAL MN= 0.5949E-36

Figure A.6.2. Concentration of C_2H_2 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



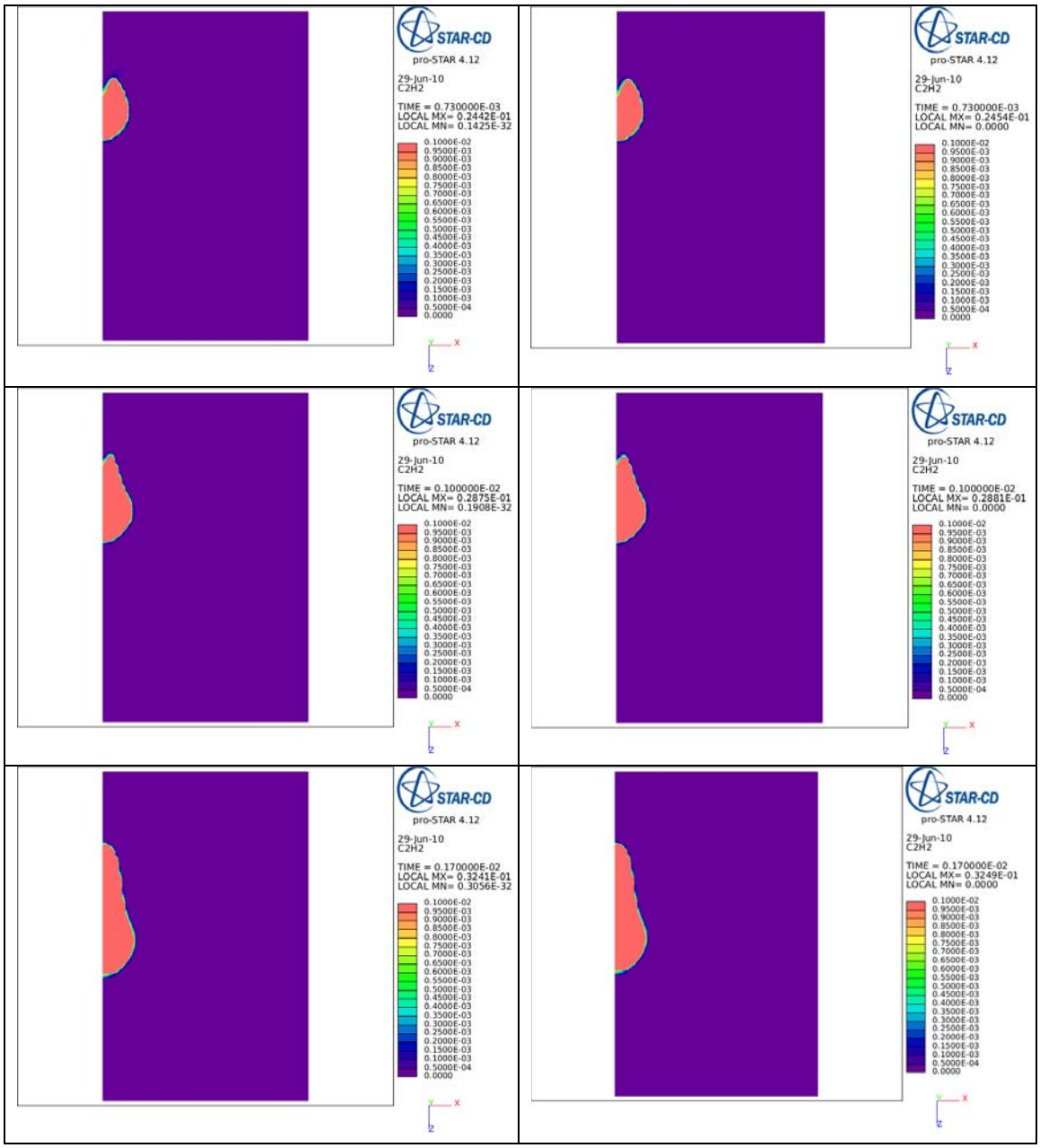
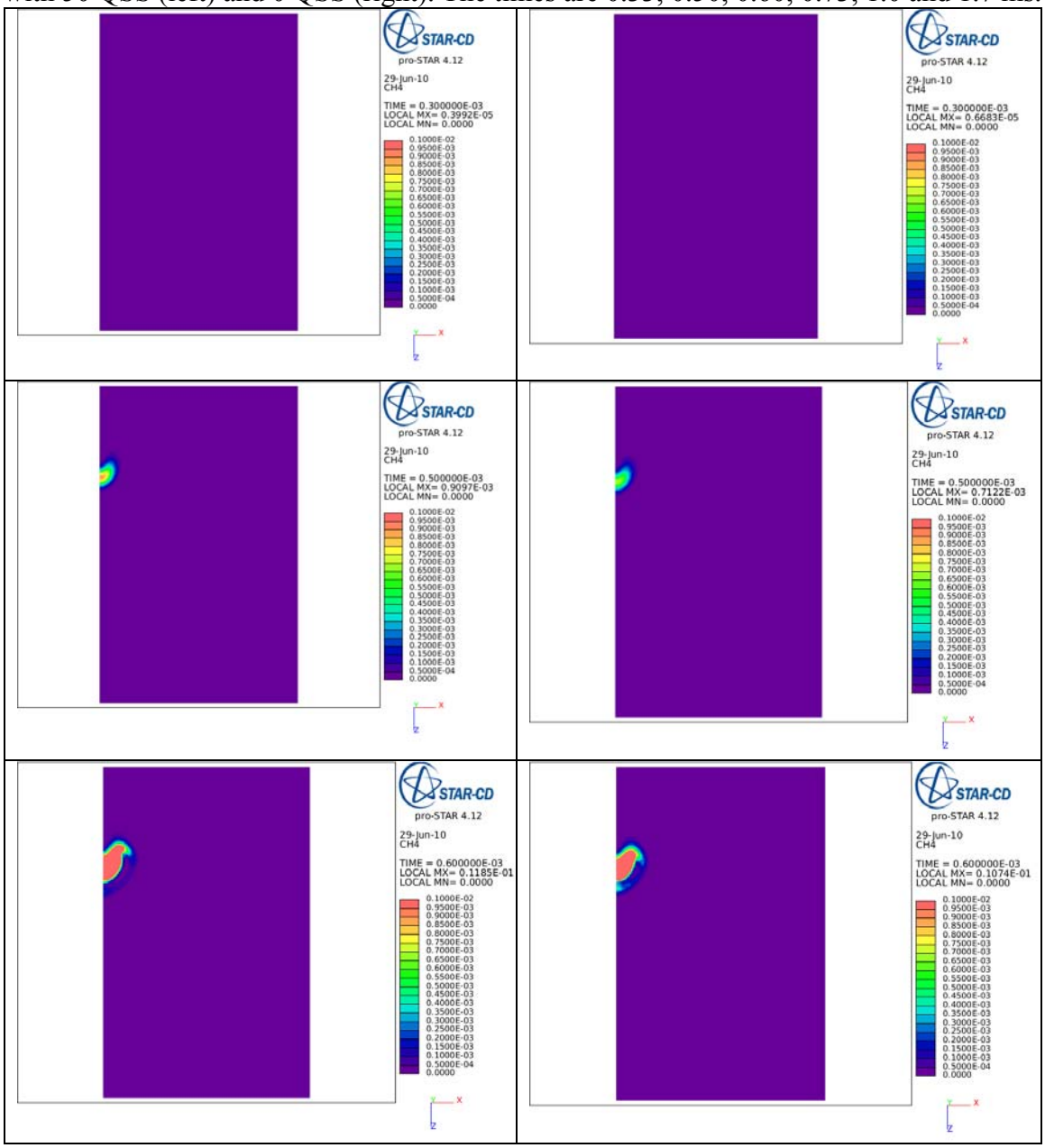

STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.730000E-03
LOCAL MX= 0.2442E-01
LOCAL MN= 0.1425E-32
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.730000E-03
LOCAL MX= 0.2454E-01
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.100000E-02
LOCAL MX= 0.2875E-01
LOCAL MN= 0.1908E-32
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.100000E-02
LOCAL MX= 0.2881E-01
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.170000E-02
LOCAL MX= 0.3241E-01
LOCAL MN= 0.3056E-32
STAR-CD
pro-STAR 4.12
29-Jun-10
C2H2
TIME = 0.170000E-02
LOCAL MX= 0.3249E-01
LOCAL MN= 0.0000

Figure A.6.4. Concentration of CH_4 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



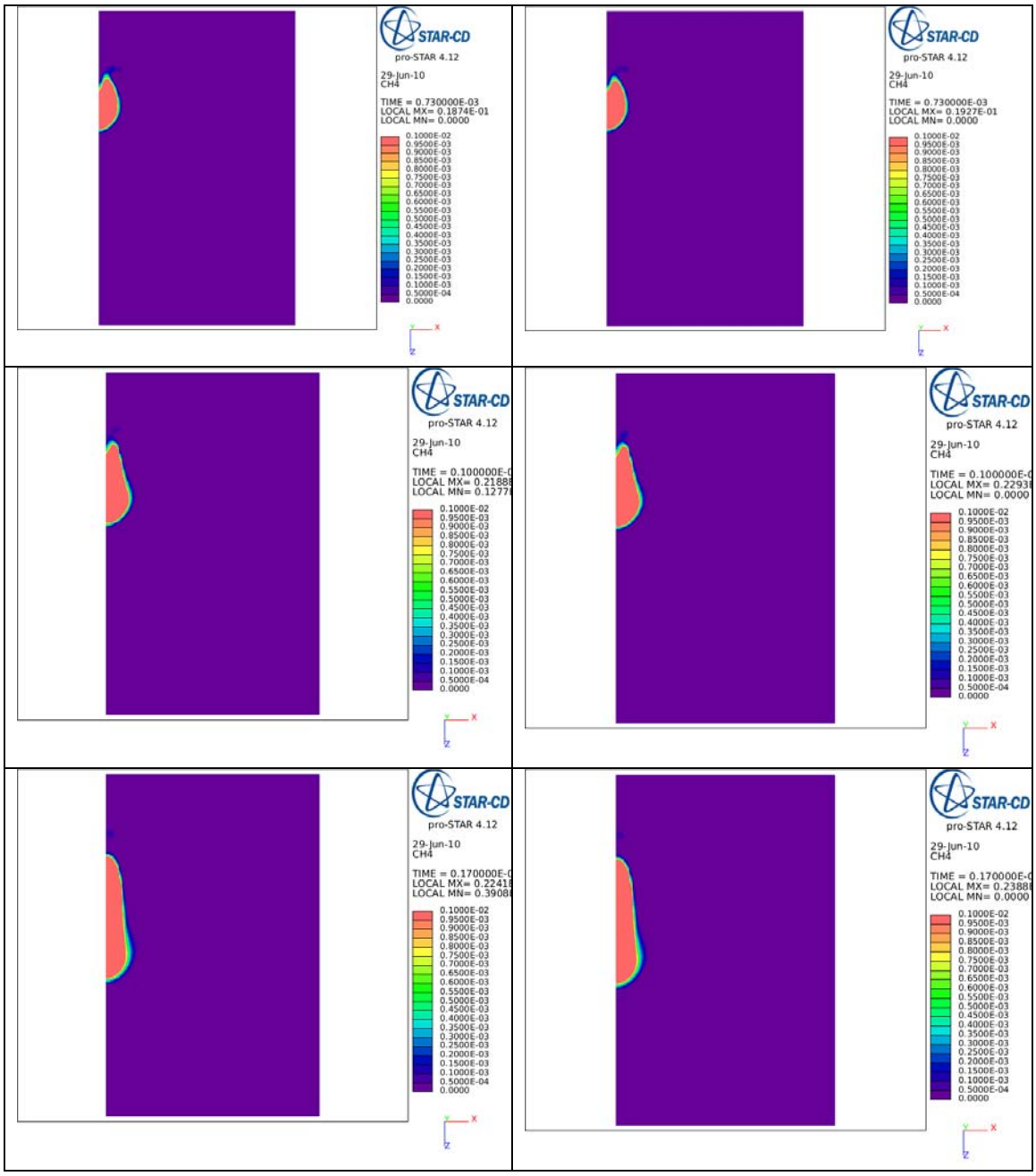
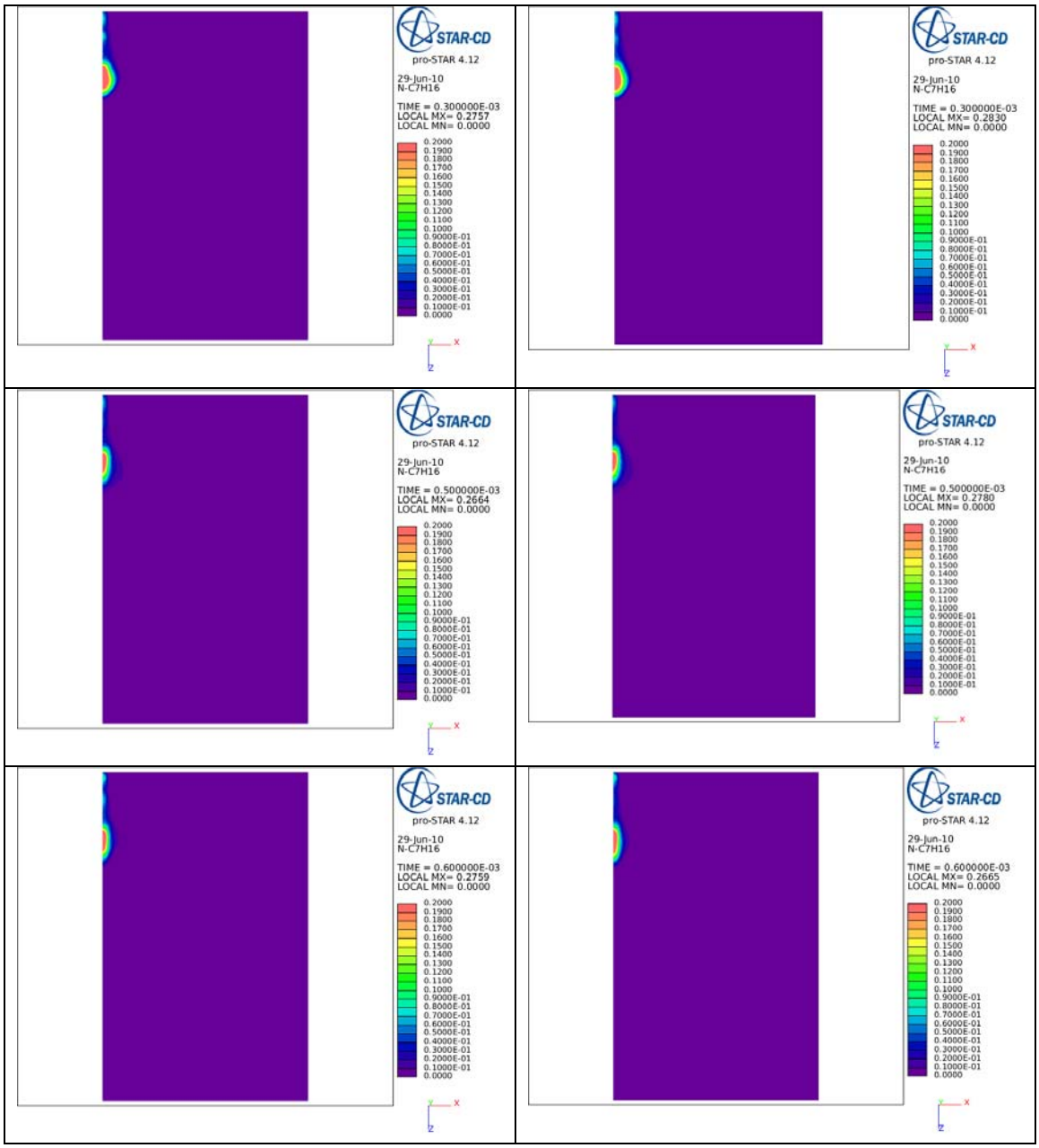

STAR-CD
pro-STAR 4.12
29-Jun-10
CH4
TIME = 0.730000E-03
LOCAL MX= 0.1874E-01
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
CH4
TIME = 0.730000E-03
LOCAL MX= 0.1927E-01
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
CH4
STAR-CD
pro-STAR 4.12
29-Jun-10
CH4
STAR-CD
pro-STAR 4.12
29-Jun-10
CH4
STAR-CD
pro-STAR 4.12
29-Jun-10
CH4

Figure A.6.5. Concentration of N-C_7H_{16} at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



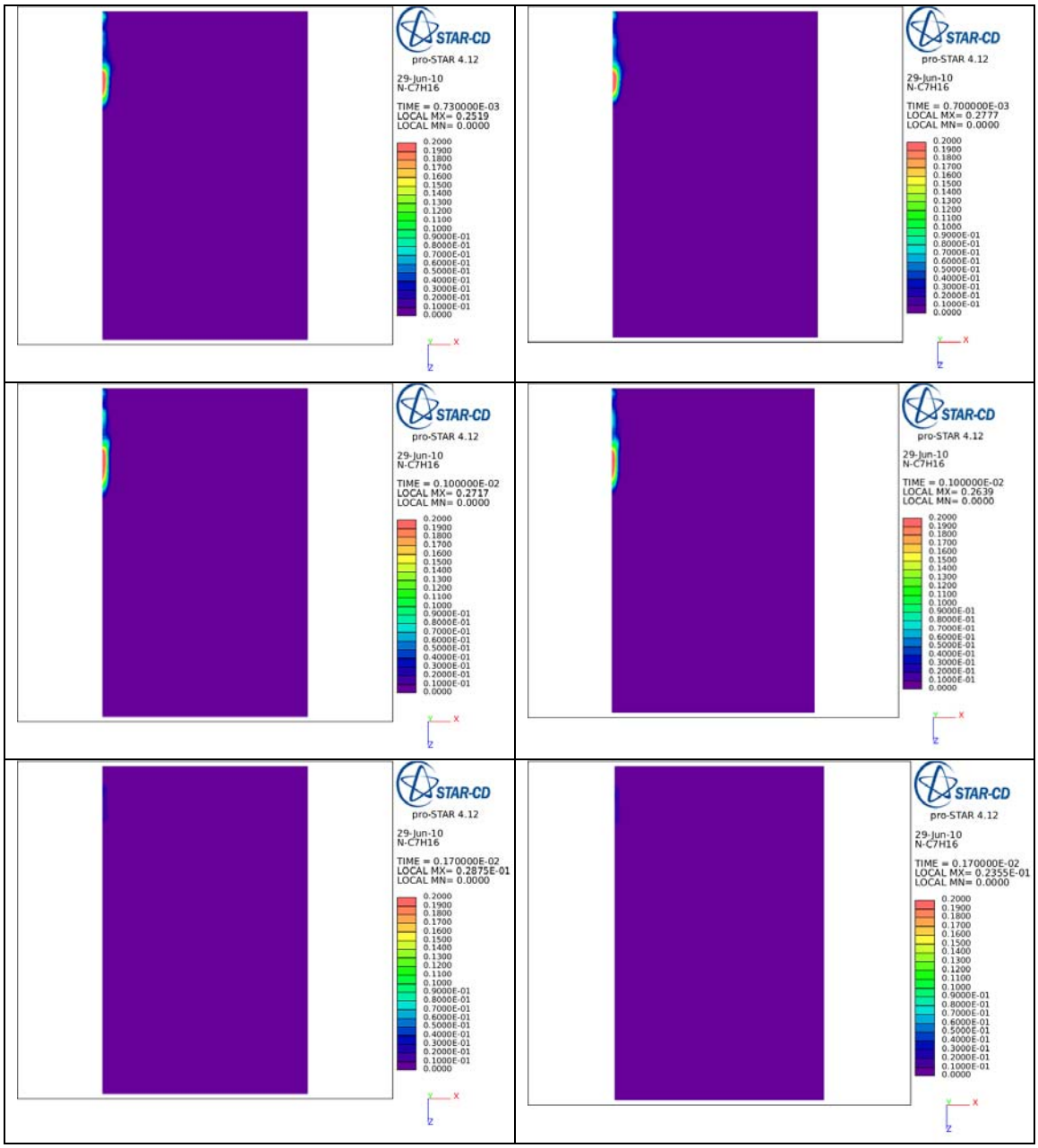
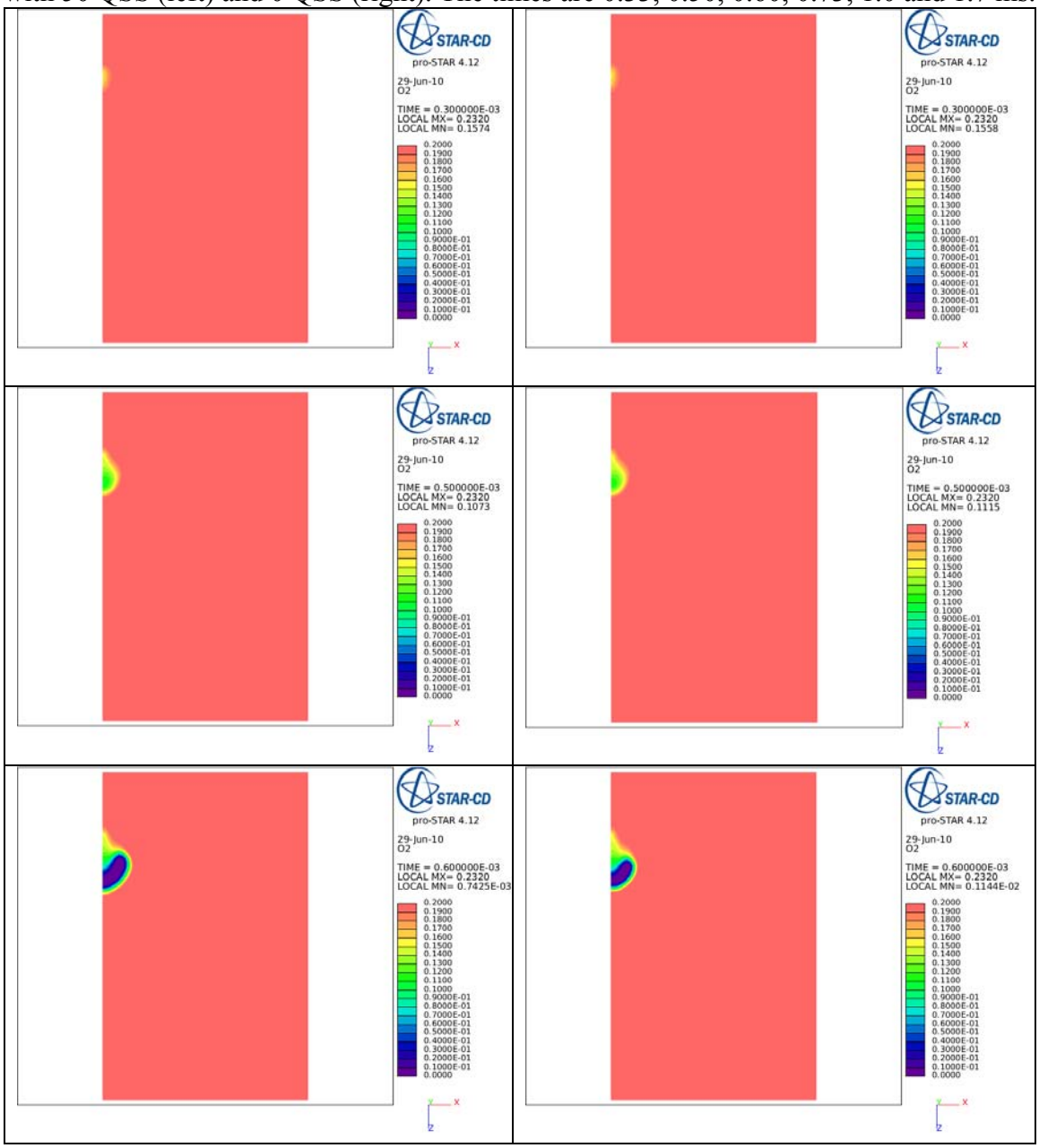

STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.730000E-03
LOCAL MX= 0.2519
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.700000E-03
LOCAL MX= 0.2777
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.100000E-02
LOCAL MX= 0.2717
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.100000E-02
LOCAL MX= 0.2639
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.170000E-02
LOCAL MX= 0.2875E-01
LOCAL MN= 0.0000
STAR-CD
pro-STAR 4.12
29-Jun-10
N-C7H16
TIME = 0.170000E-02
LOCAL MX= 0.2355E-01
LOCAL MN= 0.0000

Figure A.6.6. Concentration of O_2 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



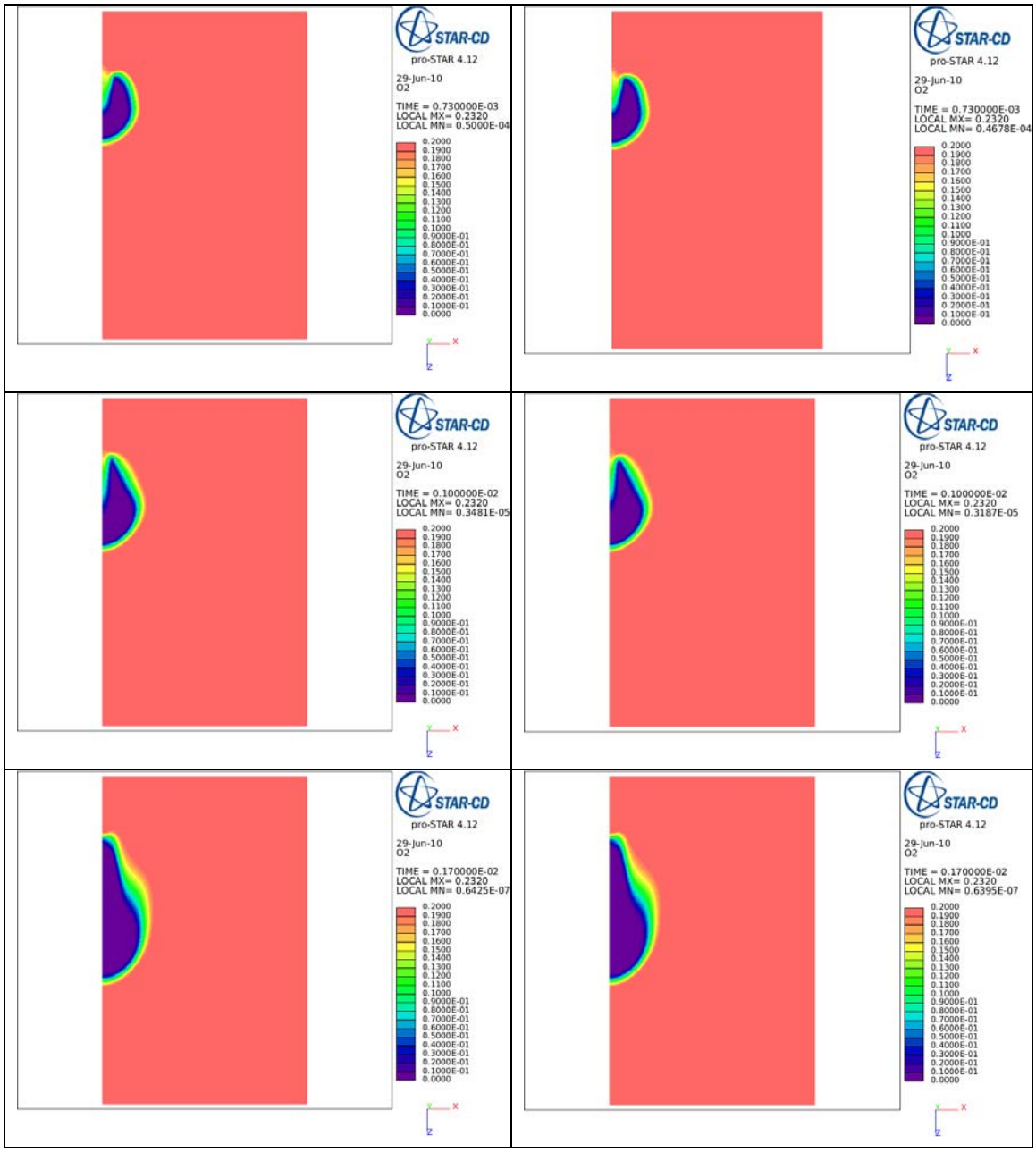
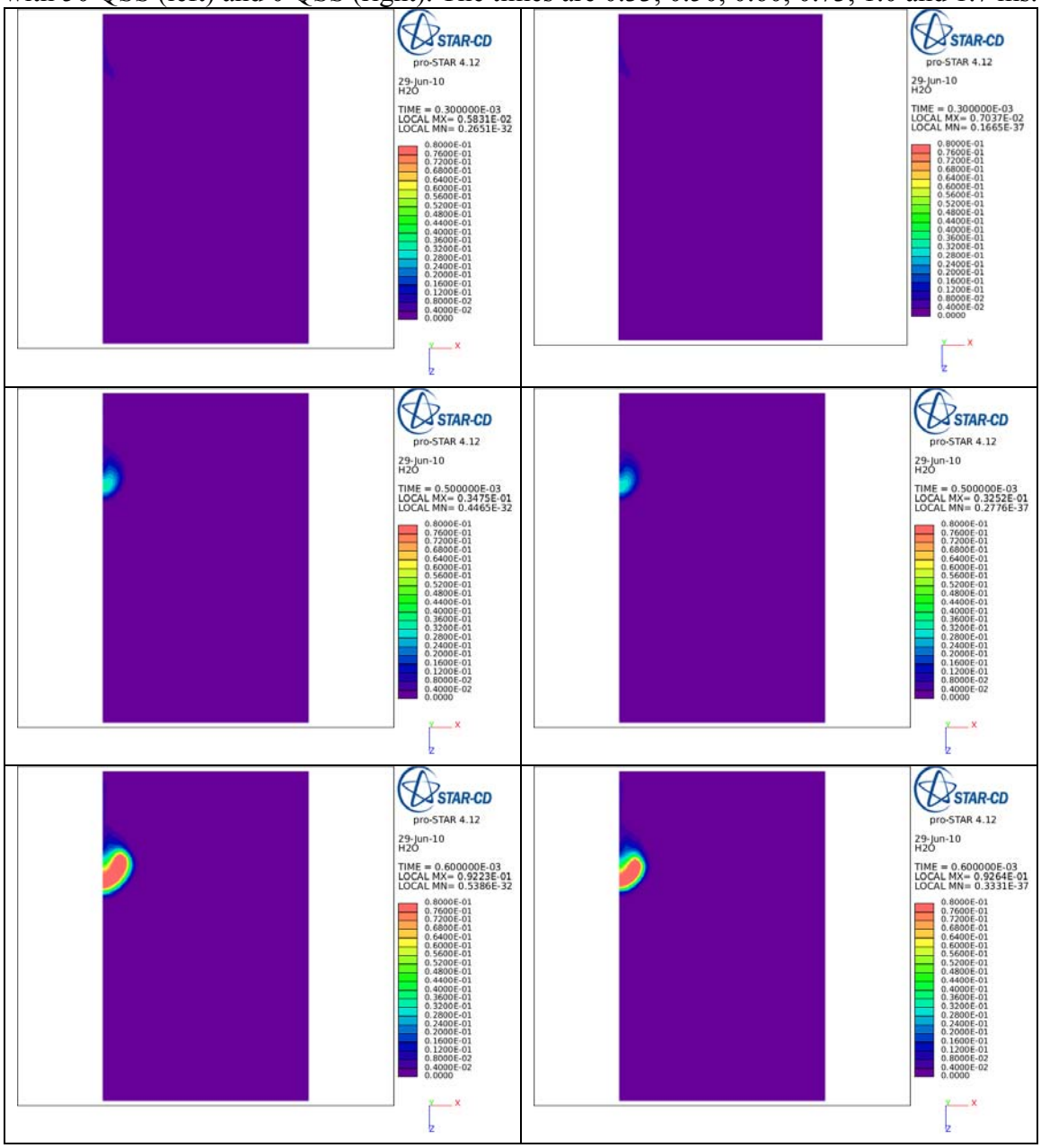

STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.730000E-03
LOCAL MX= 0.2320
LOCAL MN= 0.5000E-04
STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.730000E-03
LOCAL MX= 0.2320
LOCAL MN= 0.4678E-04
STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.100000E-02
LOCAL MX= 0.2320
LOCAL MN= 0.3481E-05
STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.100000E-02
LOCAL MX= 0.2320
LOCAL MN= 0.3187E-05
STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.170000E-02
LOCAL MX= 0.2320
LOCAL MN= 0.6425E-07
STAR-CD
pro-STAR 4.12
29-Jun-10
O2
TIME = 0.170000E-02
LOCAL MX= 0.2320
LOCAL MN= 0.6395E-07

Figure A.6.7. Concentration of H_2O at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



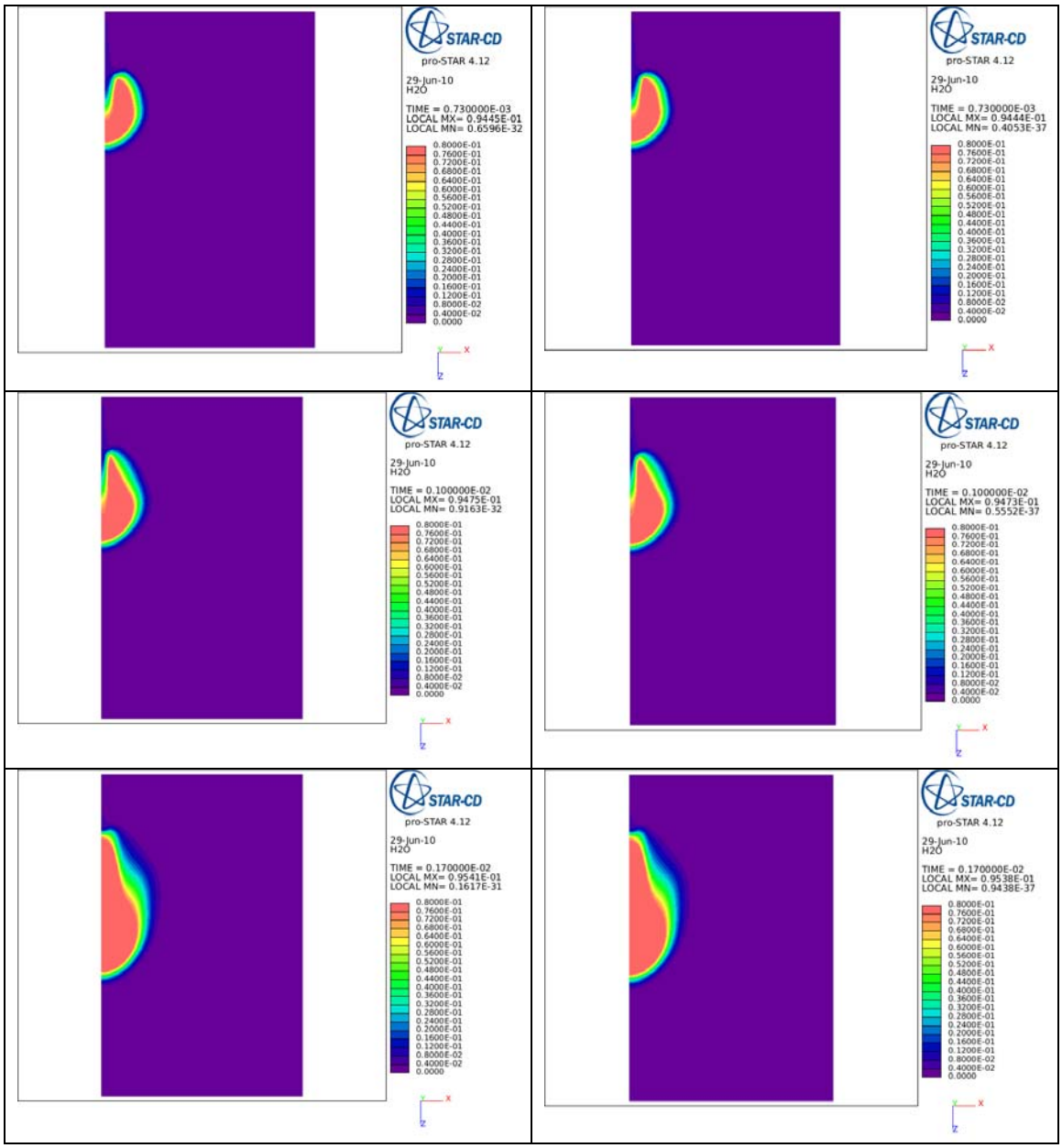
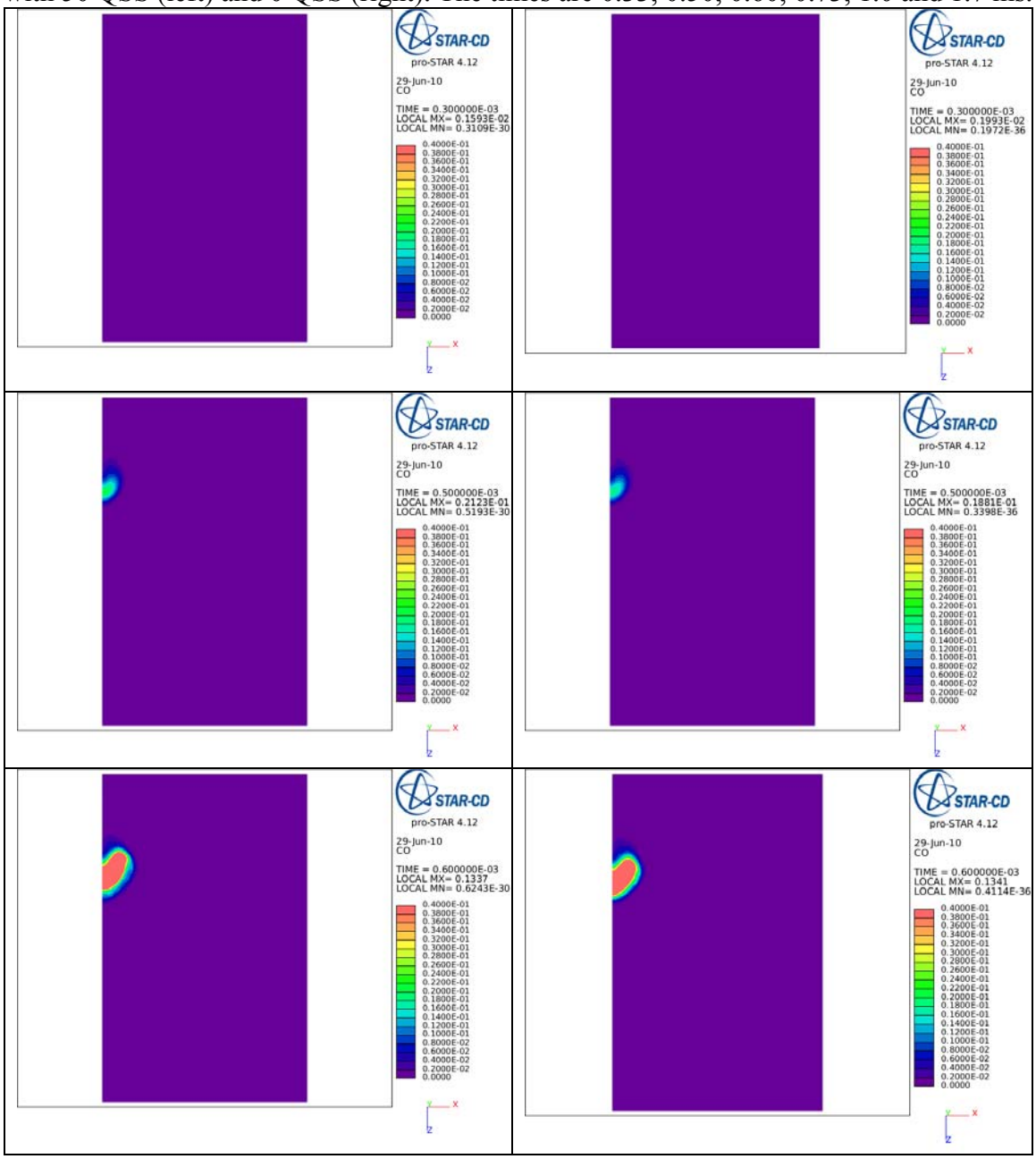

STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.730000E-03
LOCAL MX= 0.9445E-01
LOCAL MN= 0.6596E-32
STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.730000E-03
LOCAL MX= 0.9444E-01
LOCAL MN= 0.4053E-37
STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.100000E-02
LOCAL MX= 0.9475E-01
LOCAL MN= 0.9163E-32
STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.100000E-02
LOCAL MX= 0.9473E-01
LOCAL MN= 0.5552E-37
STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.170000E-02
LOCAL MX= 0.9541E-01
LOCAL MN= 0.1617E-31
STAR-CD
pro-STAR 4.12
29-Jun-10
H2O
TIME = 0.170000E-02
LOCAL MX= 0.9538E-01
LOCAL MN= 0.9438E-37

Figure A.6.8. Concentration of CO at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.



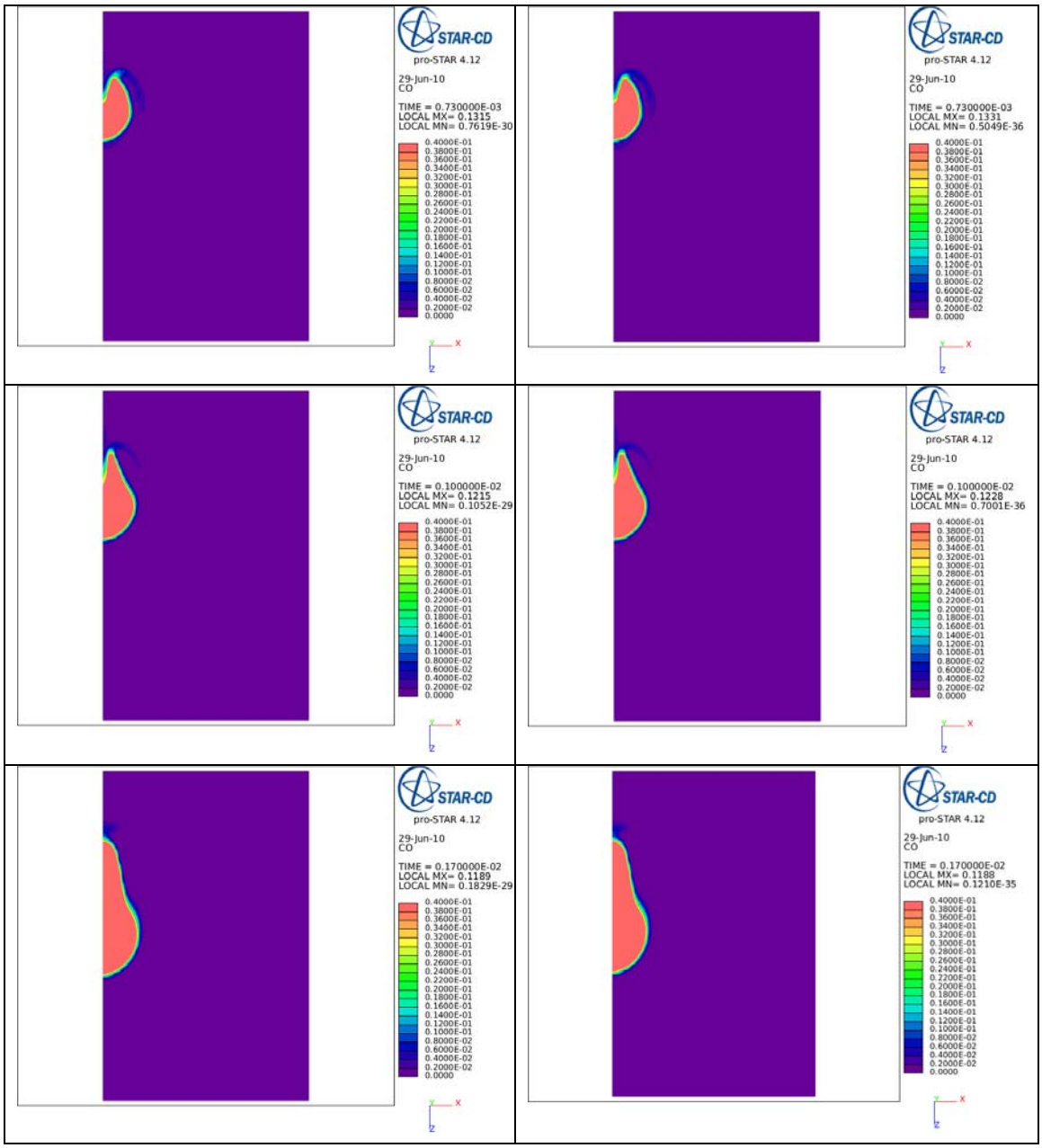
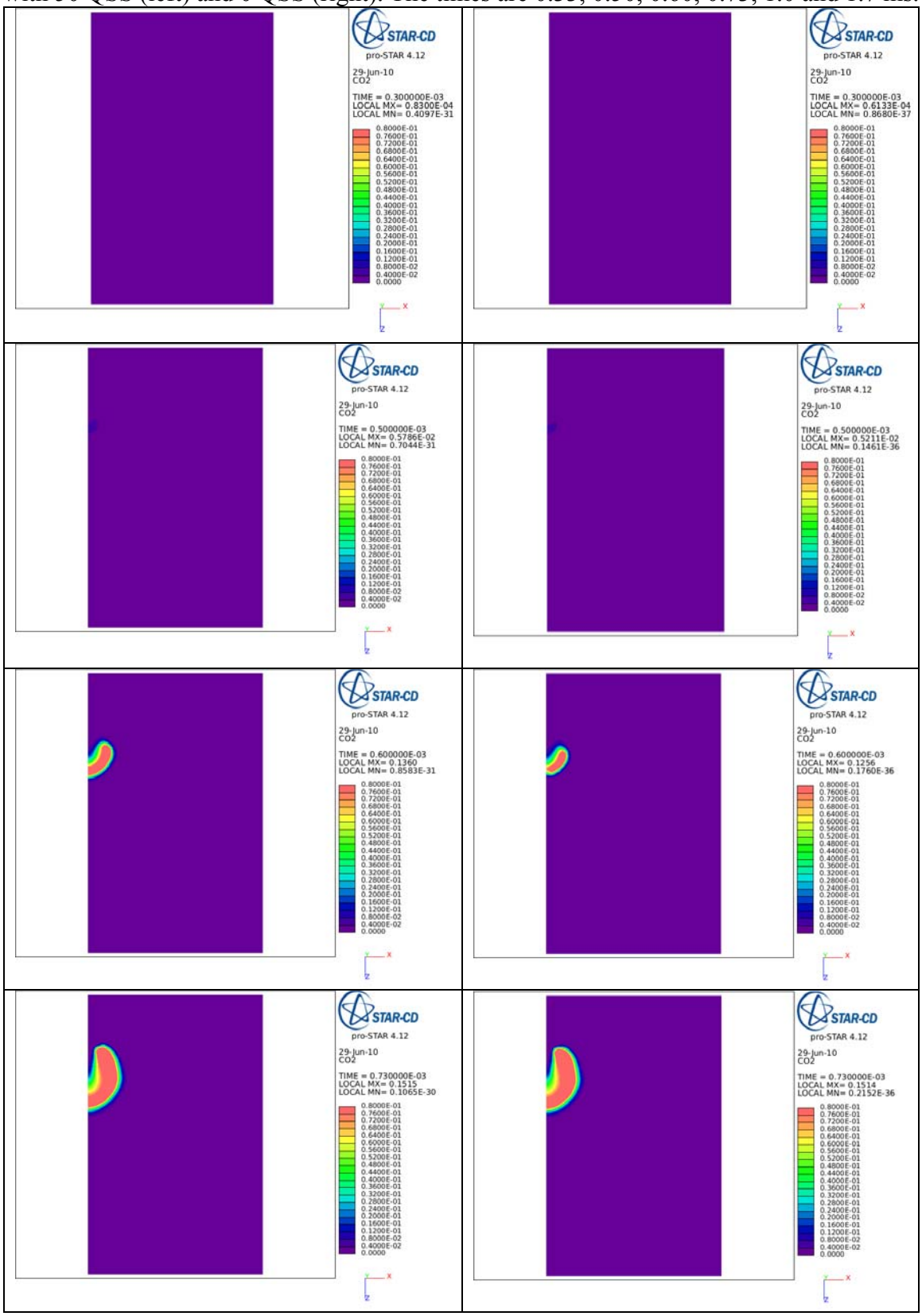
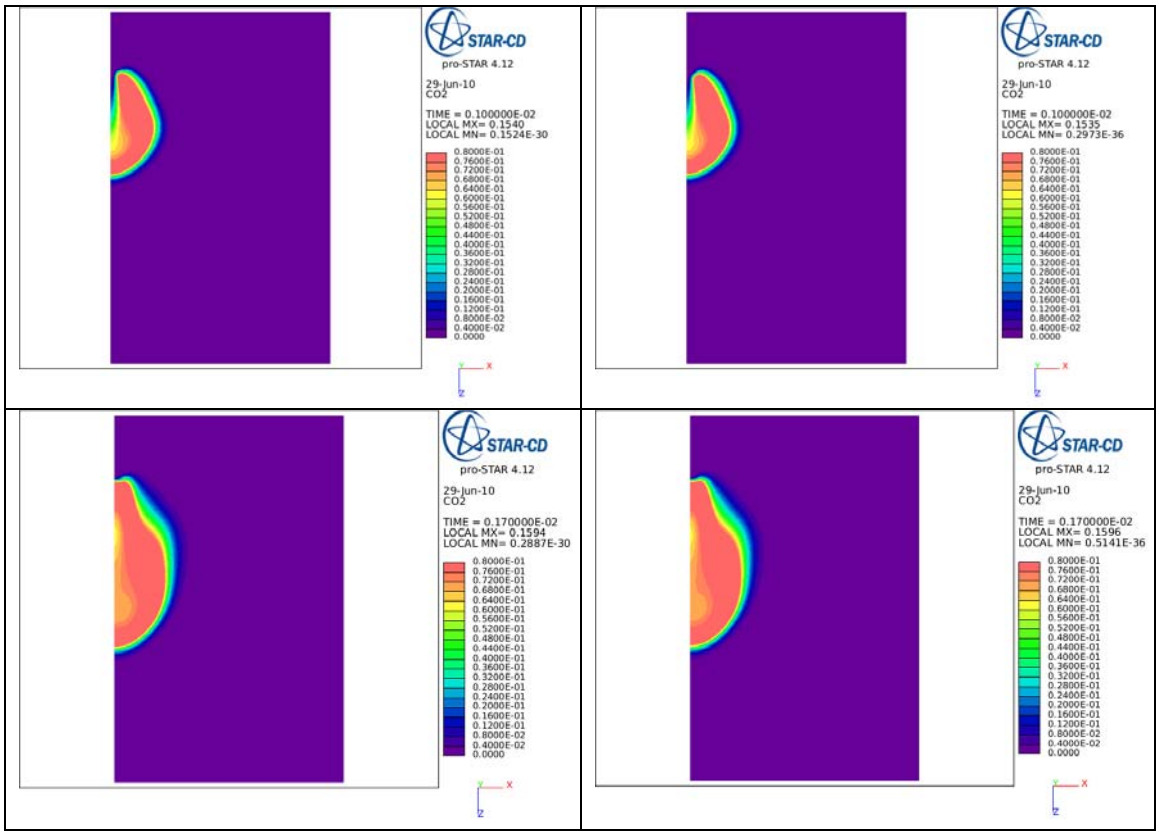

STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.730000E-03
LOCAL MX= 0.1315
LOCAL MN= 0.7619E-30
STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.730000E-03
LOCAL MX= 0.1331
LOCAL MN= 0.5049E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.100000E-02
LOCAL MX= 0.1215
LOCAL MN= 0.1052E-29
STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.100000E-02
LOCAL MX= 0.1228
LOCAL MN= 0.7001E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.170000E-02
LOCAL MX= 0.1189
LOCAL MN= 0.1829E-29
STAR-CD
pro-STAR 4.12
29-Jun-10
CO
TIME = 0.170000E-02
LOCAL MX= 0.1188
LOCAL MN= 0.1210E-35

Figure A.6.9. Concentration of CO_2 at six different times for the Newton-Newton solver with 50 QSS (left) and 0 QSS (right). The times are 0.33, 0.50, 0.60, 0.73, 1.0 and 1.7 ms.




STAR-CD
pro-STAR 4.12
29-Jun-10
CO2
TIME = 0.100000E-02
LOCAL MX= 0.1540
LOCAL MN= 0.1524E-30
STAR-CD
pro-STAR 4.12
29-Jun-10
CO2
TIME = 0.100000E-02
LOCAL MX= 0.1535
LOCAL MN= 0.2973E-36
STAR-CD
pro-STAR 4.12
29-Jun-10
CO2
TIME = 0.170000E-02
LOCAL MX= 0.1594
LOCAL MN= 0.2887E-30
STAR-CD
pro-STAR 4.12
29-Jun-10
CO2
TIME = 0.170000E-02
LOCAL MX= 0.1596
LOCAL MN= 0.5141E-36

Chapter 7.

Conclusions and Outlook

The performance of the ART and the Newton-Newton solver combination has been tested on an N-Heptane mechanism and a Methane/Propane mechanism. The CPU time and accuracy of the solution of the reduced mechanisms have been compared to a reference mechanism with 0 QSS species. The results show that the ratio between the Newton-Newton solver and the Newton solver with 0 QSS species is about 0,5 and 0,25 for the most reduced mechanism of N-Heptane and Methane/Propane respectively, while the accuracy of the solution was within acceptable limits.

The Newton-FP solver combination was tested on the N-Heptane mechanism as well and compared to the Newton-Newton solver combination. The results have also shown that the Newton-Newton solver is faster than the Newton-FP solver by a great margin and that the ratio between them, which reaches a factor 40 for the most reduced mechanisms, increases with the number of QSS species. Hence, it is advantageous to use the Newton-Newton solver instead of the Newton-FP solver and a Newton solver with 0 QSS species from a CPU time point of view.

A hypothetical solver combination with an infinitely fast inner solver is only twice as fast as the Newton-Newton solver when about 50 % of the species is in QSS. Hence, the Newton-Newton solver combination cannot be optimized much further.
The results also show that the normalized CPU time and accuracy is independent of the outer time step size for both solver combinations, which is expected since adaptive time step size is used.

Both solver combinations were tested in CFD calculations for a “real” test case, i.e. an N-Heptane injection into a constant volume vessel. A reduced mechanism with 50 QSS species was compared to an original mechanism with 0 QSS species. The result showed that the Newton-Newton solver with the reduced mechanism used about 2/3 of the CPU time that the original mechanism used, while the accuracy of the reduced mechanism was very high. This shows that the Newton-Newton solver with the reduced mechanism is advantageous to use in CFD calculations.

The result also showed that the Newton-Newton solver used about 1/50 of the CPU time that the Newton-FP solver used. This clearly rules out any use of the Newton-FP solver for CFD calculations.

The CPU time of the Newton-Newton solver depends on the Absolute and Relative tolerances. Most speed up is achieved with a low Absolute tolerance and a high Relative tolerance. Hence, the accuracy of the QSS species with low concentrations is important for the convergence of the outer solver, which in turn affects the CPU time.

One reason for the speed up of the inner solver is due to “hard coding” of the solver subroutines, which omits expensive DO-loops and IF-statements. Another reason for the speed up is the MBSA algorithm, which minimizes the number of operations in the GE and BS. The MBSA algorithm minimizes the number of operations to about 15 % of the theoretical maximum number of operations for both N-Heptane and Methane/Propane. This shows the efficiency of the MBSA algorithm.

The importance and benefits of the ART have been shown. The reduced mechanisms can be tailor made by the ART for user defined accuracy demands and physical conditions. The ART finds the species that affect the convergence of the solver, which gives the ART an advantage to traditional QSS species selection measures like LOI and LT, since those measures do not say anything about the convergence of the numerical method applied to the system of DAE.

The results show that a less reduced mechanism is obtained for larger ranges of physical conditions. This is expected since some species are important for high and others for low temperatures, pressures and fuel/air ratios.

A reduction down to 37 out of 110 species and 23 out of 118 species is achieved for the N-Heptane and Methane/Propane mechanism respectively, while the accuracy of the solution is maintained and the CPU time is significantly lower than that of the detailed mechanism. Hence, the Methane/Propane mechanism could be reduced further than the N-Heptane mechanism. The reason for this is that the N-Heptane mechanism is a skeletal mechanism, while the Methane/Propane mechanism is a detailed mechanism.

QSS species selection lists based on CF ranks the QSS species better than lists based on HF and MaxS. This is because the accuracy of the HF is affected by the accuracy of the CF. LOI ranks QSS species better than LT. This is expected since LT does not consider how sensitive a species is on the target parameter.
The best ranking performance is obtained when concentration at CF is used as a ranking measure. The reason for this is that the concentration affects all ODEs, including the ODE for HO_2, which acts as an indicator for the CF. The calculation of the species concentration cost much less than the sensitivity analysis. For this reason the concentration at CF is preferable than the LOI at CF.

Additional effects in groups of QSS species exist for the ET although the effects are small. The effects are small enough that a rough estimate of the total deviation of a

reduced mechanism can be found by simply adding the deviation of the individual QSS species included in the mechanism.

A natural step to continue this work is to;

- Use the ART to reduce various mechanisms
- Use the ART to reduce the mechanisms for problem specific physical ranges and problem specific accuracy demands
- Implement the solver combination in various CFD codes
- Do CFD simulations with various chemical mechanisms for a range of physical conditions, geometries and grid sizes.

It is interesting to investigate the CPU time gain that can be achieved for reduction of different mechanisms. Reduction of larger mechanisms is expected to give larger CPU time gain and reduction of detailed mechanism is expected to give larger CPU time gain than reduction of skeletal mechanisms.

In order to minimize to CPU time, it is important to optimize the reduction level for the specific physical ranges and specific accuracy demands of a particular problem. There are many specific problems that can be optimized.

The solver combination can be implemented in RANS, LES, DNS etc. For each CFD model the physical conditions, accuracy demands, geometries, grid sizes, chemical mechanisms and reduction level of the chemical mechanisms can be varied. This results in many possible combinations that need to be investigated and thereby simulations that must be performed.

The ART can be used for chemical mechanism with other applications than combustion, since the ART can generate a reduced mechanism from any detailed chemical mechanism. Some other applications could be chemical calculations in medicine and environmental and atmospheric science.

The Newton-Newton solver can be used in other field than combustion. The solver can basically be used for any (stiff) system of DAE where the sparseness pattern of the Jacobian is constant in time. Such systems of DAE exist within all fields of natural science, economy, applied mathematics and engineering.

Also, the MBSA algorithm has been used for image overlapping in Fluorescence Life Time Imaging applications within the field of laser diagnostics [1].

7.1. Chapter References

[1] A. Ehn, O. Johansson, J. Bood, A. Arvidsson, B. Li, M. Aldén, *Fluorescence life time imaging in a flame*, Proc. Combust. Inst., (201

Acknowledgements

I would like to thank the Centre for Combustion Science and Technology (CECOST) and the Swedish Research Council (VR) for financing the project behind my thesis. I would like to thank Professor Markus Aldén for giving me an opportunity to work at the Division of Combustion Physics and for being very flexible in order to solve several problems during my PhD studies.
I would like to thank my supervisor Professor Fabian Mauss for helping me during the PhD studies with interesting ideas and insightful comments. I would like to thank LOGE with staff for facilitating my PhD studies.
Also, a special thank you to Karin Fröjd, Anders Borg, Elna Heimdal Nilsson and Sven-Inge Möller for fruitful discussions and for the help they have given me with theoretical and practical problems during my PhD studies.
Also, a special BIG thank you to Harry Lehtiniemi who helped me more than I could have asked for.

Also, I would like to thank my physics teachers during my master education; Hans-Uno Bengtsson, Lars Gislen, Bo Söderberg and Carl-Erik Magnusson for giving me a solid master education in physics and inspiring me to further studies in physics.
Also, I would like to thank Björn Samuelsson for giving me a deeper understanding of physics and for being a reliable source of knowledge and reason in just about every aspect of science and life.

I would like to thank Ingemar Magnusson, Edward Blurock, Per-Erik Bengtsson, Alexander Konnov, Kalle Netzell and Terese Lövås for interesting discussions and their help with my thesis.
I would like to thank all the guys at the Division of Combustion Physics and Piero Iudiciani, Eric Baudoin, Holger Grosshans, Tobias Joelsson, Rixin Yu, David Sedarsky, Martin Tuner, Abdelhadi Ahmedi, Syed Sayeed Ahmed, Ngosi Ebeneezer, Pirooz Moradnia, Gladys Moréac, Thomas Zeuch and Raffaela Bellanca for being great friends.
I would like to thank Cecilia Bille, Anneli Nilsson, Eva Persson and Minna Ramkull for supporting me and helping me with practical problems during my studies.

I would like to thank Andreas Ehn, Joakim Bood and Olof Johansson for implementing the MBSA algorithm in the field of laser diagnostics and for helping me with my thesis.
Also, I would like to give them, Sven-Inge Möller and Edouard Berrocal a special THANK YOU for being great friends and for all the “intelligent” conversations at the coffee table.
Also, I would like to thank Fedor Emelianenko and AC/DC for all the happy moments in front of the computer.

Finally, I would like to thank Birgitta Arvidsson for being the best mother in the world and helping me with everything.

To the memory of my grandparents:
Bengt Arvidsson and Andrea Arvidsson