



Karlsruher Institut für Technologie  
Fakultät für Chemieingenieurwesen und Verfahrenstechnik  
Engler-Bunte-Institut  
Bereich Verbrennungstechnik  
Prof. Dr.-Ing. Henning Bockhorn

## **Bachelorarbeit**

# **Weiterentwicklung und Optimierung eines auf OpenFOAM basierten DNS Lölers zur Verbesserung der Effizienz und Handhabung**

von

**cand. chem. ing. Thorsten Zirwes**

---

Hochschullehrer: Prof. Dr.-Ing. Henning Bockhorn

Betreuer: Dipl.-Ing. Feichi Zhang

Karlsruhe, den 31. Oktober 2013





Karlsruher Institut für Technologie  
Fakultät für Chemieingenieurwesen und Verfahrenstechnik  
Engler-Bunte-Institut  
Bereich Verbrennungstechnik  
Prof. Dr.-Ing. Henning Bockhorn

## **Aufgabenstellung zur Bachelorarbeit**

and. chem. ing. Thorsten Zirwes

### **Weiterentwicklung und Optimierung eines auf OpenFOAM basierten DNS Lösers zur Verbesserung der Effizienz und Handhabung**

Ziel der Arbeit ist die Verbesserung eines vorhandenen Lösers für die Direkte Numerische Simulation (DNS) von turbulenten, reagierenden Strömungen in der Entwicklungsumgebung OpenFOAM. Dazu sollen die relevanten Routinen aus der mitverwendeten Bibliothek Cantera identifiziert und direkt in OpenFOAM integriert werden. Zudem soll der extrahierte Canteracode optimiert werden, um die Rechengeschwindigkeit des Lösers zu steigern. Danach ist der Löser anhand verschiedener Testfälle zu validieren. Die Verbesserungen des neuen Lösers bezüglich Effizienz und Handhabung im Vergleich zum vorhandenen DNS Löser sind zu verifizieren. Zur Durchführung der Simulationen soll der Supercomputer Blue Gene/Q JUQUEEN des Forschungszentrums Jülich genutzt werden.

Die Arbeit gliedert sich in die folgenden Arbeitsschritte:

- Einarbeitung in die Theorie der Verbrennung und DNS sowie in OpenFOAM und den vorhandenen Lösercode
- Direkte Implementierung der relevanten Codeabschnitte aus Cantera in den DNS Löser
- Profiling des Lösers zum Aufspüren von Leistungsentpässen
- Optimierung der Löserrechenzeit basierend auf den Profilingergebnissen
- Untersuchung der Reaktionsratenberechnung durch Mittelung über den DNS Zeitschritt sowie ohne Integration der Reaktorgleichung
- Validierung des neuen Lösers anhand von vorgemischten Wasserstoff/Luft Flammen
- Untersuchung des Einflusses verschiedener Reaktionsmechanismen auf die Rechenzeit
- Beschreibung von Theorie und Implementierung sowie Darstellung und Diskussion der Ergebnisse in einer schriftlichen Ausarbeitung
- Vorstellung der Bachelorarbeit in einem öffentlichen Seminarvortrag

Betreuer: Dipl.-Ing. Feichi Zhang

Aufgabensteller: Prof. Dr.-Ing. Henning Bockhorn

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig angefertigt habe.  
Ich habe keine anderen als die angegebenen Hilfsmittel und Quellen verwendet.

Ich erkläre mich damit einverstanden, dass meine Arbeit in die Bibliothek eingestellt  
und kopiert werden darf.

**Karlsruhe, den 31. Oktober 2013 .....**

**(Thorsten Zirwes)**

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>18</b>
1.1	Motivation . . . . .	18
1.2	Ziele . . . . .	20
1.3	Aufbau der Arbeit . . . . .	22
<b>2</b>	<b>Grundgleichungen</b>	<b>23</b>
2.1	Allgemeine Form der Erhaltungsgleichungen . . . . .	23
2.2	Gesamtmassenerhaltung . . . . .	24
2.3	Impulserhaltung . . . . .	24
2.4	Speziesmassenerhaltung . . . . .	27
2.4.1	Massenflüsse . . . . .	27
2.4.2	Mischungsgemittelte Diffusionskoeffizienten . . . . .	27
2.4.3	Korrekturgeschwindigkeit . . . . .	29
2.4.4	Reaktionsraten . . . . .	29
2.4.4.1	Geschwindigkeitskonstante der Rückreaktion . . . . .	31
2.4.4.2	Druckabhängigkeit der Geschwindigkeitskonstanten . . . . .	32
2.4.4.3	Dreistoß- und Doppelreaktionen . . . . .	33
2.5	Energieerhaltung . . . . .	34
2.5.1	Energiefluss . . . . .	34
2.5.2	Wärmeleitfähigkeit . . . . .	36
2.5.3	Wärmekapazität und Enthalpie . . . . .	38
2.5.4	Ermittlung der Temperatur . . . . .	40
2.6	Zustandsgleichung . . . . .	41

2.7	Lösung des Gleichungssystems . . . . .	41
<b>3</b>	<b>Weiterentwicklung des DNS Löser</b>	<b>43</b>
3.1	Softwarepakete und Cluster-Plattformen . . . . .	44
3.1.1	OpenFOAM . . . . .	44
3.1.2	Cantera . . . . .	45
3.1.3	Cluster-Plattformen . . . . .	46
3.2	Basislöser . . . . .	47
3.2.1	Aufbau des Basislösers . . . . .	48
3.2.2	Portierung nach den neusten Programmversionen . . . . .	51
3.2.3	Profiling . . . . .	54
3.3	Extraktion des relevanten Codes aus Cantera . . . . .	57
3.4	Optimierung des extrahierten Cantera Codes . . . . .	61
3.4.1	Spezialisierung des Codes . . . . .	61
3.4.2	Optimierung der Geschwindigkeitskonstantenberechnung . . . . .	63
3.4.3	Optimierung der Enthalpieberechnung . . . . .	63
<b>4</b>	<b>Berechnung der Reaktionsraten</b>	<b>67</b>
4.1	Mittelung der Reaktionsraten über den DNS Zeitschritt . . . . .	67
4.1.1	Interner Zeitschritt von SUNDIALS . . . . .	71
4.1.2	Einfluss der Toleranzparameter . . . . .	74
4.1.2.1	Absolute Toleranz . . . . .	74
4.1.2.2	Relative Toleranz . . . . .	76
4.2	Direkte Berechnung der Reaktionsraten . . . . .	77
<b>5</b>	<b>DNS vorgemischter Wasserstoff/Luft Flammen</b>	<b>79</b>
5.1	Flachflamme (1D) . . . . .	80
5.1.1	Numerische Einstellungen . . . . .	80
5.1.2	Vergleich der Ergebnisse mit CHEMKIN . . . . .	81
5.1.3	Einfluss der Reaktionsmechanismen auf die Rechenzeit . . . . .	84
5.2	Sphärische Flammenausbreitung im Bombenbehälter (3D) . . . . .	88
5.2.1	Experimenteller Aufbau . . . . .	88

5.2.2	Numerische Einstellungen . . . . .	89
5.2.3	Darstellung der Ergebnisse . . . . .	92
5.2.4	Einfluss der Gitterauflösung . . . . .	98
5.2.5	Skalierungsverhalten auf der HPC-Plattform JUQUEEN . . . . .	99
<b>6</b>	<b>Zusammenfassung und Ausblick</b>	<b>101</b>
6.1	Zusammenfassung . . . . .	101
6.2	Ausblick . . . . .	103
<b>A</b>	<b>Code-Ausschnitte</b>	<b>105</b>
<b>B</b>	<b>Reaktionsmechanismen</b>	<b>106</b>
B.1	Li et al. . . . .	106
B.2	Curran et al. . . . .	107
B.3	Maas und Pope . . . . .	109
B.4	Kee et al. . . . .	110
B.5	Kathrotia et al. . . . .	111

# Abbildungsverzeichnis

1.1	Energieprognose Deutschland . . . . .	19
1.2	Überblick über die verschiedenen Zweige der CFD . . . . .	20
3.1	Programmablaufplan . . . . .	50
3.2	Performanceunterschiede der Cantera-Versionen . . . . .	51
3.3	Profiling des Löser . . . . .	54
3.4	Anteil an der Gesamtlaufzeit einzelner Funktionen . . . . .	55
3.5	Ausschnitt aus dem Callstack für die Exponentialfunktion . . . . .	56
3.6	Vererbungshierarchie der Cantera-Klasse ThermoPhase . . . . .	57
3.7	Abhängigkeiten der ThermoPhase Quellcodedatei . . . . .	58
3.8	Vernetzungen in Cantera nach Ausbauen der nicht relevanten Teile . . . . .	60
4.1	SUNDIALS Flussdiagramm . . . . .	72
4.2	Verteilung der internen Zeitschritte . . . . .	73
4.3	Einfluss der absoluten Toleranzen auf die Rechenzeit . . . . .	74
4.4	Einfluss der relativen Toleranzen auf die Rechenzeit . . . . .	76
4.5	Vergleich der Rechenzeit mit direkter Berechnung der Reaktionsraten . . . . .	77
5.1	Numerische Einstellungen der Flachflamme . . . . .	80
5.2	Validierung der Reaktionskinetik . . . . .	82
5.3	Vergleich von Temperatur und Wärmefreisetzungsrate mit CHEMKIN . . . . .	83
5.4	Vergleich der Rechenzeit verschiedener Reaktionsmechanismen . . . . .	86
5.5	Vergleich der Rechenzeit verschiedener Reaktionsmechanismen (normiert) . . . . .	87
5.6	Einfluss der Speziesanzahl auf die Rechenzeit . . . . .	88

5.7	Aufbau des Bombenbehälters . . . . .	89
5.8	Vereinfachtes Rechengitter der 3D DNS . . . . .	90
5.9	Vektorfeld der Strömungsgeschwindigkeit . . . . .	92
5.10	Zeitliche Entwicklung von Temperatur und Wärmefreisetzungsrate . . .	93
5.11	Vergleich der momentanen Flammenfront mit dem Experiment . . . . .	94
5.12	Darstellung der Flammengeschwindigkeit über den Flammenradius . .	95
5.13	Zeitlicher Druckverlauf im Bombenbehälter . . . . .	96
5.14	Zeitlicher Verlauf des Druckgradienten . . . . .	97
5.15	Gitterabhängigkeit . . . . .	98
5.16	Skalierungsmessung des Lösers auf JUQUEEN . . . . .	100

# Tabellenverzeichnis

3.1	Übersicht über die Werte, die vom Löser eingelesen werden . . . . .	47
3.2	Übersicht über die Löserabschnitte . . . . .	48
4.1	Chemische Zeitschritte . . . . .	71
4.2	Fehler in Abhängigkeit absoluter Toleranzen . . . . .	75
5.1	Anfangsbedingungen der 1D Simulation . . . . .	81
5.2	Flammengeschwindigkeit und -dicke . . . . .	84
5.3	Vergleich der Reaktionsmechanismen . . . . .	85
5.4	Betriebsbedingungen von Experiment und DNS . . . . .	89
5.5	Werte der Wandbedingungen . . . . .	91
5.6	Numerische Bedingungen . . . . .	91
5.7	Flammengeschwindigkeit im Bombenbehälter . . . . .	96

# Verzeichnis der Codeausschnitte

3.1	Alte Funktion zur Berechnung der Standardbildungsenthalpie der Mischung. . . . .	64
3.2	Optimierte Funktion zur Berechnung der Standardbildungsenthalpie der Mischung. . . . .	64
3.3	Ursprüngliche Funktion zur Berechnung der thermischen Speziesenthalpie. . . . .	65
3.4	Neuen Funktionen zur Berechnung der thermischen Enthalpien und der Standardbildungsenthalpien der reinen Spezies. . . . .	66
A.1	Vereinfachter Code der Funktion <code>_update_rates_T</code> zur Berechnung der Geschwindigkeits- und Gleichgewichtskonstanten. . . . .	105

# Nomenklatur

## Lateinische Symbole

Symbol	Bedeutung	Einheit
$A$	Frequenzfaktor	variiert
$a$	Koeffizient für Polynome	variiert
$C$	molare Konzentration	mol/m <sup>3</sup>
$c_p$	isobare Wärmekapazität	J/kg/K
$\tilde{c}_p$	isobare Wärmekapazität (molar)	J/mol/K
$c_v$	isochore Wärmekapazität	J/kg/K
$\tilde{c}_v$	isochore Wärmekapazität (molar)	J/mol/K
$D_k$	mischungsgemittelter Diffusionskoeffizient	m <sup>2</sup> /s
$\mathcal{D}_{k,i}$	binäre Diffusionskoeffizienten	m <sup>2</sup> /s
$E$	Effizienz der Skalierung	1
$E_a$	Aktivierungsenergie	J/mol
$G^\circ$	Standard-Gibbs-Energie	J/mol
$h$	absolute Enthalpie	J/kg
$\tilde{h}$	absolute Enthalpie (molar)	J/mol
$h_s$	thermische Enthalpie	J/kg
$\tilde{h}_s$	thermische Enthalpie (molar)	J/mol
$I$	Einheitsmatrix	1
$\vec{j}_q$	Energiefluss	J/m <sup>2</sup> /s
$\vec{j}_k$	Massendiffusionsfluss der Spezies $k$	kg/m <sup>2</sup> /s
$K_r$	Gleichgewichtskonstante	1
$k_r$	Geschwindigkeitskonstante	variiert
$M$	Molmasse	kg/mol
$M$	Anzahl an Reaktionen	1
$M$	Kollisionspartner in Dreistoßreaktion	-
$\mathcal{M}_k$	Name der Spezies $k$	-

$m_k$	Masse eines Speziesmoleküls	kg
$m_{i,k}$	reduzierte Masse zweier Moleküle	kg
$\dot{m}$	Massenfluss	kg/m <sup>2</sup> /s
$N$	Anzahl der Spezies	1
$n$	Reaktionsordnung	1
$p$	Anzahl Prozessorkerne	1
$p$	Druck	Pa
$p_r$	reduzierter Druck	1
$\dot{q}_R$	Wärmefreisetzungsrate	J/m <sup>3</sup> /s
$R$	spezifische Gaskonstante	J/kg/K
$r$	Radius	m
$\dot{r}_k$	Reaktionsrate der Spezies $k$	kg/m <sup>3</sup> /s
$\tilde{s}$	Entropie (molar)	J/mol/K
$s_L$	laminare Flammgeschwindigkeit	m/s
$S$	Quellterm	variiert
$S$	Speedup Faktor	1
$\dot{S}_{Rad}$	Energiefluss durch Strahlung	J/m <sup>3</sup> /s
$T$	Temperatur	K
$T^*, T^{**}, T^{***}$	Parameter für Troe-Reaktionen	K
$t$	Zeit	s
$u$	innere Energie	J/kg
$\tilde{u}$	innere Energie (molar)	J/mol
$\vec{v}$	Geschwindigkeit	m/s
$\vec{V}^c$	Korrekturgeschwindigkeit	m/s
$\vec{V}_k$	Diffusionsgeschwindigkeit der Spezies $k$	m/s
$x$	Ort	m
$X$	Molanteil	mol/mol
$Y$	Massenanteil	kg/kg
$Z_{rot}$	relationale Kollisionszahl	1

## Griechische Symbole

Symbol	Bedeutung	Einheit
$\alpha$	Parameter für Troe-Reaktionen	1
$\beta$	Temperaturexponent	1
$\delta_{i,j}$	Kronecker-Delta	1
$\delta$	Flammendicke	m
$\delta^*$	reduziertes Dipolmoment	1
$\epsilon_k$	Effizienz der Spezies $k$ in einer Dreistoß-Reaktion	1
$\epsilon_k$	Tiefe des Energieminimums der Spezies $k$	J
$\epsilon_{k,i}$	reduzierte Tiefe des Energieminimums	J
$\lambda$	Wärmeleitfähigkeit	W/m/K
$\lambda$	Luftverhältnis	1
$\mu$	dynamische Viskosität	kg/m/s
$\mu_k$	Dipolmoment der Spezies $k$	C m
$\nu$	Stöchiometriekoeffizient	1
$\rho$	Dichte	kg/m <sup>3</sup>
$\sigma_k$	Lennard-Jones Kollisionsdurchmesser	m
$\sigma_{i,k}$	reduzierter Lennard-Jones Kollisionsdurchmesser	m
$\vec{\tau}$	Spannungstensor	N/m <sup>2</sup>
$\phi$	physikalische Größe	variiert
$\dot{\omega}$	Reaktionsrate	mol/m <sup>3</sup> /s
$\Omega_{i,k}^{(j,j)*}$	reduziertes Kollisionsintegral	1

## Konstanten

Symbol	Name	Wert
$\vec{g}$	Erdbeschleunigung	9.80665 m/s <sup>2</sup>
$k_B$	Boltzmann-Konstante	1.3806488 · 10 <sup>-23</sup> J/K
$N_A$	Avogadro-Konstante	6.02214129 · 10 <sup>23</sup> 1/mol
$\bar{R}$	universelle Gaskonstante	8.3144621 J/K/mol
$\pi$	Kreiszahl	3.14159 . . .

## Abkürzungen

Abkürzung	Bedeutung
1D	eindimensional
3D	dreidimensional
BLAS	<u>B</u> asic <u>L</u> inear <u>A</u> lgebra <u>S</u> ubprograms
BR	<u>B</u> atch <u>R</u> eactor
BSD	<u>B</u> erkeley <u>S</u> oftware <u>D</u> istribution
CFD	<u>C</u> omputational <u>F</u> luid <u>D</u> ynamics
CFL-Zahl	<u>C</u> ourant- <u>F</u> riedrichs- <u>L</u> ewy- <u>Z</u> ahl
CPU	<u>C</u> entral <u>P</u> rocessing <u>U</u> nit
DIC	<u>D</u> iagonal <u>i</u> ncomplete <u>C</u> holesky
DILU	<u>D</u> iagonal <u>i</u> ncomplete <u>L</u> ower <u>U</u> pper
DNS	<u>D</u> irect <u>N</u> umerical <u>S</u> imulation
EBI	<u>E</u> ngler- <u>B</u> unte- <u>I</u> nstitut
FVM	<u>F</u> inite <u>V</u> olumen <u>M</u> ethode
GPL	<u>G</u> NU General <u>P</u> ublic <u>L</u> icense
HLRS	<u>H</u> ochleistungsrechenzentrum <u>S</u> tuttgart
HPC	<u>H</u> igh <u>P</u> erformance <u>C</u> omputing
IBM	<u>I</u> nternational <u>B</u> usiness <u>M</u> achines Corporation
JSC	<u>J</u> ülich <u>S</u> upercomputing <u>C</u> entre

LAPACK	<u>L</u> inear <u>A</u> lgebra <u>P</u> ackage
LES	<u>L</u> arge <u>E</u> ddy <u>S</u> imulation
MPI	<u>M</u> essage <u>P</u> assing <u>I</u> nterface
NASA	<u>N</u> ational <u>A</u> eronautics and <u>S</u> pace <u>A</u> dmistration
OpenFOAM	<u>O</u> pen <u>S</u> ource <u>F</u> ield <u>O</u> peration and <u>M</u> anipulation
PBiCG	<u>P</u> reconditioned <u>B</u> i- <u>C</u> onjugate <u>G</u> radient
PCG	<u>P</u> reconditioned <u>C</u> onjugate <u>G</u> radient
PEV	<u>P</u> rimärenergieverbrauch
PFR	<u>P</u> lug <u>F</u> low <u>R</u> eactor
RANS	<u>R</u> eynolds- <u>a</u> veraged <u>N</u> avier- <u>S</u> tokes
RVO	<u>R</u> eturn <u>V</u> alue <u>O</u> ptimization
SUNDIALS	<u>S</u> uite of <u>n</u> onlinear and <u>d</u> ifferential/ <u>a</u> lgebraic equation <u>s</u> olvers
VBT	Lehrstuhl für <u>V</u> erbrennungstechnik

## Sub- und Superskripte

Symbol	Bedeutung
$\square_0$	Ort am Einlass
$\square_0$	bezogen auf niedrigen Referenzdruck
$\square_\infty$	bezogen auf hohen Referenzdruck
$\square_q$	Energie
$\square_F$	Brennstoff
$\square_{i,j}$	Indizes
$\square_k$	Speziesindex
$\square_M$	Index der Mischung
$\square_n$	n-te Iteration
$\square_p$	bezogen auf Anzahl der Prozessorkerne
$\square_r$	Reaktionsindex
$\square_{rot}$	Anteil durch Rotation
$\square_{trans}$	Anteil durch Translation
$\square_{vib}$	Anteil durch Vibration
$\square^*$	angeregte Spezies
$\square^*$	reduzierte Größe
$\square'$	Hinreaktion
$\square''$	Rückreaktion
$\square^T$	transponiert
$\square^\circ$	Wert bei Referenzbedingungen
$\bar{\square}$	mischungsgemittelt
$\dot{\square}$	Änderungsrate
$\tilde{\square}$	molare Größe
$\vec{\square}$	Vektorgröße

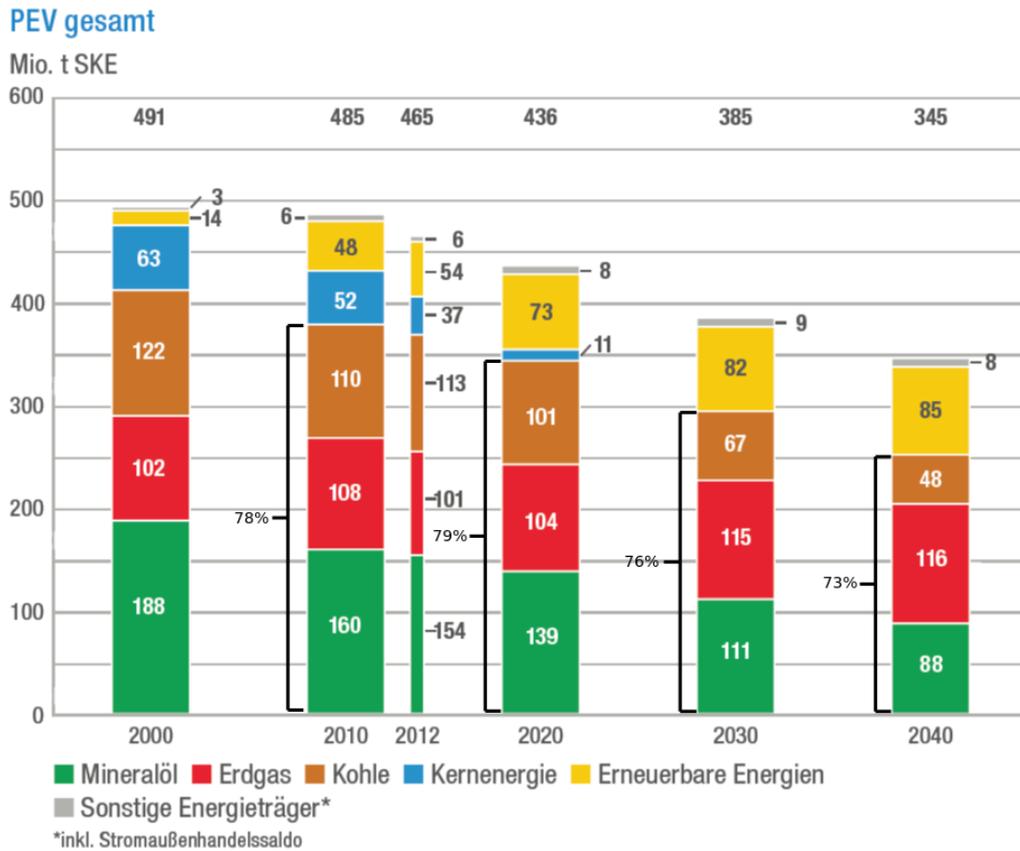
# Kapitel 1

## Einleitung

### 1.1 Motivation

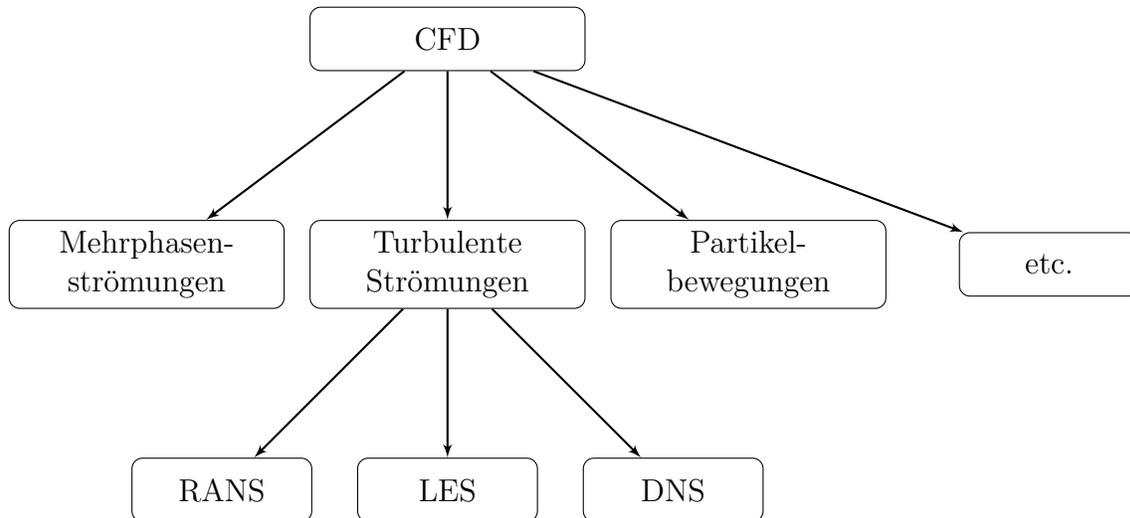
Technische Verbrennung spielt in vielen Bereichen eine große Rolle - von chemischen Prozessen in industriellem Maßstab bis hin zu alltäglichen Dingen wie der Verbrennung in Automotoren. Dabei wird ihre Rolle in Zukunft weiter an Wichtigkeit gewinnen. So wird zum Beispiel die Verbrennung fossiler Brennstoffe auch in den nächsten Jahrzehnten den Großteil des Primärenergieverbrauchs (PEV) Deutschlands ausmachen (Abbildung 1.1). Da fossile Brennstoffe eine begrenzte Ressource darstellen, ist es besonders wichtig, Verbrennungsprozesse so effizient wie möglich zu gestalten. So wird aufgrund der stetig steigenden Effizienz der Primärenergieverbrauch Deutschlands in den nächsten Jahrzehnten kontinuierlich sinken [12]. Gleichzeitig werden immer strengere Gesetze eingeführt, um die Schadstoffproduktion durch Verbrennungen zu reduzieren. Die Steigerung des Wirkungsgrades bei gleichzeitiger Senkung der Schadstoffproduktion ist nur dann erreichbar, wenn Verbrennungsprozesse auch in Zukunft besser verstanden werden.

Technische Verbrennungen finden meistens innerhalb von turbulenten Strömungen statt [40]. Sie sind oft instationär und laufen bei hohen Drücken und Temperaturen ab. Ihre Erforschung durch Experimente ist daher sehr zeit- und kostenintensiv oder gar unmöglich [41]. Als Alternative zu Experimenten bietet sich die Simulation durch numerische Strömungsmechanik (CFD) an. Mit der CFD Methode werden die Grund-



**Abbildung 1.1:** Prognose des Primärenergieverbrauchs Deutschlands bis zum Jahre 2040 [12].

gleichungen, mit denen die Strömungsverhältnisse und Verbrennungsprozesse beschrieben werden, numerisch gelöst. CFD hat neben der Simulation von Verbrennungsprozessen noch viele weitere Anwendungsgebiete wie die Simulation von mehrphasigen Strömungen oder der Partikelbewegung in Fluiden. Auch zur Simulation von turbulenten Strömungen, die für diese Arbeit relevant sind, gibt es innerhalb der CFD mehrere Ansätze. Die drei wichtigsten Verfahren sind RANS (Reynolds-gemittelte Navier-Stokes-Gleichungen), LES (Large Eddy Simulation) und DNS (Direkte Numerische Simulation). Zur Simulation von Verbrennungsprozessen muss die Wechselwirkung der chemischen Reaktionen mit der turbulenten Strömung beschrieben werden. RANS und LES stellen dabei vereinfachende Modelle für die turbulente Strömung dar, um den



**Abbildung 1.2:** Überblick über die verschiedenen Zweige der CFD.

Rechenaufwand der Simulation zu reduzieren. Bei DNS hingegen werden die grundlegenden Gleichungen ohne Vereinfachungen gelöst und genaue Ansätze für die Speziestransportgleichungen und die chemische Kinetik gewählt. Um also die genauesten Vorhersagen zu Verbrennungsprozessen treffen zu können, ist es notwendig, die Simulation mit DNS durchzuführen. Da sämtliche Erhaltungsgleichungen ohne vereinfachende Annahmen gelöst werden, ist der Rechenaufwand für DNS sehr hoch. So muss z.B. das Simulationsgebiet sehr fein aufgelöst werden, damit auch der Einfluss der kleinsten Wirbel im Strömungsfeld berücksichtigt und die Dicke der Flammenreaktionszone dargestellt werden kann. Erst durch die massive Zunahme an Rechenleistung von großen Cluster-Systemen in den letzten Jahren ist es möglich geworden, DNS für reale Probleme durchzuführen [41]. Trotzdem bleibt die Rechengeschwindigkeit von Programmen, die DNS verwenden, ein sehr wichtiger Faktor. Aus diesem Grund ist diese Arbeit initiiert worden.

## 1.2 Ziele

Ziel dieser Arbeit ist es, ein vorhandenes Programm zur Direkten Numerischen Simulation (DNS Löser) für reagierende Strömungen hinsichtlich seiner Handhabung und

Effizienz zu optimieren. Das Programm selbst ist mit C++ in OpenFOAM [38] erstellt und bedient sich der stark optimierten Bibliotheken Cantera [19] und SUNDIALS [43]. Trotzdem sind die Berechnungen der chemischen Kinetik und Transportprozesse noch sehr zeitintensiv. Ein weiteres Problem ist, dass Cantera auf allen System installiert werden muss, auf denen der Löser laufen soll, obwohl der Großteil der von Cantera unterstützten Modelle - wie Modelle für Reaktionen in flüssigen oder metallischen Phasen - nicht gebraucht wird. Dadurch wird die Handhabung des Löserters erschwert. Da Cantera einen vergleichsweise hohen Codeumfang hat, sind relevante Codeteile nicht sofort erkennbar und das Verhalten des Löserters nur schwer nachvollziehbar. Zudem ist der Canteracode nur sehr umständlich änderbar, was die Wartbarkeit des Löserters vermindert. Die Hauptaufgabe dieser Arbeit besteht daher darin, die für den DNS Löserter benötigten Codeteile aus Cantera zu identifizieren und direkt in den Löserter zu integrieren. Zudem soll die Leistung des Löserters optimiert werden, um die Rechenzeit zu reduzieren. Dafür soll das Laufzeitverhalten des Programms ausführlich analysiert werden (Profiling), um Leistungsengpässe aufzuspüren.

Sowohl das Extrahieren der benötigten Codeteile als auch deren Optimierung gestaltet sich als schwierig. Die einzelnen Module, aus denen Cantera zusammengesetzt ist, sind untereinander stark vernetzt. Entfernt man Module, so müssen an vielen Stellen im Code Modifikationen vorgenommen werden, um ein lauffähiges Programm zu erhalten. Auch die Optimierung ist aufwendig, da Cantera selbst schon stark optimierte Routinen zur Berechnung der Reaktionskinetik bereitstellt. Daher muss der Löserter ausführlich auf Leistungsengpässe untersucht und Wissen über die Funktionsweise des Löserters ausgenutzt werden, um die Rechengeschwindigkeit zu steigern. Unnötige Berechnungen, z.B. bei der Ermittlung der Speziesenthalpien oder der Geschwindigkeitskonstanten, sollen gefunden und beseitigt werden.

Der Löserter soll so modifiziert werden, dass die Reaktionsraten nicht über eine Integration der Reaktorgleichung durch SUNDIALS, sondern direkt berechnet werden. Dadurch soll der Einfluss der Reaktionskinetik auf die Rechenzeit und Stabilität des Löserters genauer untersucht und optimale Toleranzparameter für SUNDIALS gefunden werden.

Um den so optimierten DNS Löserter bezüglich der Reaktionskinetik und der Im-

plementierung des Stofftransports zu validieren, sollen die Simulationsergebnisse einer vorgemischten Wasserstoff/Luft Flachflamme mit denen des kommerziellen Programms CHEMKIN [8] verglichen werden. In einer weiteren Simulation soll ein Bombenexperiment, das am Engler-Bunte-Institut (EBI) durchgeführt wurde [2], mit einer Wasserstoff/Luft Verbrennung nachgerechnet werden, um zu zeigen, dass der neue Löser für 3D DNS verwendet werden kann. Dabei soll auch nachgewiesen werden, dass der neue Löser auf dem Supercomputer JUQUEEN ein sehr gutes Skalierungsverhalten aufweist. Zudem soll der Einfluss verschiedener Reaktionsmechanismen auf die Rechenzeit untersucht werden. Die in dieser Arbeit durchgeführten Simulationen sind mit Mischungsgemittelten Transportkoeffizienten durchgeführt worden (Kapitel 2); für Methan/Luft Flammen wurde der Reaktionsmechanismus in B.4 verwendet.

### 1.3 Aufbau der Arbeit

In Kapitel 2 werden zunächst die Grundgleichungen für DNS von reagierenden Fluiden erläutert. Kapitel 3 stellt die verwendeten Softwarepakete und Cluster-Plattformen vor. Zudem wird gezeigt, durch welche Maßnahmen die Rechengeschwindigkeit und Handhabung des DNS Löser verbessert wurden. In Kapitel 4 wird die Berechnung der Reaktionsraten durch Mittelung über den DNS Zeitschritt diskutiert und der Einfluss von Toleranzparametern gezeigt. In Kapitel 5 werden zwei Simulationen von Wasserstoff/Luft Verbrennungen zur Validierung des neuen Löser vorgestellt: Zum einen die Validierung der implementierten Reaktionskinetik mit einer vorgemischten Wasserstoff/Luft Flachflamme, zum anderen die Simulation einer 3D Verbrennung für ein Bombenexperiment [2]. Im Rahmen dieser Simulationen werden auch der Einfluss von Reaktionsmechanismen auf die Rechenzeit sowie die Auswirkung der Gitterpunktdichte auf die Form der Flammenfront gezeigt und das Skalierungsverhalten des Löser untersucht. Kapitel 6 fasst schließlich die in dieser Arbeit gewonnenen Ergebnisse zusammen und bietet einen Ausblick auf weitere Arbeit in diesem Forschungsgebiet und Verbesserungen des Löser.

# Kapitel 2

## Grundgleichungen

Bei der Direkten Numerischen Simulation (DNS) werden die Strömungsverhältnisse ohne vereinfachende Modelle berechnet. Dazu wird die allgemeine Navier-Stokes-Gleichung numerisch gelöst. Neben den Strömungsverhältnissen werden weitere Erhaltungsgleichungen benötigt, um die Verbrennungsreaktionen zu beschreiben. Dieses Kapitel gibt einen Überblick über die Grundgleichungen, die für eine solche Simulation benötigt werden. Alle im Folgenden gezeigten Gleichungen werden so auch im DNS Löser verwendet.

### 2.1 Allgemeine Form der Erhaltungsgleichungen

Die turbulente Verbrennung unterliegt mehreren Erhaltungsgleichungen. Diese können in Form von Transportgleichungen ausgedrückt werden. Die allgemeine Form einer Transportgleichung [26] ist im Folgenden dargestellt:

$$\underbrace{\rho \frac{D\phi}{Dt}}_{\text{substantielle Ableitung}} = - \underbrace{\nabla \cdot \vec{j}_\phi}_{\text{Diffusionsterm}} + \underbrace{S_\phi}_{\text{Quellterm}} \quad (2.1)$$

$\rho$  ist die Dichte der Gasphase,  $\rho \frac{D\phi}{Dt}$  ist eine substantielle Ableitung. Sie kann umgeformt werden zu

$$\rho \frac{D\phi}{Dt} = \underbrace{\frac{\partial \rho \phi}{\partial t}}_{\text{lokale Änderung}} + \underbrace{\nabla \cdot \rho \vec{v} \phi}_{\text{Konvektionsänderung}} \quad (2.2)$$

$\vec{v}$  die Geschwindigkeit der Strömung. Die allgemeine Form der Transportgleichung sieht damit wie folgt aus:

$$\frac{\partial \rho \phi}{\partial t} = -\nabla \cdot (\rho \vec{v} \phi) - \nabla \cdot \vec{j}_\phi + S_\phi \quad (2.3)$$

## 2.2 Gesamtmassenerhaltung

Die Gesamtmasse des Systems muss erhalten bleiben. Da bei einer chemischen Reaktion keine Masse entsteht und die Summe aller Speziesflüsse null ergibt, gibt es keine Quell- oder Diffusionsterme. Die Massenerhaltung wird auf folgende Weise gelöst [40]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (2.4)$$

## 2.3 Impulserhaltung

Die Impulserhaltung wird über die Navier-Stokes-Gleichung für kompressible Strömungen dargestellt [40]:

$$\frac{\partial}{\partial t} (\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot \vec{\tau} + \rho \vec{g} \quad (2.5)$$

Man erhält sie durch eine Kräftebilanz für ein infinitesimales Volumenelement.  $\nabla p$  stellt eine Oberflächenkraft dar, die durch Druckunterschiede an den Seiten des Kontrollvolumens entsteht.  $\rho \vec{g}$  beschreibt die Auftriebskraft,  $\vec{g}$  ist die Erdbeschleunigung.

Der viskose Spannungstensor  $\vec{\tau}$  beschreibt die Oberflächenkraft durch Reibung aufgrund der Fluidviskosität [40]. Für ein newtonsches Fluid hat er die folgende Form:

$$\vec{\tau} = \mu \nabla \vec{v} + \mu \left( (\nabla \vec{v})^T - \frac{2}{3} \delta_{ij} (\nabla \vec{v})^T I \right) = \mu \left( (\nabla \vec{v} + (\nabla \vec{v})^T) - \frac{2}{3} \delta_{ij} \nabla \cdot \vec{v} \right) \quad (2.6)$$

$I$  ist die Einheitsmatrix,  $\delta_{ij}$  das Kronecker-Delta:

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}, \quad i, j = 1, 2, 3 \quad (2.7)$$

Die dynamische Viskosität der Mischung  $\mu$ , die im viskosen Stresstensor in Gl. (2.6) verwendet wird, wird aus den Viskositäten  $\mu_k$  der reinen Spezies  $k$  ermittelt. Die Viskosität der reinen Spezies wird mit der Chapman-Enskog Lösung der Boltzmann-Gleichung berechnet [26]:

$$\mu_k = \frac{5}{16} \frac{\sqrt{\pi m_k k_B T}}{\pi \sigma_k^2 \Omega_{kk}^{(2,2)*}} \quad (2.8)$$

$\sigma_k$  ist der Lennard-Jones Kollisionsdurchmesser, der für jede Spezies aus einer Tabelle ausgelesen wird,  $T$  die Temperatur,  $\pi$  die Kreiszahl und  $k_B$  die Boltzmann-Konstante.  $m_k$  ist die Masse des Speziesmoleküls:

$$m_k = \frac{M_k}{N_A} \quad (2.9)$$

$N_A$  ist die Avogadro-Konstante und  $M_k$  die Molmasse der Spezies  $k$ . Die Lennard-Jones Kollisionsdurchmesser  $\sigma_k$  der Spezies und deren Molmassen  $M_k$  werden beim Programmstart als Stoffkonstanten aus einer Datei eingelesen.

$\Omega_{kk}^{(2,2)*}$  ist ein reduziertes Kollisionsintegral und beschreibt den Einfluss von Kollisionen zwischen den Spezies. Es hängt von der reduzierten Temperatur  $T^*$  und dem reduzierten Dipolmoment  $\delta^*$  ab [34].

$$T^* = \frac{k_B T}{\epsilon_{k,i}} \quad (2.10)$$

$\epsilon_k$  ist die Tiefe des Energieminimums des Lennard-Jones Potentials der Spezies  $k$ ,  $\epsilon_{k,i}$  die reduzierte Tiefe des Energieminimums des Lennard-Jones Potentials:

$$\frac{\epsilon_{k,i}}{k_B} = \sqrt{\left(\frac{\epsilon_k}{k_B}\right) \left(\frac{\epsilon_i}{k_B}\right)} \quad (2.11)$$

Das reduzierte Dipolmoment  $\delta^*$  wird aus dem Dipolmoment  $\mu_k$  der Spezies  $k$  errech-

net [34]. Dipolmoment und Tiefe des Lennard-Jones Energieminimums werden als weitere Stoffkonstanten aus einer Datenbank eingelesen.

Die Bestimmung des Kollisionsintegrals ist rechnerisch sehr aufwendig [33]. Daher werden die Viskositäten der reinen Spezies nur einmal am Programmstart vorberechnet. Mit diesen vorberechneten Werten werden Koeffizienten für ein Polynom ermittelt, mit dessen Hilfe die Speziesviskositäten zur Laufzeit approximiert werden. Dazu werden an 50 gleichverteilten Temperaturen im für die Simulation relevanten Temperaturbereich die Viskositäten nach Gl. (2.18) ermittelt und die Koeffizienten  $a_0$  bis  $a_4$  bestimmt. Das Polynom ist dabei abhängig von der logarithmischen Temperatur und hat folgende Form:

$$\sqrt{\mu_k} = T^{\frac{1}{4}} \left( a_0 + a_1 \ln T + a_2 \ln^2 T + a_3 \ln^3 T + a_4 \ln^4 T \right) \quad (2.12)$$

Hier wird nicht die Speziesviskosität  $\mu_k$ , sondern die Wurzel der Speziesviskosität  $\sqrt{\mu_k}$  durch das Polynom approximiert. Für die Wilke-Mischungsregel in Gl. (2.13) zur Bestimmung der Mischungviskosität wird sowohl die Speziesviskosität als auch die Wurzel der Speziesviskosität benötigt. Die Approximation der Wurzel der Viskosität stellt dabei eine Rechengeschwindigkeitsoptimierung dar, da das Berechnen der Wurzel deutlich aufwendiger als das Berechnen des Quadrats ist [14]. Die Mischungviskosität wird schließlich über die Wilke-Mischungsregel bestimmt [42]:

$$\mu = \sum_{k=1}^N \frac{\mu_k X_k}{\sum_{i=1}^N \Phi_{k,i} X_i} \quad (2.13)$$

$N$  ist die Anzahl der Spezies.  $\Phi_{k,i}$  ist definiert als:

$$\Phi_{k,i} = \frac{\left[ 1 + \sqrt{\left( \frac{\mu_k}{\mu_i} \sqrt{\frac{M_i}{M_k}} \right)} \right]^2}{\sqrt{8} \sqrt{1 + \frac{M_k}{M_i}}} \quad (2.14)$$

## 2.4 Speziesmassenerhaltung

Setzt man in Gl. (2.1) für  $\phi$  die Speziesmassenbrüche  $Y_k$  ein, erhält man die allgemeine Form der Erhaltungsgleichung für die Masse der Spezies  $k$  [26]:

$$\frac{\partial}{\partial t} (\rho Y_k) + \nabla \cdot \rho \vec{v} Y_k = -\nabla \cdot \vec{j}_k + \dot{r}_k \quad (2.15)$$

$\vec{j}_k$  ist der Massendiffusionsfluss der Spezies  $k$  und  $\dot{r}_k$  deren Reaktionsrate.

### 2.4.1 Massenflüsse

$\vec{j}_k$  beschreibt den Massendiffusionsfluss der Spezies  $k$  [26]:

$$\vec{j}_k = \rho Y_k \vec{V}_k \quad (2.16)$$

$\vec{V}_k$  ist die Diffusionsgeschwindigkeit der Spezies  $k$ . Da eine exakte Multikomponenten-Berechnung der Diffusionsgeschwindigkeit sehr aufwendig ist, wird der Diffusionsterm häufig als Approximation mit Mischungsgemittelten Diffusionskoeffizienten (Unterabschnitt 2.4.2) bestimmt. Mit der Vereinfachung, dass es sich um Mischungsgemittelte Diffusionskoeffizienten handelt, und unter Vernachlässigung von Thermodiffusion und Druckdiffusion ergibt sich der Diffusionsstrom nach dem ersten Fick'schen Gesetz [26]:

$$\vec{j}_k = -\rho D_k \nabla Y_k \quad (2.17)$$

$D_k$  sind dabei die Mischungsgemittelten Diffusionskoeffizienten.

### 2.4.2 Mischungsgemittelte Diffusionskoeffizienten

In dieser Arbeit wird mit Mischungsgemittelten Diffusionskoeffizienten gerechnet. Sie stellen eine Vereinfachung der exakten Multikomponenten-Koeffizienten dar und werden für jede Spezies über die binären Diffusionskoeffizienten  $\mathcal{D}_{k,i}$  für die Mischung gemittelt [26]. Die binären Diffusionskoeffizienten werden nach der Chapman-Enskog Lösung

der Boltzmann-Gleichung berechnet:

$$\mathcal{D}_{k,i}(T, p) = \frac{3}{16} \frac{\sqrt{\frac{2\pi k_B^3 T^3}{m_{i,k}}}}{p\pi\sigma_{i,k}^2\Omega_{i,k}^{(1,1)*}} \quad (2.18)$$

$\Omega_{i,k}^{(1,1)*}$  ein reduziertes Kollisionsintegral (Abschnitt 2.3),  $m_{i,k}$  die reduzierte Masse der Speziesmoleküle und  $\sigma_{i,k}$  der reduzierte Lennard-Jones Kollisionsdurchmesser.

$$m_{i,k} = \frac{m_i m_k}{m_i + m_k} = \frac{M_i M_k}{N_A (M_k + M_i)} \quad (2.19)$$

$$\sigma_{i,k} = \frac{\sigma_i \sigma_k}{\sigma_i + \sigma_k} \quad (2.20)$$

Der DNS Löser berechnet die binären Diffusionskoeffizienten - ähnlich wie die Speziesviskositäten - über diese Form nur einmal während des Programmstarts. Zur Laufzeit werden sie stattdessen über ein Polynom approximiert, um Rechenzeit zu sparen. Dazu werden die binären Diffusionskoeffizienten an 50 Temperaturen ausgewertet und damit die Koeffizienten  $a_0$  bis  $a_4$  ermittelt.

$$\mathcal{D}_{k,i} = T^{\frac{3}{2}} \left( a_0 + a_1 \ln T + a_2 \ln^2 T + a_3 \ln^3 T + a_4 \ln^4 T \right) \quad (2.21)$$

Obwohl die binären Diffusionskoeffizienten im DNS Löser prinzipiell nach Gl. (2.18) berechnet werden, gibt es einen Unterschied: Da das Polynom, mit dem die binären Diffusionskoeffizienten zur Laufzeit approximiert werden, nur von der Temperatur abhängt, werden die binären Diffusionskoeffizienten ohne den Druck als Faktor vorberechnet. Werden die binären Diffusionskoeffizienten dann später über das Polynom berechnet, wird nachträglich der Druck als Faktor berücksichtigt. So können die gemittelten Diffusionskoeffizienten auch zur Laufzeit des Programms in Abhängigkeit des Drucks approximiert werden. Die binären Diffusionskoeffizienten werden schließlich über die Mischung gemittelt:

$$\frac{1}{D_k} = \sum_{i \neq k}^N \frac{X_i}{\mathcal{D}_{k,i}} + \frac{X_k}{1 - Y_k} \sum_{i \neq k} \frac{Y_i}{\mathcal{D}_{k,i}} \quad (2.22)$$

$X_k$  ist der Molanteil der Spezies  $k$ .

### 2.4.3 Korrekturgeschwindigkeit

Die Verwendung von Mischungsgemittelten Diffusionskoeffizienten ist eine Approximation. Dadurch ist der Gesamtmassenerhalt nicht mehr garantiert. Dafür müsste die Summe aller Massenflüsse null ergeben:

$$\sum_{k=1}^N \vec{j}_k = \sum_{k=1}^N \rho Y_k \vec{V}_k = 0 \quad (2.23)$$

Um dies zu gewährleisten, muss eine Geschwindigkeitskorrektur  $\vec{V}^c$  eingeführt werden. Sie kann aus der Gesamtmassenerhaltung abgelesen werden:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v} = \nabla \cdot \underbrace{\left( \rho \sum_{k=1}^N D_k \nabla Y_k - \rho \vec{V}^c \right)}_{=0} + \underbrace{\sum_{k=1}^N \dot{r}_k}_{=0} = 0 \quad (2.24)$$

Für  $\vec{V}^c$  muss damit gelten:

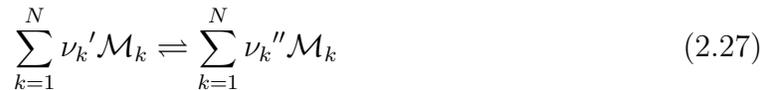
$$\vec{V}^c = \sum_{k=1}^N D_k \nabla Y_k = \sum_{k=1}^N Y_k \vec{V}_k \quad (2.25)$$

Mit der Korrekturgeschwindigkeit sieht die Erhaltungsgleichung der Speziesmasse folgendermaßen aus:

$$\frac{\partial}{\partial t} (\rho Y_k) + \nabla \cdot \rho (\vec{v} + \vec{V}^c) Y_k = \nabla \cdot \rho D_k \nabla Y_k + \dot{r}_k \quad (2.26)$$

### 2.4.4 Reaktionsraten

Der Quellterm  $\dot{r}_k$  ist die Reaktionsrate der Spezies  $k$ . Sie wird über die Reaktionskinetik ermittelt [26]. Jede chemische Reaktion hat die folgende Form:



$\mathcal{M}_k$  ist der Name der Spezies  $k$ ,  $\nu_k'$  der Stöchiometrikoeffizient der Hinreaktion und  $\nu_k''$  der Stöchiometrikoeffizient der Rückreaktion. Zudem muss insgesamt die Masse erhalten bleiben:

$$\sum_{k=1}^N \sum_{r=1}^M (\nu_{k,r}' - \nu_{k,r}'') M_k = 0 \quad (2.28)$$

Dabei ist  $N$  die Anzahl der Spezies und  $M$  die Anzahl der Reaktionen. Die Reaktionsgeschwindigkeit der Reaktion  $r$  kann über den Potenzansatz bestimmt werden:

$$\dot{\omega}_r = k_r' \prod_{k=1}^N C_k^{n_{k,r}'} - k_r'' \prod_{k=1}^N C_k^{n_{k,r}''} \quad (2.29)$$

$k_r'$  und  $k_r''$  sind die Geschwindigkeitskonstanten der Hin- und Rückreaktion,  $C_k$  die Molkonzentration der Spezies  $k$  und  $n_{k,r}$  die Reaktionsordnung im Bezug auf Spezies  $k$ . Die meisten Reaktionsmechanismen werden durch Elementarreaktionen beschrieben. Das bedeutet, dass alle Reaktionsordnungen 1 betragen bzw. den Stöchiometrikoeffizienten entsprechen. Dadurch lässt sich Gl. (2.29) vereinfachen:

$$\dot{\omega}_r = k_r' \prod_{k=1}^N C_k^{\nu_{k,r}'} - k_r'' \prod_{k=1}^N C_k^{\nu_{k,r}''} \quad (2.30)$$

Die Geschwindigkeitskonstanten werden im einfachsten Fall vom Löser über den erweiterten Arrhenius-Ansatz [29] berechnet:

$$k_r = AT^\beta e^{-\frac{E_a}{RT}} \quad (2.31)$$

$E_a$  ist die Aktivierungsenergie für die Reaktion,  $A$  der Frequenzfaktor und  $\bar{R}$  die universelle Gaskonstante.  $A$ ,  $\beta$  und  $E_a$  sind für jede Reaktion tabelliert und müssen beim Programmstart eingelesen werden.

Für die Speziesreaktionsrate  $\dot{\omega}_k$  müssen die Reaktionsraten  $\dot{\omega}_r$  über alle Reaktionen aufsummiert werden:

$$\dot{\omega}_k = \sum_{r=1}^M \dot{\omega}_r (\nu_{k,r}'' - \nu_{k,r}') \quad (2.32)$$

Die massenbezogene Reaktionsrate  $\dot{r}_k$  wird als Quellterm in der Speziestransportgleichung in Gl. (2.15) und bei der Bestimmung der Wärmefreisetzungsrate in Gl. (2.85)

benötigt und lässt sich mit Hilfe der Speziesmolmasse bestimmen:

$$\dot{r}_k = \dot{\omega}_k M_k \quad (2.33)$$

#### 2.4.4.1 Geschwindigkeitskonstante der Rückreaktion

Die Geschwindigkeitskonstante der Rückreaktion  $k_r''$  in Gl. (2.30) kann auch über den Arrhenius-Ansatz oder die Formulierung nach Troe (Unterunterabschnitt 2.4.4.2) berechnet werden. Falls aber in der Datenbank des Reaktionsmechanismus nur die nötigen Parameter für die Geschwindigkeitskonstante der Hinreaktion angegeben sind, wird die Geschwindigkeitskonstante der Rückreaktion mit Hilfe der Gleichgewichtskonstanten berechnet [26]:

$$k_r'' = \frac{k_r'}{K_r} \quad (2.34)$$

$K_r$  ist die Gleichgewichtskonstante der Reaktion. Sie wird auf folgende Weise bestimmt:

$$K_r = e^{-\frac{\Delta G_r^\circ}{RT}} \cdot \left( \frac{p^\circ}{RT} \right)^{\Delta \nu_r} \quad (2.35)$$

$p^\circ$  ist ein Referenzdruck, der - wenn nicht anders angegeben - 1 bar beträgt [1],  $\Delta G_r^\circ$  die Standard-Gibbs-Bildungsenergie und  $\Delta \nu_r$  die Differenz der Stöchiometriekoeffizienten aller Spezies der Reaktion  $r$ .

$$\Delta \nu_r = \sum_{k=1}^N (\nu_{k,r}'' - \nu_{k,r}') \quad (2.36)$$

Die Standard-Gibbs-Energie der Spezies  $k$  berechnet sich nach der Gibbs-Helmholtz-Gleichung:

$$G_k^\circ = \tilde{h}_k - T \tilde{s}_k \quad (2.37)$$

$\tilde{h}_k$  und  $\tilde{s}_k$  sind hier die molare Enthalpie und molare Entropie der Spezies  $k$  (Unterabschnitt 2.5.3) bei dem Referenzdruck  $p^\circ$  und der aktuellen Temperatur  $T$ . Man erhält schließlich  $\Delta G_r^\circ$  aus den Standard-Gibbs-Energien:

$$\Delta G_r^\circ = \sum_{k=1}^N (\nu_{k,r}'' - \nu_{k,r}') G_k^\circ \quad (2.38)$$

### 2.4.4.2 Druckabhängigkeit der Geschwindigkeitskonstanten

Druckabhängige Geschwindigkeitskonstanten werden im Löser nach einem Modell von Troe [18] berechnet. Dabei wird die Druckabhängigkeit der Geschwindigkeitskonstanten berücksichtigt. Dazu werden die Parameter  $A_0$ ,  $\beta_0$  und  $E_0$  für das Arrheniusgesetz bei geringen Drücken und  $A_\infty$ ,  $\beta_\infty$  und  $E_\infty$  als Parameter für das Arrheniusgesetz bei hohen Drücken aus einer Datenbank eingelesen und damit die Grenz-Reaktionsraten bestimmt:

$$k_0 = A_0 T^{\beta_0} e^{-\frac{E_0}{RT}} \quad (2.39)$$

$$k_\infty = A_\infty T^{\beta_\infty} e^{-\frac{E_\infty}{RT}} \quad (2.40)$$

$k_0$  hat die Einheit  $\text{m}^3/\text{mol}/\text{s}$  und  $k_\infty$  die Einheit  $1/\text{s}$  [7]. Die Geschwindigkeitskonstante der Reaktion wird mit Hilfe eines reduzierten Drucks  $p_r$  berechnet:

$$k_r = k_\infty \frac{p_r}{1 + p_r} F \quad (2.41)$$

$F$  ist eine Funktion, die abhängig von der Temperatur ist. Der reduzierte Druck wiederum ist eine Funktion der Mischungskonzentration  $C_M$ :

$$p_r = \frac{k_0 C_M}{k_\infty} \quad (2.42)$$

Falls es sich um eine Dreistoß-Reaktion handelt (Unterunterabschnitt 2.4.4.3), kann die effektive Mischungskonzentration über der echten Konzentration liegen. Über die Mischungskonzentration wird der Einfluss des Drucks auf die Geschwindigkeitskonstante berücksichtigt. Die Funktion  $F$  ist durch die Troe-Form gegeben:

$$\log F = \left[ 1 + \left[ \frac{\log p_r + c}{n - d(\log p_r + c)} \right]^2 \right]^{-1} \log F_{cent} \quad (2.43)$$

mit

$$c = -0.4 - 0.67 \log F_{cent} \quad (2.44)$$

$$n = 0.75 - 1.27 \log F_{cent} \quad (2.45)$$

$$d = 0.14 \quad (2.46)$$

$$F_{cent} = (1 - \alpha)e^{-\frac{T}{T^{***}}} + \alpha e^{-\frac{T}{T^*}} + e^{-\frac{T}{T}} \quad (2.47)$$

Die vier Parameter  $\alpha$ ,  $T^*$ ,  $T^{**}$  und  $T^{***}$  werden für jede druckabhängige Reaktion aus einer Datenbank gelesen.

#### 2.4.4.3 Dreistoß- und Doppelreaktionen

Neben allgemeinen Elementarreaktionen mit zwei Stoßkörpern gibt es auch Dreistoß-Reaktionen [26]. Dabei handelt es sich um Reaktionen, bei denen eine Spezies  $M$  einen Beitrag zur Reaktionsgeschwindigkeit leistet, aber nicht in der netto Reaktionsgleichung auftaucht. Eine Dreistoß-Reaktion liegt immer dann vor, wenn das Produkt einer chemischen Reaktion durch Erhöhung der Schwingungsenergie instabil wird. Da Impuls und kinetische Energie bei einer Reaktion konstant sind, wird die freiwerdende Bindungsenergie in Schwingungsenergie umgewandelt. Diese kann durch Kollision an ein anderes Speziesmolekül abgegeben werden, sodass das Produkt der Reaktion stabilisiert wird. Ein Kollisionspartner  $M$  kann aber auch Energie liefern, um eine bestehende Bindung zu brechen. Im Prinzip kommt als Kollisionspartner  $M$  jede Spezies in Frage. Die Effektivität, mit der eine Spezies die Schwingungsenergie aufnehmen bzw. Energie zur Verfügung stellen kann, wird mit  $\epsilon_k$  bezeichnet. Dadurch muss Gl. (2.30) angepasst werden:

$$\dot{\omega}'_r = k'_r \sum_{k=1}^N (\epsilon_k C_k) \prod_{k=1}^N C_k^{\nu'_{k,r}} \quad (2.48)$$

Bei einer Dreistoß-Reaktion haben alle Spezies eine Grundeffektivität von  $\epsilon = 1$  [7]. Effektivitäten anderer Spezies, die über diesem Wert liegen, werden für jede Reaktion aus einer Datenbank gelesen. Die effektive Konzentration einer Spezies kann damit höher sein als ihre wirkliche Konzentration. Die Reaktionsgeschwindigkeit einer Dreistoß-

Reaktion lässt sich somit folgendermaßen bestimmen:

$$\dot{\omega}_r = \sum_{k=1}^N (\epsilon_k C_k) \left( k_r' \prod_{k=1}^N C_k^{\nu_{k,r}'} - k_r'' \prod_{k=1}^N C_k^{\nu_{k,r}''} \right) \quad (2.49)$$

Zusätzlich kann es vorkommen, dass innerhalb des Reaktionsmechanismus die gleiche Reaktion auf verschiedenen Reaktionspfaden erreicht wird. In Abhängigkeit des vorherigen Reaktionspfades können sich die Parameter zur Berechnung der Geschwindigkeitskonstanten der Folgereaktion stark unterscheiden [7]. Um dies zu berücksichtigen, können im Löser Reaktionen als **duplicate** definiert werden. So können verschiedene Reaktionsparameter für die formal gleiche Reaktion zur Verfügung gestellt werden.

## 2.5 Energieerhaltung

Die allgemeine Form der Energieerhaltung über eine Enthalpiebilanz für die thermische Enthalpie [17] sieht folgendermaßen aus [41]:

$$\frac{\partial}{\partial t} (\rho h_s) + \nabla \cdot (\rho h_s \vec{v}) = \frac{Dp}{Dt} + \vec{\tau} : \nabla \vec{v} - \nabla \cdot \vec{j}_q + \dot{q}_R + \dot{S}_{Rad} \quad (2.50)$$

$h_s$  ist die thermische Enthalpie und  $\vec{j}_q$  der Energiefluss.  $\frac{Dp}{Dt}$  ist der reversible Energiefluss durch Kompression, während  $\vec{\tau} : \nabla \vec{v}$  die irreversible Erwärmung durch Reibung beschreibt. Letztere wird im DNS Löser vernachlässigt. Als Quellterme gibt es die Wärmefreisetzungsrates  $\dot{q}_R$  durch die chemische Reaktion und den Energieeintrag durch Wärmestrahlung  $\dot{S}_{Rad}$ . Aktuell wird im DNS Löser kein Strahlungsmodell verwendet und dieser Quellterm somit vernachlässigt.

### 2.5.1 Energiefluss

Der Energiefluss  $\vec{j}_q$  setzt sich aus zwei Teilen zusammen; zum einen dem Wärmestrom durch den Temperaturgradienten nach dem Fourier'schen Gesetz, zum anderen der

Wärmefluss durch die Massendiffusion der Spezies:

$$\vec{j}_q = \underbrace{-\lambda \nabla T}_{\text{Fourier'sches Gesetz}} + \underbrace{\sum_{k=1}^N \vec{j}_k h_{s,k}}_{\text{Energiefluss durch Massendiffusion}} \quad (2.51)$$

$h_{s,k}$  ist dabei die spezifische, thermische Enthalpie der Spezies  $k$  (Unterabschnitt 2.5.3) und  $\lambda$  die Wärmeleitfähigkeit. Die Wärmeleitung lässt sich mit folgender Beziehung in Abhängigkeit der Enthalpie umformen:

$$\nabla h = c_p \nabla T + \sum_{k=1}^N h_k \nabla Y_k \quad (2.52)$$

$h$  ist die absolute Enthalpie der Mischung und  $h_k$  die absolute Enthalpie der Spezies  $k$ . Die thermische Enthalpie lässt sich aus der absoluten Enthalpie darstellen [41]:

$$h_{s,k} = h_k - h_k^\circ = \int_{T^\circ}^T c_{p,k} dT \quad (2.53)$$

$h_k^\circ$  ist die Standardbildungsenthalpie der Spezies  $k$ :

$$h_k^\circ = h_k(T^\circ) = \int_{0 \text{ K}}^{T^\circ} c_{p,k} dT \quad (2.54)$$

$T^\circ$  ist eine Referenztemperatur von 298.15 K. So lässt sich der Diffusionsterm in Abhängigkeit der thermischen Enthalpie darstellen [4]:

$$\begin{aligned} \lambda \nabla T &= \lambda \frac{1}{c_p} \nabla h - \lambda \sum_{k=1}^N \frac{1}{c_p} h_k \nabla Y_k \\ &= \frac{\lambda}{c_p} \nabla h_s + \underbrace{\sum_{k=1}^N \frac{\lambda}{c_p} h_k^\circ \nabla Y_k - \sum_{k=1}^N \frac{\lambda}{c_p} h_k^\circ \nabla Y_k}_{=0} - \sum_{k=1}^N \frac{\lambda}{c_p} h_{s,k} \nabla Y_k \\ &= \frac{\lambda}{c_p} \nabla h_s - \sum_{k=1}^N \frac{\lambda}{c_p} h_{s,k} \nabla Y_k \end{aligned} \quad (2.55)$$

$c_p$  ist die isobare Wärmekapazität. Der Energiefluss ergibt sich damit zu

$$\vec{j}_q = -\frac{\lambda}{c_p} \nabla h_s + \sum_{k=1}^N \frac{\lambda}{c_p} h_{s,k} \nabla Y_k + \sum_{k=1}^N \vec{j}_k h_{s,k} \quad (2.56)$$

$\vec{j}_k$  ist dabei die Speziesmassendiffusion aus Unterabschnitt 2.4.3.

## 2.5.2 Wärmeleitfähigkeit

Die Wärmeleitfähigkeit  $\lambda$  in Gl. (2.56) kann, wie die binären Diffusionskoeffizienten und die Viskosität, über die Chapman-Enskog Beziehung errechnet werden [28]:

$$\lambda_k = \frac{25}{32\pi^{\frac{1}{2}}} \left( \frac{k_B T}{m_k} \right)^{\frac{1}{2}} \frac{\tilde{c}_v}{\sigma_k^2 N_A \Omega_{kk}^{(2,2)*}} \quad (2.57)$$

$\tilde{c}_v$  ist die isochore Wärmekapazität. Sie lässt sich über die Mayer-Beziehung bestimmen:

$$c_v = \frac{1}{M} (\tilde{c}_p - \bar{R}) \quad (2.58)$$

Statt über die Chapman-Enskog Beziehung wird die Wärmeleitfähigkeit im DNS Löser aber über die Speziesviskosität  $\mu_k$  und den Selbstdiffusionskoeffizienten  $\mathcal{D}_{k,k}$  errechnet, um Rechenzeit zu sparen [26]:

$$\lambda_k = \frac{\mu_k}{M_k} (f_{trans} \tilde{c}_{v,trans} + f_{rot} \tilde{c}_{v,rot} + f_{vib} \tilde{c}_{v,vib}) \quad (2.59)$$

Die Faktoren  $f_{trans}$ ,  $f_{rot}$ ,  $f_{vib}$  werden wie folgt bestimmt:

$$f_{vib} = \frac{\rho \mathcal{D}_{k,k}}{\mu_k} \quad (2.60)$$

$$f_{trans} = \frac{5}{2} \left( 1 - \frac{2}{\pi} \frac{\tilde{c}_{v,rot} A}{\tilde{c}_{v,trans} B} \right) \quad (2.61)$$

$$f_{rot} = f_{vib} \left( 1 + \frac{2A}{\pi B} \right) \quad (2.62)$$

$$A = \frac{5}{2} - f_{vib} \quad (2.63)$$

$$B = Z_{rot} + \frac{2}{\pi} \left( \frac{5}{3} \frac{\tilde{c}_{v,rot}}{R} + f_{vib} \right) \quad (2.64)$$

Der Anteil an der translationalen Energie  $\tilde{c}_{v,trans}$  ist ein konstanter Wert:

$$\tilde{c}_{v,trans} = \frac{3}{2} \bar{R} \quad (2.65)$$

$\tilde{c}_{v,rot}$  ist für ideale Gase abhängig von der Molekülstruktur:

$$\tilde{c}_{v,rot}(\text{eiatomig}) = 0 \quad (2.66)$$

$$\tilde{c}_{v,rot}(\text{linear}) = \bar{R} \quad (2.67)$$

$$\tilde{c}_{v,rot}(\text{nicht linear}) = \frac{3}{2} \bar{R} \quad (2.68)$$

$\tilde{c}_{v,vib}$  wird über die Schließbedingung und die Mayer-Beziehung in Gl. (2.58) bestimmt:

$$\tilde{c}_v = \tilde{c}_{v,vib} + \tilde{c}_{v,rot} + \tilde{c}_{v,trans} \quad (2.69)$$

Insgesamt vereinfacht sich damit der Ausdruck für  $c_{v,vib}$ :

$$\tilde{c}_{v,vib} = -\tilde{c}_{v,rot} + \tilde{c}_p - \frac{2}{5} \bar{R} \quad (2.70)$$

$Z_{rot}$  in Gl. (2.64) ist die relationale Kollisionszahl. Sie ist abhängig von der Temperatur und beschreibt, wie viele Kollisionen nötig sind, um ein angeregtes Molekül wieder in den Grundzustand zu bringen. Wird für den Löser der Multikomponenten-Ansatz

gewählt, dann berechnet sich  $Z_{rot}$  nach folgender Gleichung:

$$Z_{rot} = Z_{rot}(T = 298 \text{ K}) \frac{F(T = 298 \text{ K})}{F(T)} \quad (2.71)$$

$F(T)$  ist dabei definiert als

$$F(T) = 1 + \frac{\pi^{\frac{3}{2}}}{2} \sqrt{\left(\frac{\epsilon}{k_B T}\right)} + \left(\frac{\pi^2}{4} + 2\right) \left(\frac{\epsilon}{k_B T}\right) + \pi^{\frac{3}{2}} \left(\frac{\epsilon}{k_B T}\right)^{\frac{3}{2}} \quad (2.72)$$

Im Mischungsgemittelten Transportmodell wird  $Z_{rot}$  dagegen als konstant angesehen:

$$Z_{rot}(T) \equiv Z_{rot}(T = 298 \text{ K}) \quad (2.73)$$

Die Stoffkonstante  $Z_{rot}(T = 298 \text{ K})$  wird für jede Spezies aus einer Datenbank ausgelesen.

Auch die Wärmeleitfähigkeiten der Spezies werden zur Laufzeit des DNS Löser - wie auch die binären Diffusionskoeffizienten und die Speziesviskositäten - über ein Polynom mit folgender Form approximiert:

$$\lambda_k = \sqrt{T} \left( a_0 + a_1 \ln T + a_2 \ln^2 T + a_3 \ln^3 T + a_4 \ln^4 T \right) \quad (2.74)$$

Die Wärmeleitfähigkeit der Mischung wird schließlich über alle Spezies gemittelt [26]:

$$\lambda = \frac{1}{2} \left( \sum_{k=1}^N X_k \lambda_k + \frac{1}{\sum_{k=1}^N \frac{X_k}{\lambda_k}} \right) \quad (2.75)$$

### 2.5.3 Wärmekapazität und Enthalpie

Die isobare Wärmekapazität in Gl. (2.52) ist in Abhängigkeit der Enthalpie definiert:

$$c_{p,k} = \left( \frac{\partial h_k}{\partial T} \right)_p \quad (2.76)$$

Die Berechnungen der isobaren Wärmekapazität und der Enthalpie werden im DNS Löser approximiert. Dazu werden zunächst die molare Enthalpie  $\tilde{h}_k$  und die molbezoge-

ne isobare Wärmekapazität  $\tilde{c}_p$  über NASA-Polynome berechnet [35]. Auch die molare Entropie  $\tilde{s}_k$  wird über ein solches Polynom berechnet. Die molare Entropie wird bei der Berechnung der Gibbs-Energie in Unterabschnitt 2.4.4 und der inneren Energie in Abschnitt 4.1 verwendet.

$$\frac{\tilde{c}_{p,k}}{\bar{R}} = a_0 + a_1T + a_2T^2 + a_3T^3 + a_4T^4 \quad (2.77)$$

$$\frac{\tilde{h}_k}{\bar{R}T} = a_0 + \frac{a_1}{2}T + \frac{a_2}{3}T^2 + \frac{a_3}{4}T^3 + \frac{a_4}{5}T^4 + a_5 \quad (2.78)$$

$$\frac{\tilde{s}_k}{\bar{R}} = a_0 \ln T + a_1T + \frac{a_2}{2}T^2 + \frac{a_3}{3}T^3 + \frac{a_4}{4}T^4 + a_6 \quad (2.79)$$

Die Koeffizienten  $a_0$  bis  $a_6$  der Polynome werden dabei für jede Spezies aus einer Datenbank gelesen. Um eine möglichst hohe Genauigkeit bei der Berechnung dieser Werte zu erhalten, gibt es sowohl Koeffizienten für niedrige Temperaturbereiche, als auch Koeffizienten für hohe Temperaturbereiche. Die Umrechnung der molbezogenen Größen in massenbezogene Größen, wie sie in der Enthalpiebilanz auftauchen, ist im Folgenden gezeigt:

$$c_{p,k} = \frac{\tilde{c}_{p,k}}{\bar{M}} \quad (2.80)$$

$$h_k = \frac{\tilde{h}_k}{\bar{M}} \quad (2.81)$$

$\bar{M}$  ist die mittlere Molmasse der Mischung:

$$\bar{M} = \sum_{k=1}^N X_k M_k = \left( \sum_{k=1}^N \frac{Y_k}{M_k} \right)^{-1} \quad (2.82)$$

Enthalpie und isobare Wärmekapazität werden schließlich für die Mischung gemittelt:

$$h = \frac{1}{\bar{M}} \sum_{k=1}^N X_k \tilde{h}_k = \sum_{k=1}^N Y_k h_k \quad (2.83)$$

$$c_p = \frac{1}{\bar{M}} \sum_{k=1}^N X_k \tilde{c}_{p,k} = \sum_{k=1}^N Y_k c_{p,k} \quad (2.84)$$

Der Quellterm  $\dot{q}_R$  der Energieerhaltung ist die Wärmefreisetzungsrate der chemischen Reaktion und wird berechnet über die Speziesreaktionsraten aus Gl. (2.33):

$$\dot{q}_R = - \sum_{k=1}^N \dot{r}_k h_k^\circ \quad (2.85)$$

### 2.5.4 Ermittlung der Temperatur

Die Energieerhaltungsgleichung wurde in Gl. (2.56) so umgeformt, dass sie in Abhängigkeit der thermischen Enthalpie und nicht mehr in Abhängigkeit der Temperatur gelöst wird. Dadurch muss ein weiterer Zusammenhang zwischen Temperatur und Enthalpie geschaffen werden, um die Temperaturänderung aus der Mischungsenthalpie  $h_s$  zu berechnen. Die Temperatur lässt sich als Funktion der Mischungsenthalpie darstellen:

$$c_p = \left( \frac{\partial h}{\partial T} \right)_p \quad (2.86)$$

Anders als bei der Energieerhaltung wird dazu nicht die thermische Enthalpie, sondern die absolute Enthalpie verwendet. Nach der Lösung der Energieerhaltung gibt es eine neue Mischungsenthalpie:

$$h_s = \sum_{k=1}^N Y_k h_{k,s}$$

Daher wird eine Temperatur gesucht, bei der sich für die aktuelle Mischungszusammensetzung die gleiche Mischungsenthalpie ergibt, wie durch die Energieerhaltung berechnet. Die Iterationsvorschrift kann dabei (vereinfacht) aus Gl. (2.86) abgelesen werden:

$$T_{n+1} = T_n + \frac{h_0 - h_n}{c_{p,n}} \quad (2.87)$$

$h_0$  ist die Mischungsenthalpie, die durch die Lösung der Energieerhaltung berechnet wurde. Mit der alten Temperatur  $T_n$  wird dann eine Mischungsenthalpie  $h_n$  und eine Wärmekapazität  $c_{p,n}$  mit Gl. (2.77) bestimmt und die Temperatur der nächsten Iteration  $T_{n+1}$  berechnet. Die Iteration wird so lange durchgeführt, bis das Abbruchkriterium

erreicht ist:

$$|T_{n+1} - T_n| \leq 10^{-4} \text{ K} \quad (2.88)$$

## 2.6 Zustandsgleichung

Keine der bisher genannten Erhaltungsgleichungen liefert einen direkten Zusammenhang zwischen Druck und Dichte. Da die Reaktionsphase als Mischung idealer Gase angesehen wird, ist diese Beziehung durch die thermische Zustandsgleichung idealer Gase gegeben:

$$p = \rho RT \quad (2.89)$$

$R$  ist die spezifische Gaskonstante der Mischung. Sie wird über die universelle Gaskonstante  $\bar{R}$  und mittlere Molmasse der Mischung  $\bar{M}$  aus Gl. (2.82) bestimmt:

$$R = \frac{\bar{R}}{\bar{M}} \quad (2.90)$$

## 2.7 Lösung des Gleichungssystems

Mathematisch ergeben die Erhaltungsgleichungen ein Gleichungssystem aus partiellen Differentialgleichungen. Dieses besteht aus der Gesamtmassenerhaltung Gl. (2.91), der Impulserhaltung in allen drei Raumrichtungen Gl. (2.92), (N-1) Speziesmassenerhaltungen Gl. (2.93) und der Energieerhaltung Gl. (2.94). Die Schließung des Gleichungssystems erfolgt über die Zustandsgleichung Gl. (2.95). Die folgenden Gleichungen, aus denen das gesamte Gleichungssystem besteht, sind so notiert, wie sie auch im aktuellen

DNS Löser berechnet werden:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (2.91)$$

$$\frac{\partial}{\partial t} (\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot \mu \left( (\nabla \vec{v} + (\nabla \vec{v})^T) - \frac{2}{3} \delta_{ij} \nabla \cdot \vec{v} \right) + \rho \vec{g} \quad (2.92)$$

$$\frac{\partial}{\partial t} (\rho Y_k) + \nabla \cdot \rho (\vec{v} + \vec{V}^c) Y_k = \nabla \cdot \rho D_k \nabla Y_k + \dot{r}_k \quad (2.93)$$

$$\frac{\partial}{\partial t} (\rho h_s) + \nabla \cdot (\rho h_s \vec{v}) = \nabla \cdot \frac{\lambda}{c_p} \nabla h_s - \sum_{k=1}^N \nabla \cdot \frac{\lambda}{c_p} h_{s,k} \nabla Y_k \quad (2.94)$$

$$- \sum_{k=1}^N \nabla \cdot (h_{s,k} \vec{j}_k) + \frac{Dp}{Dt} + \dot{q}_R$$

$$p = \rho RT \quad (2.95)$$

Dieses partielle Differentialgleichungssystem ist z.B. wegen der exponentiellen Abhängigkeit der Reaktionsraten zur Temperatur nicht linear und somit analytisch nicht lösbar. Stattdessen muss es numerisch gelöst werden. Im DNS Löser wird dafür die Finite Volumen Methode (FVM) genutzt, die häufig für CFD Probleme verwendet wird. Dafür wird das Rechengebiet in eine endliche Anzahl von Volumenelementen (Zellen) unterteilt. Die Erhaltungsgleichungen werden dadurch in integraler Form diskretisiert. So kann ihre Ableitung durch die diskreten Werte der Nachbarstellen approximiert werden. Die Erhaltungsgleichungen werden für jedes Volumenelement gelöst, wobei die einzelnen Volumenelemente über ihre Oberfläche mit den Nachbarelementen verbunden sind. Das dabei entstehende algebraische Gleichungssystem muss iterativ gelöst werden [13].

# Kapitel 3

## Weiterentwicklung des DNS Löser

Der Schwerpunkt dieser Arbeit liegt in der Weiterentwicklung eines vorhandenen DNS Löser [5]. Obwohl der Löser bereits erfolgreich validiert worden ist, gibt es noch Nachteile, die mit der Verwendung der Bibliothek Cantera zusammenhängen:

- Bevor der Löser verwendet werden kann, muss erst Cantera installiert werden
- Man ist an die installierte Canteraversion gebunden und kann z.B. nicht den Einfluss verschiedener Compiler-Einstellungen testen, ohne Cantera jedes Mal neu installieren zu müssen
- Der Canteracode ist nur sehr umständlich änderbar und kann deshalb nicht für den Löser angepasst werden. Bei jeder Änderung im Code muss Cantera neu installiert werden; zudem ist der Codeumfang von Cantera so groß, dass relevante Teile für den Löser nur schwer ersichtlich sind
- Der Canteracode daher ist nur schwer einsehbar, wodurch das Verhalten des Löser nicht immer nachvollziehbar ist
- Die Rechengeschwindigkeit ist wegen unnötiger Berechnungen suboptimal (Unterabschnitt 3.4.1)

Das Ziel dieser Arbeit ist es daher, die für den Löser relevanten Codeteile aus Cantera zu extrahieren und direkt in OpenFOAM zu implementieren. Im Folgenden werden

zunächst die verwendeten Softwarepakete näher vorgestellt und dann anhand von Beispielen beschrieben, durch welche Maßnahmen der DNS Löser in dieser Arbeit verbessert wurde. Die Messungen der Rechenzeit in diesem und folgenden Kapiteln sind mit mehreren Reaktionsmechanismen durchgeführt worden.

## 3.1 Softwarepakete und Cluster-Plattformen

Die drei wichtigsten Softwarepakete, die vom Löser verwendet werden, sind OpenFOAM [38], Cantera [19] und SUNDIALS [43]. OpenFOAM löst alle Erhaltungsgleichungen und berechnet die Strömung. Cantera liefert die Grundlagen zur Berechnung druck- und temperaturabhängiger Stoffwerte und verwaltet die Reaktionskinetik. SUNDIALS schließlich löst die steife, nicht-lineare Reaktorgleichung zur Berechnung der Reaktionsraten, die in OpenFOAM als Quellterm in der Speziesmassenerhaltung in Gl. (2.93) verwendet werden.

### 3.1.1 OpenFOAM

Der Hauptteil des DNS Lösers ist mit dem Softwarepaket OpenFOAM implementiert. OpenFOAM steht für „Open Source Field Operation and Manipulation“. Es stellt das Grundgerüst des Programms dar und übernimmt dabei hauptsächlich das Lösen der Erhaltungsgleichungen. OpenFOAM ist ein frei verfügbares open-source Projekt, das unter der GNU General Public License (GPL) lizenziert und aktuell in der Version 2.2.2 verfügbar ist. Es findet in vielen CFD Bereichen Anwendung, wodurch es vielfach validiert worden ist [37]. OpenFOAM ist in der Programmiersprache C++ geschrieben und bietet eine leicht verständliche Schnittstelle. Weitere Vorteile von OpenFOAM sind

- Umfangreiche numerische Hilfsmittel für CFD Anwendungen wie z.B. Diskretisierungen, Randbedingungen und iterative Löser
- eine große Anzahl an vorhandenen physikalischen Modellen wie z.B. Turbulenz- oder Verbrennungsmodelle
- die einfache Implementierung eigener Modelle durch die gut strukturierte Code-dokumentation mit Doxygen [11]

Eine andere Stärke von OpenFOAM ist, dass die Arbeit des Löser leicht auf eine Vielzahl von Prozessorkernen verteilt und gleichzeitig eine sehr gute Skalierbarkeit erreicht werden kann. Dadurch ist es möglich, Simulationen auf Supercomputern wie dem IBM-Computer JUQUEEN [22] laufen zu lassen und den größtmöglichen Nutzen aus der parallelen Architektur zu ziehen. OpenFOAM unterteilt das Simulationsgebiet dazu in Untergebiete, die dann den verfügbaren Prozessorkernen zugewiesen werden. Die Kommunikation bzw. der Austausch von Daten zwischen den Prozessorkernen geschieht dabei über ein „Message Passing Interface“ (MPI). OpenFOAM verwendet dafür standardmäßig Open MPI [39]; es können aber auch spezielle MPI Systeme benutzt werden, die für die Zielarchitektur optimiert worden sind [37].

### 3.1.2 Cantera

Cantera [19] wurde ursprünglich von Prof. David Goodwin vom California Institute of Technology entwickelt. Sein Ziel war es, eine kostenfreie, open-source Alternative zu dem kommerziellen Programm CHEMKIN [8] anzubieten. Cantera ist in C++ geschrieben und bietet eine Schnittstelle zu weiteren Programmiersprachen wie Python, Fortran und Matlab. Es ist lizenziert unter der „Berkeley Software Distribution 3-Clause License“ (BSD-3).

Cantera besteht aus insgesamt 648 C++ Quellcode-dateien und bietet Funktionalitäten für die unterschiedlichsten chemischen Bereiche an: Es unterstützt über 20 Modelle für Reaktionen in verschiedenen Phasen – von idealen Gasphasen bis zu metallischen Phasen –, 8 Methoden, um die thermodynamischen Eigenschaften der Spezies zu berechnen (z.B. über NASA-Polynome), und 8 Modelle für verschiedene Reaktionstypen. Die Routinen, die dafür zur Verfügung gestellt werden, sind stark optimiert, um optimales Laufzeitverhalten zu gewährleisten. Die chemischen Reaktionen werden dabei in einem idealisierten Reaktor simuliert. Cantera bietet die Möglichkeit, Reaktornetze aus diversen Reaktortypen wie idealen Rührkesseln (BR) oder Pfropfstromreaktoren (PFR) zu erstellen und Randbedingungen wie Ein- und Auslässe und Temperaturgradienten zu definieren.

Cantera bedient sich zusätzlich der externen Bibliotheken SUNDIALS [43] und BLAS/LAPACK. SUNDIALS steht für „Suite of nonlinear and differential/algebraic

equation solvers“ und liefert verschiedene, hochoptimierte Routinen zum Lösen von gewöhnlichen Differentialgleichungen. SUNDIALS selbst hat direkt nichts mit der Berechnung von chemischen Quelltermen zu tun. Stattdessen wird die Reaktorgleichung von Cantera so vorbereitet, dass sie von SUNDIALS gelöst werden kann. Der Grund dafür ist, dass die Reaktorgleichung ein nicht-lineares und steifes Differentialgleichungssystem darstellt, das numerisch nur aufwendig zu lösen ist [26]. Im DNS Löser wird nur der Teil von SUNDIALS verwendet, der gewöhnliche Differentialgleichungen als Anfangswertproblem löst (CVODE). SUNDIALS ist in der Version 2.5.0 verfügbar und unter der BSD Lizenz lizenziert [21].

BLAS (Basic Linear Algebra Subprograms) und LAPACK (Linear Algebra Package) sind Bibliotheken zur Lösung von Problemen der linearen Algebra. Dabei gibt es viele BLAS/LAPACK Versionen, die teilweise stark für einzelne Prozessorarchitekturen, wie die des IBM Computers JUQUEEN, optimiert sind. BLAS/LAPACK wird von Cantera hauptsächlich für die Multiplikation großer Matrizen verwendet.

Cantera selbst hat auch einen Parser zum Einlesen von Werten aus Dateien im CHEMKIN Format und einen Konverter, der die CHEMKIN Dateien in ein Cantera eigenes Format umwandelt. Cantera steht aktuell in der Version 2.0.2 zur Verfügung [20]. Im DNS Löser werden sämtliche temperatur- und druckabhängigen Stoffwerte mit Funktionen berechnet, die aus Cantera übernommen wurden.

### 3.1.3 Cluster-Plattformen

Der in dieser Arbeit optimierte DNS Löser ist auf zwei Cluster-Plattformen weiterentwickelt und validiert worden; zum einen auf dem Institutscluster des Engler-Bunte-Instituts im Bereich Verbrennungstechnik (VBT) und zum anderen auf dem Supercomputer JUQUEEN. Der Institutscluster besteht aus 196 Prozessorkernen mit einem Gesamtarbeitsspeicher von 608 GB und hat eine x86-Prozessorarchitektur [47]. Als Betriebssystem wird GNU/Linux verwendet. Das Clustersystem ist „Beowulf“ [3]. Der IBM-Computer JUQUEEN Blue Gene/Q des Jülich Supercomputing Centre's (JSC) [22] besteht aus 458752 Prozessorkernen und hat einen Hauptspeicher von 448 TB. Mit einer Höchstleistung von 5.9 Petaflops<sup>1</sup> ist JUQUEEN damit auf Platz 7 der schnell-

---

<sup>1</sup>10<sup>15</sup> Fließkommazahloperationen pro Sekunde

sten Supercomputer der Welt und Platz 1 in Europa [46]. Die Prozessorarchitektur ist PowerPC, als Betriebssystem wird ebenfalls GNU/Linux verwendet. Der DNS Löser wurde auf JUQUEEN mit bis zu 8192 CPU-Kernen getestet (Unterabschnitt 5.2.5).

## 3.2 Basislöser

Diese Arbeit basiert auf der am EBI durchgeführten Bachelorarbeit von Bonart [4], in der der ursprüngliche DNS Löser in OpenFOAM implementiert wurde. Zudem wurde eine Kopplungsbibliothek erstellt, die als Schnittstelle zwischen OpenFOAM und Cantera dient. Dazu wurde OpenFOAM in der Version 2.1 und Cantera in der Version 1.8 genutzt. Seitdem wurde der Löser weiterentwickelt, sodass nun OpenFOAM in der Version 2.2.2 und Cantera in der Version 2.0 [5] unterstützt wird. Letzteres hat einen Gewinn von 30 % der Rechengeschwindigkeit ergeben (Unterabschnitt 3.2.2).

**Tabelle 3.1:** *Kurzübersicht über die Stoff- und Reaktionsdaten, die vom Löser beim Programmstart aus einer Datenbank eingelesen werden müssen.*

<b>Transportdaten</b>
Molekülgeometrie Gl. (2.66) - Gl. (2.68)
Tiefe des Energieminimums des Lennard-Jones Potentials $\epsilon_k$ Gl. (2.11)
Lennard-Jones Kollisionsdurchmesser $\sigma_k$ Gl. (2.20)
Dipolmoment $\mu_k$
relationale Kollisionszahl $Z_{rot}$ Gl. (2.64), Gl. (2.71)
<b>Thermodynamische Stoffdaten</b>
NASA-Koeffizienten ( $a_0 - a_6$ ) für je zwei Temperaturbereiche Gl. (2.77) - Gl. (2.79)
<b>Reaktionsdaten</b>
Arrheniusparameter $A, \beta, E_a$ Gl. (2.31)
Effektivitäten der Dreistoß-Reaktionen $\epsilon_k$ Gl. (2.49)
Arrheniusparameter für Grenzdrücke $\beta_0, \beta_\infty, E_0, E_\infty, A_0, A_\infty$ Gl. (2.39) und Gl. (2.40)
Parameter für Troe Reaktionen $\alpha, T^*, T^{**}, T^{***}$ Gl. (2.47)

### 3.2.1 Aufbau des Basislösers

Die Erhaltungsgleichungen aus Kapitel 2 werden im DNS Löser von OpenFOAM gelöst. Die Berechnung der druck- und temperaturabhängigen Größen wie z.B. Reaktionsraten oder Transporteigenschaften geschieht mit Hilfe von Cantera. Eine Kopplungsbibliothek dient als Schnittstelle zwischen OpenFOAM und Cantera. In Kapitel 2 wurde genannt, welche Stoffwerte beim Programmstart aus einer Datenbank gelesen werden müssen. Diese sind in Tabelle 3.1 noch einmal zusammengefasst.

Um die Beschreibung des Lösers übersichtlich zu halten, wird der Löser im Folgenden in die Dateien gegliedert, die im Hauptprogramm eingebunden werden. Auch das Ablaufdiagramm (Abbildung 3.1) ist mit dieser Einteilung strukturiert. Eine kurze Übersicht darüber, welche Aufgaben von den jeweiligen Dateien übernommen werden, ist in Tabelle 3.2 aufgeführt.

Im ersten Abschnitt des DNS Lösers - **preCalc.H** - werden zwei Größen vorberechnet: zum einen die mittlere molare Masse der Mischung nach Gl. (2.82), zum anderen die thermischen Enthalpien der reinen Spezies nach Gl. (2.78) und Gl. (2.53). Die thermischen Enthalpien der Spezies werden später in der Enthalpiebilanz für den Wärmestrom in Gl. (2.56) verwendet.

Im Abschnitt **chemistry.H** werden die mittleren Reaktionsraten der Spezies und die Wärmefreisetzungsrate nach Gl. (2.85) ermittelt. Die hier bestimmten Reaktionsraten werden als Quellterm in der Speziesmassenerhaltung in Gl. (2.93), die Wärme-

**Tabelle 3.2:** *Kurzübersicht über die Abschnitte im DNS Löser. Die Einteilung in Abschnitte erfolgt nach den eingebundenen Quellcodedateien.*

Datei	Beschreibung
<b>preCalc.H</b>	Berechnet Enthalpien und mittlere Molmasse
<b>chemistry.H</b>	Löst die Reaktorgleichung zur Bestimmung der Reaktionsraten
<b>rhoEqn.H</b>	Löst die Massenerhaltung
<b>UEqn.H</b>	Löst die Impulserhaltung
<b>YEqn.H</b>	Löst die Speziesmassenerhaltung
<b>hSEqn.H</b>	Löst die Energieerhaltung und berechnet Transportkoeffizienten
<b>pEqn.H</b>	Berechnet Druckkorrekturen

freisetzungsrates als Quellterm in der Enthalpiebilanz in Gl. (2.94) verwendet. Je nach Komplexität des Reaktionsmechanismus nimmt dieser Programmteil 50 % bis 70 % der Gesamtrechnzeit in Anspruch (Unterabschnitt 3.2.3). Zur Berechnung der Reaktionsraten wird für jede Zelle im Simulationsgebiet ein idealer Rührkessel (BR) erstellt. Dann wird die Reaktorgleichung für diesen Reaktor durch SUNDIALS iterativ gelöst, indem die Reaktionskinetik innerhalb des DNS Zeitschritts mehrfach ausgewertet wird. Schließlich werden die Reaktionsraten über den DNS Zeitschritt gemittelt. Der Grund für die hohe Rechenzeit dieses Programmteils ist, dass die Lösung der Reaktorgleichung sehr aufwendig ist. Auf welche Weise die Reaktionsraten ermittelt werden und warum die Lösung der Reaktorgleichung so zeitaufwendig ist, wird in Abschnitt 4.1 diskutiert. Da die Berechnungen in **chemistry.H** hauptsächlich durch Cantera gemacht werden, liegt der Schwerpunkt der Optimierungen in dieser Arbeit auf dem Canteracode.

In **rhoEqn.H** wird die Massenerhaltung nach Gl. (2.4) gelöst. Da keine temperatur- oder druckabhängigen Größen neu berechnet werden müssen und die Erhaltungsgleichung keine Quell- oder Diffusionsterme enthält, ist ihr Einfluss auf die Gesamtrechnzeit mit weniger als 0.1 % vernachlässigbar.

Danach wird in **UEqn.H** die Impulserhaltung nach Gl. (2.5) gelöst. Die Mischungviskosität, die als Term im Spannungstensor in Gl. (2.6) erscheint, wird nicht hier berechnet, sondern ist nach dem Lösen der Energieerhaltung in **hsEqn.H** des letzten Zeitschritts bestimmt worden.

Für die Lösung der Speziesmassenerhaltung in **YEqn.H** nach Gl. (2.93) werden als temperatur- und druckabhängige Größen die Reaktionsraten und die Diffusionskoeffizienten der Spezies benötigt. Allerdings werden in diesem Abschnitt keine temperatur- und druckabhängigen Werte neu berechnet: die Reaktionsraten sind in **chemistry.H**, die Diffusionskoeffizienten nach dem Lösen der Enthalpiebilanz des letzten Zeitschritts berechnet worden.

Im Abschnitt **hsEqn.H** wird die Energieerhaltungsgleichung in Form der Enthalpiebilanz nach Gl. (2.94) gelöst. Nachdem diese gelöst ist, wird die Temperatur aus den neuen Enthalpiewerten berechnet (Unterabschnitt 2.5.4). Mit der neuen Temperatur werden dann die Transportgrößen neu berechnet: die Wärmeleitfähigkeit  $\lambda$  nach Gl. (2.75), die Viskosität  $\mu$  nach Gl. (2.13), und die Diffusionskoeffizienten  $D_k$  nach

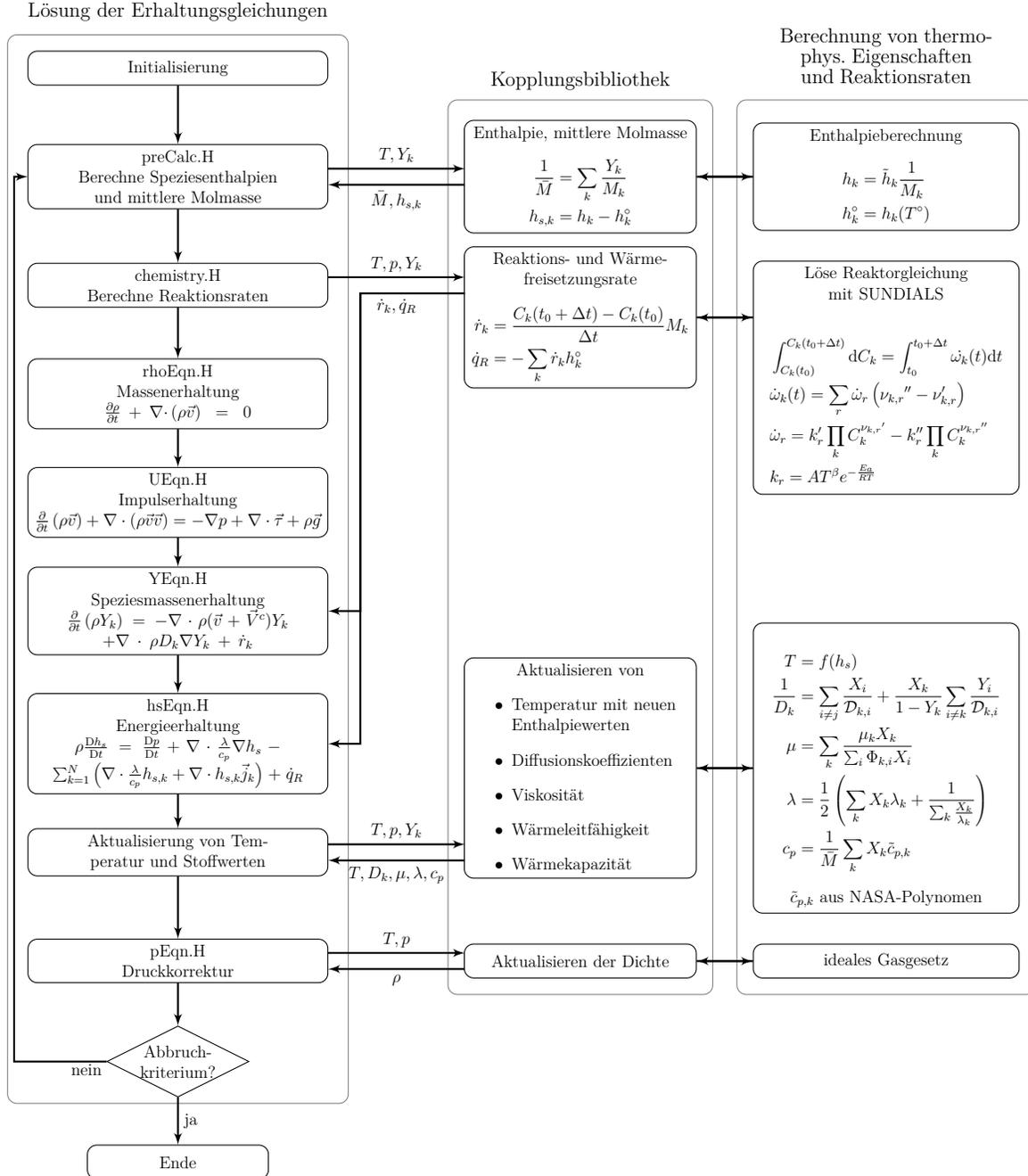


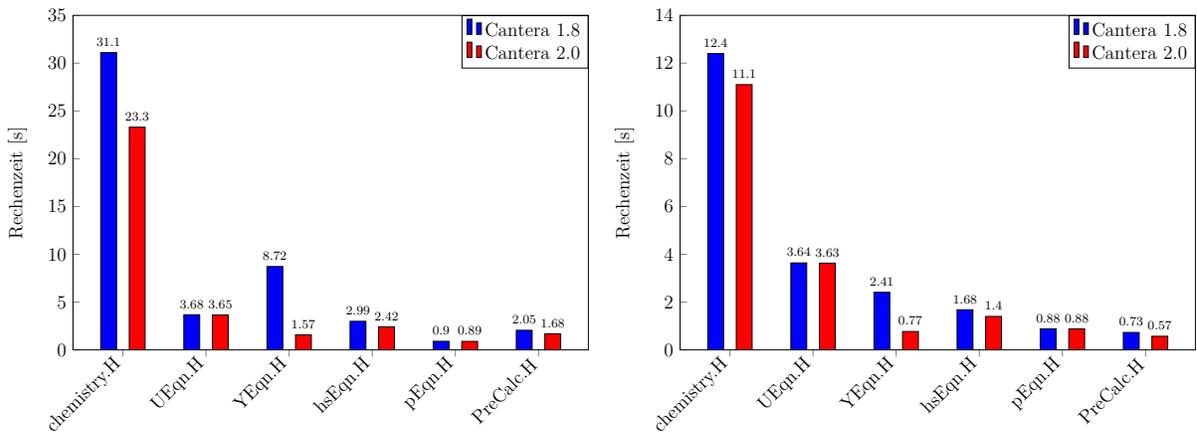
Abbildung 3.1: Vereinfachter Ablaufplan des DNS Löser.

Gl. (2.22). Zuletzt wird in **pEqn.H** eine Korrekturgleichung [13] für den Druck gelöst und die Dichte der Mischung mit Hilfe der idealen Gasgleichung neu berechnet.

Abbildung 3.1 fasst die Funktionsweise des DNS Löser als Fließdiagramm zusammen. Links sind die Erhaltungsgleichungen aufgeführt, die von OpenFOAM gelöst werden. Rechts sind die Berechnungen der druck- und temperaturabhängigen Stoffgrößen sowie die der Reaktionsraten aufgeführt, die im Programm von Cantera übernommen werden. In der Mitte ist die Kopplungsbibliothek dargestellt, die den Datenaustausch zwischen OpenFOAM und Cantera übernimmt.

### 3.2.2 Portierung nach den neusten Programmversionen

Der ursprüngliche DNS Löser wurde für Cantera in der Version 1.8 entwickelt. Seitdem ist er für Cantera 2.0 portiert worden [5]. Wird Cantera 2.0 genutzt, ergibt sich ein Leistungsgewinn von ca. 30 % gegenüber Cantera 1.8. Der Unterschied in der Rechenzeit der einzelnen Löserabschnitte für den Reaktionsmechanismus einer Methan/Luft Verbrennung (B.4) und einer Wasserstoff/Luft Verbrennung nach einem Mechanismus von Maas und Pope (B.3) ist in Abbildung 3.2 dargestellt. Die Aufgaben jedes Abschnitts



**Abbildung 3.2:** Gemittelte Rechenzeiten für die Löserabschnitte eines Zeitschritts mit den Cantera Versionen 1.8 und 2.0, gemessen auf JUQUEEN mit 1024 Prozessorkernen und einem 2 Millionen Zellen Gitter. **Links:** Methan/Luft Verbrennung (B.4). **Rechts:** Wasserstoff/Luft Verbrennung (B.3).

sind in Tabelle 3.2 zusammengefasst.

Die Berechnung der Reaktionsraten in **chemistry.H** ist in Cantera 2.0 effizienter implementiert als in Cantera 1.8. Ein weiterer Unterschied liegt in der Lösung der Speziesmassenerhaltung in **YEqn.H**. Im Falle der Methan/Luft Verbrennung dauert dieser Schritt bis zu sechsmal länger. Der Grund dafür liegt darin, wie in Cantera 1.8 die Diffusionskoeffizienten ermittelt werden. Der Diffusionsterm wird nach Gl. (2.17) gelöst:

$$\vec{j}_k = -\rho D_k \nabla Y_k$$

Dabei sind  $D_k$  die massenbezogenen Diffusionskoeffizienten. In der Kopplungsbibliothek für Cantera 1.8 gibt es nicht die Möglichkeit, massenbezogene Diffusionskoeffizienten zu berechnen. Stattdessen werden molbezogene Diffusionskoeffizienten benutzt. Der Term für die Massendiffusion sieht damit wie folgt aus:

$$\vec{j}_k = -\rho \tilde{D}_k \nabla X_k \quad (3.1)$$

Da im DNS Löser mit Massenbruchgradienten gerechnet wird, muss diese Gleichung umgeformt werden. Mit der Beziehung

$$X_k = \frac{\bar{M}}{M_k} Y_k \quad (3.2)$$

ergibt sich der Molenbruchgradient in Abhängigkeit der Massenbrüche:

$$\nabla X_k = \nabla \frac{\bar{M}}{M_k} Y_k = \frac{\bar{M}}{M_k} \nabla Y_k + Y_k \nabla \frac{\bar{M}}{M_k} \quad (3.3)$$

Mit der Definition der mittleren Molmasse der Mischung

$$\bar{M} = \sum_{k=1}^N X_k M_k \quad (3.4)$$

lässt sich der Molenbruchgradient weiter umformen [4]:

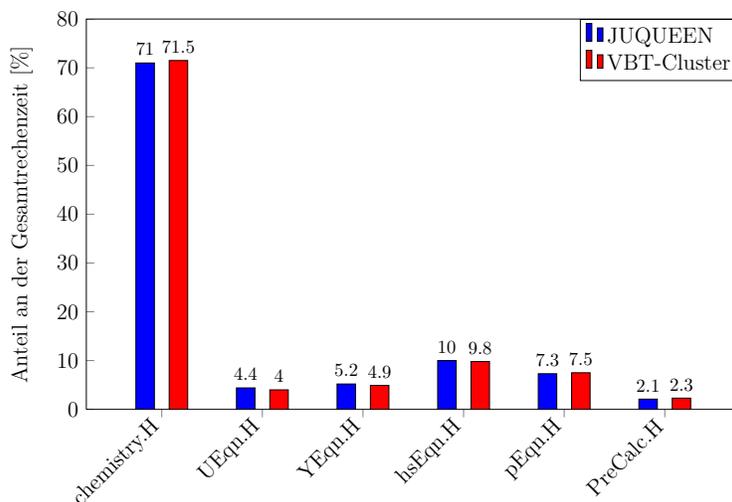
$$\begin{aligned}
\nabla X_k &= \frac{\bar{M}}{M_k} \nabla Y_k + Y_k \nabla \frac{\bar{M}}{M_k} \\
&= \frac{\bar{M}}{M_k} \nabla Y_k + \frac{Y_k}{M_k} \nabla \left( \sum_{i=1}^N \frac{Y_i}{M_i} \right)^{-1} \\
&= \frac{\bar{M}}{M_k} \nabla Y_k - \frac{Y_k}{M_k} \underbrace{\left( \frac{1}{\sum_{i=1}^N \frac{Y_i}{M_i}} \right)^2}_{\bar{M}^2} \nabla \sum_{i=1}^N \frac{Y_i}{M_i} \\
&= \frac{\bar{M}}{M_k} \nabla Y_k - \frac{Y_k}{M_k} \bar{M}^2 \sum_{i=1}^N \frac{1}{M_i} \nabla Y_i
\end{aligned} \tag{3.5}$$

Der Massenfluss kann nun in Abhängigkeit der Massenbrüche dargestellt werden:

$$\vec{j}_k = \rho \tilde{D}_k \frac{\bar{M}}{M_k} \nabla Y_k - \rho \tilde{D}_k \frac{Y_k}{M_k} \bar{M}^2 \sum_{i=1}^N \frac{1}{M_i} \nabla Y_i \tag{3.6}$$

Im Gegensatz zu Gl. (2.17) ist in diesem Ausdruck der Rechenaufwand deutlich höher. Zur Bestimmung des Massenflusses einer Spezies muss über den Massenbruchgradienten aller Spezies summiert werden. Berechnet man die Massenflüsse aller N Spezies, so wird N-Mal über alle N Speziesmassenbruchgradienten summiert, wodurch der Rechenaufwand quadratisch mit der Speziesanzahl steigt. Da bei der Methan/Luft Verbrennung mehr Spezies beteiligt sind als bei der Wasserstoff/Luft Verbrennung in Abbildung 3.2, erklärt dies auch, warum die Lösung der Speziesmassenerhaltung bei der Methan/Luft Verbrennung verhältnismäßig länger dauert als bei der Wasserstoff/Luft Verbrennung.

Eine Portierung auf die OpenFOAM Version 2.2 hat keine signifikante Verbesserung der Rechengeschwindigkeit gezeigt. Bei der Verwendung von Zeitdiskretisierungen höherer Ordnung läuft der Löser aber numerisch stabiler, wodurch seine Genauigkeit verbessert werden kann.



**Abbildung 3.3:** Profilingergebnisse des Löser für eine Methan/Luft Verbrennung, gemessen mit Cantera 2.0 auf JUQUEEN und dem VBT-Cluster.

### 3.2.3 Profiling

Für den ursprünglichen DNS Löser wurde ein ausführliches Profiling<sup>2</sup> durchgeführt, um Leistungsengpässe aufzuspüren. Dazu wurde eine speziell modifizierte Version des Löser erstellt, in der die Rechenzeiten für die einzelnen Löserabschnitte protokolliert werden. In Abbildung 3.3 ist der Anteil an der Gesamtlaufzeit der verschiedenen Programmteile für die Verbrennung von Methan in Luft dargestellt. Der DNS Löser verbringt ca. 70 % der Rechenzeit mit der Lösung der Reaktionskinetik und weitere 10 % mit der Berechnung der Transportgrößen. Die größte Arbeit in diesen beiden Abschnitten wird von Cantera übernommen. Daher liegt der Schwerpunkt der Optimierungen in dieser Arbeit in dem Codeteil, der von Cantera gestellt wird. Verwendet man den weniger komplexen Reaktionsmechanismus einer Wasserstoff/Luft Verbrennung nach Li et al. (B.1), reduziert sich der Anteil von **chemistry.H** an der Gesamtzeit auf ca. 60 %.

Nach diesem groben Überblick über den Anteil der Löserabschnitte an der Gesamtzeit, wurde mit dem Tool Callgrind des Programmes Valgrind [36] untersucht, welchen Anteil an der Gesamtlaufzeit einzelne Programmfunktionen haben. Als Beispiel

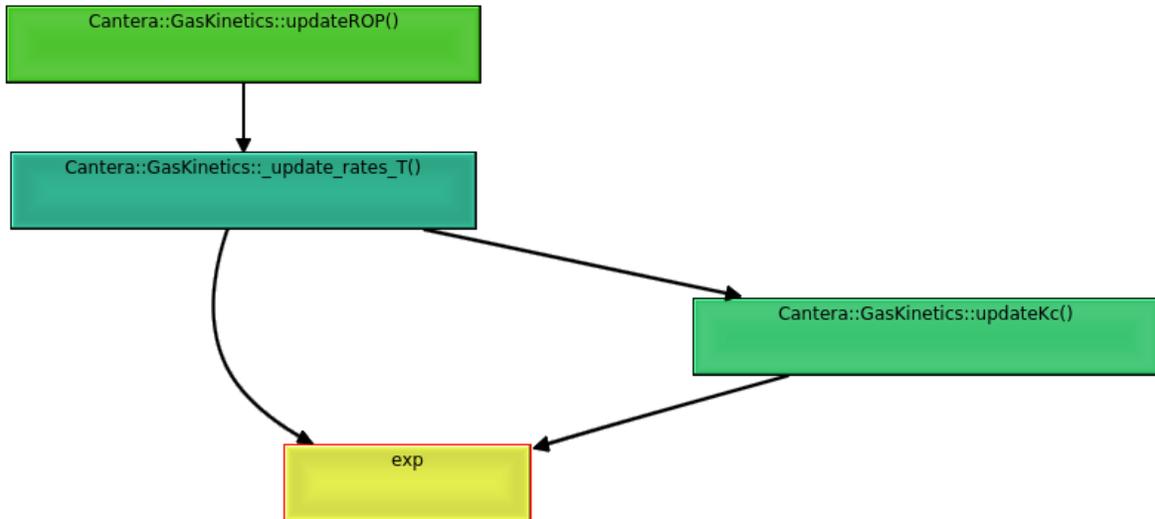
<sup>2</sup>Analyse des Laufzeitverhaltens eines Programms. In dieser Arbeit die Untersuchung, welche Programmteile die meiste Rechenzeit benötigen.

Incl.	Self	Called	Function	Location
100.00	0.00	(0)	0x0000000000000b00	ld-2.11.2.so
100.00	0.00	1	0x0000000000427a70	buoyantDiffReactingFoamOrigin...
100.00	0.00	1	(below main)	libc-2.11.2.so
100.00	0.11	1	main	buoyantDiffReactingFoamOriginal
75.40	0.03	98	Foam::canteraChemistryModel<Foam::psiChemistryModel, Foam::canteraTh...	libcanteraFoamModelStatic.so
75.12	0.00	12 250 000	Cantera::ReactorNet::advance(double)	libcanteraFoamModelStatic.so: ...
73.08	0.00	12 250 000	Cantera::CVodesIntegrator::integrate(double)	libcanteraFoamModelStatic.so: ...
73.08	0.05	12 250 000	CVode	libcanteraFoamModelStatic.so: ...
65.22	0.04	58 881 711	cvStep	libcanteraFoamModelStatic.so
63.85	0.01	59 006 711	cvNls	libcanteraFoamModelStatic.so
63.84	0.05	59 006 711	cvNlsNewton	libcanteraFoamModelStatic.so
51.05	0.03	53 832 533	cvDenseSetup	libcanteraFoamModelStatic.so
47.14	0.04	358 122 063	cvodes_rhs	libcanteraFoamModelStatic.so: ...
47.10	0.08	358 122 063	Cantera::ReactorNet::eval(double, double*, double*, double*)	libcanteraFoamModelStatic.so: ...
40.25	0.64	358 122 063	Cantera::Reactor::evalEqs(double, double*, double*, double*)	libcanteraFoamModelStatic.so: ...
38.83	0.03	358 122 063	Cantera::GasKinetics::getNetProductionRates(double*)	libcanteraFoamModelStatic.so: ...
34.67	2.98	358 122 063	Cantera::GasKinetics::updateROP()	libcanteraFoamModelStatic.so: ...
33.35	0.13	12 375 000	cvDlsDenseDQJac	libcanteraFoamModelStatic.so
26.94	2.63	358 122 063	Cantera::GasKinetics::_update_rates_T()	libcanteraFoamModelStatic.so: ...
17.35	17.35	49 420 853 244	exp	libm-2.11.2.so
16.10	2.49	358 122 063	Cantera::GasKinetics::updateKc()	libcanteraFoamModelStatic.so: ...
14.16	0.01	53 832 533	DenseGETRF	libcanteraFoamModelStatic.so
14.16	13.29	53 832 533	denseGETRF	libcanteraFoamModelStatic.so
9.00	1.26	3 842 885 582	Cantera::IdealGasPhase::_updateThermo() const	libcanteraFoamModelStatic.so: ...
7.28	1.50	1 182 947 022	Cantera::NasaThermo::update(double, double*, double*, double*) const	libcanteraFoamModelStatic.so: ...
7.26	0.53	384 092 063	Cantera::ThermoPhase::setState_HPporUV(double, double, double, bool)	libcanteraFoamModelStatic.so: ...
7.13	0.10	1 205 602 099	Cantera::IdealGasPhase::enthalpy_mole() const	libcanteraFoamModelStatic.so: ...
7.07	0.06	370 372 063	Cantera::ReactorNet::updateState(double*)	libcanteraFoamModelStatic.so: ...
7.01	0.30	370 372 063	Cantera::Reactor::updateState(double*)	libcanteraFoamModelStatic.so: ...
5.48	0.02	12 250 000	cvHin	libcanteraFoamModelStatic.so
5.32	5.32	21 293 046 396	Cantera::NasaPoly1::updateProperties(double const*, double*, double*, doub...	libcanteraFoamModelStatic.so: ...
5.29	0.17	246 960 000	Foam::canteraMixture<Foam::canteraThermo>::Hs(int, double) const	libcanteraFoamModelStatic.so
5.17	0.02	36 750 000	cvYddNorm	libcanteraFoamModelStatic.so
4.90	0.04	98	Foam::canteraHsPsiMixtureThermo<Foam::canteraMixture<Foam::canteraTh...	libcanteraFoamModelStatic.so
4.86	0.00	97	Foam::canteraHsPsiMixtureThermo<Foam::canteraMixture<Foam::canteraTh...	libcanteraFoamModelStatic.so
4.54	0.05	59 006 711	cvNewtonIteration	libcanteraFoamModelStatic.so
4.47	0.05	324 207	mpi_request_default_wait_all	libmpi.so.1.0.1
4.46	0.71	2 584 044 341	opal_progress	libmpi.so.1.0.1
4.46	0.00	63 503	Foam::UPstream::waitRequests(int)	libPstream.so
4.46	0.00	63 503	MPI_Waitall	libmpi.so.1.0.1
4.31	0.00	370 372 063	Cantera::ThermoPhase::setState_UV(double, double, double)	libcanteraFoamModelStatic.so: ...
4.13	4.13	358 122 063	Cantera::ReactionStoichMgr::getNetProductionRates(unsigned long, double c...	libcanteraFoamModelStatic.so: ...

Abbildung 3.4: Anteil an der Gesamtlauzeit einzelner Funktionen. Hervorgehoben sind die Funktionen `exp`, `_update_rates_T` und `Hs`, deren Optimierung im Folgenden beschrieben wird. Abbildung erstellt mit Valgrind [36], visualisiert durch KCachegrind [25].

dafür, wie Leistungsengpässe durch diese Art von Profiling aufgespürt werden können, dient Abbildung 3.4. Dargestellt sind die einzelnen Funktionen, die vom Löser benutzt werden. Der Funktionsname steht in der Spalte **Function**. Die linke Spalte (**Incl.**) zeigt, wie viel Prozent der Gesamtrechenzeit durch die jeweilige Funktion und alle weiteren, von dieser Funktion aufgerufenen Funktionen, eingenommen wird. In der Spalte **Called** ist aufgeführt, wie oft die Funktion im Löser aufgerufen wird.

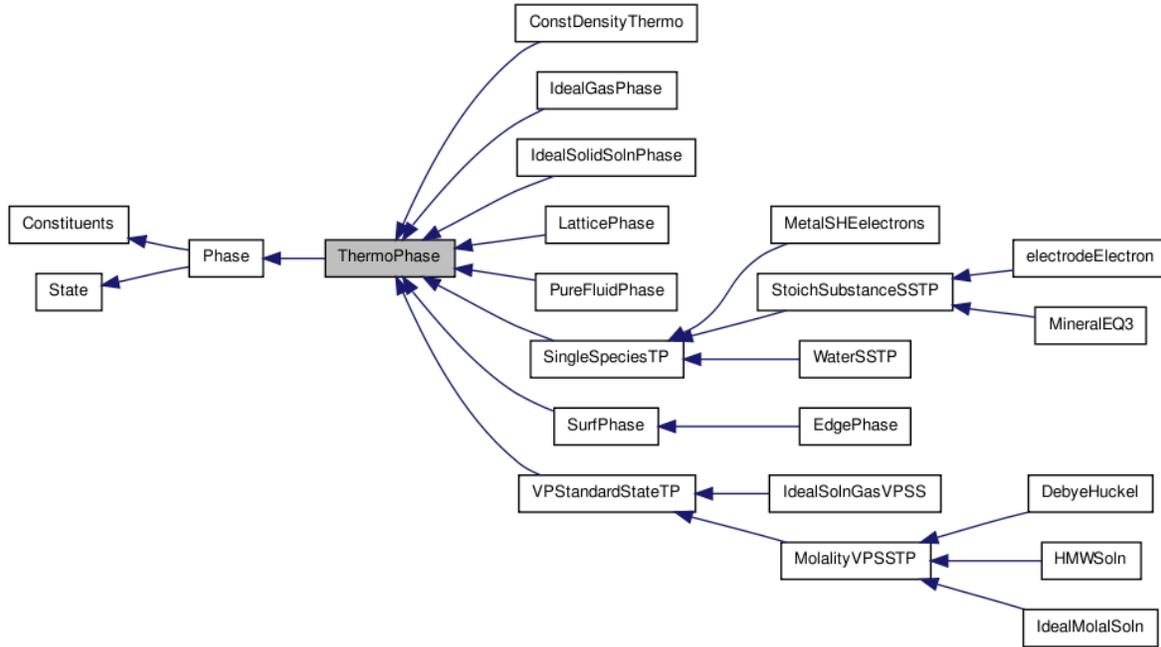
Auffällig ist, dass die Funktion `exp` mit am häufigsten aufgerufen wird und hier 17 % der Gesamtlauzeit ausmacht. Dabei handelt es sich um die Exponentialfunktion der



**Abbildung 3.5:** Ausschnitt aus dem Callstack für die Exponentialfunktion `exp`. Abbildung erstellt mit Valgrind und KCachegrind.

C++ Standardbibliothek. Um zu untersuchen, von welchen Funktionen die Exponentialfunktion verwendet wird, wurde der Callstack des Löser mit Valgrind analysiert. Der Callstack ist die Abfolge aller Funktionen, die von einem Programm aufgerufen werden. Der für die Exponentialfunktion relevante Ausschnitt aus dem Callstack ist in Abbildung 3.5 dargestellt. Die Exponentialfunktion wird sowohl von der Funktion `_update_rates_T` als auch der Funktion `updateKc` aufgerufen. `_update_rates_T` ist dabei die Funktion, die den temperaturabhängigen Teil der Geschwindigkeitskonstanten mit Hilfe des Arrheniusansatzes nach Gl. (2.31) berechnet. Sie nimmt insgesamt 27 % der Gesamtrechnzeit in Anspruch (Abbildung 3.4). `updateKc` aktualisiert die Gleichgewichtskonstanten und wird direkt von `_update_rates_T` aufgerufen.

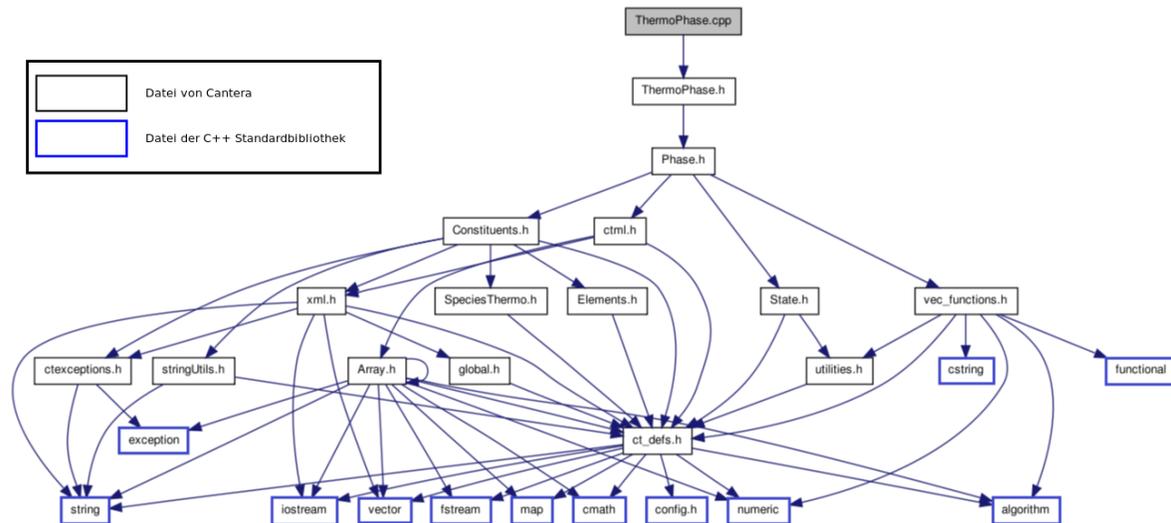
Da die Exponentialfunktion als Teil der C++ Standardbibliothek unveränderbar und schon stark optimiert ist, muss also die Funktion `_update_rates_T` optimiert werden, um den Anteil der Exponentialfunktion an der Gesamtlaufzeit zu reduzieren. Diese Optimierung ist in Unterabschnitt 3.4.2 beschrieben.



**Abbildung 3.6:** Übersicht über die Vererbungshierarchie der Cantera-Klasse **ThermoPhase**. Jedes Kästchen stellt eine Klasse aus Cantera dar [44].

### 3.3 Extraktion des relevanten Codes aus Cantera

Zur Verbesserung der Handhabung des DNS Löser und als Grundlage weiterer Optimierungen muss der für den Löser relevante Code aus Cantera extrahiert und nach OpenFOAM portiert werden. Wie in Unterabschnitt 3.1.2 bereits erwähnt, deckt Cantera eine Vielzahl chemischer Reaktionsbereiche ab und besteht aus 648 C++ Quellcode-dateien. Um sich einen Überblick über die Struktur von Cantera zu verschaffen, wurde analysiert, welche Abhängigkeiten zwischen den Klassen in Cantera bestehen. Cantera bietet dazu auf seiner Homepage eine automatisch generierte Visualisierung an. Als Beispiel wurde die Canteraklasse **ThermoPhase** gewählt. Sie stellt eine allgemeine Basisklasse für verschiedene Phasen im Reaktionsgebiet dar. Die Abhängigkeiten der Vererbungshierarchie der **ThermoPhase**-Klasse sind in Abbildung 3.6 dargestellt. Jedes Kästchen stellt dabei eine Klasse von Cantera dar. **ThermoPhase** selbst erbt von der Klasse **Phase**. Von **ThermoPhase** erben dann sämtliche von Cantera unterstützten Phasenmodelle wie metallische Phasen, Flüssigphasen und ideale Gasphasen.



**Abbildung 3.7:** Übersicht über die Abhängigkeiten der *ThermoPhase*-Quelldatei zu weiteren Dateien. [45].

Da der DNS Löser zur Simulation von Reaktionen in idealen Gasphasen verwendet werden soll, können also die meisten Klassen in dieser Vererbungshierarchie entfernt werden.

In einem nächsten Schritt wurden die Abhängigkeiten zwischen den Quellcodedateien untersucht. In Abbildung 3.7 wird die Quellcodedatei der Klasse **ThermoPhase** betrachtet. Dargestellt ist, welche weiteren Dateien von dieser Quellcodedatei benötigt werden. Jedes Kästchen enthält den Namen einer Datei. Dabei stellen Namen in Kästchen mit schwarzem Rand Dateien von Cantera dar, während Namen in Kästchen mit blauem Rand für Dateien aus der C++ Standardbibliothek stehen. Auch hier zeigt sich eine starke Vernetzung der Canteradateien untereinander.

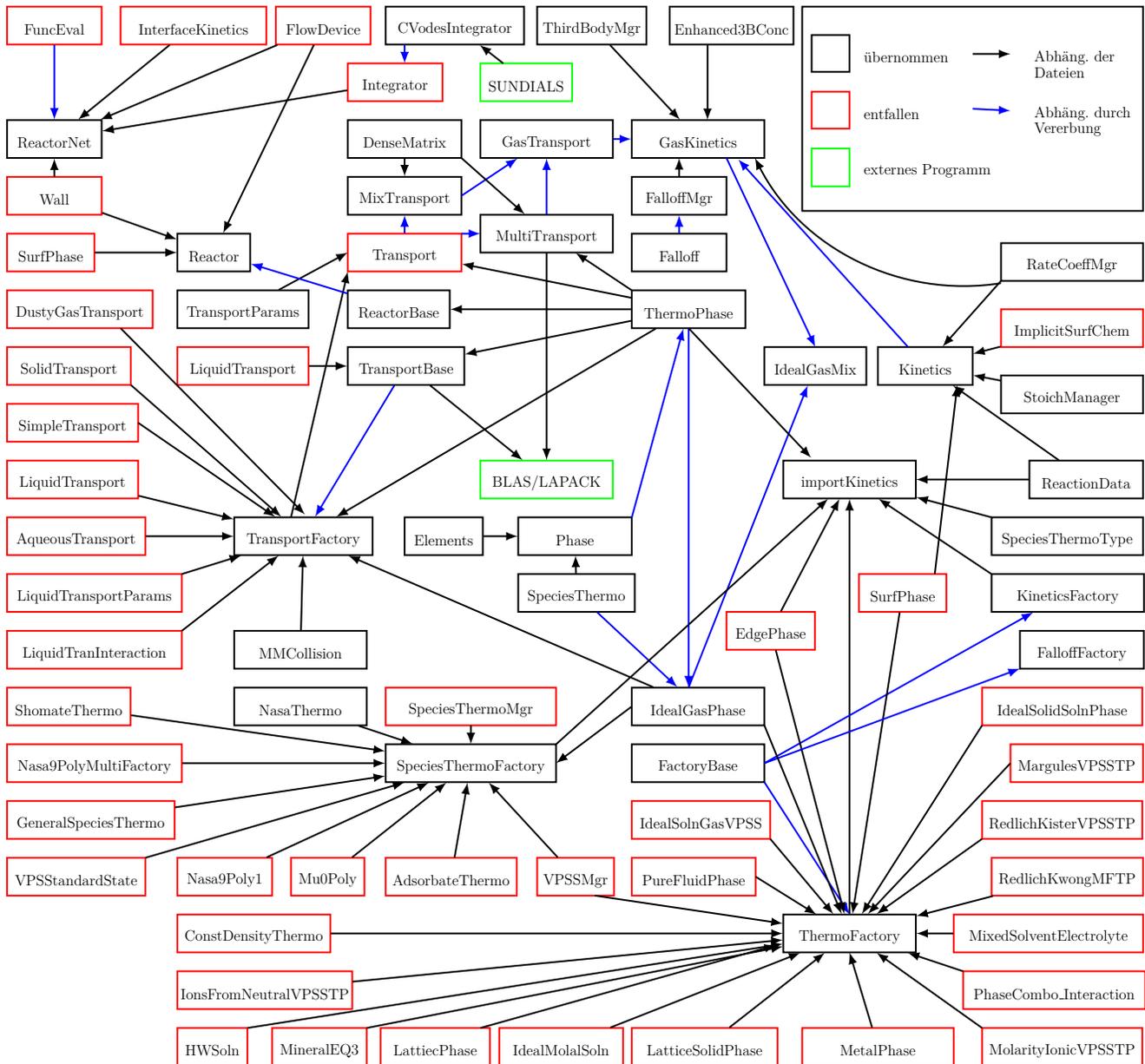
Um die relevanten Codeabschnitte aus Cantera direkt in den DNS Löser zu integrieren, müssen zunächst die im Löser benutzten Klassen und deren Quellcodedateien identifiziert werden. Diese müssen dann solange modifiziert werden, bis alle Abhängigkeiten zu nicht relevanten Teilen von Cantera - wie z.B. metallische Reaktionsphasen - entfernt worden sind und so ein alleine lauffähiges Programm entsteht. Nach diesem Schritt liegt ein Programm vor, das nur die Dateien aus Cantera enthält, die essentiell für die Funktionalität des DNS Löfers sind. Abbildung 3.8 verdeutlicht diesen Zustand

in vereinfachter Form. Jede Box stellt sowohl eine Klasse als auch deren Quellcodedatei aus Cantera dar. Schwarze Pfeile bedeuten, dass zwischen den Klassen und deren Dateien eine Abhängigkeit der Quellcodedateien besteht. Blaue Pfeile bedeuten, dass zwei Klassen zusätzlich in einer Klassen-Vererbungshierarchie liegen. Dies verdeutlicht noch einmal die starke Verknüpfung der Teilbereiche untereinander. Klassen von Cantera, die in Boxen mit schwarzem Rand stehen, sind direkt in den DNS Löser eingebaut worden. Boxen mit rotem Rand sind Klassen, die vom DNS Löser nicht verwendet werden. Zu ihnen wurden sämtliche Abhängigkeiten in den Quellcodeateien entfernt, die in den Löser integriert worden sind (Boxen mit schwarzem Rand). Dabei sind nur die Klassen aus Cantera dargestellt, die eine direkte Abhängigkeit zu den relevanten Klassen aufweisen.

Auch Klassen, die zum Parser gehören (siehe Unterabschnitt 3.1.2), sind in der Übersicht weggelassen. Neben ihnen gibt es noch viele weitere Klassen, die aus Gründen der Übersicht nicht aufgeführt sind. Boxen mit grünem Rand bedeuten, dass externe Programme verwendet werden; in diesem Fall SUNDIALS und BLAS/LAPACK.

Durch das Entfernen der nicht benötigten Cantera-Teile ist die Anzahl der C++ Quellcodedateien von 648 auf 76 zurückgegangen. Innerhalb dieser 76 Quellcodedateien liegt aber immer noch Code vor, der nicht verwendet wird. Um den Lösercodeumfang weiter zu reduzieren, wurden die Ergebnisse der Callstack Analyse benutzt (Unterabschnitt 3.2.3). Da im Callstack genau die Funktionen enthalten sind, die vom Löser benutzt werden, können alle Funktionen aus den Quellcodedateien entfernt werden, die nicht im Callstack vorhanden sind.

In einem letzten Schritt wurde das System zur Fehlerbehandlung entfernt. Dazu verwendet Cantera eine interne Logger-Klasse, die den Verlauf von aufgetretenen Fehlern verwaltet. Zusätzlich verwendet Cantera bei Fehlern, die das Programm in einen ungültigen Zustand versetzen, die Ausnahmen der C++ Standardbibliothek. Da OpenFOAM ein eigenes System zur Fehlerverwaltung bereitstellt, sind sowohl die Logger-Funktionalität sowie die Verwendung der Ausnahmen aus dem extrahierten Canteracode entfernt und mit Fehlermeldungen ersetzt worden, die an den DNS Löser angepasst sind. Dadurch wurde der Codeumfang weiter reduziert. Neben den Teilen von Cantera, die für die Berechnung der Reaktionskinetik und der Transportgrößen notwendig



**Abbildung 3.8:** Vereinfachte Übersicht über die Vernetzungen der verschiedenen Cante-rateile untereinander. Jede Box steht für eine Klasse und deren zugehörige Quellcodedatei. Klassennamen in Boxen mit rotem Rand bedeuten, dass sie in dieser Arbeit nicht mit in den DNS Löser übernommen worden sind. Namen in Boxen mit schwarzem Rand sind die Quell-codedateien, die in den neuen DNS Löser integriert wurden. Dargestellt sind nur Klassen, die eine direkte Abhängigkeit zu den für den Löser relevanten Klassen aufweisen.

sind, wurde auch der Parser in den Löser integriert, der die tabellierten Stoffwerte und Reaktionsdaten aus einer Datei liest.

## 3.4 Optimierung des extrahierten Cantera Codes

Durch den Ausbau der Codeteile aus Cantera, wie im letzten Abschnitt beschrieben, ist es möglich, den Quellcode direkt zu optimieren. Der Code von Cantera selbst ist schon stark optimiert. Daher ist ein ausführliches Profiling nötig, um Leistungsengpässe aufzuspüren. Nur durch das Spezialisieren des Cantera Codes für den DNS Löser ist es möglich, signifikante Leistungsverbesserungen zu erreichen; denn Cantera muss die Reaktionskinetik im Hinblick auf viele mögliche Reaktionsmodelle lösen können. Der DNS Löser hingegen wird nur für Verbrennung in idealen Gasphasen verwendet. Daher werden in Cantera Aufgaben durchgeführt, die für den DNS Löser nicht relevant sind. Ein weiterer Ansatzpunkt für die Verbesserung der Rechengeschwindigkeit ist die Interaktion zwischen Cantera und OpenFOAM. Im Folgenden werden die Maßnahmen beschrieben, die zu einer Leistungssteigerung des Programms geführt haben.

### 3.4.1 Spezialisierung des Codes

Der aus Cantera ausgebaute Code wurde an einigen Stellen für den DNS Löser optimiert, um die Rechengeschwindigkeit zu erhöhen. So wurde z.B. die Ausnahmebehandlung entfernt (Abschnitt 3.3). Das Vorhandensein von Ausnahmen zwingt den Compiler, zusätzliche Instruktionen in das Programm einzubauen, um auf die Auslösung einer Ausnahme reagieren zu können. Kommen diese zusätzlichen Instruktionen in Funktionen vor, die schon einen Leistungsengpass darstellen, wird ihr Einfluss spürbar. Zudem verhindern Ausnahmen spezielle Compiler Optimierungen wie die Return Value Optimization (RVO) [6]. Dabei kann der Compiler das Kopieren von Rückgabewerten von Funktionen vermeiden.

Cantera benutzt an vielen Stellen das „Factory pattern“ [16]. Dieses Entwurfsmuster stellt eine Programmieretechnik dar, bei der spezialisierte Objekte durch Hilfsklassen erzeugt werden. Die spezialisierten Objekte werden dann durch Basisobjekte, von denen sie abgeleitet sind, verwendet. So gibt es in Cantera die Basisklasse **ThermoPha-**

se, durch die spezialisierte Klassen wie **IdealGasPhase** oder **MetallicPhase** benutzt werden. Der Zugriff auf die spezialisierten Objekte geschieht also nur indirekt über die Basisobjekte. Da im DNS Löser nur ideale Gasphasen benutzt werden, wurde der Programmcode so geändert, dass direkt auf das Objekt **IdealGasPhase** zugegriffen wird. Dies hat zwei Vorteile: Zum einen muss das Programm nicht während der Laufzeit feststellen, welches Objekt sich hinter dem allgemeinen **ThermoPhase** Objekt verbirgt [31]. Zum anderen werden weitere Optimierungen durch den Compiler wie „function inlining“ ermöglicht. Beim „function inlining“ werden Funktionsaufrufe komplett entfernt und der Code der Funktion direkt in das Hauptprogramm integriert. Dadurch werden weitere Optimierungen wie eine verbesserte Sprungvorhersage<sup>3</sup> ermöglicht [23]. Auf diese Weise wurden an mehreren Stellen Indirektionen eliminiert. Zum Beispiel wird nun die Funktion zur Lösung der Reaktorgleichung für einen idealen Rührkessel (BR) direkt von SUNDIALS aufgerufen. Vorher war sie eine virtuelle Funktion in einer Klasse, die von einer allgemeinen Integratorklasse abgeleitet war und so Optimierungen wie „function inlining“ behindert hat [15]. Zudem wird nun als Reaktor immer ein idealer Rührkessel verwendet, anstatt von einem allgemeinen Reaktorobjekt auf ein Rührkessel-Objekt zuzugreifen.

Auch das Modell, wie Cantera die Reaktorgleichung auswertet, wurde für den DNS Löser optimiert. Cantera bietet die Möglichkeit, Reaktoren zu Reaktornetzen zusammenzusetzen und Randbedingungen wie Ein-/ und Auslässe zu definieren. Immer, wenn die Reaktorgleichung ausgewertet wird, muss über alle Reaktoren im Reaktornetz iteriert und sämtliche Randbedingungen ausgewertet werden. Für den DNS Löser stellt dies eine unnötige Arbeit dar, da jede Zelle des Simulationsgebietes als genau ein Rührkessel betrachtet wird. Daher sind das Reaktornetzsystem von Cantera und sämtliche dazugehörigen Randbedingungen entfernt worden. So entfällt der Mehraufwand für das Verwalten mehrerer Reaktoren und diverser Randbedingungen bei der Lösung der Reaktorgleichung.

---

<sup>3</sup>Sprungvorhersage ist ein Mechanismus der CPU, der die Wahrscheinlichkeit erhöht, dass bei Verzweigungen im Code die richtigen Instruktionen geladen werden.

### 3.4.2 Optimierung der Geschwindigkeitskonstantenberechnung

In Unterabschnitt 3.2.3 wurde festgestellt, dass die Exponentialfunktion im Löser sehr häufig aufgerufen wird und mit 17 % einen erheblichen Teil der Gesamtrechnzeit ausmacht. Zudem wurde gezeigt, dass die Funktion `_update_rates_T` zur Berechnung der Geschwindigkeits- und Gleichgewichtskonstanten einen ersten Ansatzpunkt für eine Optimierung darstellt. Da der dazugehörige Quellcode nach der Extraktion des Cantera-codes nun dem Löser beiliegt, kann dieser direkt eingesehen werden. Es zeigt sich, dass jedes Mal, wenn die Reaktionskinetik ausgewertet wird, sämtliche Geschwindigkeits- und Gleichgewichtskonstanten neu berechnet werden. Sie werden sogar dann neu berechnet, wenn die Temperatur sich seit der letzten Berechnung nicht geändert hat und ihre Werte somit konstant geblieben sind. Es zeigt sich außerdem, dass für die meisten Zellen im Reaktionsgebiet der Temperaturgradient sehr klein ist. Der Code wurde daraufhin so modifiziert, dass die Geschwindigkeits- und Gleichgewichtskonstanten nur dann neu berechnet werden, wenn die Temperatur sich seit der letzten Berechnung um mehr als  $10^{-5}$  K geändert hat. Durch diese Änderung konnte keine Abweichung der Simulationsergebnisse festgestellt werden. Erst, wenn diese Toleranz weiter erhöht wird, sind Abweichungen zu erkennen. Der Leistungsgewinn für den Löser durch diese Optimierung beträgt je nach Reaktionsmechanismus bis zu 15 %.

### 3.4.3 Optimierung der Enthalpieberechnung

Die Berechnung der thermischen Enthalpie für jede Spezies geschieht durch die Kopplungsbibliothek. Dabei werden die absoluten Enthalpien bei der aktuellen Temperatur berechnet und von diesen die Standardbildungsenthalpien abgezogen.

$$h_{k,s} = h_k - h_k^\circ \quad (2.53)$$

Obwohl die Standardbildungsenthalpien der reinen Spezies konstante Größen sind, werden sie in der ursprünglichen Version der Kopplungsbibliothek jedes Mal über die NASA-Polynome neu berechnet, wenn die thermischen Enthalpien angefordert werden. In der neuen Version der Kopplungsbibliothek hingegen wurde der Container `Hc.i` eingeführt. Beim Programmstart werden die Standardbildungsenthalpien der Spezies ein-

mal berechnet und in diesem Container gespeichert. Um die Standardbildungsenthalpie der Mischung zu berechnen, werden dann die vorberechneten Standardbildungsenthalpien mit den Speziesmassenbrüchen gewichtet. Der neue Code im Codeausschnitt 3.2 verdeutlicht dies (Code leicht vereinfacht).

<pre> 1 //Berechnung der Mischungsenthalpie bei T° 2 scalar canteraThermo::Hc() 3 { 4     return H(Tstd); 5     // jedes Mal, wenn die Funktion 6     // aufgerufen wird: 7     // berechne alle Speziesenthalpien 8     // über NASA-Polynome bei T=298.15 K 9     // neu und gewichte sie mit 10    // aktuellen den Massenbrüchen. 11 }</pre>	<pre> 1 //Berechnung der Mischungsenthalpie bei T° 2 scalar canteraThermo::Hc() 3 { 4     return gas().mean.Y(&amp;Hc_i[0]); 5     // Hc_i enthält die vorberechneten 6     // Standardbildungsenthalpien. 7     // Diese werden mit den aktuellen 8     // Massenbrüchen gewichtet. Es werden 9     // keine Enthalpien neu berechnet 10 11 }</pre>
--	--

**Codeausschnitt 3.1:** *Alte Funktion zur Berechnung der Standardbildungsenthalpie der Mischung.*

**Codeausschnitt 3.2:** *Optimierte Funktion zur Berechnung der Standardbildungsenthalpie der Mischung.*

Auch die Berechnung der Standardbildungsenthalpien und der thermischen Enthalpien der reinen Spezies, wie sie z.B. für die Energieerhaltungsgleichung in Gl. (2.94) gebraucht werden, ist im Löser mit unnötigem Mehraufwand verbunden. Denn in der ursprünglichen Kopplungsbibliothek gibt es nicht die Möglichkeit, direkt an die Speziesenthalpien zu gelangen. Stattdessen wird jedes Mal, wenn die Speziesenthalpien benötigt werden, eine neue Reaktionsphase erstellt. Bei dieser sind die Massenbrüche aller Spezies auf null gesetzt. Nur der Massenbruch der Spezies, deren Enthalpie bestimmt werden soll, wird auf eins gesetzt. Von dieser Phase wird dann die Mischungsenthalpie ermittelt, die nun der Enthalpie der reinen Spezies entspricht. Codeausschnitt 3.3 zeigt, wie eine temporäre Reaktionsphase zur Ermittlung der thermischen Enthalpie der Spezies  $k$  erstellt wird.

---

```

1  template<class ThermoType>
2  scalar canteraMixture<ThermoType>::Hs(label k, scalar T)
3  {
4      forAll(yTemp_, i)    // setze alle Speziesmassenbrüche der
5                          yTemp_[i]=0;    // neuen Phase auf null
6
7      yTemp_[k] = 1.0;    // nur der Massenbruch der k-ten Spezies ist eins
8
9      // erstelle die neue Phase mit der Zusammensetzung yTemp_
10     initializeThermo<ThermoType>::canteraData().setState(T,p,yTemp_);
11
12     // berechne die thermische Mischungsenthalpie der neuen Phase,
13     // die der thermischen Enthalpie der Spezies k entspricht
14     return initializeThermo<ThermoType>::canteraData().Hs(T);
15 }

```

---

**Codeausschnitt 3.3:** *Ursprüngliche Funktion zur Berechnung der thermischen Speziesenthalpie.*

Auch die Standardbildungsenthalpien der reinen Spezies werden über eine temporäre Reaktionsphase berechnet. Die Funktion **setState** übergibt die aktuelle Temperatur und den aktuellen Druck, sowie die temporäre Zusammensetzung an die neue Reaktionsphase. Bevor die neuen Massenbrüche, die als Parameter an **setState** übergeben werden, für die neue Reaktionsphase übernommen werden, werden sie erst normiert:

$$Y_k = \frac{Y_k^{\text{Parameter}}}{\sum_{i=1}^N Y_i^{\text{Parameter}}} \quad (3.7)$$

In diesem Falle ist es jedoch unnötig, die Massenbrüche zu normieren. Da alle Massenbrüche den Wert null haben und ein Massenbruch explizit auf 1 gesetzt wird, ist garantiert, dass ihre Summe eins ergibt. In der optimierten Version der Kopplungsbibliothek sind daher zwei neue Funktionen eingeführt worden, um die thermischen Enthalpien und die Standardbildungsenthalpien der reinen Spezies zu berechnen.

---

```

1 // neue Funktion zur Berechnung der thermischen Speziesenthalpien
2 scalar canteraThermo::Hs(scalar T, label i)
3 {
4     setTemperature(T); // Berechne die thermischen Enthalpien bei der Temperatur T
5     // thermische Enthalpie = absolute Enthalpie - Standardbildungsenthalpie
6     return gas().enthalpy_mass_i(i) - Hc(i);
7 }
8
9 // neue Funktion zur Berechnung der Standardbildungsenthalpien
10 scalar canteraThermo::Hc(label i)
11 {
12     return Hc.i[i]; // die Standardbildungsenthalpien können dem Container
13                    // Hc.i entnommen werden, der die vorberechneten Werte enthält
14 }

```

---

**Codeausschnitt 3.4:** *Neuen Funktionen zur Berechnung der thermischen Enthalpien und der Standardbildungsenthalpien der reinen Spezies.*

Die Funktion **enthalpy\_mass\_i** im Codeausschnitt 3.4 wurde dabei neu erstellt. Sie gibt direkten Zugriff auf die absoluten Speziesenthalpien, die über ein NASA-Polynom ermittelt werden. Die Speziesenthalpien können somit direkt berechnet werden und müssen nicht indirekt über eine temporäre Phase als Mischungsenthalpie ermittelt werden. Damit erfolgt die Berechnung der Speziesenthalpien in **preCalc.H** (Tabelle 3.2) nun 68 % schneller und die Lösung der Energieerhaltung in **hsEqn.H** 29 % schneller als im ursprünglichen Löser. Da **preCalc.H** und **hsEqn.H** zusammen nur 13 % der gesamten Rechenzeit ausmachen (Abbildung 3.3), liegt die Geschwindigkeitsverbesserung für den gesamten Löser damit bei ca. 4 %.

Die Rechenzeit konnte für den neuen DNS Löser durch die hier beschriebenen Maßnahmen deutlich reduziert werden: Die Optimierung der Geschwindigkeitskonstantenberechnung hat die Rechenzeit um 14 % bis 15 % gesenkt, die Optimierungen der Enthalpieberechnung um 4 %. Die Vereinfachungen des Canteracodes und das Entfernen unnötiger Berechnungen haben weitere 2 % Leistungssteigerung gebracht. Durch weitere, kleinere Optimierungen konnte die Gesamtrechenzeit um ca. 1 % verbessert werden. Insgesamt läuft der in dieser Arbeit optimierte Löser um etwa **20 %** schneller, als der ursprüngliche Löser (Abschnitt 3.2).

# Kapitel 4

## Berechnung der Reaktionsraten

Der in dieser Arbeit vorgestellte DNS Löser verbringt bis zu 70 % der Gesamtrechnungszeit mit der Bestimmung der Reaktionsraten. Der Grund dafür wird in Abschnitt 4.1 beschrieben. In Abschnitt 4.2 wird die Möglichkeit diskutiert, die Reaktionsraten ohne iterative Lösung der Reaktorgleichung direkt zu berechnen.

### 4.1 Mittelung der Reaktionsraten über den DNS Zeitschritt

Es gibt zwei relevante Zeitmaße für den DNS Löser: den DNS Zeitschritt für die Aufintegration der Strömungsgleichungen und den chemischen Zeitschritt für die Auflösung der Reaktionskinetik. Ein Kriterium für den DNS Zeitschritt ist das CFL-Kriterium [13] (Courant–Friedrichs–Lewy-Kriterium):

$$CFL = \frac{v \cdot \Delta t}{\Delta x} < 1 \quad (4.1)$$

Das CFL-Kriterium beschreibt, um wie viele Zellen pro Zeitschritt sich eine Größe maximal bewegen darf, damit die Lösung der Strömungsgleichungen numerisch stabil bleibt.  $v$  ist die Strömungsgeschwindigkeit,  $\Delta t$  der Löserzeitschritt und  $\Delta x$  der Zellenabstand, der für den vorliegenden Löser meist konstant ist. Das CFL-Kriterium lässt sich z.B. aus der Stabilitätsbedingung für das explizite Finite Differenzenverfahren herleiten. Im

vorliegenden DNS Löser wird der Zeitschritt  $\Delta t$  für eine CFL-Zahl von 0.1 gewählt, um eine höhere numerische Genauigkeit zu erreichen.

Die Reaktionsraten, die als Quellterm in der Speziesmassenerhaltung und als Teil der Wärmefreisetzungsrate in der Enthalpiebilanz verwendet werden, werden im DNS Löser nicht direkt über die Reaktionskinetik berechnet (Unterabschnitt 2.4.4). Stattdessen wird die Reaktorgleichung für jeden DNS Zeitschritt iterativ gelöst und die Reaktionsraten somit mehrfach innerhalb des DNS Zeitschritts berechnet. Die Reaktionsraten werden schließlich über den gesamten Zeitschritt gemittelt. Bei der Lösung der Reaktorgleichung muss die Iterationsschrittweite klein genug gewählt werden, um die Änderungsrate für die Spezies mit der kleinsten Zeitskala mit ausreichender Genauigkeit auflösen zu können. Diese Iterationsschrittweite entspricht dem kleinsten chemischen Zeitmaß.

Die Reaktorgleichung, mit der die Reaktionsraten im optimierten DNS Löser berechnet werden, ist numerisch nur aufwendig zu lösen: zum einen ist das Differentialgleichungssystem nicht linear, da die Reaktionsraten exponentiell von der Temperatur abhängen. Zum anderen ist es sehr steif [17]: um eine ausreichende Stabilität bei der Lösung zu garantieren, muss die Iterationsschrittweite deutlich kleiner gewählt werden, als für die numerische Genauigkeit nötig wäre. Der Grund dafür ist, dass die chemischen Zeitskalen der Spezies um bis zu 10 Größenordnungen auseinander liegen [48]. In typischen chemischen Reaktionen ist der chemische Zeitschritt  $t_c$  deutlich kleiner als der DNS Zeitschritt  $\Delta t$ , der zur stabilen Lösung der Erhaltungsgleichungen benutzt werden kann. Daher muss die Reaktorgleichung innerhalb des DNS Zeitschritts mehrfach ausgewertet und iterativ gelöst werden, um eine mittlere Rate zu erhalten.

Die Mittelung der Reaktionsraten geschieht im DNS Löser auf folgende Weise: Jede Zelle im Simulationsgebiet wird als ein idealer Rührkessel betrachtet. Zuerst werden die Reaktionsraten aller Spezies für die aktuelle Temperatur und die aktuelle Mischungszusammensetzung bestimmt (Unterabschnitt 2.4.4). Mit den Reaktionsraten wird ein chemischer Zeitschritt gewählt, der klein genug ist, um auch die Spezies mit dem kleinsten chemischen Zeitmaß aufzulösen. Der chemische Zeitschritt ist gleichzeitig die Schrittweite für die Iteration zur Lösung der Reaktorgleichung. Mit den Reaktionsraten werden dann die Spezieskonzentrationen aktualisiert. Für die neue Mischungszusammensetzung

wird bei konstanter innerer Energie und konstantem Volumen die neue Temperatur bestimmt. Dabei wird die Beziehung

$$du = c_v dT \quad (4.2)$$

ausgenutzt. Die innere Energie  $u$  wird vom DNS Löser über die Fundamentalgleichung berechnet:

$$u = \frac{1}{M} \left( \sum_{k=1}^N X_k \tilde{h}_k - \bar{R}T \right) \quad (4.3)$$

Die isochore Wärmekapazität  $c_v$  wird mit der Mayer-Beziehung berechnet [41]:

$$c_v = \frac{1}{M} (\tilde{c}_p - \bar{R}) \quad (4.4)$$

Um die Temperaturänderung durch die Reaktion zu berücksichtigen, wird ähnlich wie in Gl. (2.87) eine Temperatur gesucht, bei der die innere Energie für die neue Mischungszusammensetzung gleich der inneren Energie zu Beginn des DNS Zeitschritts ist. Innerhalb des Iterationsschritts für die Lösung der Reaktorgleichung muss daher auch die Temperatur iterativ bestimmt werden. Die Iterationsvorschrift zur Änderung der Temperatur ergibt sich aus Gl. (4.2):

$$T_{n+1} = T_n + \frac{u_0 - u(T_n)}{c_v(T_n)} \quad (4.5)$$

$u_0$  die innere Energie zu Beginn des DNS Zeitschritts und  $T_n$  die Temperatur der aktuellen Iteration.  $T_{n+1}$  ist die Temperatur der nächsten Iteration. Die Iteration ist abgeschlossen, wenn gilt:

$$|T_{n+1} - T_n| \leq 10^{-7} \text{ K} \quad (4.6)$$

$T_{n+1}$  ist somit die Temperatur für den nächsten Iterationsschritt zur Lösung der Reaktorgleichung. Für den nächsten chemischen Zeitschritt werden dann wieder die Reaktionsraten bestimmt und die oben beschriebene Vorschrift zur Auswertung der Reaktorgleichung beginnt erneut. Dies geschieht so lange, bis der DNS Zeitschritt  $\Delta t$  erreicht ist. Nachdem die Reaktorgleichung gelöst ist, werden die neu berechneten Temperaturen verworfen. Nur die Konzentration der letzten Iteration wird verwendet, um die

Reaktionsraten über den gesamten DNS Zeitschritt zu mitteln:

$$\dot{r}_k = \frac{dm_k}{dt} \approx \frac{C_k(t + \Delta t) - C_k(t)}{\Delta t} M_k \quad (4.7)$$

Aufgrund der Nichtlinearität und der hohen Steifheit wird die Reaktorgleichung für Verbrennungsprobleme häufig von spezialisierten Softwarepaketen übernommen [26]. Im DNS Löser wird diese Aufgabe von SUNDIALS (Unterabschnitt 3.1.2) übernommen. SUNDIALS stellt hochoptimierte Routinen zur Verfügung, um gewöhnliche Differentialgleichungen der Form

$$\frac{\partial \phi}{\partial t} = f(\phi, t) \quad (4.8)$$

zu lösen [21].  $\phi$  ist in eine beliebige physikalische Größe. Die Reaktorgleichung eines idealen Rührkessels hat die Form:

$$\frac{dC_k}{dt} = \dot{\omega}_k(t) \quad (4.9)$$

Gelöst wird sie in integraler Form:

$$\int_{C_k(t_0)}^{C_k(t_0+\Delta t)} dC_k = \int_{t_0}^{t_0+\Delta t} \dot{\omega}_k(t) dt \quad (4.10)$$

Die molaren Konzentrationen  $C_k$  lassen sich dabei folgendermaßen berechnen:

$$C_k = X_k \frac{\rho}{M} \quad (4.11)$$

SUNDIALS selbst hat direkt nichts mit der Bestimmung chemischer Quellterme zu tun, sondern löst allgemeine Differentialgleichungen aus mathematischer Sicht. Um SUNDIALS für die Lösung der Reaktorgleichung einsetzen zu können, muss daher die Funktion  $f(\phi, t)$  aus Gl. (4.8) vom DNS Löser definiert und an SUNDIALS übergeben werden. Die Funktion  $f(\phi, t)$  setzt die Differentialgleichungen, die von SUNDIALS gelöst werden, in den Kontext der Reaktionskinetik.  $f(\phi, t)$  enthält im DNS Löser die Berechnungsvorschriften für die Reaktionsraten und der Temperatur, wie sie oben beschrieben wurden. Die Mittelung der Reaktionsraten über den DNS Zeitschritt erlaubt es, den Löserzeitschritt an den DNS Zeitschritt anzupassen, ohne Rücksicht auf den

chemischen Zeitschritt nehmen zu müssen. Da der DNS Zeitschritt deutlich über dem chemischen Zeitschritt liegt, kann dadurch die Rechenzeit stark reduziert werden.

Eine vereinfachte Übersicht darüber, wie die Reaktionsraten für den Löserzeitschritt gemittelt werden, ist in Abbildung 4.1 dargestellt.

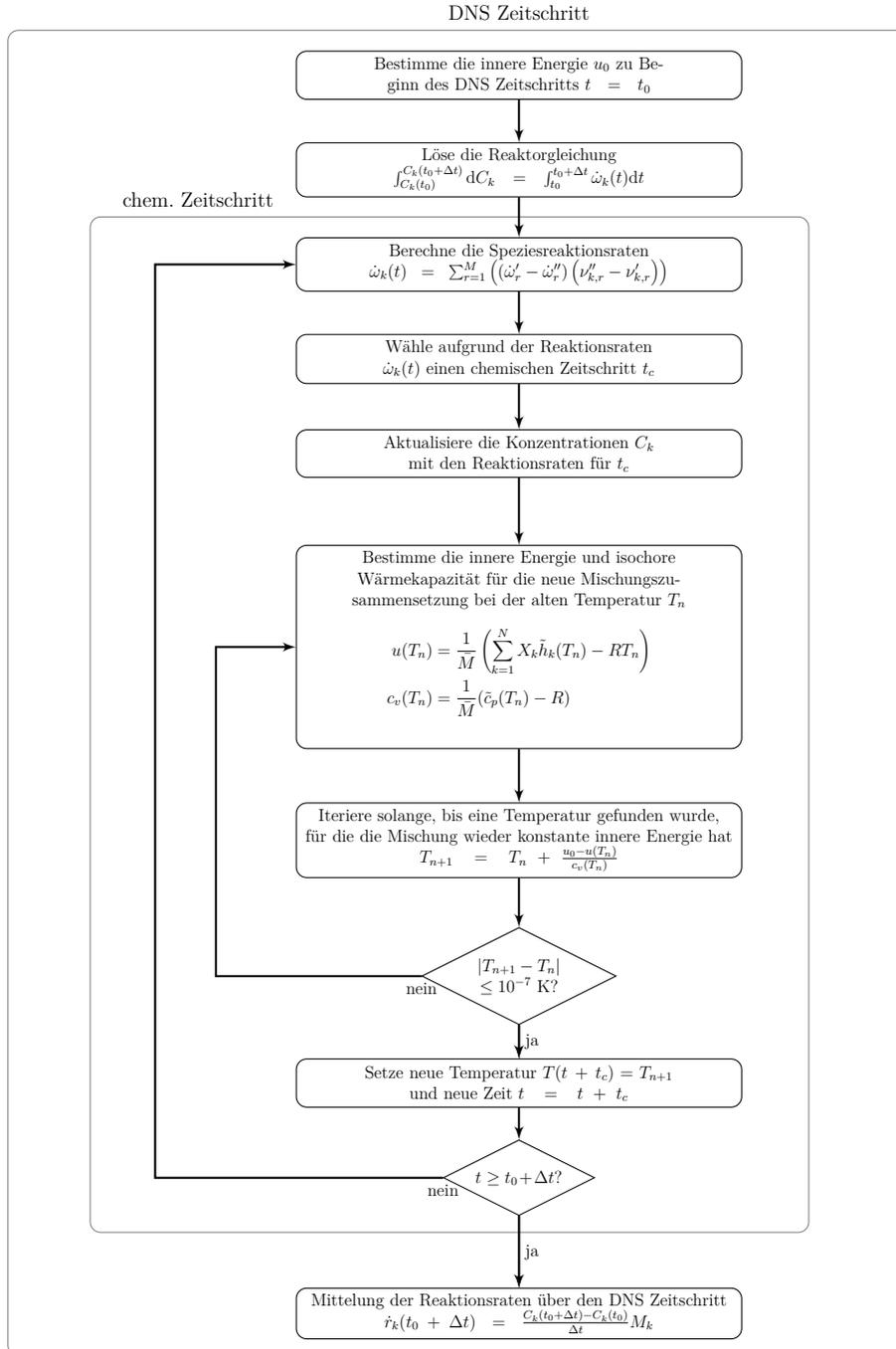
### 4.1.1 Interner Zeitschritt von SUNDIALS

Ein großer Vorteil von SUNDIALS ist, dass der chemische Zeitschritt, der intern von SUNDIALS für die Lösung der Reaktorgleichung gewählt wird, sehr flexibel ist. Er wird für jeden Iterationsschritt und jede Zelle im Rechenggebiet individuell angepasst. Zudem bietet SUNDIALS die Möglichkeit, die Lösung der Differentialgleichungen über Toleranzparameter zu steuern (Unterabschnitt 4.1.2).

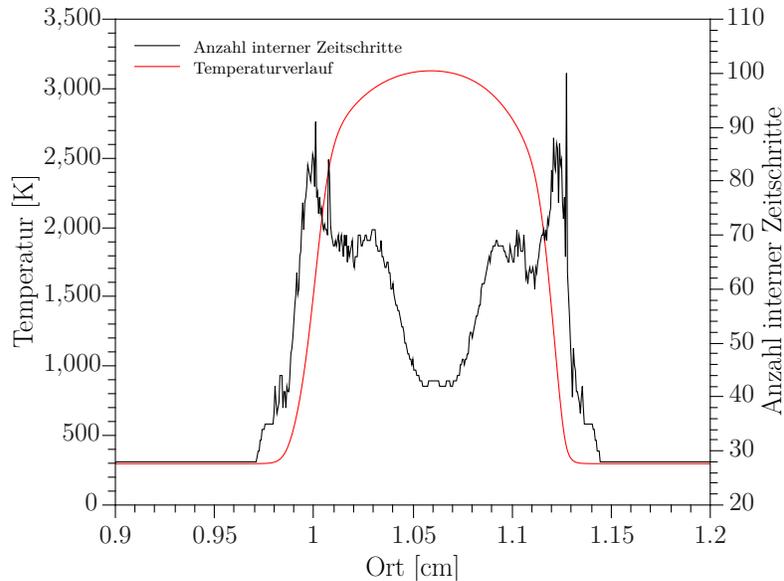
Der interne Zeitschritt ist in dieser Arbeit in Abhängigkeit des Reaktionsmechanismus und der absoluten Toleranz untersucht worden. Dazu wurde die Zündung einer stöchiometrischen 1D Methan/Luft Flamme mit 6000 Zellen und einem Zeitschritt von  $\Delta t = 5 \cdot 10^{-7}$  s durchgeführt. Mit einer speziell dafür modifizierten Version des Löser ist gezählt worden, wie oft SUNDIALS für jede Zelle innerhalb des Löserzeitschritts die Reaktionsraten auswertet. In Tabelle 4.1 ist der durchschnittliche chemische Zeitschritt für alle Zellen und der kleinste chemische Zeitschritt einer Zelle dargestellt. Der durchschnittliche chemische Zeitschritt, der von SUNDIALS ausgewählt wird, liegt bei  $1.7 \cdot 10^{-8}$  s für eine absolute Toleranz von  $10^{-12}$ . Sie liegen damit deutlich unter dem Zeitschritt des Löser von  $5 \cdot 10^{-7}$  s. Für Zellen, die in einem Gebiet mit großem Tem-

**Tabelle 4.1:** *Chemische Zeitschritte innerhalb des DNS Zeitschritts von  $5 \cdot 10^{-7}$  s.*

Absolute Toleranz	durchschn. chem. Zeitschritt [s]	kleinster chem. Zeitschritt[s]
$10^{-15}$	$1.22 \cdot 10^{-8}$	$1.37 \cdot 10^{-9}$
$10^{-13}$	$1.61 \cdot 10^{-8}$	$1.86 \cdot 10^{-9}$
$10^{-12}$	$1.72 \cdot 10^{-8}$	$2.81 \cdot 10^{-9}$
$10^{-11}$	$1.79 \cdot 10^{-8}$	$3.57 \cdot 10^{-9}$
$10^{-8}$	$1.85 \cdot 10^{-8}$	$7.04 \cdot 10^{-9}$
$10^{-4}$	$1.85 \cdot 10^{-8}$	$1.61 \cdot 10^{-8}$



**Abbildung 4.1:** Vereinfachte Übersicht über die Bestimmung der Reaktionsraten für den DNS Zeitschritt.



**Abbildung 4.2:** Verteilung der internen Zeitschritte und deren Korrelation mit dem Temperaturgradienten. Messung mit 6000 Zellen der Zündphase einer 1D Methan/Luft Flamme bei einem Zeitschritt von  $5 \cdot 10^{-7}$  s.

peraturgradienten liegen, wählt SUNDIALS einen noch kleineren internen Zeitschritt. Bei einer absoluten Toleranz von  $10^{-12}$  ergibt sich ein minimaler chemischer Zeitschritt von  $2.8 \cdot 10^{-9}$  s.

Werden die Messungen mit dem Mechanismus der Wasserstoff/Luft Verbrennung nach Maas und Pope für eine Flamme mit Luftüberschuss durchgeführt, zeigt sich, dass die internen Zeitschritte deutlich größer ausfallen. Der chemische Zeitschritt liegt im Schnitt bei  $2.8 \cdot 10^{-8}$  s, aber damit immer noch deutlich unter dem Löserzeitschritt. Innerhalb des Löserzeitschritts werden die Reaktionsraten für eine absolute Toleranz von  $10^{-12}$  im Durchschnitt nur 18 Mal pro Zelle berechnet, bei der Methan/Luft Verbrennung 29 Mal. Dies verdeutlicht, dass die chemischen Zeitskalen sich je nach Reaktionsbedingungen stark unterscheiden können.

Nur ein kleiner Teil der Zellen liegt über dem Durchschnittswert des chemischen Zeitschritts (Abbildung 4.2). Die internen Zeitschritte orientieren sich dabei näherungsweise am Gradienten der Temperatur in der Reaktionszone der Flamme. Dadurch wird ein weiterer Vorteil von SUNDIALS deutlich: Anstatt eines fixen Zeitschritts für alle

Zellen wird die Reaktionskinetik nur für die Zellen häufiger ausgewertet, die in Bereichen mit hohem Temperaturgradienten liegen. Je größer der Temperaturgradient ist, desto geringer ist der chemische Zeitschritt, der von SUNDIALS gewählt wird.

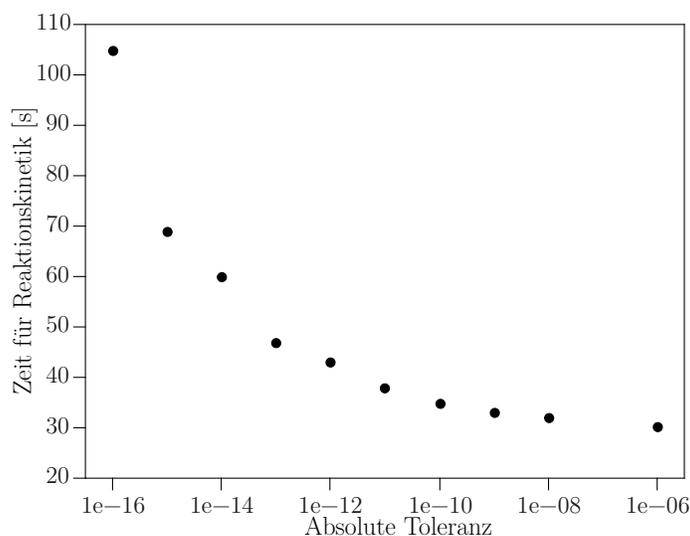
## 4.1.2 Einfluss der Toleranzparameter

SUNDIALS bietet die Möglichkeit, über zwei Parameter die Genauigkeit zu bestimmen, mit der die Reaktorgleichung gelöst wird: absolute und relative Toleranzen. Der Einfluss dieser Parameter auf Genauigkeit und Rechenzeit wird im Folgenden beschrieben.

### 4.1.2.1 Absolute Toleranz

Der wichtigste Parameter für die Lösung der Reaktorgleichung ist die absolute Toleranz. Ihr Einfluss auf die Rechenzeit ist in Abbildung 4.3 dargestellt.

Dabei hängt die Rechenzeit stark von der absoluten Toleranz ab. Der Einfluss dieses Parameters auf die Genauigkeit der Simulationsergebnisse ist in Tabelle 4.2 dargestellt.



**Abbildung 4.3:** Einfluss der absoluten Toleranzen auf die Rechenzeit, gemessen mit einer relativen Toleranz von  $10^{-4}$  und einem Zeitschritt von  $5 \cdot 10^{-7}$  s für ein Gitter mit 2 Millionen Zellen für eine Methan/Luft Flamme mit 12 Prozessorkernen auf dem VBT-Institutscluster (Unterabschnitt 3.1.3).

**Tabelle 4.2:** Maximaler Fehler in % der Reaktionsraten in Abhängigkeit der absoluten Toleranz bei einer relativen Toleranz von  $10^{-4}$  für eine stöchiometrische Methan/Luft Flamme. Als Referenzwert dient die absolute Toleranz  $10^{-15}$ .

Spezies	Reaktionsrate [kg/m <sup>3</sup> /s]	Fehler in % in Abhängigkeit absoluter Toleranzen						
		$10^{-13}$	$10^{-12}$	$10^{-11}$	$10^{-10}$	$10^{-8}$	$10^{-6}$	$10^{-4}$
H <sub>2</sub> O <sub>2</sub>	$4.9 \cdot 10^{-5}$	0.0016	0.0056	0.063	0.43	2.1	28.8	145
HO <sub>2</sub>	$5.9 \cdot 10^{-5}$	0.30	0.69	1.06	1.6	5.3	20.7	13.7
CH	$7.4 \cdot 10^{-7}$	0.40	0.43	0.51	0.63	6.77	10.4	65.2
OH*	$4.3 \cdot 10^{-10}$	0.84	0.90	113	7189	4350	563	205000

Der Fehler wurde dabei auf folgende Weise ermittelt: Bei einem 2 Millionen Zellen Gitter wurde der Durchschnittswert aller Zellen in einem Zeitschritt von allen Reaktionsraten ermittelt. Als Referenzwert dienen die Reaktionsraten, die bei einer absoluten Toleranz von  $10^{-15}$  ermittelt wurden. Die vier größten Abweichungen in Prozent der Reaktionsraten sind in Tabelle 4.2 dargestellt. Sie treten dort auf, wo die Werte der Reaktionsraten am kleinsten sind.

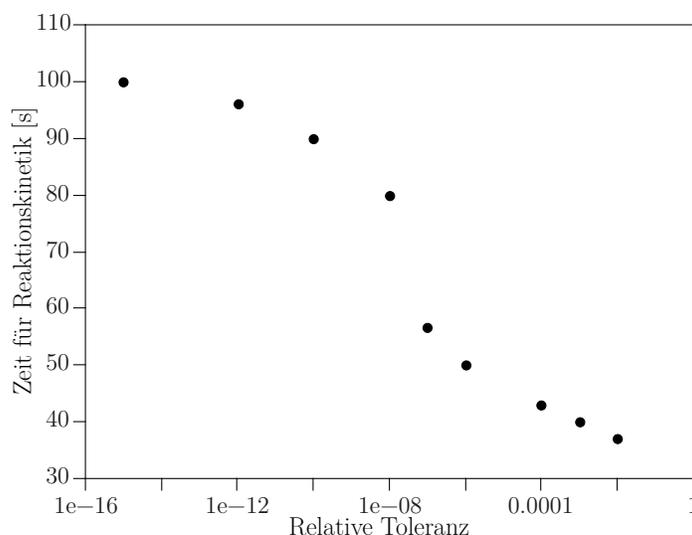
Bei einer absoluten Toleranz von  $10^{-12}$  ist der größte Fehler von allen Reaktionsraten kleiner als 1 %. Bei weiterer Verringerung der absoluten Toleranz steigt die Rechenzeit des Programms so stark, dass die Genauigkeitsverbesserung nicht gerechtfertigt ist. So steigt die Rechenzeit bei der Verringerung der absoluten Toleranz von  $10^{-12}$  auf  $10^{-13}$  um 10 %, der Fehler verringert sich aber nur um maximal 0.3 % (Abbildung 4.3 und Tabelle 4.2).

Auf der anderen Seite zeigt sich, dass bei einer absoluten Toleranz von  $10^{-11}$  die Abweichung der Reaktionsrate von angeregten Hydroxyl-Radikalen (OH\*) bei 113 % liegt. Die Reaktionsrate der OH\*-Speziesmoleküle ist im Vergleich zu den anderen Spezies sehr klein. Sie hat eine Größenordnung von  $10^{-10}$  und liegt damit fast im Bereich der absoluten Toleranz, wodurch der hohe Fehler zu erklären ist. Senkt man die absolute Toleranz weiter, so stellt man fest, dass Fehler von über 10 % erst dann auftauchen, wenn die absolute Toleranz ca. eine Größenordnung unter dem Absolutwert der Reaktionsrate liegt. Die absolute Toleranz sollte damit immer mindestens zwei Größenordnungen unter dem Wert der kleinsten Reaktionsrate liegen. In der Praxis hat sich eine absolute Toleranz von  $10^{-12}$  als bester Kompromiss aus Genauigkeit und Rechenzeit

herausgestellt.

#### 4.1.2.2 Relative Toleranz

Die relative Toleranz hat wenig Einfluss auf die Genauigkeit, aber einen großen Einfluss auf die Rechenzeit (Abbildung 4.4). So dauert die Lösung der Reaktionskinetik bei einer absoluten Toleranz von  $10^{-12}$  und einer relativen Toleranz von  $10^{-4}$  43 s. Bei gleicher absoluter Toleranz, aber einer relativen Toleranz von  $10^{-8}$ , dauert die Lösung 80 s. Da der maximal beobachtbare Fehler zwischen den beiden Einstellungen kleiner als 0.4 % ist, ist die Verdopplung der Laufzeit also nicht gerechtfertigt. Eine relative Toleranz von  $10^{-3}$  verringert die Laufzeit nur auf 40 Sekunden. Der maximal beobachtete Fehler steigt aber auf 2.5 % und tritt wieder bei der Reaktionsrate der angeregten Hydroxyl-Radikale auf, die den kleinsten Absolutwert aller Reaktionsraten haben. Der Fehler zu den Referenzwerten bleibt bei einer relativen Toleranz von  $10^{-4}$  immer kleiner als 1 %. Daher wird dieser Wert für weitere Simulation verwendet.

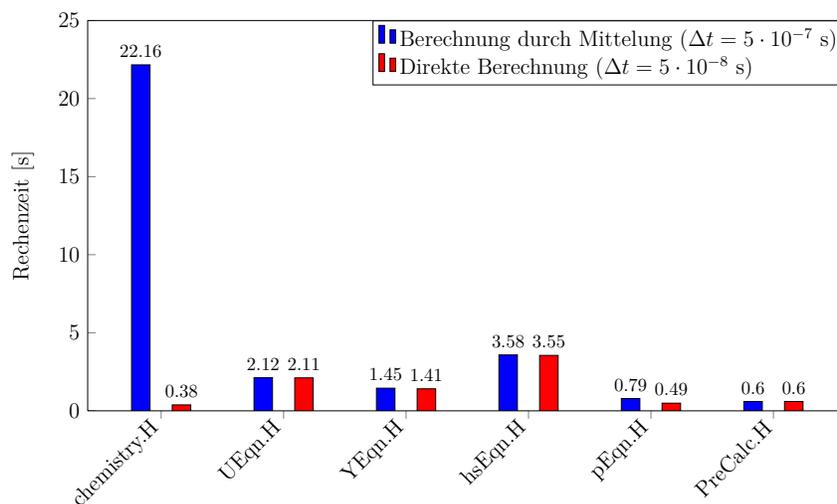


**Abbildung 4.4:** Einfluss der relativen Toleranzen auf die Rechenzeit. Simulationsbedingungen wie in Abbildung 4.3.

## 4.2 Direkte Berechnung der Reaktionsraten

Um den Einfluss der Reaktionskinetik auf die Rechenzeit weiter zu untersuchen, wurde eine neue Löservariante erstellt, bei der die Reaktionsraten direkt und damit ohne iterative Lösung der Reaktorgleichung ermittelt werden. Dabei kann sich der Löserzeitschritt nicht an den DNS Zeitschritt richten, sondern muss an das kleinste chemische Zeitmaß angepasst werden, um eine ausreichende Stabilität bei der Verbrennung zu garantieren. Die Abhängigkeit des Löserzeitschritts von den chemischen Zeitskalen führt dazu, dass die Simulation schnell instabil wird, wenn große Temperaturgradienten auftauchen. Messungen mit einem 125 Millionen Zellen Gitter für eine Wasserstoff/Luft Flamme mit Luftüberschuss haben gezeigt, dass der Löser bei einem Zeitschritt von über  $1 \cdot 10^{-7}$  s instabil ist. Bei einer stöchiometrischen Methan/Luft Verbrennung läuft der Löser sogar bis zu einem Zeitschritt von  $7 \cdot 10^{-8}$  s noch instabil. Werden die Reaktionsraten hingegen über den gesamten Löserzeitschritt mit SUNDIALS gemittelt (Kapitel 4), kann ein Zeitschritt von  $1 \cdot 10^{-6}$  s gewählt werden, ohne dass die DNS Rechnung instabil wird.

Abbildung 4.5 zeigt die Auswirkung der direkten Berechnung der Reaktionsraten



**Abbildung 4.5:** Vergleich der Rechenzeiten für direkte Berechnung der Reaktionsraten und Mittelung über den DNS Zeitschritt. Rechnung für eine Methan/Luft Flamme auf JUQUEEN (Unterabschnitt 3.1.3) mit einem 6.75 Millionen Zellen Gitter mit 1024 CPU-Kernen.

auf die Rechenzeit. Da die Reaktionskinetik hier direkt gelöst wird und die Reaktionsraten somit genau einmal pro Zeitschritt bestimmt werden, nimmt die Rechenzeit für den Abschnitt **chemistry.H** von 22.2 s auf 0.4 s ab und nimmt anstatt 70 % nur noch 5 % der Gesamtrechenzeit ein. Außer bei der Druckkorrekturberechnung in **pEqn.H** ist die Rechenzeit für die anderen Löserabschnitte nur wenig abhängig vom gewählten Zeitschritt  $\Delta t$ . Die Rechenzeit für den gesamten Zeitschritt reduziert sich von 31 s auf 8.5 s. Obwohl die Rechenzeit für die Reaktionskinetik um mehr als Faktor 50 abgenommen hat, hat sich die Rechenzeit für den Löser als Ganzes also nur um Faktor 4 verringert. Damit die Simulation bei der direkten Berechnung der Reaktionsraten stabil läuft, muss der Löserzeitschritt von  $1 \cdot 10^{-6}$  s auf mindestens  $1 \cdot 10^{-7}$  s gesenkt werden. Obwohl jeder einzelne Zeitschritt viermal schneller berechnet wird als bei der Mittelung der Reaktionsraten, müssen zehnmal mehr Zeitschritte berechnet werden. Dabei ist  $1 \cdot 10^{-7}$  s der Maximalwert für numerische Stabilität. Um genauere Werte zu errechnen, müsste der Zeitschritt weiter reduziert werden; der durchschnittliche, chemische Zeitschritt liegt nämlich bei  $5 \cdot 10^{-8}$  s (Unterabschnitt 4.1.1). Dadurch läuft der Löser bei direkter Berechnung der Reaktionsraten insgesamt deutlich langsamer als bei der Mittelung über den Löserzeitschritt. Denn der geschwindigkeitsbestimmende Schritt bei der direkten Berechnung ist nun nicht mehr die Lösung der Reaktionskinetik, sondern die Erhaltungsgleichungen, deren Lösungszeit nur wenig abhängig vom gewählten Zeitschritt ist.

Abschließend lässt sich feststellen, dass die Mittelung der Reaktionsrate über den DNS Zeitschritt für den Löser sehr sinnvoll ist. Die Stabilität des Löser steigt und der Löserzeitschritt kann an den DNS Zeitschritt angepasst werden, wodurch die Rechengeschwindigkeit stark steigt. Die Verwendung von SUNDIALS als Integrator bietet ein hohes Maß an Flexibilität: Obwohl der chemische Zeitschritt abhängig von den Reaktionsbedingungen ist, muss er im Löser nicht explizit angegeben werden, sondern wird für jede Zelle und jeden Zeitschritt individuell angepasst. Dadurch wird die Reaktorgleichung nur für die Zellen häufiger ausgewertet, die in Bereichen mit hohem Temperaturgradienten liegen.

# Kapitel 5

## DNS vorgemischter Wasserstoff/Luft Flammen

Um die Allgemeingültigkeit des Lösers zu zeigen, soll die Validierung des weiterentwickelten Lösers mit einer Wasserstoff/Luft Verbrennung geschehen. In der früheren Arbeit von Bonart [4] wurde der ursprüngliche Löser mit einer Methan/Luft Verbrennung validiert. Ein weiterer Unterschied ist, dass bei der Validierung des ursprünglichen Lösers der Molekulartransport über einen Multikomponenten-Ansatz berechnet wurde (Kapitel 2); im weiterentwickelten Löser werden Mischungsgemittelte Transportkoeffizienten zur Berechnung des Molekulartransports verwendet. Der Grund dafür ist, dass die Berechnung von Transportkoeffizienten mit dem Multikomponenten-Ansatz numerisch sehr aufwendig ist und im Allgemeinen daher Mischungsgemittelte Größen genutzt werden [26].

Zur Validierung des in dieser Arbeit neu implementierten Lösers sind zwei Simulationen durchgeführt worden. Um die Reaktionskinetik und den Molekulartransport zu validieren, wurden die Ergebnisse einer vorgemischten 1D Flamme mit den Ergebnissen aus CHEMKIN verglichen. Zudem wurde der Einfluss des gewählten Reaktionsmechanismus auf die Rechengeschwindigkeit untersucht. In einer zweiten Simulation wurde ein 3D Fall berechnet, um die Anwendbarkeit des Lösers für komplexe 3D Gitter zu zeigen. Der berechnete Fall ist dabei experimentell am Engler-Bunte-Institut untersucht worden [2], sodass die Ergebnisse des Experiments zur weiteren Validierung herange-

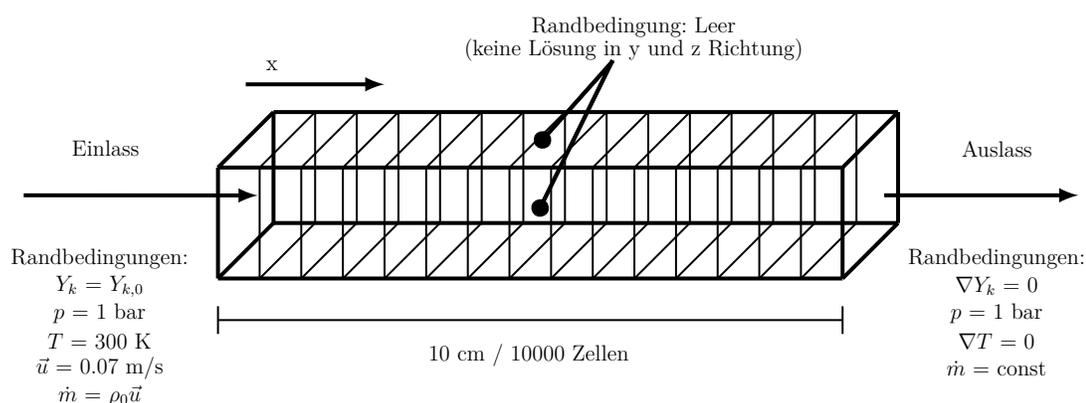
zogen werden können. Darüber hinaus wurde der Einfluss der Gitterauflösung auf die DNS Rechnung untersucht.

## 5.1 Flachflamme (1D)

Um die Berechnung der Reaktionsraten und temperatur- und druckabhängigen Stoffgrößen des optimierten Löser zu validieren, wurden die Simulationsergebnisse einer vorge-mischten Flachflamme mit den Ergebnissen des kommerziellen Programms CHEMKIN verglichen. Dazu wurde eine Wasserstoff/Luft Verbrennung mit dem Reaktionsmecha-nismus nach Li et al. (B.1) durchgeführt.

### 5.1.1 Numerische Einstellungen

Das Simulationsgebiet ist eine 1D Strecke von 10 cm, an deren Anfangspunkt ( $x = 0$ ) sich ein Gaseinlass für Frischgas befindet. Das Frischgas strömt mit einer Geschwin-digkeit von 0.07 m/s und einer Temperatur von 300 K bei einem Druck von 1 bar in das Simulationsgebiet. Am rechten Rand ( $x = 10 \text{ cm}$ ) befindet sich der Auslass. Dieser Aufbau und die jeweiligen Randbedingungen sind in Abbildung 5.1 abgebildet. Das Rechengitter besteht aus 10000 äquidistanten Zellen. Der Abstand zweier Zellen beträgt



**Abbildung 5.1:** Randbedingungen für die Simulation der 1D Wasserstoff/Luft Flamme.

somit  $\Delta x = 0.01$  mm. Obwohl das Simulationsgebiet dreidimensional abgebildet ist, ist für die 1D Simulation nur die Länge in x-Richtung relevant. Am Einlass werden feste Werte vorgegeben, am Auslass herrscht ein Null-Gradient für die Speziesmassenbrüche und die Temperatur. Die Geschwindigkeit wird so eingestellt, dass sich ein konstanter Massenfluss  $\dot{m}$  ergibt. Der Druck wird konstant auf 1 bar gehalten. Die Simulation wurde mit dem in dieser Arbeit optimierten DNS Löser durchgeführt.

Um die Zusammensetzung des Frischgases zu charakterisieren, wird die Luftzahl  $\lambda$  verwendet:

$$\lambda = \frac{\frac{X_{Luft}}{X_{Brennstoff}}}{\frac{X_{Luft,stöch}}{X_{Brennstoff,stöch}}} \quad (5.1)$$

Im Frischgas herrscht ein Luftüberschuss bei  $\lambda = 3$  und einer Luftzusammensetzung von  $Y_{O_2} = 0.233$  und  $Y_{N_2} = 0.767$ . Eine Übersicht über die Randbedingungen für das Frischgas ist in Tabelle 5.1 dargestellt. Als Startwerte wurden die Ergebnisse von CHEMKIN für die stationäre Flamme übernommen. Von dieser Lösung aus wurden 0.05 s mit einem Zeitschritt von  $5 \cdot 10^{-7}$  s, also 100000 Zeitschritte, weitergerechnet und schließlich wieder mit CHEMKIN verglichen.

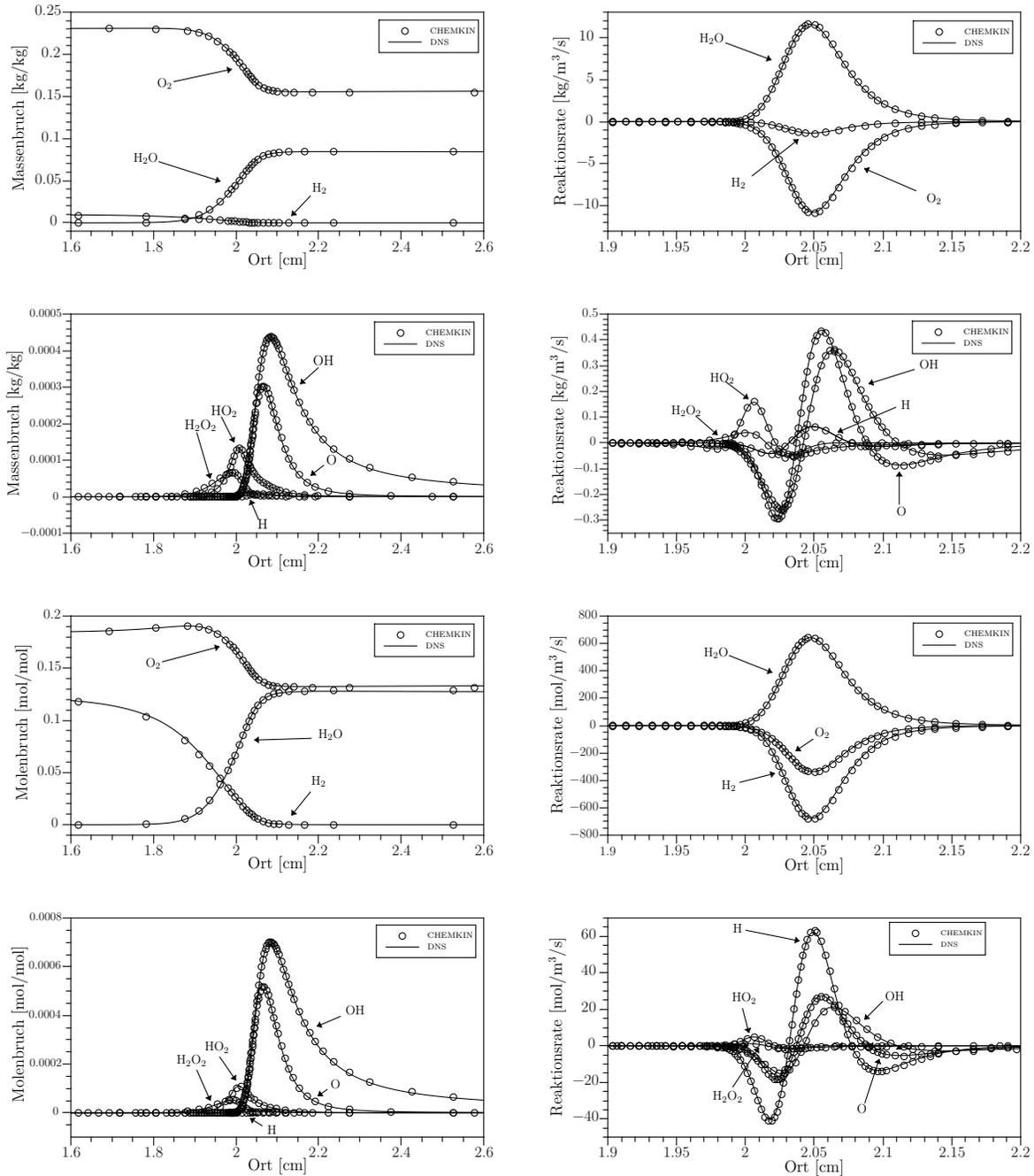
**Tabelle 5.1:** *Randbedingungen für das Frischgas am Einlass des Rechengitters.*

$p_0$	$T_0$	$\lambda_0$	$Y_{H_2,0}$	$Y_{O_2,0}$	$Y_{N_2,0}$	$\vec{u}_0$
1 bar	300 K	3	0.0097	0.2307	0.7596	0.07 m/s

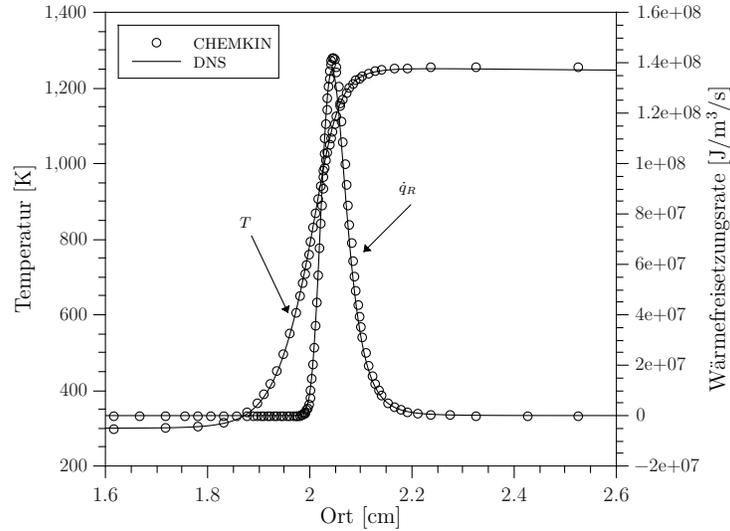
### 5.1.2 Vergleich der Ergebnisse mit CHEMKIN

In Abbildung 5.2 werden die berechneten Spezieskonzentrationen und deren Reaktionsraten aus der DNS mit den Werten aus CHEMKIN verglichen. Oben sind die Massenbrüche und Reaktionsraten dargestellt; unten die Molenbrüche und molaren Reaktionsraten, jeweils für die Haupt- und Nebenspezies. Der Vergleich zeigt eine sehr gute Übereinstimmung.

Beim Verlauf der Hauptspeziesmolenbrüche zeigt sich ein Effekt von bevorzugter Diffusion: Da Wasserstoff einen sehr hohen Diffusionskoeffizienten hat, diffundiert es



**Abbildung 5.2:** Vergleich der Flammenprofile mit CHEMKIN. *Oben:* Massenbrüche und Reaktionsraten. *Unten:* Molenbrüche und molare Reaktionsraten.



**Abbildung 5.3:** Vergleich der Temperatur und der Wärmefreisetzungsrates mit CHEMKIN.

mit höherer Geschwindigkeit in die Flammenfront, als andere Spezies. Bevor die Flammenfront erreicht ist, nimmt daher der Molenbruch von Wasserstoff ab. Gleichzeitig erhöht sich dadurch der Molanteil von Sauerstoff. Auch das Temperaturprofil und die Wärmefreisetzungsrates zeigen eine gute Übereinstimmung mit den CHEMKIN Ergebnissen: Die Temperatur des verbrannten Gases beträgt im neuen DNS Löser 1257.0 K, in CHEMKIN ergibt sich ein Wert von 1256.8 K. Die maximale Wärmefreisetzungsrates ist im DNS Löser  $1.37 \cdot 10^8 \text{ W/m}^3$  und in CHEMKIN  $1.42 \cdot 10^8 \text{ W/m}^3$  (Abbildung 5.3). Ein Grund für diese Abweichung ist die Gittergröße. Die Zellen des äquidistanten Gitters des DNS Lösers haben einen Abstand von  $\Delta x = 0.01 \text{ mm}$ . Das adaptive Gitter von CHEMKIN weist an der dichtesten Stelle einen Abstand von  $\Delta x = 0.008 \text{ mm}$  auf. Ein weiterer Unterschied ist, dass CHEMKIN zur Lösung der Erhaltungsgleichungen nicht die Finite Volumen Methode verwendet, sondern die Finite Differenzen Methode.

Die Flammengeschwindigkeit  $s_L$  lässt sich für eine stehende Flamme aus der Massenbilanz mit der Kontinuitätsgleichung ableiten:

$$\dot{m} = \rho_0 s_L \quad (5.2)$$

$\rho_0$  ist die Dichte des unverbrannten Gases. Der Massenfluss des Brennstoffs  $\dot{m}_F$  kann

zusätzlich über folgende Beziehung dargestellt werden:

$$\dot{m}_F = \rho_0 s_L Y_{F,0} = \int_{-\infty}^{\infty} \dot{r}_F dx \quad (5.3)$$

$\dot{r}_F$  ist dabei die Reaktionsrate des Brennstoffs,  $Y_{F,0}$  der Massenanteil des Brennstoffs im Frischgas. Mit Gl. (5.2) und Gl. (5.3) lässt sich die Flammengeschwindigkeit folgendermaßen darstellen [41]:

$$s_L = -\frac{1}{\rho_0 Y_{F,0}} \int_{-\infty}^{\infty} \dot{r}_F dx \quad (5.4)$$

Die Flammendicke beschreibt näherungsweise die Breite der Reaktionszone. Sie ist definiert als [41]:

$$\delta = \frac{T_1 - T_0}{\max\left(\left|\frac{\partial T}{\partial x}\right|\right)} \quad (5.5)$$

$T_0$  ist die Temperatur des unverbrannten Gases,  $T_1$  die Temperatur des verbrannten Gases und  $\max\left(\left|\frac{\partial T}{\partial x}\right|\right)$  der maximale Temperaturgradient. Der Vergleich der Flammgeschwindigkeit und -dicke ist in Tabelle 5.2 dargestellt. Die Ergebnisse stimmen dabei sehr gut überein

**Tabelle 5.2:** Vergleich der Flammgeschwindigkeit und -dicke zwischen dem DNS Löser und CHEMKIN.

	$s_L$ [m/s]	$\delta$ [mm]
<b>CHEMKIN</b>	0.078	1.266
<b>DNS Löser</b>	0.077	1.244

### 5.1.3 Einfluss der Reaktionsmechanismen auf die Rechenzeit

Um das Verhalten des DNS Löser besser nachvollziehen zu können, wurde der Einfluss verschiedener Reaktionsmechanismen unter den in Unterabschnitt 5.1.1 beschriebenen Bedingungen untersucht. Dabei wurde sowohl eine Methan/Luft Verbrennung (B.4), als auch die Wasserstoff/Luft Verbrennung nach Li et al (B.1), Curran et al. (B.2), und Maas und Pope (B.3) betrachtet. Grundsätzlich unterscheiden sich die Reaktionsmechanismen in vier Aspekten:

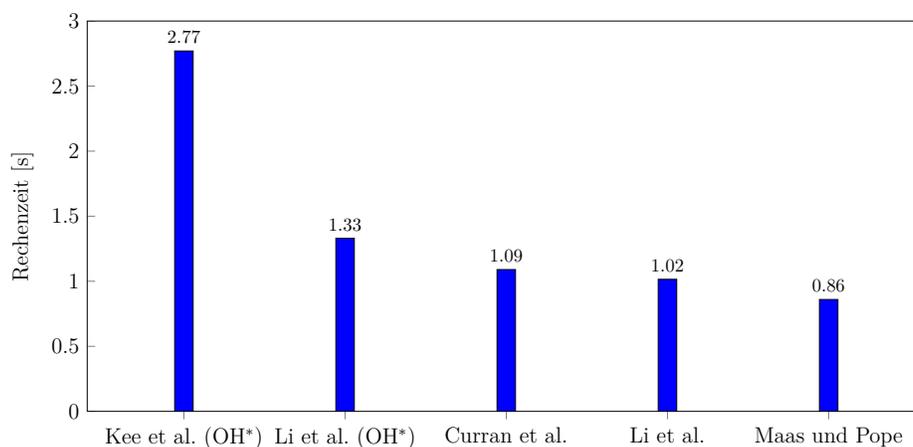
1. Die Anzahl der Spezies, die an der Reaktion beteiligt sind, inklusive inerten Spezies. Die Methan/Luft Verbrennung umfasst 18 Spezies, während die hier betrachteten Mechanismen der Wasserstoff/Luft Verbrennung 9 Spezies umfassen. Zusätzlich wurde die Wasserstoff/Luft Verbrennung nach Li et al. untersucht, die um angeregte OH\*-Speziesmoleküle erweitert wurde und somit 10 Spezies enthält (B.5).
2. Die Anzahl an einzelnen Reaktionen, die den Reaktionsmechanismus bilden. Sind die Rückreaktionen explizit angegeben, werden sie trotzdem im Folgenden nur einfach gezählt.
3. Druckabhängige Geschwindigkeitskonstanten. Die Berechnung von druckabhängigen Geschwindigkeitskonstanten nach Gl. (2.41) ist rechnerisch deutlich aufwendiger als die Berechnung mit dem normalen Arrhenius Gesetz in Gl. (2.31).
4. Angabe der Parameter für die Rückreaktion. Wenn ein Mechanismus für jede Reaktion sowohl die Parameter der Hin- sowie der Rückreaktion bereitstellt, dann können die Geschwindigkeitskonstanten der Rückreaktion direkt berechnet werden. Ansonsten muss für jede Reaktion zuerst die Gleichgewichtskonstante berechnet werden, um damit die Geschwindigkeitskonstante der Rückreaktion nach Gl. (2.34) zu ermitteln.

Eine Übersicht der untersuchten Reaktionsmechanismen ist in Tabelle 5.3 gegeben.

Die Rechenzeit, die für die Ermittlung der Reaktionsraten aufgewendet werden muss, ist stark abhängig vom gewählten Reaktionsmechanismus. Die Rechenzeiten für die Lö-

**Tabelle 5.3:** *Vergleich der Reaktionsmechanismen.*

Mechanismus	Spezies	Reakt.	druckabh.	Rückr.
<b>Kee et al. (OH*)</b> B.4,B.5	18	69	nein	nein
<b>Li et al. (OH*)</b> B.1,B.5	10	26	ja	nein
<b>Curran et al.</b> B.2	9	20	ja	ja
<b>Li et al.</b> B.1	9	19	ja	nein
<b>Maas und Pope</b> B.3	9	19	nein	ja

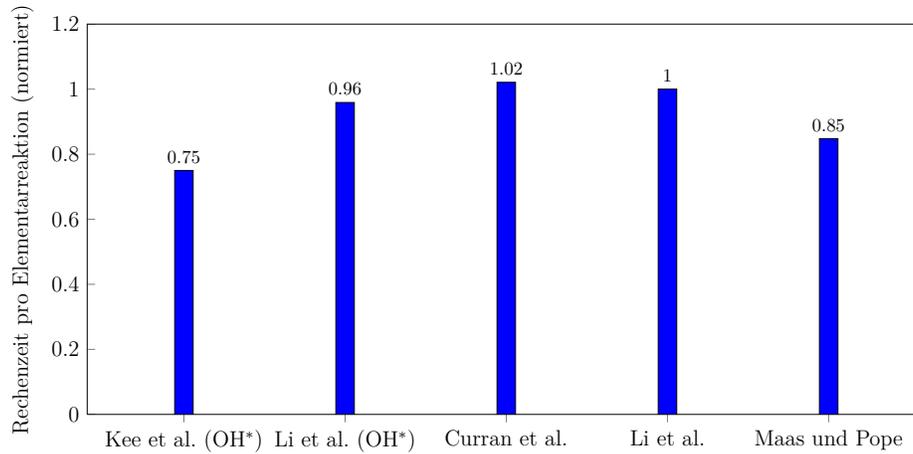


**Abbildung 5.4:** Durchschnittliche Zeit, die benötigt wird, um die Reaktionskinetik in Abhängigkeit des gewählten Mechanismus zu lösen. Mechanismen absteigend nach Anzahl der Elementarreaktionen sortiert. Messung auf dem VBT-Cluster [47] mit einem CPU-Kern.

sung der Reaktionskinetik sind in Abhängigkeit des Reaktionsmechanismus in Abbildung 5.4 dargestellt. Die Rechenzeit steigt mit zunehmender Anzahl an Elementarreaktionen. Teilt man die Rechenzeit für die Bestimmung der Reaktionsraten durch die Anzahl der Elementarreaktionen der Mechanismen und normiert sie, so ergibt sich das Bild in Abbildung 5.5.

Die Mechanismen nach Curran et al., Li et al. und Li et al. mit angeregten OH-Speziesmolekülen ( $\text{OH}^*$ ) weisen fast die gleiche Rechenzeit pro Elementarreaktion auf. Dies lässt den Schluss zu, dass die Rechenzeit zur Lösung der Reaktionskinetik näherungsweise linear mit der Anzahl der Elementarreaktionen steigt. Auch die Angabe expliziter Parameter zur Berechnung der Geschwindigkeitskonstanten der Rückreaktion hat keinen großen Einfluss auf die Rechenzeit.

Die drei Mechanismen haben gemeinsam, dass sie druckabhängige Geschwindigkeitskonstanten nutzen. Im Gegensatz dazu nutzen der Mechanismus von Maas und Pope und der der Methan/Luft Verbrennung keine druckabhängigen Geschwindigkeitskonstanten, sondern nur das Arrhenius Gesetz. Die Rechenzeit der Reaktionskinetik für den Mechanismus nach Maas und Pope beträgt daher nur das 0.85 fache der Rechenzeit der anderen drei Mechanismen der Wasserstoff/Luft Verbrennung. Der Mechanismus der Methan/Luft Verbrennung benötigt nur das 0.75 fache der Zeit des Mechanismus



**Abbildung 5.5:** Durchschnittliche Zeit pro Elementarreaktion, die benötigt wird, um die Reaktionskinetik in Abhängigkeit des gewählten Mechanismus zu lösen. Die Rechenzeit ist normiert auf die Zeit, die für den Mechanismus nach Li et al. benötigt wird.

nach Li et al. Der Grund dafür ist, dass der Mechanismus der Methan/Luft Verbrennung mit 69 Elementarreaktionen mehr als dreimal so viele Elementarreaktionen hat als die anderen hier untersuchten Mechanismen. Daher nimmt der Anteil der Lösung der Reaktionskinetik, der unabhängig von der Elementarreaktionszahl ist (z.B. die Temperaturbestimmung wie in Abschnitt 4.1) überproportional stark ab.

Untersucht man den Einfluss der Reaktionsmechanismen auf die anderen Löserabschnitte, so stellt sich heraus, dass die Lösung der Speziestransportgleichung und die Berechnung der thermischen Enthalpien quasi-linear von der Speziesanzahl abhängen (Abbildung 5.6). Die Rechenzeiten für diese beiden Abschnitte sind für alle Reaktionsmechanismen mit 9 Spezies innerhalb der Messungenauigkeit identisch. Sie sind doppelt so hoch mit dem Mechanismus mit 18 Spezies verglichen mit denen mit 9 Spezies. Bei dem erweiterten Mechanismus mit 10 Spezies lässt sich dagegen eine um 10 % gesteigerte Rechenzeit feststellen. Die Lösungszeit für die Energieerhaltung und die Neuberechnung der Transportgrößen dauert 2.4 Mal solange für den Mechanismus mit 18 Spezies, verglichen mit denen mit 9 Spezies. Dieser Abschnitt skaliert also nicht linear mit der Speziesanzahl.

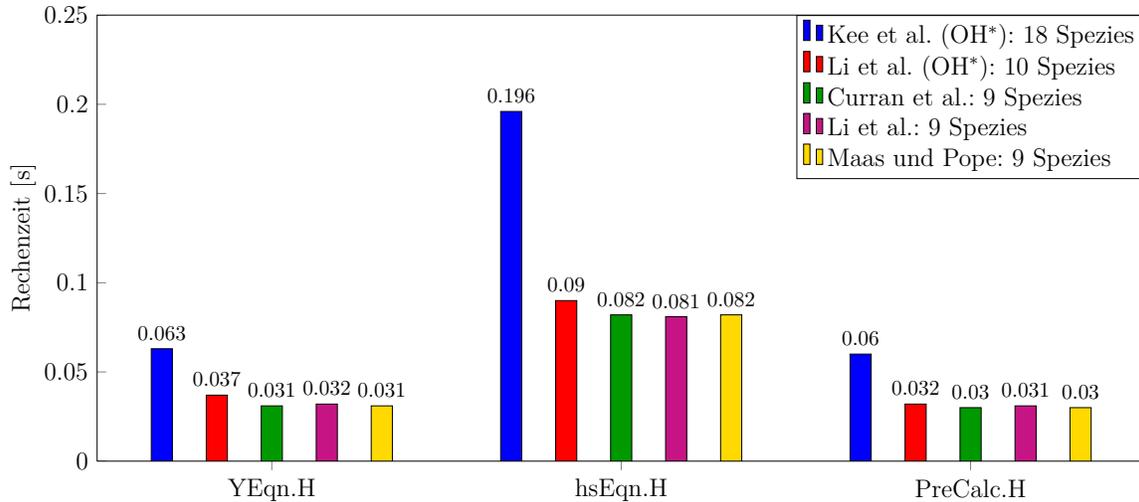


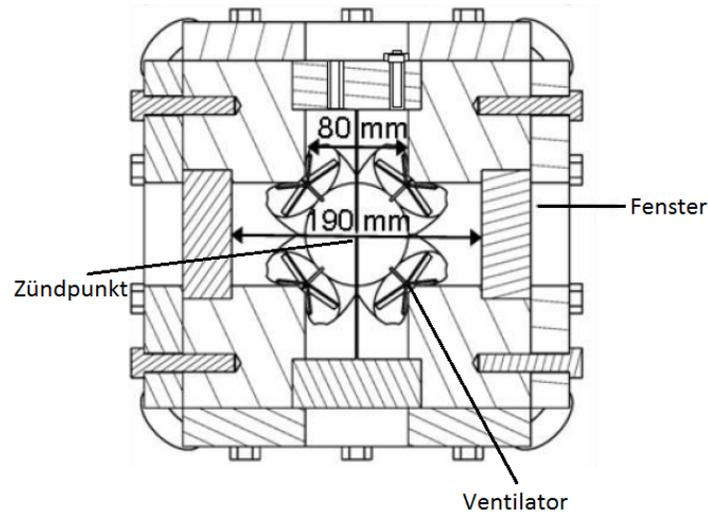
Abbildung 5.6: Abhängigkeit der Rechenzeit von der Speziesanzahl. Gemessen auf dem VBT-Cluster für den 1D Fall mit einem CPU-Kern.

## 5.2 Sphärische Flammenausbreitung im Bombenbehälter (3D)

### 5.2.1 Experimenteller Aufbau

Um die Anwendbarkeit des neuen Lösers für 3D Verbrennungsvorgänge zu zeigen, wurde die sphärische Ausbreitung einer Wasserstoff/Luft Flamme in einem Bombenbehälter mittels DNS durchgeführt. Die Bedingungen wurden dabei so gewählt, dass sie mit einem am EBI/VBT durchgeführten Bombenexperiment [2] vergleichbar sind. Der Bombenbehälter ist näherungsweise kugelförmig und hat ein Hohlvolumen von 2.28 L bei einem Durchmesser von 8 cm, in der ein isochorer Verbrennungsdruck von bis zu 150 bar erreicht werden kann (Abbildung 5.7). Im Mittelpunkt des Kugelvolumens befinden sich zwei Zündelektroden, mit denen die Gasmischung gezündet wird. Als Messtechnik stehen ein Laser zur Verfügung, mit dem die Oberfläche der Flamme aufgenommen werden kann, sowie ein Drucksensor, der den Druckverlauf aufzeichnet. Zusätzlich gibt es Sichtfenster zur Beobachtung der Ausbreitung der Flammenfront sowie acht Ventilatoren, mit denen eine turbulente Strömung im Verbrennungsraum induziert werden kann.

Im Experiment wurde unter anderem die Verbrennung eines Wasserstoff/Luft Ge-



**Abbildung 5.7:** *Aufbau des Bombenbehälters* [2].

misches bei 1 bar und einer Luftzahl von  $\lambda = 3$  untersucht (Tabelle 5.4), die schon im 1D Fall (Abschnitt 5.1) simuliert wurde. Die Starttemperatur beträgt 300 K und findet im ruhenden Fluid statt. Diese Bedingungen werden auch für die DNS übernommen.

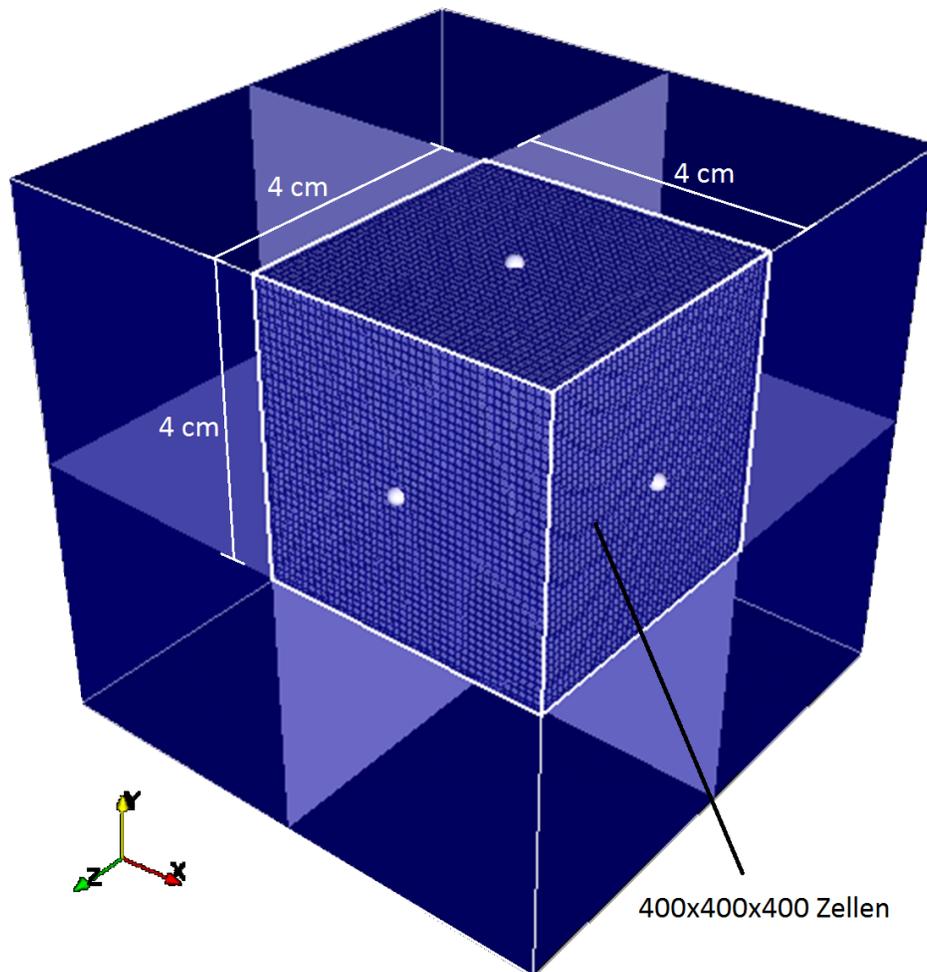
**Tabelle 5.4:** *Betriebsbedingungen des Experiments sowie der 3D DNS.*

$p_0$	$T_0$	$\lambda_0$	$Y_{H_2,0}$	$Y_{O_2,0}$	$Y_{N_2,0}$	$\vec{u}_0$
1 bar	300 K	3	0.0097	0.2307	0.7596	0

## 5.2.2 Numerische Einstellungen

Zur DNS der Wasserstoff/Luft Flamme im Bombenbehälter mit dem optimierten Löser wurden dieselben physikalischen Bedingungen gewählt wie die des Experiments (Tabelle 5.4). Die Anfangswerte sind dabei im gesamten Simulationsgebiet homogen verteilt.

Im Gegensatz zum Bombenbehälter ist das Simulationsgebiet keine Kugel, sondern ein Würfel. Bei der Simulation wird zusätzlich ausgenutzt, dass die Flamme sich näherungsweise kugelsymmetrisch um den Zündpunkt ausbreitet. Daher wird nur ein Achtel der gesamten Flamme berücksichtigt (Abbildung 5.8). Das Rechengitter ist somit ein Würfel mit einer Kantenlänge von 4 cm. Das Simulationsgebiet besteht aus insgesamt



**Abbildung 5.8:** 3D Gitter der DNS des Bombenbehälters. Zur besseren Übersicht ist nur ein Teil der 64 Millionen Zellen dargestellt.

$400 \times 400 \times 400 = 64$  Millionen äquidistanten Zellen; in jede Raumrichtung gibt es 400 Zellen. Die Gitterweite beträgt  $\Delta x = \Delta y = \Delta z = 0.1$  mm. Da nur ein Achtel des gesamten Verbrennungsraums berücksichtigt wird, haben die drei inneren Würfelseiten eine Symmetrierandbedingung, die äußeren drei Seiten sind als Wand festgelegt. Die festen Werte für die verschiedenen Größen an der Wand sind in Tabelle 5.5 aufgeführt.

Zum Zeitpunkt  $t = 0$  s wird das Gasmischung gezündet. Die Zündung des Gases geschieht, indem in einem Kugelvolumen mit einem Durchmesser von 1 mm ein zusätzlicher Quellterm von  $5 \cdot 10^9$  W/m<sup>3</sup> für 0.11 ms in der Energieerhaltung berücksichtigt

**Tabelle 5.5:** Übersicht über die festen Werte der Wandbedingung.

$p_0$	$T_0$	$\lambda_0$	$Y_{H_2,0}$	$Y_{O_2,0}$	$Y_{N_2,0}$	$Y_{\text{Rest}}$	$\vec{u}_0$
$\nabla p = 0$	300 K	3	0.0097	0.2307	0.7596	0	0

wird. Die gesamte Zündenergie beträgt somit 2.3 mJ. Der Mittelpunkt der Zündung liegt im Ursprung des Simulationsgebietes.

Zur Beschreibung der chemischen Reaktion wird der Reaktionsmechanismus nach Li et al. verwendet (B.1), da in diesem Mechanismus die Druckabhängigkeit der Geschwindigkeitskonstanten berücksichtigt wird. Die gewählten Bedingungen für die DNS sind in Tabelle 5.6 dargestellt. Die Benennung der Diskretisierungsverfahren ist dabei von OpenFOAM übernommen [37]. Insgesamt wurde die Ausbreitung der Flammenfront von der Zündung bei  $t = 0$  s bis zum Zeitpunkt  $t = 0.07$  s bei einem Zeitschritt von  $1 \cdot 10^{-6}$  s simuliert. Die Rechnung wurde ca. 6 Tage lang auf dem Supercomputer CRAY XE6 HERMIT des Hochleistungsrechenzentrums Stuttgart [10] (HLRS) mit 4096 CPU Kernen durchgeführt (insgesamt ca. 590000 CPU Stunden).

**Tabelle 5.6:** Übersicht über die numerischen Bedingungen der DNS [37].

Ableitungsschema	Diskretisierung	Ordnung
Zeitableitung $\frac{\partial}{\partial t}$	backward implicit	2
Gradient $\nabla \phi$	gauss linear	2
Konvektion $\nabla \cdot \phi$	gauss cubic	4
Diffusion $\nabla \cdot \nabla \phi$	gauss linear	2

Größe	Löser	Vorkonditionierung
$\rho, p$	PCG	DIC
$\vec{u}, h_s, Y_k$	PBiCG	DILU

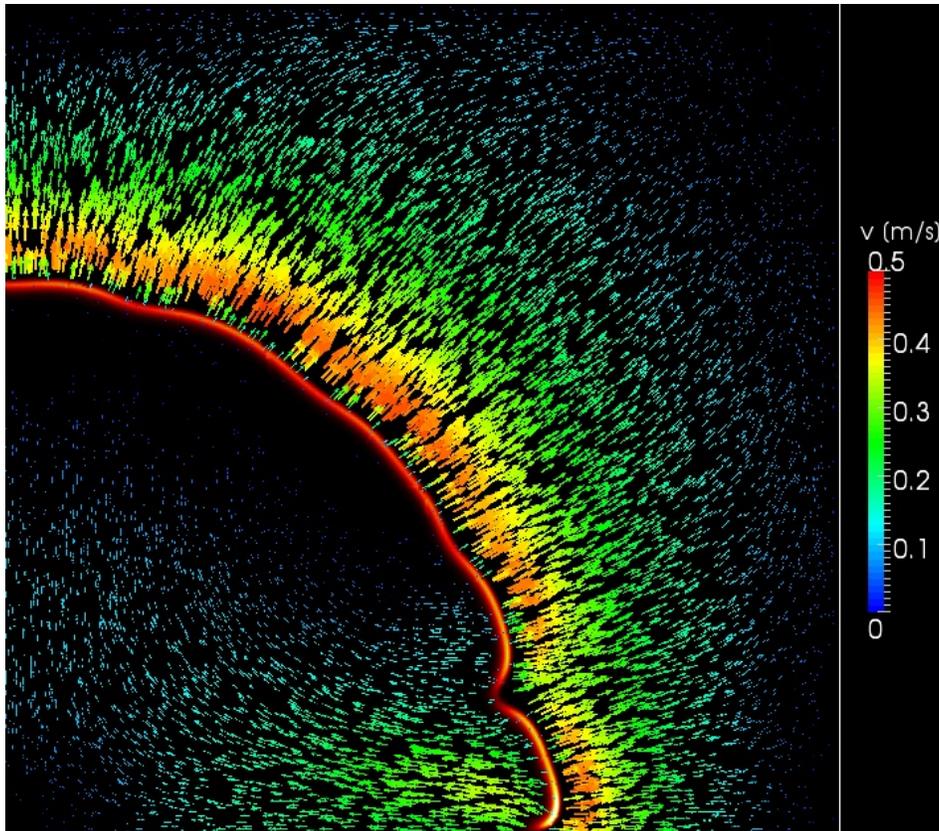
  

Abmessung	Zellen	$\Delta t$ [s]	$\Delta x$ [m]
4 cm $\times$ 4 cm $\times$ 4 cm	$64 \cdot 10^6$	$1 \cdot 10^{-6}$	$1 \cdot 10^{-4}$

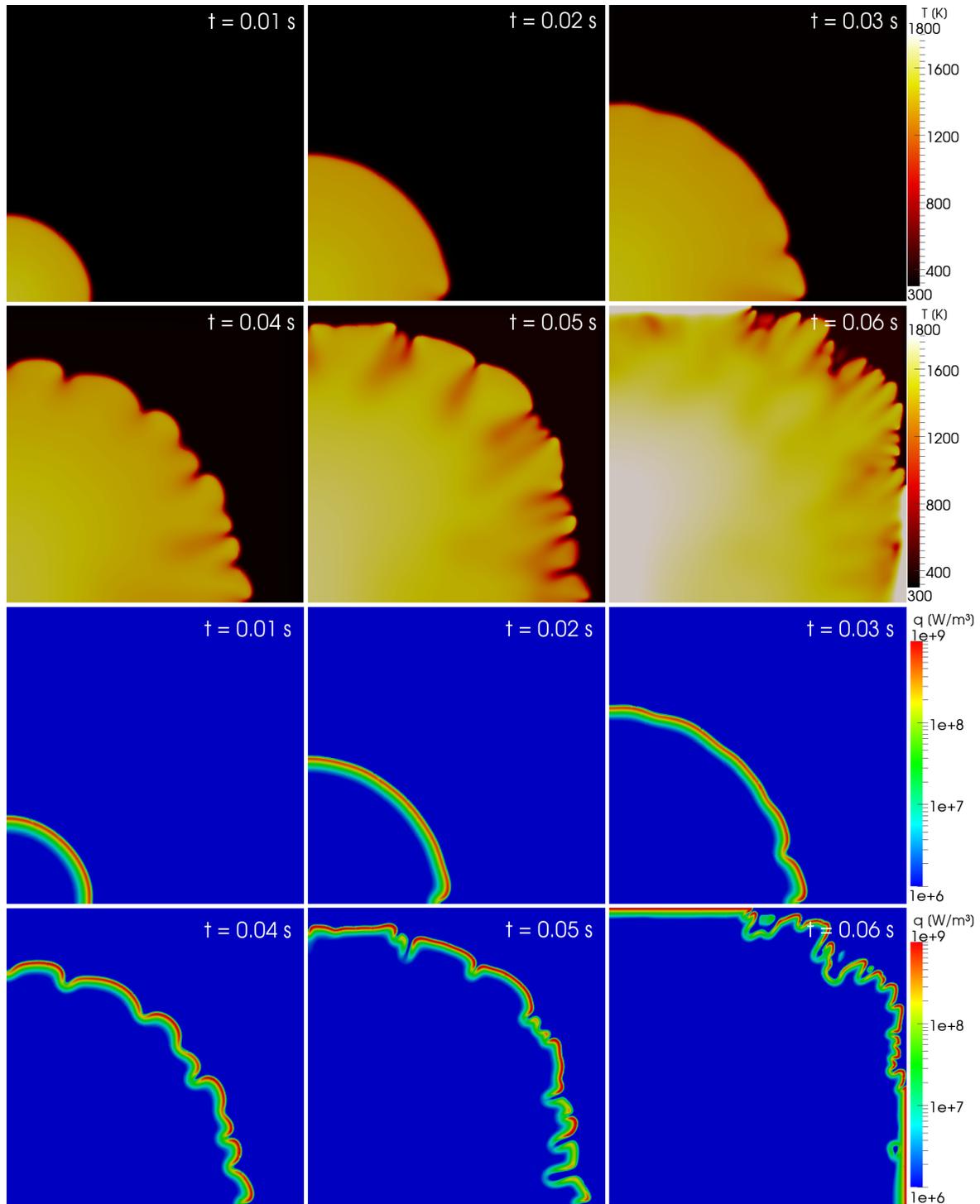
### 5.2.3 Darstellung der Ergebnisse

Zur Auswertung der DNS werden die zeitlichen Verläufe des Drucks, der Temperatur und der Wärmefreisetzungsrates untersucht. Zusätzlich werden die Flammengeschwindigkeit und die Form der Flammenfront mit den Ergebnissen des Experiments verglichen. Nach der Zündung breitet sich die laminare Flammenfront näherungsweise mit konstanter Geschwindigkeit als ideale Kugel aus. Der Ablauf der Flammenausbreitung ist in Abbildung 5.10 am Beispiel der Temperatur und Wärmefreisetzungsrates dargestellt.

Ab dem Zeitpunkt  $t = 0.02$  s wird die Flammenfront instabil und beginnt sich zu falten. Die Geschwindigkeit der Flammenfront ist aufgrund des Auftriebs am oberen Rand



**Abbildung 5.9:** Vektorfeld der Strömungsgeschwindigkeit bei  $t=0.03$  s. Am unteren Teil der Flammenfront stellt sich ein Rückstrom in die Flamme ein.



**Abbildung 5.10:** Zeitliche Entwicklung der Temperatur und der Wärmefreisetzungsrate. Ab  $t = 0.02$  s weist die Flammenfront erste Faltungen auf.



**Abbildung 5.11:** Form der Flammenfront bei einem flächenbezogenen Flammenradius von 2 cm. **Links:** Aufnahme des verbrannten Gases aus dem Experiment [2]. **Rechts:** Massenbruch von Wasserstoff aus der DNS.

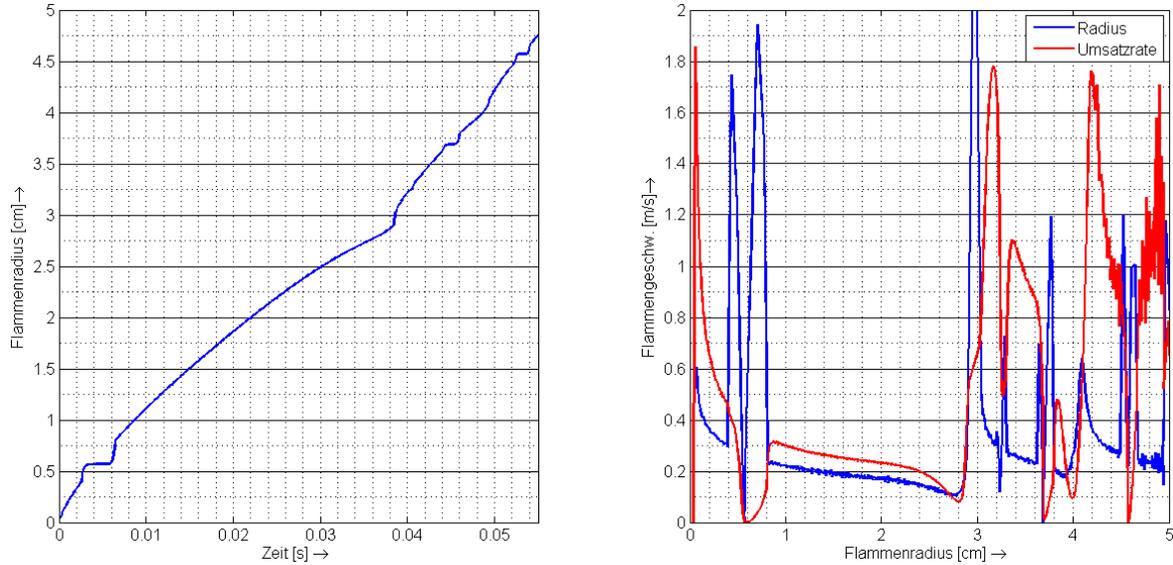
der Flamme größer als am Unteren. Daher stellt sich eine Rückströmung in den unteren Teil der Flamme ein, der zu ersten Wellen in der Flammenfront führt (Abbildung 5.9). Diese Faltung wurde auch im Experiment beobachtet. Ein Vergleich der Form der Flammenfront zwischen Experiment und DNS ist in Abbildung 5.11 dargestellt.

Die Flammengeschwindigkeit wurde mit zwei Methoden ausgewertet. Bei der ersten Methode wird die Umsatzrate zur Bestimmung der Flammengeschwindigkeit nach Gl. (5.4) benutzt:

$$\dot{m}_F = \rho_0 s_L Y_{F,0} = \int_{-\infty}^{\infty} \dot{r}_F dx$$

Dazu wird eine Raumdiagonale definiert, die vom Zündpunkt bis zum gegenüberliegenden Eckpunkt verläuft. Auf dieser 1D Diagonalen kann wie im 1D Fall die Flammengeschwindigkeit bestimmt werden.

Bei der zweiten Methode wird die Bewegung der Flammenfront über die Zeit betrachtet. Aus der Position der Flammenfront lässt sich die radiale Ausbreitungsgeschwindigkeit der Flamme  $\frac{dr}{dt}$  bestimmen. Diese kann dann in die Flammengeschwin-



**Abbildung 5.12:** Ausbreitung der Flammenfront. **Links:** Position der Flammenfront über die Zeit. **Rechts:** Flammengeschwindigkeit über den Flammenradius  $r$ . In Rot ist die Flammengeschwindigkeit über die Berechnung der Umsatzrate nach Gl. (5.4), in Blau die Flammengeschwindigkeit aus der Bewegung der Flammenfront nach Gl. (5.6) dargestellt.

digkeit umgerechnet werden [41]:

$$s_L = \frac{\rho_1}{\rho_0} \frac{dr}{dt} \quad (5.6)$$

$\rho_0$  ist die Dichte des unverbrannten Gases und  $\rho_1$  die Dichte des verbrannten Gases. Das Dichteverhältnis lässt sich über das ideale Gasgesetz bestimmen:

$$\frac{\rho_1}{\rho_0} = \frac{\frac{p_1 \bar{M}_1}{RT_1}}{\frac{p_0 \bar{M}_0}{RT_0}} = \frac{p_1 \bar{M}_1}{p_0 \bar{M}_0} \cdot \frac{T_0}{T_1} \quad (5.7)$$

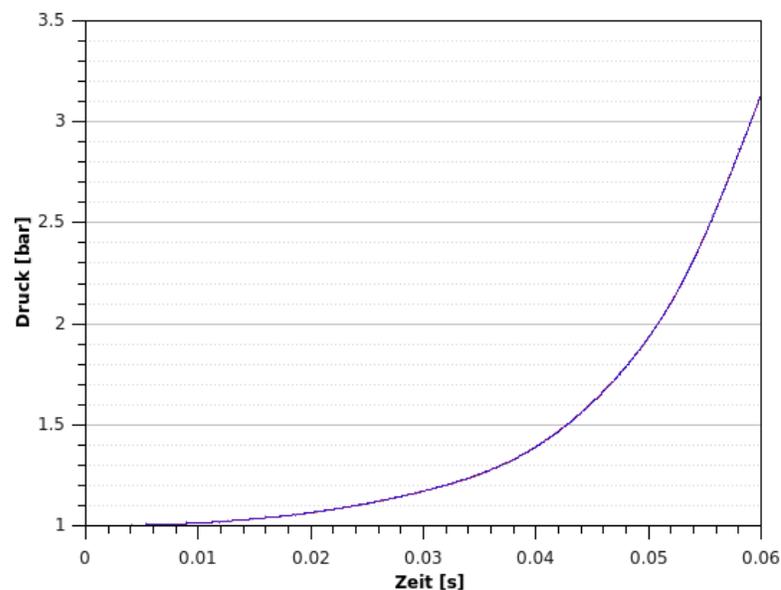
Der Verlauf der Flammenfront und die Auswertung der Flammengeschwindigkeit sind in Abbildung 5.12 dargestellt. Der blaue Graph im rechten Bild stellt die Flammengeschwindigkeit dar, die über die Bewegung der Flammenfront ermittelt wurde. Der rote Graph ist die Flammengeschwindigkeit aus der Massenbilanz mit der Umsatzrate. In dieser Darstellung gibt es eine Unstetigkeit zu Beginn der Flammenfrontausbrei-

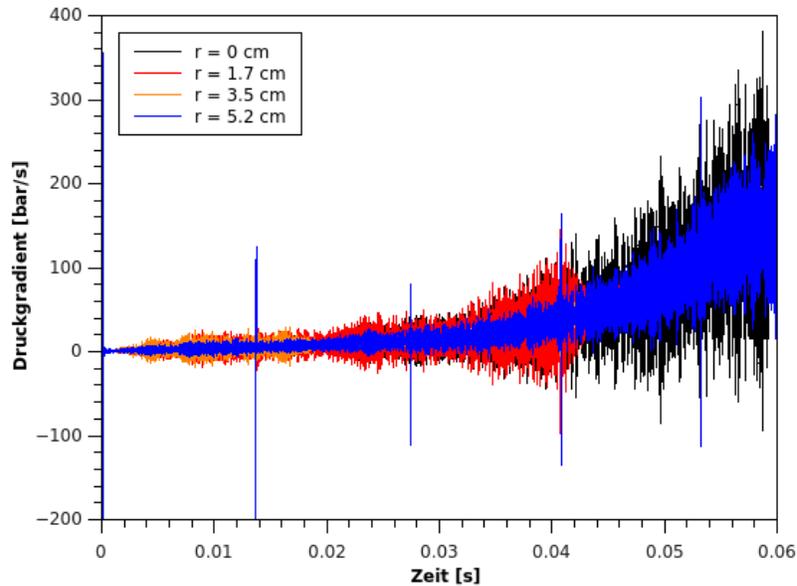
**Tabelle 5.7:** *Flammengeschwindigkeit im Bombenbehälter.*

Umsatzrate $-\frac{1}{\rho_0 Y_{F,0}} \int_{-\infty}^{\infty} \dot{r}_F dx$	Flammenfrontgeschw. $\frac{\rho_1}{\rho_0} \frac{dr}{dt}$	Experiment [2]
0.21 m/s	0.18 m/s	0.18 m/s

zung: Zwischen 0.002 s und 0.006 s kommt die Reaktion und damit die Bewegung der Flammenfront zum Erliegen. Danach gibt es einen sprunghaften Anstieg der Flammenfrontposition und -geschwindigkeit. Der Grund dafür ist, dass bei dieser Simulation die Raumdiagonale, die zur Bestimmung der Flammenfrontposition verwendet wurde, einen Punkt lokaler Löschung in der Flammenfront schneidet. Da die Hauptraumdiagonale den Punkt der Flammenfront durchstößt, der am weitesten von den Symmetrierandbedingungen entfernt ist, herrscht dort die größte, lokale Streckung und somit die größte Wahrscheinlichkeit für eine kurzzeitige Löschung der Flamme.

Für den Bereich der stabilen Flammenausbreitung zwischen den Flammenradien 1.5 cm und 2.5 cm weisen beide Methoden eine geringe Abnahme der Flammengeschwindigkeit von 0.20 m/s auf 0.17 m/s auf. Dabei ist die Flammengeschwindigkeit

**Abbildung 5.13:** *Zeitlicher Verlauf des Drucks an vier verschiedenen Orten.*



**Abbildung 5.14:** Zeitlicher Verlauf des Druckgradienten an vier Orten entlang der Hauptdiagonalen.

mit der Radiusmethode nach Gl. (5.6) kleiner, als die, die über die Massenbilanz mit der Umsatzratenmethode nach Gl. (5.4) bestimmt wurde. Bei beiden Methoden ist zudem deutlich der Einfluss der Faltung der Flammenfront ab einem Flammenradius von ca. 3 cm. erkennbar. Durch die Faltung bzw. Streckung der Flammenfront kommt es immer wieder zur Löschung und Zündung der Flamme. Auch im Experiment wurde die Flammengeschwindigkeit für den Bereich der stabilen, laminaren Ausbreitung bei einem Flammenradius von 2 cm bestimmt [2]. Ein Vergleich der beiden Bestimmungsmethoden für eine Flamme mit einem Radius von 2 cm mit dem im Experiment gemessenen Wert ist in Tabelle 5.7 dargestellt. Auch die Flammengeschwindigkeit zeigt eine gute Übereinstimmung mit dem Experiment.

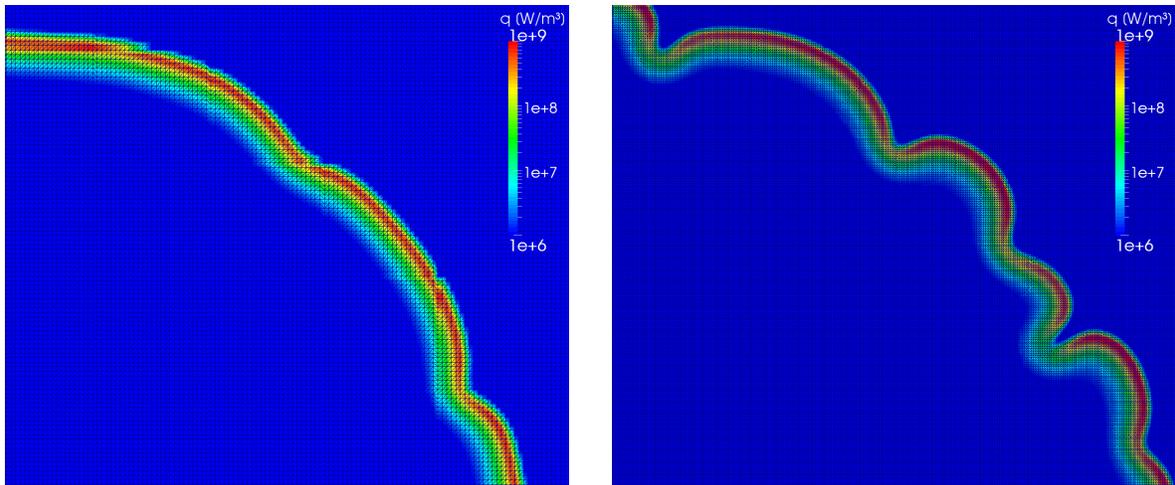
Im Laufe der Verbrennung steigt der Druck im Bombenbehälter. Der Druckverlauf ist für vier verschiedenen Orte in Abbildung 5.13 dargestellt. Der erste Ort liegt im Zündpunkt. Die anderen drei Orte befinden sich im Abstand von 1.7 cm, 3.5 cm und 5.2 cm vom Zündpunkt auf der Hauptdiagonalen. Die Druckverläufe fallen dabei quasi zusammen. Der Grund dafür ist, dass sich der Druck mit Schallgeschwindigkeit ausbreitet, welche um Größenordnungen schneller als die Strömungsgeschwindigkeit ist.

Diese Ergebnisse werden auch durch das Experiment qualitativ bestätigt [2]. Obwohl der Druckverlauf für alle Orte fast identisch ist, gibt es kleine Druckschwankungen zwischen den Orten. In Abbildung 5.14 ist der Druckgradient für die vier verschiedenen Orte über die Zeit dargestellt.

### 5.2.4 Einfluss der Gitterauflösung

Für die gleiche Rechengebiet wie in Unterabschnitt 5.2.2 mit denselben numerischen Bedingungen wurde auch eine Simulation mit einem gröberen Rechengitter mit 8 Millionen Zellen durchgeführt. In diesem Rechengitter gibt es 200 Zellen in jede Raumrichtung mit einer Gitterweite von 0.2 mm. Zu Beginn der Flammenausbreitung ergeben sich für beide Rechengitter sehr ähnliche Verläufe. Erst bei Einsetzen der Faltung der Flammenfront zeigen sich deutliche Unterschiede: Im feineren Rechengitter gibt es mehr Faltungen, die zudem stärker ausgeprägt sind. Abbildung 5.15 zeigt den Vergleich zwischen dem groben und dem feinen Gitter mit der Flammenfront bei  $t = 40$  ms am Beispiel der Wärmefreisetzungsrates. Links ist das Gitter mit 8 Millionen Zellen und rechts das feine Gitter mit 64 Millionen Zellen.

Der Vergleich zeigt, dass die Ergebnisse der DNS für die einsetzende Faltung der



**Abbildung 5.15:** Vergleich der Flammenfront für zwei Rechengitter am Beispiel der Wärmefreisetzungsrates zum Zeitpunkt  $t = 40$  ms. **Links:** Rechengitter mit 8 Millionen Zellen. **Rechts:** Rechengitter mit 64 Millionen Zellen.

Flammenfront noch immer abhängig von der Gitterauflösung sind. Weitere Simulationen mit noch feineren Gittern sind daher sinnvoll, um diese Abhängigkeit weiter zu untersuchen.

### 5.2.5 Skalierungsverhalten auf der HPC-Plattform JUQUEEN

Die Skalierbarkeit mit der Anzahl eingesetzter CPU-Kerne ist ein sehr wichtiger Faktor für einen DNS Löser. Da DNS sehr rechenintensiv ist, müssen Supercomputer eingesetzt werden. Diese zeichnen sich durch eine große Anzahl CPUs aus, mit denen die Rechenoperationen zur Lösung von CFD Problemen durchgeführt werden können. Nur wenn die Rechenzeit mit steigender CPU Zahl möglichst linear sinkt, kann der Löser sinnvoll auf Supercomputern eingesetzt werden.

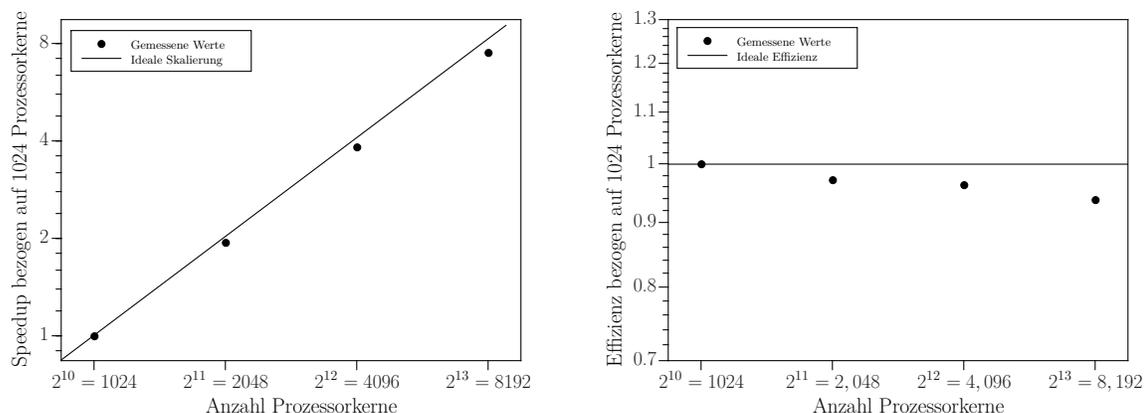
In der Arbeit von Bonart [4] konnte für den Löser eine superlineare Skalierung mit bis zu 8192 CPU-Kernen auf der High Performance Computing (HPC) Plattform Cray XE6 HERMIT des Hochleistungsrechenzentrums Stuttgart [10] (HLRS) gezeigt werden. In dieser Arbeit ist das Skalierungsverhalten des optimierten DNS Löser auf dem Supercomputer JUQUEEN (Unterabschnitt 3.1.3) als HPC Plattform mit dem 64 Millionen Zellen Gitter der 3D DNS aus diesem Kapitel untersucht worden. Die Simulation wurde dabei parallel mit 1024, 2048, 4096 und 8192 Prozessorkernen durchgeführt. Der relative Rechenzeitgewinn  $S$  (Speedup) hat folgende Definition:

$$S = \frac{t_{p_1}}{t_{p_2}} \quad (5.8)$$

$t_{p_1}$  ist die Rechenzeit für ein Programm mit  $p_1$  Prozessorkernen. Für diese Messung sind für  $p_1$  aus praktischen Gründen 1024 CPU-Kerne, für  $p_2$  2048, 4069 und 8192 CPU-Kerne gewählt worden. Ein idealer Speedup verläuft dabei linear mit der Anzahl der CPU-Kerne [49]. Die Effizienz  $E$  ist definiert als

$$E = \frac{p_1}{p_2} \frac{t_{p_1}}{t_{p_2}} \quad (5.9)$$

Ein Programm mit idealem Skalierungsverhalten hat somit eine Effizienz von 1. Die Ergebnisse der Skalierungsanalyse sind in Abbildung 5.16 dargestellt. Der DNS



**Abbildung 5.16:** Skalierungsverhalten auf der HPC-Plattform JUQUEEN. **Links:** Relativer Speedup des DNS Löser. **Rechts:** Effizienz der Parallelisierung für die Rechengeschwindigkeitssteigerung.

Löser weist auch bei 8192 noch eine Effizienz von 94 % auf. Damit ist gezeigt, dass der DNS Löser auch auf der parallelen IBM Architektur von JUQUEEN effizient eingesetzt werden kann.

Da der extrahierte Canteracode nun dem Löser beiliegt, ist man nicht mehr an die vorinstallierte Canteraversion gebunden (Abschnitt 3.3). So ist es möglich, gezielt den Einfluss verschiedener Compiler zu testen. Auf JUQUEEN hat sich gezeigt, dass der IBM-Compiler bgxlc++ ein deutlich performanteres Programm für die IBM-Architektur erzeugt als der GNU g++ Compiler. Die Rechenzeit lässt sich bei der Verwendung des IBM-Compilers um bis zu 40 % reduzieren.

# Kapitel 6

## Zusammenfassung und Ausblick

### 6.1 Zusammenfassung

Das Ziel dieser Arbeit ist es, einen Löser für Direkte Numerische Simulation in den Punkten Effizienz und Handhabung zu verbessern. Der ursprüngliche Löser benutzt die externe Bibliothek Cantera zur Berechnung der Reaktionsraten und druck- und temperaturabhängiger Stoffgrößen; daher muss Cantera auf jeder Plattform, auf der der Löser benutzt werden soll, installiert werden. Da Cantera viel mehr Funktionalitäten anbietet, als der DNS Löser benötigt, sind relevante Codestellen für den Löser nicht sofort erkennbar. Deshalb sind die relevanten Routinen aus Cantera in dieser Arbeit identifiziert und direkt in den Löser implementiert worden. Dadurch muss auf neuen Systemen, auf denen der Löser verwendet werden soll, nicht mehr Cantera installiert werden. Zudem ist man nicht an die vorinstallierte Canteraversion gebunden, wodurch sich leichter Parameter wie z.B. der verwendete Compiler untersuchen lassen. Weitere Vorteile sind, dass nun der Canteracode dem Löser direkt beiliegt. Dadurch kann das Programmverhalten besser nachvollzogen und Anpassungen für den DNS Löser gemacht werden. Dies erhöht die Wartbarkeit des Lösers. Mittels ausführlichem Profiling wurden zudem Leistungsengpässe im DNS Löser gesucht. Mit Optimierungen für die Berechnung der Speziesenthalpien und der Geschwindigkeitskonstanten sowie spezielle Anpassungen des Canteracodes an den Löser konnte die Rechengeschwindigkeit um 20 % gesteigert werden.

Auch die Berechnung der Reaktionsraten durch eine Mittelung über den Zeitschritt wurde untersucht. Dabei hat sich die Verwendung der externen Bibliothek SUNDIALS für die Lösung der Reaktorgleichung als sehr sinnvoll herausgestellt: die Integration der Reaktorgleichung durch SUNDIALS erhöht die Stabilität des Löser und ermöglicht es, größere DNS Zeitschritte zu wählen. Dies verbessert die Rechengeschwindigkeit des Programms. Zudem macht die Verwendung von SUNDIALS den DNS Löser sehr flexibel: Die Reaktorgleichung wird nur in Zellen mit den kleinsten chemischen Zeitmaßen besonders häufig gelöst. Außerdem passt sich der chemische Zeitschritt automatisch an die aktuellen Reaktionsbedingungen an, sodass der chemische Zeitschritt nicht durch den Nutzer festgelegt werden muss. Im Rahmen dieser Untersuchungen wurden auch Werte für die Toleranzparameter ermittelt, die einen guten Kompromiss zwischen Genauigkeit und Rechengeschwindigkeit bieten.

Für die Validierung des neu implementierten Löser wurde zuerst eine 1D Simulation einer vorgemischten Wasserstoff/Luft Flachflamme durchgeführt. Die Verläufe der Reaktionsraten und Spezieskonzentrationen sowie der Temperatur und Flammgeschwindigkeit dieser Simulation wurden mit den Werten des kommerziellen Programms CHEMKIN verglichen. Dabei konnte eine sehr gute Übereinstimmung gezeigt werden. Im Rahmen dieser Simulationen wurde zusätzlich gezeigt, dass die Rechenzeit zur Bestimmung der Reaktionsraten näherungsweise linear mit der Anzahl der Elementarreaktionen im verwendeten Reaktionsmechanismus zunimmt. Die Rechenzeit für die Lösung der Speziestransportgleichung und der Berechnung der Speziesenthalpien hängen linear von der Speziesanzahl ab.

Danach wurde die 3D Simulation einer Wasserstoff/Luft Verbrennung in einem Bombenbehälter durchgeführt. Die Simulationsergebnisse konnten mit experimentell ermittelten Daten verglichen werden. Der Vergleich der Flammgeschwindigkeit, des Druckverlaufs und der Form der Flammenfront decken sich dabei mit dem Experiment. Durch den Vergleich der DNS Rechnung mit einem größeren Gitter konnte gezeigt werden, dass die Gitterauflösung eine wichtige Rolle für die Faltung der Flammenfront spielt. Für diesen Fall wurde auch die Skalierbarkeit auf dem Supercomputer JUQUEEN Jülich Blue Gene/Q des Forschungszentrums Jülich (JSC) untersucht. Dabei konnte eine fast ideale Skalierung der Rechengeschwindigkeit mit der Anzahl der CPU Kerne für bis zu

8192 Prozessorkerne gezeigt werden.

Abschließend lassen sich die Vorteile des neuen DNS Löser folgendermaßen zusammenfassen:

- Da der Löser auf open-source Softwarepaketen wie OpenFOAM und Cantera basiert ist, ist er frei zugänglich
- Er ist in dieser und in weiteren Arbeiten für verschiedene Verbrennungsfälle validiert worden [4][5]
- Der Löser kann leicht um weitere physikalische Modelle wie z.B. Strahlungsmodelle erweitert werden, da solche Modelle bereits in OpenFOAM implementiert sind
- Er erreicht hohe Effizienz auf parallelen Hochleistungsplattformen und kann daher für komplexe Verbrennungsprobleme eingesetzt werden

## 6.2 Ausblick

Obwohl der Löser in dieser Arbeit bereits stark optimiert worden ist, bietet er weiteren Raum für Verbesserungen. Neben einer weiteren Steigerung der Rechengeschwindigkeit kann der Löser um zusätzliche Funktionalitäten erweitert werden. So können neben den bereits implementierten Reaktionstypen bei Bedarf weitere Typen aus Cantera übernommen werden. Zudem können Modelle wie Wärmestrahlung oder Thermodiffusion implementiert und validiert werden. Eine weitere Verbesserung der Handhabung wäre, die genutzten Routinen aus der externen Bibliothek BLAS/LAPACK (Unterabschnitt 3.1.2) direkt in den Löser zu implementieren; denn wenn Mischungsgemittelte Transportkoeffizienten genutzt werden, benötigt der Löser nur eine sehr kleine Teilmenge aus BLAS/LAPACK. Da die Rechengeschwindigkeit immer einen wichtigen Aspekt darstellt, können weitere Bemühungen um Optimierungen unternommen werden. Eine Möglichkeit wäre es, anstelle von äquidistanten Gittern dynamische Gitter zu benutzen.

Neben den Verbesserungen am Löser selbst kann der hier berechnete 3D Fall weiter ausgewertet werden. Zudem können mit dem Löser Simulationen für komplexere

Verbrennungsprozesse durchgeführt werden, mit deren Hilfe die grundlegenden Mechanismen der Verbrennungs-Strömungs-Interaktion erforscht und Verbrennungsmodelle wie LES oder RANS verbessert werden können [41].

# Anhang A

## Code-Ausschnitte

---

```
1 // Funktion, die die Gleichgewichtskonstanten und Geschwindigkeitskonstanten berechnet
2 void KITGasKinetics::update_rates_T()
3 {
4     doublereal T = thermo().temperature(); //die aktuelle Temperatur der Zelle
5
6     if ( std::abs(T-m_temp) < 1.e-5 ) // falls sich die Temperatur seit der letzten
7     { // Berechnung um weniger als 1e-5 K
8         return; // geändert hat, dann berechne nichts neu
9     }
10
11     m_temp = T; // Zwischenspeichern der aktuellen Temperatur
12
13     doublereal logT = std::log(T); // Berechne den natürlichen Logarithmus von T
14
15     if (!m_rfn.empty()) {
16         m_rates.update(T, logT, &m_rfn[0]); // Berechne die Geschwindigkeitskonstanten
17         // mit dem Arrheniusansatz
18
19     if (!m_rfn_low.empty())
20     { // Berechne die
21         m_falloff_low_rates.update(T, logT, &m_rfn_low[0]); // Geschwindigkeitskonstanten
22         m_falloff_high_rates.update(T, logT, &m_rfn_high[0]); // k_0 und k_inf nach Troe
23     }
24
25     if (!falloff_work.empty())
26         m_falloffn.updateTemp(T, &falloff_work[0]); //berechne k(T,p) nach Troe
27
28     updateKc(); // Berechne die Gleichgewichtskonstanten
29                // für die Reaktionen, die keine Arrhenius-
30                // paramter für die Rückreaktion angegeben haben
31 }
```

---

**Codeausschnitt A.1:** Vereinfachter Code der Funktion `_update_rates_T` zur Berechnung der Geschwindigkeits- und Gleichgewichtskonstanten.

# Anhang B

## Reaktionsmechanismen

### B.1 Li et al.

Der Reaktionsmechanismus nach Li et al. [30] beschreibt die Wasserstoff/Luft Verbrennung und wurde in dieser Arbeit für die Validierungsrechnungen eingesetzt.

```
ELEMENTS H O N END
SPECIES H2 H H2O2 H2O HO2 OH O2 O N2 END
REACTIONS          CAL/MOLE
H+O2=O+OH          3.547E+15      -0.406      1.6599E+4
O+H2=H+OH          0.508E+05       2.67        0.629E+04
H2+OH=H2O+H        0.216E+09       1.51        0.343E+04
O+H2O=OH+OH        2.97E+06        2.02        1.340E+04
H2+M=H+H+M         4.577E+19      -1.40        1.0438E+05
  H2/2.5/ H2O/12/
O+O+M=O2+M         6.165E+15      -0.50        0.000E+00
H2/2.5/ H2O/12/
O+H+M=OH+M         4.714E+18      -1.00        0.000E+00
  H2/2.5/ H2O/12/
H+OH+M=H2O+M       3.800E+22      -2.00        0.000E+00
  H2/2.5/ H2O/12/
H+O2(+M)=HO2(+M)   1.475E+12       0.60        0.000E+00
  LOW/6.366E+20 -1.72 5.248E+02/
  TROE/0.8 1E-30 1E+30 1.0000E+100 /
  H2/2.0/ H2O/11./ O2/0.78/
HO2+H=H2+O2        1.66E+13        0.00        0.823E+03
HO2+H=OH+OH         7.079E+13       0.00        2.95E+02
HO2+O=O2+OH         0.325E+14       0.00        0.00E+00
HO2+OH=H2O+O2       2.890E+13       0.00       -4.970E+02
HO2+HO2=H2O2+O2    4.200E+14       0.00        1.1982E+04
```

```

DUPLICATE
HO2+HO2=H2O2+O2          1.300E+11      0.00      -1.6293E+03
DUPLICATE
H2O2(+M)=OH+OH(+M)        2.951E+14      0.00      4.843E+04
LOW/1.202E+17 0.00 4.55E+04/
TROE/0.5 1E-30 1E+30 1.0000E+100 /
H2/2.5/ H2O/12/
H2O2+H=H2O+OH             0.241E+14      0.00      0.397E+04
H2O2+H=HO2+H2             0.482E+14      0.00      0.795E+04
H2O2+O=OH+HO2             9.550E+06      2.00      3.970E+03
H2O2+OH=HO2+H2O          1.000E+12      0.00      0.000
DUPLICATE
H2O2+OH=HO2+H2O          5.800E+14      0.00      9.557E+03
DUPLICATE
END

```

## B.2 Curran et al.

Die Wasserstoff/Luft Verbrennung nach dem Reaktionsmechanismus nach Curran et al. [9] ist im Folgenden aufgeführt.

```

ELEMENTS H O N END
SPECIES H H2 O O2 OH H2O N2 HO H2O2 END
REACTIONS          CAL/MOLE
H+O2 = O+OH        1.915E+14      0.00      1.644E+04
REV /              5.481E+11      0.39      -2.930E+02 /
O+H2 = H+OH        5.080E+04      2.67      6.292E+03
REV /              2.667E+04      2.65      4.880E+03 /
OH+H2 = H+H2O      2.160E+08      1.51      3.430E+03
REV /              2.298E+09      1.40      1.832E+04 /
O+H2O = OH+OH      2.970E+06      2.02      1.340E+04
REV /              1.465E+05      2.11      -2.904E+03 /
H2+M = H+H+M       4.577E+19      -1.40      1.044E+05
REV /              1.146E+20      -1.68      8.200E+02 /
H2/2.5/ H2O/12.0/
O2+M = O+O+M       4.515E+17      -0.64      1.189E+05
REV /              6.165E+15      -0.50      0.000E+00 /
H2/2.5/ H2O/12.0/
OH+M = O+H+M       9.880E+17      -0.74      1.021E+05
REV /              4.714E+18      -1.00      0.000E+00 /
H2/2.5/ H2O/12.0/
H2O+M = H+OH+M     1.912E+23      -1.83      1.185E+05
REV /              4.500E+22      -2.00      0.000E+00 /
H2/0.73/ H2O/12.0/

```

H+O2(+M) = HO2(+M)	1.475E+12	0.60	0.000E+00
LOW / 3.4820E+16 -4.1100E-01 -1.1150E+03 /			
TROE/ 0.5 1.0000E-30 1.0000E+30 1.0000E+100 /			
H2/1.3/ H2O/14.0/			
HO2+H = H2+O2	1.660E+13	0.00	8.230E+02
REV / 3.164E+12 0.35 5.551E+04 /			
HO2+H = OH+OH	7.079E+13	0.00	2.950E+02
REV /	2.027E+10	0.72	3.684E+04 /
HO2+O = OH+O2	3.250E+13	0.00	0.000E+00
REV / 3.252E+12 0.33 5.328E+04 /			
HO2+OH = H2O+O2	2.890E+13	0.00	-4.970E+02
REV /	5.861E+13	0.24	6.908E+04 /
H2O2+O2 = HO2+HO2	4.634E+16	-0.35	5.067E+04
REV /	4.200E+14	0.00	1.198E+04 /
DUPLICATE			
H2O2+O2 = HO2+HO2	1.434E+13	-0.35	3.706E+04
REV /	1.300E+11	0.00	-1.629E+03 /
DUPLICATE			
H2O2(+M) = OH+OH(+M)	2.951E+14	0.00	4.843E+04
LOW / 1.202E+17 0.00 45500. /			
TROE /0.5 1.0E-30 1.0E+30 1.0E+100/			
H2/2.5/ H2O/12.0/			
H2O2+H = H2O+OH	2.410E+13	0.00	3.970E+03
REV /	1.269E+08	1.31	7.141E+04 /
H2O2+H = H2+HO2	6.025E+13	0.00	7.950E+03
REV /	1.041E+11	0.70	2.395E+04 /
H2O2+O = OH+HO2	9.550E+06	2.00	3.970E+03
REV /	8.660E+03	2.68	1.856E+04 /
H2O2+OH = H2O+HO2	1.000E+12	0.00	0.000E+00
REV /	1.838E+10	0.59	3.089E+04 /
DUPLICATE			
H2O2+OH = H2O+HO2	5.800E+14	0.00	9.557E+03
REV /	1.066E+13	0.59	4.045E+04 /
DUPLICATE			
END			

## B.3 Maas und Pope

Im Folgenden dargestellt ist der Mechanismus für die Wasserstoff/Luft Verbrennung nach Maas und Pope [32].

```

ELEMENTS H O N END
SPECIES O2 H OH O H2 H2O HO2 H2O2 N2 END
REACTIONS          CAL/MOL
O2+H=>OH+O        2.00E+14      0.00      16790.88
OH+O=>O2+H        1.47E+13      0.00      501.58
H2+O=>OH+H        5.06E+04      2.67      6281.65
OH+H=>H2+O        2.24E+04      2.67      4394.77
H2+OH=>H2O+H      1.00E+08      1.60      3296.07
H2O+H=>H2+OH      4.46E+08      1.60      18415.03
OH+OH=>H2O+O      1.50E+09      1.14      95.54
H2O+O=>OH+OH      1.51E+10      1.14      17101.37
H+H+M=>H2+M      1.80E+18     -1.00      0.00
H2+M=>H+H+M      6.98E+18     -1.00     104136.86
H+OH+M=>H2O+M    2.20E+22     -2.00      0.00
H2O+M=>H+OH+M    3.80E+23     -2.00     119279.69
O+O+M=>O2+M      2.90E+17     -1.00      0.00
O2+M=>O+O+M      6.78E+18     -1.00     118563.14
H+O2+M=>HO2+M    2.30E+18     -0.80      0.00
HO2+M=>H+O2+M    2.66E+18     -0.80     49250.04
HO2+H=>OH+OH      1.50E+14      0.00      1003.15
OH+OH=>HO2+H      1.63E+13      0.00     37737.67
HO2+H=>H2+O2      2.50E+13      0.00      692.65
H2+O2=>HO2+H      8.39E+13      0.00     55603.35
HO2+H=>H2O+O      3.00E+13      0.00      1719.69
H2O+O=>HO2+H      3.29E+13      0.00     55460.04
HO2+O=>OH+O2      1.80E+13      0.00     -406.04
OH+O2=>HO2+O      2.67E+13      0.00     52617.77
HO2+OH=>H2O+O2    6.00E+13      0.00      0.00
H2O+O2=>HO2+OH    8.97E+14      0.00     70029.65
HO2+HO2=>H2O2+O2  2.50E+11      0.00     -1242.00
OH+OH+M=>H2O2+M  3.25E+22     -2.00      0.00
H2O2+M=>OH+OH+M  2.11E+24     -2.00     49393.35
H2O2+H=>H2+HO2    1.70E+12      0.00      3749.88
H2+HO2=>H2O2+H    9.35E+11      0.00     21782.75
H2O2+H=>H2O+OH    1.00E+13      0.00      3582.69
H2O+OH=>H2O2+H    2.66E+12      0.00     73469.03
H2O2+O=>OH+HO2    2.80E+13      0.00      6401.07
OH+HO2=>H2O2+O    6.80E+12      0.00     22547.06
H2O2+OH=>H2O+HO2  5.40E+12      0.00      1003.15
H2O+HO2=>H2O2+OH  1.32E+13      0.00     34154.98
END

```

## B.4 Kee et al.

Alle in dieser Arbeit durchgeführten Rechnungen mit Methan/Luft Flammen sind mit dem folgenden Mechanismus nach Kee et al. [27] durchgeführt worden, der zusätzlich um angeregte OH\*-Speziesmoleküle erweitert wurde (B.5).

```

ELEMENTS H O C N END
SPECIES H2 O2 H2O H O OH HO2 CO CO2 H2O2 CH4 CH3 CH2 CH CH2O HCO N2 END
REACTIONS          CAL/MOL
CH3+H+M=CH4+M    8.00E26      -3.000      0
CH4+O2=CH3+HO2   7.90E13      0.000     56000
CH4+H=CH3+H2     2.20E04      3.000     8750
CH4+O=CH3+OH     1.60E06      2.360     7400
CH4+OH=CH3+H2O  1.60E06      2.100     2460
CH3+O=CH2O+H    6.80E13      0.000      0
CH3+OH=CH2O+H2  1.00E12      0.000      0
CH3+OH=CH2+H2O  1.50E13      0.000     5000
CH3+H=CH2+H2    9.00E13      0.000    15100
CH2+H=CH+H2     1.40E19     -2.000      0
CH2+OH=CH2O+H   2.50E13      0.000      0
CH2+OH=CH+H2O   4.50E13      0.000     3000
CH+O2=HCO+O     3.30E13      0.000      0
CH+O=CO+H       5.70E13      0.000      0
CH+OH=HCO+H     3.00E13      0.000      0
CH+CO2=HCO+CO   3.40E12      0.000     690
CH2+CO2=CH2O+CO 1.10E11      0.000    1000
CH2+O=CO+H+H    3.00E13      0.000      0
CH2+O=CO+H2     5.00E13      0.000      0
CH2+O2=CO2+H+H  1.60E12      0.000    1000
CH2+O2=CH2O+O   5.00E13      0.000    9000
CH2+O2=CO2+H2   6.90E11      0.000     500
CH2+O2=CO+H2O   1.90E10      0.000   -1000
CH2+O2=CO+OH+H  8.60E10      0.000    -500
CH2+O2=HCO+OH   4.30E10      0.000    -500
CH2O+OH=HCO+H2O 3.43E09      1.180    -447
CH2O+H=HCO+H2   2.19E08      1.770    3000
CH2O+M=HCO+H+M  3.31E16      0.000   81000
CH2O+O=HCO+OH   1.81E13      0.000    3082
HCO+OH=CO+H2O   5.00E12      0.000      0
HCO+M=H+CO+M    1.60E14      0.000   14700
HCO+H=CO+H2     4.00E13      0.000      0
HCO+O=CO2+H     1.00E13      0.000      0
HCO+O2=HO2+CO   3.30E13     -0.400      0
CO+O+M=CO2+M    3.20E13      0.000   -4200
CO+OH=CO2+H     1.51E07      1.300    -758
CO+O2=CO2+O     1.60E13      0.000   41000

```

HO2+CO=CO2+OH	5.80E13	0.000	22934
H2+O2=2OH	1.70E13	0.000	47780
OH+H2=H2O+H	1.17E09	1.300	3626
H+O2=OH+O	5.13E16	-0.816	16507
O+H2=OH+H	1.80E10	1.000	8826
H+O2+M=HO2+M	3.61E17	-0.720	0
H2O/18.6/ CO2/4.2/ H2/2.86/ CO/2.11/ N2/1.26/			
OH+HO2=H2O+O2	7.50E12	0.000	0
H+HO2=2OH	1.40E14	0.000	1073
O+HO2=O2+OH	1.40E13	0.000	1073
2OH=O+H2O	6.00E08	1.300	0
H+H+M=H2+M	1.00E18	-1.000	0
H+H+H2=H2+H2	9.20E16	-0.600	0
H+H+H2O=H2+H2O	6.00E19	-1.250	0
H+H+CO2=H2+CO2	5.49E20	-2.000	0
H+OH+M=H2O+M	1.60E22	-2.000	0
H2O/5.0/			
H+O+M=OH+M	6.20E16	-0.600	0
H2O/5.0/			
H+HO2=H2+O2	1.25E13	0.000	0
HO2+HO2=H2O2+O2	2.00E12	0.000	0
H2O2+M=OH+OH+M	1.30E17	0.000	45500
H2O2+H=HO2+H2	1.60E12	0.000	3800
H2O2+OH=H2O+HO2	1.00E13	0.000	1800
END			

## B.5 Kathrotia et al.

Der Mechanismus nach Kathrotia et al. [24] beschreibt die Kettenreaktionen angeregter Hydroxyl-Radikale (OH\*). Da die Konzentration von OH\*-Moleküle aufgrund ihrer Chemilumineszenz leicht messbar ist, kann sie im Experiment überprüft werden.

REACTIONS	KJOULES/MOLE MOLES		
H+O+M=OH*+M	1.500E+13	0.00	25.000
H2 /1.0 /H2O /6.5 /O2 /0.4 /AR /0.35 /N2 /0.40 /CO /0.75 /CO2 /1.5 /CH4 /0.66 /			
CH+O2=CO+OH*	1.800E+11	0.0	0.0
OH*=OH	1.450E+06	0.0	0.0
OH*+O2=OH+O2	2.100E+12	0.5	-2.02
OH*+H2O=OH+H2O	5.930E+12	0.5	-3.61
OH*+H2=OH+H2	2.950E+12	0.5	-1.86
OH*+CO2=OH+CO2	2.760E+12	0.5	-4.06
OH*+CO=OH+CO	3.230E+12	0.5	-3.30
OH*+CH4=OH+CH4	3.360E+12	0.5	-2.66
OH*+OH=OH+OH	6.010E+12	0.5	-3.19

---

OH*+H=OH+H	1.310E+13	0.5	-0.7
OH*+AR=OH+AR	1.690E+12	0.0	17.32
END			

# Literaturverzeichnis

- [1] P. Atkins; J. Paula. *Atkins' Physical Chemistry*. Oxford University Press, Achte Edition, 2006.
- [2] T. Baust. Druck- und Turbulenzeinfluss auf den Marksteinzahl-Effekt einer instationären sphärischen Flammenfront. Diplomarbeit, Karlsruher Institut für Technologie, 2013.
- [3] Beowulf. <http://www.beowulf.org/>. [Online; besucht 26. Oktober 2013].
- [4] H. Bonart. Implementation and Validation of a Solver for Direct Numerical Simulations of Turbulent Reacting Flows in OpenFOAM. Bachelorarbeit, Karlsruher Institut für Technologie, 2012.
- [5] H. Bonart; F. Zhang; P. Habisreuther; H. Bockhorn. Development of an OpenFOAM based DNS solver for massive parallel computing of chemically reacting flows on complex geometries. *Computers and Fluids*, 2013. eingereicht.
- [6] D. Bulka; D. Mayhew. *Efficient C++ Performance Programming Techniques*. Addison-Wesley, 1999.
- [7] *CHEMKIN - A Software package for the analysis of gas-phase chemical and plasma kinetics*, 2000.
- [8] CHEMKIN. <http://www.reactiondesign.com/products/chemkin/>. [Online; besucht 26. Oktober 2013].

- [9] M. O. Connaire; H. J. Curran; J. M. Simmie; W. J. Pitz; C. K. Westbrook. A comprehensive modeling study of hydrogen oxidation. *International Journal of Chemical Kinetics*, 2004.
- [10] CRAY XE6 (HERMIT). <http://www.hlrs.de/systems/platforms/cray-xe6-hermit/>. [Online; besucht 30. Oktober 2013].
- [11] Doxygen. <http://www.stack.nl/~dimitri/doxygen/>. [Online; besucht 26. Oktober 2013].
- [12] ExxonMobil. *Energieprognose Deutschland 2013-2040*, 2013.
- [13] J. H. Ferziger; M. Perić. *Computational Methods for Fluid Dynamics*. Springer, Dritte Edition, 2002.
- [14] A. Fog. *Instruction tables: Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel, AMD and VIA CPUs*. Copenhagen University College of Engineering, 2012.
- [15] A. Fog. *Optimizing software in C++*. Copenhagen University College of Engineering, 2012.
- [16] E. Gamma; R. Helm; R. Johnson; J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1994.
- [17] P. Gerlinger. *Numerische Verbrennungssimulation: Effiziente numerische Simulation turbulenter Verbrennung*. Springer, 2005.
- [18] R. G. Gilbert; K. Luther; J. Troe. *Ber. Bunsenges. Phys. Chem.*, 1983. 87: 169.
- [19] D. Goodwin. Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics and transport processes. <http://code.google.com/p/cantera/>. [Online; besucht 26. Oktober 2013].
- [20] D. G. Goodwin. *Cantera C++ User's Guide*. California Institute of Technology, 2002.

- 
- [21] A. C. Hindmarsh; R. Serban. *User Documentation for CVODE v2.7.0*. Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2012.
- [22] Institute for Advanced Simulation (IAS) and Jülich Supercomputing Centre (JSC). JUQUEEN - Jülich Blue Gene/Q. [http://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/JUQUEEN/JUQUEEN\\_node.html](http://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/JUQUEEN/JUQUEEN_node.html). [Online; besucht 26. Oktober 2013].
- [23] Intel. *Intel 64 and IA-32 Architectures Optimization Reference Manual*, 2012.
- [24] T. Kathrotia; U. Riedel; A. Seipel; K. Moshhammer; A. Brockhinke. Experimental and numerical study of chemiluminescent species in low-pressure flames. *Applied Physics B*, 107, 2012.
- [25] KCachegrind - call graph viewer. <http://kcachegrind.sourceforge.net/html/Home.html>. [Online; besucht 26. Oktober 2013].
- [26] R. J. Kee; M. E. Coltrin; P. Glarborg. *Chemically Reacting Flow*. Wiley-Interscience, 2003.
- [27] R. J. Kee; J. F. Gracr; M. D. Smooke; J. A. Miller. *SAND Report 85-8240*, 1985.
- [28] M. Klein; H. J. M. Hanley; F. J. Smith; P. Holland. *Tables of Collision Integrals and Second Virial Coefficients for the (m,6,8) Intermolecular Potential Function*. U.S. Department of Commerce, 1974.
- [29] K. K. Kuo. *Principles of combustion*. John Wiley and Sons, Inc., Zweite Edition, 2005.
- [30] J. Li; Z. Zhao; A. Kazakov; F. L. Dryer. An updated comprehensive kinetic model of hydrogen combustion. *International Journal of Chemical Kinetics*, 2004.
- [31] S. B. Lippman. *Inside the C++ Object Model*. Addison-Wesley Professional, 1996.
- [32] U. Maas; S. B. Pope. Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space. *Comb. and Flame*, 1992.

- 
- [33] T. R. Marerro; E. A. Mason. *J. Phys. Chem. Ref. Data*, 1, 1972.
- [34] E. A. Mason; L. Monchick. Transport properties of polar gases. *J. Chem. Phys.*, 1961.
- [35] B. J. McBride; S. Gordon; M. A. Reno. *Coefficients for Calculating Thermodynamic and Transport Properties of Individual Species*. National Aeronautics and Space Administration, 1993. NASA Technical Memorandum 4513.
- [36] N. Nethercote; R. Walsh; J. Fitzhardinge. IEEE international symposium on workload characterization, 2006.
- [37] Open-FOAM Foundation. *OpenFOAM: The Open Source CFD Toolbox, User Guide 2.1.1*, 2012.
- [38] OpenFOAM: The Open Source CFD Toolbox. <http://www.openfoam.com/>. [Online; besucht 26. Oktober 2013].
- [39] Open MPI: Open source high performance computing. <http://www.open-mpi.de/>. [Online; besucht 26. Oktober 2013].
- [40] N. Peters. *Turbulent Combustion*. Cambridge monographs on mechanics. Cambridge University Press, 2004.
- [41] T. Poinsot; D. Veynante. *Theoretical and numerical combustion*. R.T. Edwards, Inc., Zweite Edition, 2005.
- [42] B. E. Poling; J. M. Prausnitz; J. P. O'Connell. *The Properties of Gases and Liquids*. The McGraw-Hill Companies, Fünfte Edition, 2004.
- [43] SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. <http://computation.llnl.gov/casc/sundials/main.html>. [Online; besucht 26. Oktober 2013].
- [44] ThermoPhase class reference. [http://stuff.mit.edu/afs/athena/course/2/2.62/www/cantera/doc/classCantera\\_1\\_1ThermoPhase.html](http://stuff.mit.edu/afs/athena/course/2/2.62/www/cantera/doc/classCantera_1_1ThermoPhase.html). [Online; besucht 26. Oktober 2013].

- 
- [45] ThermoPhase.cpp file reference. [http://stuff.mit.edu/afs/athena/course/2/2.62/www/cantera/doc/ThermoPhase\\_8cpp.html](http://stuff.mit.edu/afs/athena/course/2/2.62/www/cantera/doc/ThermoPhase_8cpp.html). [Online; besucht 26. Oktober 2013].
- [46] Top500 list - June 2013. <http://www.top500.org/list/2013/06/>. [Online; besucht 26. Oktober 2013].
- [47] Information über den Linux-Cluster an der Verbrennungstechnik. [http://vbt.ebi.kit.edu/index.pl/Haupt\\_Menu\\_Verschiedenes\\_M35/clusterInfo/](http://vbt.ebi.kit.edu/index.pl/Haupt_Menu_Verschiedenes_M35/clusterInfo/). [Online; besucht 26. Oktober 2013].
- [48] J. Warnatz; U. Maas; R. W. Dibble. *Combustion: Physical and Chemical Fundamentals, Modeling and Simulation, Experiments, Pollutant Formation*. Springer, Vierte Edition, 2006.
- [49] A. Williams. *C++ Concurrency in Action - Practical Multithreading*. Manning Publications Co., 2012.