

Diskontinuierliche Galerkin-Verfahren für die operationelle Wettervorhersage

Dissertation
zur Erlangung des Grades

„Doktor
der Naturwissenschaften“

am Fachbereich Physik, Mathematik und Informatik
der Johannes Gutenberg-Universität
in Mainz

Dieter Schuster
geb. in Kleinkopisch

Mainz, den 10. Juli 2014

Dekan:

1. Gutachter: *Die Namen des Dekans und der Gutachter wurde zum Schutze der*
2. Gutachter: *Daten in der elektronischen Fassung entfernt.*

Tag der mündlichen Prüfung: 7. November 2014

D77 - Mainzer Dissertationen

An die Freiheit

Ich habe die jetzt als Dissertation vorgelegte Arbeit selbständig verfasst. Es wurden ausschließlich die angegebenen Quellen und Hilfsmittel verwendet. Von der Ordnung zur Sicherung guter wissenschaftlicher Praxis in Forschung und Lehre und vom Verfahren zum Umgang mit wissenschaftlichem Fehlverhalten habe ich Kenntnis genommen.

(Dieter Schuster)

Kurzfassung

In dieser Arbeit wird ein neuer Dynamikkern entwickelt und in das bestehende numerische Wettervorhersagesystem COSMO integriert. Für die räumliche Diskretisierung werden diskontinuierliche Galerkin-Verfahren (DG-Verfahren) verwendet, für die zeitliche Runge-Kutta-Verfahren. Hierdurch ist ein Verfahren hoher Ordnung einfach zu realisieren und es sind lokale Erhaltungseigenschaften der prognostischen Variablen gegeben. Der hier entwickelte Dynamikkern verwendet geländefolgende Koordinaten in Erhaltungsform für die Orographiemodellierung und koppelt das DG-Verfahren mit einem Kessler-Schema für warmen Niederschlag. Dabei wird die Fallgeschwindigkeit des Regens, nicht wie üblich implizit im Kessler-Schema diskretisiert, sondern explizit im Dynamikkern. Hierdurch sind die Zeitschritte der Parametrisierung für die Phasenumwandlung des Wassers und für die Dynamik vollständig entkoppelt, wodurch auch sehr große Zeitschritte für die Parametrisierung verwendet werden können. Die Kopplung ist sowohl für Operatoraufteilung, als auch für Prozessaufteilung realisiert.

Anhand idealisierter Testfälle werden die Konvergenz und die globalen Erhaltungseigenschaften des neu entwickelten Dynamikkerns validiert. Die Masse wird bis auf Maschinengenauigkeit global erhalten. Mittels Bergüberströmungen wird die Orographiemodellierung validiert. Die verwendete Kombination aus DG-Verfahren und geländefolgenden Koordinaten ermöglicht die Behandlung von steileren Bergen, als dies mit dem auf Finite-Differenzenverfahren-basierenden Dynamikkern von COSMO möglich ist. Es wird gezeigt, wann die volle Tensorproduktbasis und wann die Minimalbasis vorteilhaft ist. Die Größe des Einflusses auf das Simulationsergebnis der Verfahrensordnung, des Parametrisierungszeitschritts und der Aufteilungsstrategie wird untersucht. Zuletzt wird gezeigt dass bei gleichem Zeitschritt die DG-Verfahren aufgrund der besseren Skalierbarkeit in der Laufzeit konkurrenzfähig zu Finite-Differenzenverfahren sind.

Abstract

In this work a new dynamical core is developed and integrated in the existing numerical weather prediction system COSMO. The spatial discretisation uses discontinuous Galerkin methods (DG methods) and the temporal integration uses Runge-Kutta methods. Due to this, it is easy to realise a high order method and to have local conservation properties for the prognostic variables. The developed dynamical core uses terrain following coordinates in conservation form for the modelling of the orography. The DG method is coupled with a Kessler scheme for warm precipitation. Here, the sedimentation velocity of the rain is explicitly discretised in the dynamical core, not as usual implicitly in the Kessler scheme. Due to this, the time steps of the dynamics and of the parametrisation for the phase change of water are fully decoupled, which results in the possibility to use very long time steps for the parametrisations. The coupling is realised by operator splitting or process splitting.

With the help of idealised test cases the convergence and the global conservation properties of the new developed core are validated. The mass is conserved globally down to machine precision. The orography is validated by mountain overflows. The used combination of DG methods and terrain following coordinates makes it possible to deal with steeper mountains as with the finite-difference method based dynamical core of COSMO. It is shown in which case the full tensor product basis are advantageous compared to the minimal basis. The size of the influence on the simulation result of the order of the method, the time step of the parametrisation and the two coupling strategies is studied. Finally it is shown, that when the same time step is used, the DG method is competitive to the finite-difference method due to a better scalability.

Danksagung

An dieser Stelle möchte ich mich bei den Menschen bedanken, die mich während der Erstellung dieser Arbeit unterstützt haben.

Der persönliche Teil der Danksagung wurde zum Schutze der persönlichen Daten, insbesondere der Namen, in der elektronischen Fassung entfernt.

Diese Arbeit ist im Rahmen des DFG-Schwerpunktprogramms „MetStröm“ entstanden. An dieser Stelle danke ich der DFG für die Finanzierung und dem Deutschen Wetterdienst für die Bereitstellung der Infrastruktur und des Büros.

Inhaltsverzeichnis

1. Einleitung	17
1.1. Stand der Wissenschaft der diskontinuierlichen Galerkin-Verfahren	19
1.2. COSMO und dessen Dynamikkern	23
1.3. Ziele und Beiträge dieser Arbeit	25
2. Bilanzgleichungen des physikalischen Modells	27
2.1. Physikalisches Modell	27
2.1.1. Kessler-Schema und Sättigungsadjustierung	31
2.2. Referenzzustand	35
2.3. Bilanzgleichungen in geländefolgenden Koordinaten	36
3. Diskontinuierliche Galerkin-Verfahren	41
3.1. Modales diskontinuierliches Galerkin-Verfahren	43
3.2. Lokales DG für Viskosität	46
3.3. Zeitintegration	48
3.4. Randbedingungen	50
3.5. Anfangsbedingungen	52
3.6. Einzelheiten zur Implementierung	54
4. Numerische Ergebnisse für die trockene Atmosphäre	57
4.1. Schwerewellentestfall von Baldauf/Brdar 2013	58
4.1.1. Konvergenzanalyse	60
4.1.2. Erhaltungseigenschaften	62
4.1.3. Vergleich des vollen Tensorprodukts mit der Minimalbasis	64
4.2. Ruhende hydrostatische Atmosphäre und Wohlbalanciertheitsproblem	66
4.3. Fallende Kälteblase nach Straka et. al. 1993	67
4.4. Bergüberströmungen	71
4.5. Berg in ruhender Atmosphäre	74
5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit	77
5.1. Feuchtkonvektion nach Weisman und Klemp 1982	77
5.2. Skalierbarkeit	88
6. Zusammenfassung und Ausblick	93
6.1. Zusammenfassung	93
6.2. Ausblick	94
A. Mathematischer Anhang	97
A.1. Begriffe und Notation	97
A.2. Allgemeine Koordinaten	97
A.3. Legendre-Polynome	98
A.4. Skalierbarkeit, Effizienz	99
Literatur	101

Symbolverzeichnis

\mathbf{x}	globale Koordinaten, bei 2d: $\mathbf{x} = (x, z)$ oder $\mathbf{x} = (x, \zeta)$, bei 3d: $\mathbf{x} = (x, y, z)$ oder $\mathbf{x} = (x, y, \zeta)$
i, j, k	Zähler in x -, y -, z -Richtung
l	Zähler der Basisfunktionen
r	sonstiger Zähler
u, v	horizontaler Wind in Richtung der x und y -Achse
w	vertikaler Wind in Richtung der z - oder der ζ -Achse
\mathbf{u}	Windvektor, bei 2d: $\mathbf{u} = (u, w)^\top$, bei 3d: $\mathbf{u} = (u, v, w)^\top$
θ, θ_m	potentielle Temperatur und feucht potentielle Temperatur
p	Druck
ρ	Massendichte
ρ_v, q_v	Partialdichte des Wasserdampfs und spezifische Feuchte
ρ_c, q_c	Partialdichte des Wolkenwassers und spezifischer Wolkenwassergehalt
ρ_r, q_r	Partialdichte des Regenwassers und spezifischer Regenwassergehalt
R_d, R_m	spezifische Gaskonstante für trockene und feuchte Luft
\mathbf{q}	Vektor der prognostischen Variablen
\mathbf{F}	Vektor der Flüsse der prognostischen Variablen
\mathbf{S}	Vektor der Quellterme
ϕ_l	l -te Basisfunktion des Ansatz- und Testraums V_h
Ω	Berechnungsgebiets
Ω_i, Ω_{ijk}	Gitterelement
Γ_i, Γ_{ijk}	Randflächen/-kanten eines Gitterelements
D	Dimension des physikalischen Raums
d	Dimension von V_h , d. h. Anzahl der Basisfunktionen
κ	maximaler Polynomgrad der Basisfunktionen
V_h	Ansatz- und Testraum
\mathbb{R}	Menge der Reellen Zahlen
\mathbf{J}_k	Transformationsmatrix vom Referenzelement auf ein Gitterelement
J_k	Jakobi-Determinante
$\Delta t, \Delta \tau$	Dynamik- und Physikzeitschritt
σ	Verhältnis des Physikzeitschritts zum Dynamikzeitschritt

1. Einleitung

Für den Betrieb eines numerischen Wettervorhersagesystems (NVW-Systems) muss die Atmosphäre vermessen werden können und es ist Wissen über die theoretische Meteorologie nötig, insbesondere über die atmosphärische Dynamik, Thermodynamik und weitere physikalische Prozesse. Hierfür werden Näherungen an die physikalischen Bilanzgleichungen bestimmt, die alle wesentlichen Prozesse modellieren. Sie werden aufgeteilt in einen Teil der die Strömung simuliert; dieser Teil des Vorhersagesystems heißt *dynamischer Kern* oder *Dynamikkern*. Diese Terme werden auf einer vorab gewählten räumlichen und zeitlichen Skala numerisch integriert. Die subskalige Dynamik und physikalische Prozesse, wie etwa die Phasenumwandlung von Wasser, werden durch *Parametrisierungen* berechnet, die an die skalige Dynamik gekoppelt werden, indem die prognostischen Variablen als Eingabegrößen verwenden werden und die Ergebnisse der Parametrisierungen in die weitere Integration der Dynamik wieder einfließt. Um mit einem Vorhersagesystem eine Wetterprognose erstellen zu können müssen Messwerte aus Beobachtungen in das System *assimiliert* werden. Eine Einführung in die mesoskalige Modellierung und hierfür geeignete numerische Verfahren gibt etwa [68]. Die Qualität der Wettervorhersage kann folglich weiter verbessert werden durch: bessere Parametrisierungen, eine verbesserte Datenassimilation oder durch einen verbesserten Dynamikkern. Die Gittermaschenweite des Atmosphärenmodells ist überwiegend durch die zur Verfügung stehende Rechenzeit bestimmt. Gewünscht ist die Wettervorhersage auf immer kleineren Skalen, um feinere Strömungsstrukturen auflösen zu können, wofür rechenstärkere Großrechner benötigt werden. Die Steigerung der Qualität der NVW-Systeme ist also auch von der Entwicklung der Großrechner abhängig.

Die Dynamikkerne bisheriger operationeller NVW-Systeme lassen sich im Wesentlichen in zwei Klassen einteilen:

spektrale Verfahren. Hierzu gehören etwa die Globalmodelle „IFS“ [49] des Europäischen Zentrums für Mittelfristvorhersagen (EZMW), „GFS“ (Information gibt es unter: <http://www.emc.ncep.noaa.gov/gmb/moorthi/gam.html>) der US-amerikanischen National Oceanic and Atmospheric Administration (NOAA), sowie das Regionalmodell „ALADIN“ und dessen Nachfolger „AROME“ des ALADIN-Konsortiums (<http://www.cnrm.meteo.fr/aladin/>) um Meteo France. Diese Modelle verwenden in der Horizontalen einen spektralen Ansatz, und Finite Differenzen in der Vertikalen. Die zeitliche Diskretisierung erfolgt über ein semi-Lagrange'sches und semi-implizites Verfahren. Für die Berechnung der Parametrisierungen wird auf ein Punktgitter transformiert.

Finite-Differenz- (FD) und Finite-Volumenverfahren (FV). Hierzu zählen das Globalmodell „GME“ des Deutschen Wetterdienstes (DWD). Auch das neue Modell „ICON“ des DWD, das bei Fertigstellung dieser Arbeit für den operationellen Betrieb vorbereitet wurde, zählt hierzu. Im Vergleich zum GME erhält es die Masse und den Impuls. Für die zeitliche Diskretisierung wird ein Prediktor-Korrektorverfahren verwendet, räumlich werden die Bilanzgleichungen vertikal implizit und horizontal explizit integriert. Auch das „Unified Model“ des UK MetOffice (<http://www.metoffice.gov.uk>) basiert auf FD- und FV-Verfahren. Sie verwenden jedoch ein semi-Lagrange'sches und semi-implizites Integrationsverfahren. Das Unified Modell ist massenerhaltend. In diese Klasse gehört auch das Regionalmodell „WRF“ [81] der US-amerikanischen Institute University Corporation for Atmospheric Research (UCAR), des National Center for

1. Einleitung

Atmospheric Research (NCAR) und des National Centers for Environmental Prediction (NCEP). Letztes betreibt WRF auch operationell. WRF verwendet die Gleichungen in Flussform und erhält die Masse und die trockene Entropie. Impulserhaltung ist im WRF-Modell nicht gegeben [80]. Das letzte Modell, das an dieser Stelle erwähnt werden soll, ist das Regionalmodell „COSMO“ (vgl. [33, 32, 73]) des COSMO-Konsortiums, das sich um den Deutschen Wetterdienst (DWD) gebildet hat. Dieses Modell ist ähnlich zu WRF, jedoch integriert es die Bilanzgleichungen in Advektionsform, und hat daher keine Erhaltungseigenschaften. COSMO wird im folgenden noch ausführlicher beschrieben.

Ein Übersichtsartikel über die zeitliche Entwicklung der Dynamikkerne globaler Modelle, einschließlich der verschiedenen Gitter ist etwa [92]. Um die Parametrisierungen und die Dynamik miteinander zu koppeln gibt es verschiedene Strategien. Besonders semi-Langrange'sche Verfahren, in denen ein sehr großer Zeitschritt verwendet wird, sind sensitiv gegenüber der verwendeten Strategie. In [83] und [84] stellen Staniforth, Wood und Côté ein Verfahren vor zum Vergleich von unterschiedlichen Physik-Dynamik-Kopplungsstrategien. Dieses arbeitet auf idealisierten einfachen Gleichungen, deren Terme eine physikalische Parametrisierung oder die Dynamik repräsentieren. Es werden verschiedene Kombinationen aus expliziter und impliziter Kopplung auf Genauigkeit, Stabilität und Effizienz untersucht. Dubal, Wood und Staniforth erweitern in [34] diese Untersuchungen auf Strategien für Kopplungen aus mehreren Parametrisierungen mit der Dynamik. Diese Strategien lassen sich in drei Kategorien einteilen: 1. Physik und Dynamik laufen hintereinander ab; 2. Physik und Dynamik laufen parallel zueinander ab; 3. ein Teil der Parametrisierungen und der Dynamik läuft hintereinander, der andere Teil parallel zueinander ab.

In vielen Euler'schen Gitterpunktmodellen, wie etwa COSMO, findet eine sehr einfache Kopplung statt. Die einzelnen Parametrisierungen und die Physik laufen hintereinander ab. Nur der Reihenfolge wird besondere Aufmerksamkeit geschenkt. Diese Strategie ist auch als *Operatoraufteilung* bekannt und ist besonders einfach zu implementieren. In [93] wird die Kopplung der physikalischen Parametrisierungen mit der Dynamik im Community Climate Model-3 für die beiden Strategien Operator- und *Prozessaufteilung*, d. h. Physik und Dynamik laufen parallel ab, verglichen. Der Unterschied in den Ergebnissen durch die Kopplungsstrategie ist für die Klimasimulation nur gering. In [91] wird der Einfluss des Zeitschritts auf die Parametrisierungen im Klimamodell CAM4 untersucht. Anlass war das Auftreten von numerisch induziertem Starkniederschlag, der auf den unterschiedlichen Zeitschritt von konkurrierenden Parametrisierungen zurückgeführt wird. Anhand von idealisierten Gleichungen wird verdeutlicht, dass ein Modell sensitiv von einer ungünstigen Wahl der Parametrisierungszeitschritte abhängen kann. Eine erweiterte Diskussion der Physik-Dynamik-Kopplung findet man etwa in [92].

Mit neuen Großrechnern ist zwar eine höhere Spitzenrechenleistung erreichbar, jedoch wächst die Rechenleistung einer einzelnen CPU¹ kaum noch weiter. Die höhere Spitzenrechenleistung wird überwiegend über eine größere Anzahl an CPUs erreicht. Dies liegt vor allem daran, dass die Leistungsaufnahme und die Abführung der Abwärme einer einzelnen CPU zu einem immer größeren Problem wird, d. h. es ist nicht zu erwarten, dass sich in naher Zukunft dieser Trend ändert. Um mit mehreren CPUs gleichzeitig eine Simulation zu berechnen, müssen zwischen den CPUs Daten ausgetauscht werden. Diese Kommunikation ist vergleichsweise langsam, insbesondere wenn die CPUs in verschiedenen Rechnerknoten verbaut sind und über ein Netzwerk miteinander kommunizieren müssen. Folglich ist diese Kommunikation der Flaschenhals für die real erreichbare Rechenleistung bei den meisten Anwendungen. Dies trifft insbesondere auf die numerische Wettervorhersage zu. Die Simula-

¹englische Abkürzung für „Central Processing Unit“; deutsch: Zentrale Recheneinheit. Also der Teil eines Prozessors in einem Rechner, der die eigentliche Arithmetik ausführt.

tionsprogramme müssen also numerische Verfahren verwenden, die wenig Datenaustausch zwischen den CPUs benötigen. Damit sind die spektralen Verfahren, die oft nicht nur mit den Nachbar-CPU's kommunizieren müssen, sondern auch global, auf kommenden Großrechnern kaum noch effizient einsetzbar. Aber auch die FD- und die FV-Verfahren stoßen an Skalierbarkeitsgrenzen, insbesondere wenn Verfahren höherer Ordnung eingesetzt werden.

Darüber hinaus ist es schwierig mit FD-Verfahren Erhaltungseigenschaften zu realisieren. FV-Verfahren sind hingegen erhaltend. Beide Verfahren haben Schwierigkeiten mit einer hohen Verfahrensordnung und unstrukturierten Gittern, insbesondere wenn diese dynamisch verfeinert werden. Daher wird die Abbildung der Orographie über geländefolgende Koordinaten (vgl. etwa [38]) realisiert, damit wieder auf einem kartesischen Gitter gerechnet werden kann. Diese Verfahren reduzieren jedoch die Genauigkeit der Lösung (vgl. Abschnitt 4.5 oder auch [40]).

Aufgrund dieser Umstände sind in den letzten Jahren Finite-Elementverfahren, stetige wie unstetige, in der Meteorologie populär geworden. So kann etwa das IFS-Modell seit November 2013 die vertikalen Terme auch mit einem auf kubischen B-Splines basierenden Finite-Elementverfahren integrieren (siehe Homepage). Insbesondere die *diskontinuierlichen Galerkin-Verfahren* haben das Potential die meisten der vorherigen angesprochenen Probleme zu beseitigen. Diese Verfahren erhalten momentan erhöhte Aufmerksamkeit in der Weiterentwicklung und sind auf einem guten Weg, den akademischen Bereich zu verlassen und in ein paar Jahren in der operationellen numerischen Wettervorhersage Einzug zu finden. Diese Arbeit soll hierzu einen Beitrag leisten, indem ein neuer Dynamikkern für COSMO entwickelt wird, der auf den diskontinuierlichen Galerkin-Verfahren (kurz: *DG-Verfahren*²) basiert. Mit ihnen erstellt man auf direktem Wege Systeme mit lokalen Erhaltungseigenschaften und einer hohen Verfahrensordnung. Unstrukturierte Gitter bereiten bei DG-Verfahren keine Probleme. Es kann sogar mit wenig Aufwand sowohl das Gitter als auch die Ordnung lokal adaptiert werden. Der Diskretisierungsskern bleibt auch bei hoher Ordnung kompakt [67, 20], dies begünstigt die gute Skalierbarkeit³ der DG-Verfahren. Der Nachteil der DG-Verfahren besteht in der starken Zeitschrittbeschränkung (vgl. Abschnitt 3.3) wodurch sie auf wenigen CPUs langsam sind. Verglichen mit Finite-Differenzenverfahren haben DG-Verfahren eine höhere Komplexität wodurch die Implementation komplizierter wird.

1.1. Stand der Wissenschaft der diskontinuierlichen Galerkin-Verfahren

Diskontinuierliche Galerkin-Verfahren gehören zu den Ansatzverfahren. Das zu lösende Gleichungssystem wird hierbei in eine integrale Form überführt, die elementweise gelöst wird. Hierbei kann auf eine stetige Lösung über die Elementgrenzen hinweg verzichtet werden. Dadurch bleibt die Massenmatrix des Verfahrens lokal, im Vergleich zu den stetigen Finiten-Elementverfahren, deren Massenmatrizen global aufgestellt werden müssen. Einen Überblick über DG-Verfahren kann man sich etwa in [25, 26, 47] verschaffen. DG-Verfahren können in unterschiedlichen Variationen implementiert werden. Dabei können die zu diskretisierenden Gleichungen in der sogenannten *starken Form* oder in der *schwachen Form* (siehe Kapitel 3) formuliert werden. Diese Gleichungen sind analytisch äquivalent, haben aber numerische Unterschiede [39].

Neben dieser Unterscheidung lassen sich DG-Verfahren durch die Wahl der Ansatzfunktionen klassifizieren. Diese Funktionen sind Polynome, die aus elementweise definierten

²im Sprachgebrauch hat sich der Begriff „DG“ sowohl auf deutsch, also auch auf englisch eingebürgert, daher wird an dieser Stelle der Begriff „diskontinuierlich“ anstelle von „unstetig“ verwendet, damit „DG“ als kanonische Abkürzung entsteht.

³Für eine Definition der Skalierbarkeit wird auf Anhang A.4 verwiesen.

1. Einleitung

Basispolynomen gebildet werden. Entstehen die Polynome durch Interpolation geeigneter Raumpunkte (etwa Lagrange-Polynome), so spricht man von *nodalen* DG-Verfahren. Bei *modalen* DG-Verfahren werden Polynome angesetzt, deren Koeffizienten keine direkte physikalische Interpretation besitzen (meistens Legendre-Polynome). Diese Klasse hat Vorzüge etwa bei den Erhaltungseigenschaften. Eine gute und praktische Einführung, mit Schwerpunkt auf den nodalen Verfahren und vielen technischen Einzelheiten, gibt [47]. Der Vergleich der beiden Formen wird auch in [39] diskutiert. Ein weiteres, die Verfahren prägendes Merkmal, ist der verwendete numerische Fluss, mit dem der Informationsaustausch zwischen den Elementen modelliert wird. Im Prinzip kann jeder aus dem Bereich der Finite-Volumenverfahren bekannte numerische Fluss verwendet werden. In [69] werden häufig verwendete numerische Flüsse im DG-Kontext miteinander in der Laufzeit und Genauigkeit verglichen.

Für die Behandlung von Ableitungen zweiter Ordnung, wie sie etwa bei den Navier-Stokes-Gleichungen auftreten, haben sich mehrere Varianten herausgebildet. Eine einfach zu implementierende Variante ist etwa das Lokale Diskontinuierliche Galerkin-Verfahren (LDG) [28], das auf einer Transformation der Differentialgleichungen höherer Ordnung auf ein System von Differentialgleichungen erster Ordnung basiert. Die so entstehenden zusätzlichen Gleichungen lassen sich aber durch geschicktes Umformen und Einsetzen wieder vermeiden. Dies wird etwa durch das Compact Discontinuous Galerkin-Verfahren (CDG) [67] ausgenutzt. Beim CDG-Verfahren ist der Diskretisierungsskalar auch für Gleichungen höherer Ordnung nur auf die direkten Nachbarelemente beschränkt. Brdar hat zusammen mit Dedner und Klöforn die effizientere Verbesserung CDG2 von CDG entwickelt [20].

DG-Verfahren werden in den Ingenieurwissenschaften schon seit längerem für Probleme der Gasdynamik entwickelt und haben dort auch schon eine gewisse Reife erlangt (siehe etwa das EU-Projekt ADIGMA des sechsten Rahmenprogramms [58]). In den Geowissenschaften, insbesondere in der Meteorologie, ist diese Verfahrensklasse noch relativ neu. Im Vergleich zu ingenieurwissenschaftlichen Fragestellungen spielen in der Meteorologie die hydrostatische Balance und die notwendigen Parametrisierungen eine wichtige Rolle. Für die Meteorologie relevante Arbeiten sollen hier kurz vorgestellt werden, ohne dass die folgende Aufzählung den Anspruch auf Vollständigkeit erhebt; sie dient nur dem Überblick über den Entwicklungsstand der DG-Verfahren.

Das „Discontinuous Galerkin Coastal Ocean Model“ (DGCOM) (siehe [1, 43] oder die Homepage <http://faculty.nps.edu/fgirald/projects/DGCOM>) verwendet ein nodales DG-Verfahren auf einem unstrukturierten Gitter. Es ist ein zweidimensionales Flachwassermodell (FW-Modell) für die Simulation küstennaher Ozeanprozesse. Der Tsunami vom 26. Dezember 2004 im indischen Ozean wurde mit der realen Bathymetrie modelliert [1], d. h. die Simulation ging über einen idealisierten Charakter hinaus. Anhand von Messungen der Gezeitenpegel lokaler Stationen und von Satellitendaten wird gezeigt, dass DGCOM die Ankunftszeiten der Tsunamiwellen gut reproduzieren kann. Auch Blasie und St-Cyr simulieren die FW-Gleichungen mit DG [13], verwenden jedoch ein Verfahren mit lokal adaptivem unstrukturiertem Gitter (h -adaptiv) und lokal adaptiver Ordnung (p -adaptiv) und validieren dieses mit dem Tsunami im Februar 2010 vor Chile, sowie mit einer globalen stationären nicht-linearen zonalen geostrophischen Strömung und einer zonalen Strömung über einen isolierten Berg. Giraldo, Hesthaven und Warburton diskretisieren die FW-Gleichungen mit nodalem DG [41]. Sie validieren ihr Modell neben den vorherig erwähnten beiden zonalen Strömungen mit weiteren Testfällen. Darüber hinaus vergleichen sie die starke mit der schwachen Form auf Genauigkeit und Effizienz. Nair, Thomas und Loft haben ein modales DG-Verfahren für die FW-Gleichungen, mit Massenerhaltung bis auf Maschinengenauigkeit und guter Energieerhaltung, entwickelt auf der „würfelförmigen Spähren“⁴-Gitterstruktur

⁴oft mit dem englischen Begriff „cubed sphere“ bezeichnet

[64].

Hall und Nair [46] simulieren mit einem nodalen DG-Verfahren die Advektionsgleichungen auf der Sphäre mit einem Yin-Yang-Gitter, das mit DG-Verfahren weniger Schwierigkeiten bereitet als mit FV-Verfahren. Sie prüfen auf Konvergenz und Massenerhaltung, die zwischen den beiden Teilgittern des Yin-Yang-Gitters verletzt wird.

Für die Euler- und Navier-Stokes-Gleichungen sollen folgende Arbeiten erwähnt werden. In [42] werden von Giraldo und Restelli DG-Verfahren für drei unterschiedliche Gleichungssysteme implementiert und getestet. Dabei werden folgende prognostischen Variablen verwendet: System 1: Exner-Druck, Impuls, und potentielle Temperatur; System 2: Dichte, Impuls und potentielle Temperatur; System 3: Dichte, Impuls und totale Energie. Das erste System ist in Advektionsform, also nicht erhaltend, die letzten beiden Systeme sind in Erhaltungsform formuliert. Diese drei Systeme werden durch nodale DG-Verfahren (und durch Spektralelemente-Verfahren) diskretisiert und die Lösungen werden anhand sieben idealisierter Testfälle auf Genauigkeit, Erhaltungseigenschaften und Effizienz verglichen. Unter diesen Testfällen sind die Schwerewellenausbreitung im Kanal nach Skamarock und Klemp [79], die fallende Kälteblase nach Straka et. al. [85] und Bergüberströmungen. System 3 ist das einzige energieerhaltende System. Damit auch die Diskretisierung energieerhaltend ist, muss das DG-Verfahren in der schwachen Form verwendet werden. System 3 ist aufgrund der Variablenwahl etwas effizienter als System 2. Restelli und Giraldo lösen in [72] die Navier-Stokes-Gleichungen für ein vertikal geschichtetes Medium mit nodalen DG-Verfahren. Für die Zeitintegration wird ein semi-implizites Runge-Kutta-Verfahren (RK-Verfahren) verwendet. Dadurch kann die Schwerewellenausbreitung von Skamarock und Klemp [79] mit CFL-Zahl 5,1, eine lineare-hydrostatische Bergüberströmung sogar mit der CFL-Zahl 18,6 integriert werden. Restelli stellt in [71] ein Modell vor, das eine räumliche DG-Diskretisierung mit einer zeitlichen semi-impliziten, semi-Lagrange'schen Diskretisierung verbindet. Für die lineare hydrostatische Bergüberströmung wird mit der CFL-Zahl 16 gerechnet, damit kommt das Verfahren in die gleiche Region wie das semi-implizite Verfahren in [72]. Für die Advektion sind die Fehler vergleichbar mit DG-Verfahren mit Euler'scher Runge-Kutta-Zeitintegration wenn die gleiche CFL-Zahl verwendet wird. In der Gruppe von Kröner [19, 20] werden DG-Verfahren auf theoretischer und praktischer Ebene weiterentwickelt und ebenfalls auf geophysikalische Probleme angewandt. Hier ist insbesondere die Arbeit von Brdar [18] zu erwähnen. In der Arbeit wird ein DG-Löser für atmosphärische Strömungen entwickelt. Dieser basiert auf der DUNE-Bibliothek [11, 10, 15, 14, 30, 31] und kann auf strukturierten, unstrukturierten und h -adaptiven Gittern rechnen. Verwendet wird ein modales DG-Verfahren, das explizite und semi-implizite RK-Verfahren zur Zeitintegration verwendet. Es werden mehrere trockene zweidimensionale Testfälle betrachtet, wie die lineare und nicht-lineare und hydrostatische und nicht-hydrostatische Bergüberströmung und die fallende Kälteblase nach Straka et. al. [85]. Darüber hinaus kann Wasserdampf und Wolkenwasser, jedoch kein Niederschlag simuliert werden. Für die Phasenumwandlung sind im DUNE-Löser analytische ad-hoc-Funktionen implementiert, die über einen freien Parameter justiert werden können. Anhand einer zweidimensionalen Feuchtkonvektion in einer Wasserdampf-gesättigten Atmosphäre nach [22] wird das Feuchte-Schema validiert. Die Bergüberströmung kann mit den semi-impliziten RK-Verfahren um den Faktor 2,4 beschleunigt werden im Vergleich zu den expliziten RK-Verfahren. Müller [62] arbeitet an DG-Verfahren mit h -adaptiven Gittern, dessen Gitter über raumfüllende Kurven bearbeitet werden, und untersucht anhand aufsteigender Wärmeblasen den Unterschied in Genauigkeit und Laufzeit der adaptiven Lösung zur Lösung auf uniformen Gittern, mit der gleichen Auflösung wie die feinste Auflösung des adaptiven Gitters. Tabelle 1.1 fasst die vorherig aufgeführten Modelle zusammen.

Durch die hohe Lokalität der DG-Verfahren erwartet man, dass sie sehr gut mit der Anzahl der CPUs skalieren. Hierzu wurde in [27, Tabelle 1.1] die schwache Skalierbarkeit für einen

1. Einleitung

Modell	Gleichungen	DG-Variante	Gitter	Zeitintegration
Hall, Nair [46]	Advektion	nodal, stark	Yin-Yang	RK beliebige Ordnung
DGCOM [1, 43]	FW	nodal, stark	unstrukturiert	explizit RK3
Blasi, St-Cyr [13]	FW	nodal, schwach	unstrukturiert, h - p -adaptiv	explizit RK3
Giraldo, Hesthaven, Warburton [41]	FW	nodal, stark, schwach	icosahedrisch	explizit RK3, Filter
Nair, Thomas, Loft [64]	FW	modal, schwach	würfelförmige Sphäre	explizit RK3
Giraldo, Restelli [42]	Euler	nodal, stark, schwach	unstrukturiert	explizit RK3, Filter
Restelli, Giraldo [72]	Euler	nodal, stark	strukturiert	semi-implizit RK3, Filter
Restelli [71]	Euler	nodal, stark	strukturiert	semi-implizit, semi-Lagrange, Filter
DUNE [18]	Euler	modal, schwach	strukturiert, unstrukturiert, h -adaptiv	explizit RK, semi-implizit RK
Müller (et al.) [63, 62]	Euler	nodal, stark	unstrukturiert, h -adaptiv	semi-implizit RK2

Tabelle 1.1.: Übersicht über die erläuterten DG-Verfahren, mit der Bezeichnung „Euler“ sind Euler-Gleichungen mit und ohne Diffusionsterme gemeint. RK3 bezeichnet ein Runge-Kutta-Verfahren dritter Ordnung.

zweidimensionalen Fall der skalaren Gleichung $\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0$ getestet. Für 256 CPUs liegt die Effizienz noch immer über 99 %. Klöforn zeigt in [54] die *starke Skalierbarkeit* (vgl. Anhang A.4) des DUNE-Lösers, und erläutert Einzelheiten zur Implementierung, wie diese zu erreichen ist. Bei 4096 CPUs beträgt die parallele Effizienz 91 % und bei 65 536 CPUs noch einen guten Wert von 72 %. Beide Werte verwenden 512 CPUs als Referenz.

Die obigen aufgeführten Arbeiten, die atmosphärische Strömungen modellieren, behandeln Probleme mit idealisiertem Charakter. Meteorologische Parametrisierungen sind bisher kaum bei den DG-Verfahren berücksichtigt.

Die Bestrebung ein operationelles Modell auf DG-Basis zu entwickeln verfolgen folgende Projekte: Das in den USA von der Naval Postgraduate School und dem Naval Research Laboratory entwickelte neue Atmosphärenmodell „Non-hydrostatic Unified Model of the Atmosphere“ (kurz NUMA; siehe etwa [52, 44] oder die Homepage <http://faculty.nps.edu/fgirald/projects/NUMA>) verwendet stetige Galerkin- als auch diskontinuierliche Galerkin-Verfahren. Ziele von NUMA sind: hydrostatische und nicht-hydrostatische Simulation, skalenübergreifende Verwendung (d. h. sowohl global als auch auf der Meso-Skala) und einheitliche Numerik für alle Anwendungen. NUMA ist masse-, impuls- und energieerhaltend. Das FW-Modell „HOMME“ des UCAR/NCARs (siehe <https://www.homme.ucar.edu>) hat neben

einem Dynamikkern, der auf spektralen Elementen basiert auch einen DG-Kern. HOMME ist der Dynamikkern des „Community Climate System Model“ und des „Community Earth System Model“. Mit der vorliegenden Arbeit unternimmt auch der Deutsche Wetterdienst den ersten Schritt in Richtung eines Dynamikkerns für die operationelle Wettervorhersage, der auf DG-Verfahren basiert.

Da der in dieser Arbeit entwickelte Löser auch warmen Niederschlag simulieren kann, soll die Arbeit von Gaberšek et al. [37] erwähnt werden, die jedoch auf spektralen Elementen-Verfahren und nicht auf DG-Verfahren basiert. Neben zweidimensionalen trockenen Testfällen wird die Wolken- und Niederschlagsentwicklung in einem zweidimensionalen Testfall untersucht. Hierfür wird ein nodales Modell mit nicht exakten Quadraturformeln und einer semi-impliziten Zeitintegration verwendet. Die trockenen Lösungen konvergieren unter Gitterverfeinerung. Für die feuchte Simulation ändert sich die Lösung unter Gitterverfeinerung ohne zu konvergieren. Die Autoren stellen die Hypothese auf, dass die starke nicht-Linearität der feuchten Bilanzgleichungen hierfür verantwortlich ist.

1.2. COSMO und dessen Dynamikkern

Das „Consortium for small scale modelling“ (COSMO) entwickelt und betreibt das gleichnamige Wettervorhersagemodell COSMO (vgl. [33, 32, 73] und die Webseite <http://www.cosmo-model.org>, hier ist der „COSMO Newsletter“ besonders erwähnenswert, da er einen Überblick über aktuelle Weiterentwicklungen und Forschungsergebnisse liefert.) auf dem diese Arbeit aufbaut. Die COSMO-Entwicklung reicht von der operationellen Anwendung bis hin zu theoretischen Studien (vgl. etwa [7]). COSMO ist ein Regionalmodell und für die operationelle Kurzfristvorhersage entwickelt, d. h. es wird mehrfach täglich zur Datenassimilation und Wettervorhersage laufen gelassen. Es ist die Weiterentwicklung des LM (Lokalmodell) des Deutschen Wetterdienstes, der auch heute noch den größten Teil der Entwicklungsarbeiten leistet. Darüber hinaus sind die Wetterdienste aus Griechenland, Italien, Polen, Rumänien, Russland und der Schweiz an der Entwicklung beteiligt. Im täglichen Betrieb wird es in diesen Ländern, sowie u. a. in Brasilien, Israel und dem Oman eingesetzt.

COSMO ist ein nicht-hydrostatisches, voll kompressibles Modell. Es verwendet die Euler-Gleichungen in Advektionsform mit den prognostischen Variablen: Abweichung des Drucks p' von einem gegebenen Referenzdruck, die zonale, meridionale und vertikale Windgeschwindigkeit u , v , w , sowie der Abweichung der totalen Temperatur T' von einer gegebenen Referenztemperatur. Für die Feuchte werden Wasserdampf, Wolkenwasser, Regenwasser, Eis, Schnee und Graupel prognostisch modelliert. Das von COSMO verwendete Gitter ist quadrilateral und vom Arakawa-C-Typ: die Größen p' und T' werden in den Elementmittelpunkten berechnet und u , v bzw. w in den Mittelpunkten der Seitenflächen orthogonal zur zonalen, meridionalen bzw. vertikalen Richtung. Das Gitter ist horizontal homogen und vertikal nach oben hin gestreckt.

COSMO enthält momentan zwei Dynamikkerne, den Bocksprung-Kern (oft mit der englischen Bezeichnung „leapfrog“ bezeichnet) und den Runge-Kutta-Kern. Darüber hinaus werden seit 2008, im Rahmen eines COSMO-Prioritäts-Projekts, zwei Dynamikkerne entwickelt (eine Übersicht gibt [4]), die auf FV-Verfahren basieren und dadurch Erhaltungseigenschaften haben. Der eine ist der EULAG-Kern, der anstelle der voll-kompressiblen Gleichungen die anelastischen Gleichungen verwendet. Dieser Kern ist soweit gereift, dass bereits semi-reale Simulationen möglich sind, etwa Alpenüberströmungen [4]. Der zweite Kern „CONSOL“ stammt aus der Aerodynamik und verwendet die voll-kompressiblen Euler-Gleichungen, mit voll-impliziter Zeitintegration. Mit CONSOL sind erste idealisierte Testfälle möglich [4].

In der operationellen Vorhersage wird der Runge-Kutta-Kern seit dem 26. Juli 2010 [90] vom

1. Einleitung

Parametrisierung	n
Subskalige Turbulenz	1
Oberflächenflüsse	1
Wolken und Niederschlag auf der Gitterskala	1
Feuchtkonvektion	10
Subskalige Wolken	1
Strahlung	36
Boden und Vegetation	1
Subskalige Orographische Prozesse	5

Tabelle 1.2.: Physikalische Parameterisierungen von COSMO und deren Zeitschritte. Die Zeitschritte sind in Vielfachen n des Zeitschritts Δt für die langsamen Prozesse des Dynamikkerns angegeben. Subskalige Prozesse bezeichnen hier Prozesse die nicht auf der Gitterskala aufgelöst werden können.

DWD operationell verwendet, dieser soll hier kurz beschrieben werden. Eine ausführliche Beschreibung der beiden operationell verwendbaren Dynamikkerne COSMOs findet man in [33]. Um ein effizientes Verfahren zu erhalten wird ein split-explizites Verfahren verwendet (vgl. [53, 78]): Es werden die schnellen Prozesse von den langsamen getrennt und mit unterschiedlichen Zeitschritten integriert. Dabei zählen zu den schnellen Prozessen die Schall- und Schwerewellen, die restlichen Terme gehören zu den langsamen Prozessen.

Für die zeitliche Diskretisierung wird ein dreistufiges RK-Verfahren zweiter Ordnung verwendet [6]. Die räumliche Diskretisierung erfolgt mit Finiten-Differenzenverfahren. Die schnellen Prozesse werden mit symmetrischen Differenzen zweiter Ordnung gerechnet. Für die langsamen Prozesse gibt es in der horizontalen eine Auswahl bis zur sechsten Ordnung. Im operationellen Betrieb wird ein Aufwindschema fünfter Ordnung verwendet. Die vertikalen Prozesse werden mit zentrierten Differenzen von zweiter Ordnung gerechnet. Damit der Zeitschritt nicht durch die geringe vertikale Ausdehnung der Gitterzellen beschränkt ist, wird vertikal ein implizites Zeitintegrationsverfahren verwendet, horizontal wird explizit integriert. Durch diese Aufteilung und die implizite Integration kann für die COSMO-DE-Gittermaschenweite (2,8 km horizontal, vertikal zwischen ca. 10 m und 1000 m) ein Zeitschritt für die langsamen Wellen von 25 s verwendet werden. Für die schnellen Wellen wird ein Zeitschritt von $4\frac{1}{6}$ s verwendet.

Um die Orographie abzubilden verwendet COSMO *geländefolgende Koordinaten* wodurch zusätzliche metrische Terme in den Bilanzgleichungen auftreten. Hierauf wird in Abschnitt 2.3 näher eingegangen. Für die Abbildung der physikalischen Prozesse, die nicht durch den Dynamikkern dargestellt werden können, werden eine Reihe von physikalischen Parameterisierungen verwendet [32], die zum Teil einen größeren Zeitschritt als der Dynamikkern haben. Tabelle 1.2 gibt einen Überblick über die Parameterisierungen von COSMO und mit welchem Zeitschritt sie aufgerufen werden. COSMO verwendet für die Physik-Dynamik-Kooplung Operatoraufteilung (vgl. Abschnitt 3.3).

COSMO wird für Skalen verwendet, in denen die Konvektion durch die Numerik direkt aufgelöst wird (etwa COSMO-DE), und für Skalen mit parametrisierter Konvektion (etwa COSMO-EU, horizontal etwa 7 km Gittermaschenweite). Es hat sich auch eine Wissenschaftlergruppe (CLM-Community) gebildet, die COSMO für regionale Klimasimulationen eingesetzt. Einen Überblick über deren Aktivitäten gibt der „CLM-Community Newsletter“ (<http://www.clm-community.eu/index.php?menuid=162>) und die Referenzen darin.

1.3. Ziele und Beiträge dieser Arbeit

Das Ziel dieser Arbeit ist die Entwicklung eines neuen Dynamikkerns „DG-COSMO“ für ein bestehendes numerisches Wettervorhersagesystem, das operationell eingesetzt wird. Da die Arbeit beim Deutschen Wetterdienst durchgeführt wurde, ist das Regionalmodell COSMO eine naheliegende Wahl. Im Vergleich zu NUMA wird also nicht ein komplett neues Modell entwickelt, sondern ein vorhandenes erweitert. Der neue Dynamikkern soll lokal und global die Masse, den Impuls und die potentielle Temperatur erhalten und eine hohe Verfahrensordnung ermöglichen. Darüber hinaus sollen die physikalischen Parametrisierungen, exemplarisch am Beispiel der Wolkenmikrophysik, mit einbezogen werden. Der in dieser Arbeit entwickelte DG-Löser unterscheidet sich von den anderen DG-Lösern durch:

1. Im Vergleich zu NUMA ist DG-COSMO ein modales Verfahren. Hierdurch lässt sich DG-COSMO einfach um p -Adaptivität erweitern, da eine hierarchische Basis vorliegt.
2. Für die Physik-Dynamik-Kopplung stehen die beiden Strategien Prozess- und Operatoraufteilung zur Verfügung.
3. Die Orographie wird durch geländefolgende Koordinaten modelliert, dadurch kann ein quadrilaterales Gitter auch bei vorhandener Orographie verwendet werden. Durch das quadrilaterale Gitter ist kein aufwendiger Gittergenerator nötig, die Implementierung des DG-Verfahrens wird vereinfacht und es kann ein effizientes Verfahren implementiert werden.
4. DG-COSMO ist mit einem Kessler-Schema für warmen Niederschlag (d. h. ohne Eisphase) gekoppelt. Die Sättigungsadjustierung die im Kessler-Schema verwendet wird, ist aus dem Energieerhaltungssatz hergeleitet und nicht durch ad-hoc gewählte Funktionen modelliert wie in [18].

Diese Arbeit leistet zu folgenden offenen Fragen einen Beitrag:

- Welche Vorteile bringen diskontinuierliche Galerkin-Verfahren für den Dynamikkern im Vergleich zu einem Dynamikkern der auf Finite-Differenzenverfahren basiert?
- Bringt eine höhere Verfahrensordnung einen Vorteil?
- Welcher der beiden Kopplungsstrategien *Operatoraufteilung* und *Prozessaufteilung* ist der Vorzug zu geben, um die physikalischen Parametrisierungen an den DG-Dynamikkern zu koppeln?

Diese Arbeit hat einen interdisziplinären Charakter, da die erfolgreiche Übertragung der Methoden aus den Ingenieurwissenschaften und der Mathematik auf die numerische Wettervorhersage zur Beantwortung vorheriger Fragestellungen neben meteorologischen Kenntnisse auch vertieftes Wissen in der mathematischen, geophysikalischen Modellierung, der numerischen Mathematik und der Informatik erfordern. Die Beiträge dieser Arbeit sind:

- Die Entwicklung eines DG-Dynamikkerns mit geländefolgenden Koordinaten und einem warmen Regenmodell vom Kessler-Typ. Dieser Kern hat formal eine beliebig hohe räumliche Ordnung und zwischen zweiter und vierter Ordnung in der zeitlichen Diskretisierung. Die experimentelle Konvergenzordnung erreicht die formale. Diese Arbeit hat auch einen exploratorischen Charakter. Sie stellt einen weiteren Ansatz zu bestehenden atmosphärischen DG-Lösern zur Verfügung, und hilft hierdurch eine für die operationelle Vorhersage geeignete DG-Variante zu finden.
- Untersuchungen der Skalierbarkeit und der Laufzeit im Vergleich zum bestehenden COSMO-Modell, d. h. eines FD-Dynamikkerns. Die Skalierbarkeit des DG-Kerns ist höher als die des RK-Kerns von COSMO. Wird für beide Kerne der gleiche Zeitschritt gewählt, so ist für die trockene Atmosphäre für eine dreidimensionale Simulation ab etwa 4220 CPUs der DG-Kern schneller. Bei zweidimensionalen Simulationen ist der DG-Kern schon bei deutlich weniger CPUs schneller.

1. Einleitung

- Ein Algorithmus zu Bestimmung der hydrostatischen Balance an den Quadraturpunkten des DG-Verfahrens.
- Untersuchungen des Wohlbalanciertheitsproblems. Für atmosphärische DG-Verfahren tritt durch den Dissipationsterm im Rusanov-Fluss eine zusätzliche Störung bei ruhender Atmosphäre auf. Die Größe dieser Störung lässt sich aber über die Verfahrensordnung kontrollieren. In einer ruhenden Atmosphäre mit einem isolierten Berg bleibt der DG-Kern stabil für Steigungen, bei denen COSMO bereits abstürzt. Über eine Erhöhung der Ordnung lässt sich der Fehler unter den Fehler der COSMO-Lösung bringen.
- Ein Vergleich der Operatoraufteilung und der Prozessaufteilung bei unterschiedlichen Verfahrensordnungen und Zeitschritten der Parametrisierung. Erst ab einem größeren Zeitschritt für die Parametrisierungen ist ein signifikanter Unterschied zwischen den Aufteilungsstrategien zu erkennen.

Diese Arbeit ist wie folgt aufgebaut: In Kapitel 2 werden die Bilanzgleichungen beschrieben, die das physikalische Modell bilden, das dieser Arbeit zu Grunde liegt. Hierzu gehören auch die Phasenumwandlung des Wassers einschließlich eines Kessler-Schemas für warmen Regen. Kapitel 3 gibt eine kurze technische Herleitung des DG-Ansatzes und beschreibt anschließend im Einzelnen die DG-Variante die in DG-COSMO implementiert wurde. Kapitel 4 beschreibt die Validierung von DG-COSMO anhand von trockenen idealisierten Testfällen. In Kapitel 5 wird das Kessler-Schema validiert und die Kopplung der physikalischen Parametrisierung untersucht, sowie die Skalierbarkeit von DG-COSMO und COSMO getestet. In Kapitel 6 wird resümiert und ein Ausblick für den weiteren Weg bis zur operationellen Reife von DG-COSMO gegeben.

2. Bilanzgleichungen des physikalischen Modells

Um ein System zu erhalten, mit dem prinzipiell Wettervorhersagen möglich sind, werden in dieser Arbeit die feuchte Luft, Wolken, Niederschlag sowie die Orographie modelliert. Hiermit lassen sich schon einige Wettersituationen simulieren, wenn bei weitem auch nicht alle. Auch kann bei dieser Auswahl nicht erwartet werden, dass die Ergebnisse realer Wettersimulationen in der Qualität vergleichbar sind mit Modellen, die die operationelle Einsatzbereitschaft erlangt haben. Aber für idealisierte Testfälle ist direkte Vergleichbarkeit gegeben. In den Simulationen in den Kapiteln 4 und 5 werden nur Testfälle betrachtet, bei denen die Erdkrümmung vernachlässigt werden kann. Auch sind die Integrationszeiträume so kurz, dass die Corioliskraft keine Rolle spielt. Dies gilt nicht für die Bergüberströmung, hier ist jedoch der Testfall so entworfen, dass ohne Corioliskraft gerechnet wird. Daher wird in dieser Arbeit sowohl die Erdkrümmung also auch die Corioliskraft vernachlässigt.

In diesem Kapitel werden die Bilanzgleichungen für die Atmosphärendynamik dargestellt. Die meisten Modellentscheidungen, die hier getroffen werden, sind üblich in der Meteorologie. Diskontinuierliche Galerkin-Verfahren erhalten die prognostischen Variablen lokal bis auf Maschinengenauigkeit. Daher werden die Bilanzgleichungen in Flussform mit möglichst wenigen Quelltermen verwendet. Für die Orographie wird ein geländefolgendes Koordinatensystem eingeführt, wodurch zusätzliche Quellterme auftreten, wenn man alle Variablen in diesem System formuliert. Ein gemischtes Koordinatensystem, das in den freien Koordinaten geländefolgend und in den prognostischen Variablen kartesisch ist, schafft hier Abhilfe.

2.1. Physikalisches Modell

Zuerst wird das physikalische Modell für die trockene Dynamik beschrieben. Trockene Luft wird hier, wie in der Meteorologie üblich als homogenes ideales Gas betrachtet, daher werden die spezifischen Wärmekapazitäten als konstant angenommen¹ und es gilt für die Gaskonstante für trockene Luft $R_d = c_{pd} - c_{vd}$, wobei c_{pd} bzw. c_{vd} die isobare bzw. isochore spezifische Wärmekapazität von trockener Luft sind. Tabelle 2.1 gibt eine Übersicht für die in dieser Arbeit verwendeten Werte der Gaskonstanten, spezifischen Wärmekapazitäten (und der Gravitationskonstanten). Es wird für die trockene Atmosphäre somit folgende Gasgleichung verwendet:

$$p = \rho R_d T,$$

wobei p den Luftdruck und T die absolute Temperatur bezeichnet. Mit der potentiellen Temperatur $\theta = T \left(\frac{p_{00}}{p} \right)^{\frac{R_d}{c_{pd}}}$ lässt sich diese Gleichung schreiben als:

$$p = p_{00} \left(\frac{R_d}{p_{00}} \rho \theta \right)^{\frac{c_{pd}}{c_{vd}}},$$

wobei wie üblich $p_{00} := 10^5$ Pa.

¹In der Realität ist Luft kein perfektes ideales Gas und die Wärmekapazitäten sind schwach temperaturabhängig, zwischen 250 K und 300 K bei 1013,25 hPa schwankt beispielsweise c_{vd} um etwa 0,7‰ und c_{pd} um etwa 2,4‰ [59].

2. Bilanzgleichungen des physikalischen Modells

Für die trockene Atmosphäre werden als prognostische Variablen die Massendichte ρ , die Impulsdichte $\rho\mathbf{u}$ und das Produkt aus Massendichte und potentieller Temperatur $\rho\theta$ verwendet. Die molekulare kinematische Viskosität der Luft bei 273,15 K und 1013,3 hPa ist $\nu = 1,56 \cdot 10^{-5} \frac{\text{m}^2}{\text{s}}$ (vgl. [76]). Dies ist um fünf Zehnerpotenzen kleiner als die turbulente kinematische Viskosität von Luft, deren typischer Wert bei etwa $4 \frac{\text{m}^2}{\text{s}}$ (vgl. [36]) liegt, daher wird die Luft ohne molekulare Reibung modelliert. So bilden die Euler-Gleichungen in Erhaltungsform die Grundlage dieser Arbeit:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.1)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p I_3) = -\rho g \mathbf{k}, \quad (2.2)$$

$$\frac{\partial \rho \theta}{\partial t} + \nabla \cdot (\rho \theta \mathbf{u}) = 0, \quad (2.3)$$

wobei \otimes das dyadische Produkt, I_3 die dreidimensionale Einheitsmatrix und \mathbf{k} der Einheitsvektor in z -Richtung ist. Da die Turbulenz auf praktisch relevanten Skalen nicht aufgelöst werden kann, wird eine Reynolds-Mittelung obiger Gleichungen im Raum und in der Zeit durchgeführt. Dabei soll die Zerlegung $\psi = \bar{\psi} + \psi'$ gelten mit:

$$\bar{\psi} := \frac{1}{\Delta t \Delta x \Delta y \Delta z} \int_t^{t+\Delta t} \int_x^{x+\Delta x} \int_y^{y+\Delta y} \int_z^{z+\Delta z} \psi(\tau, \xi_1, \xi_2, \xi_3) d\xi_3 d\xi_2 d\xi_1 d\tau$$

ist das Mittel und ψ' die Abweichung vom Mittel, dabei wird $\overline{\psi'} = 0$ gefordert. Für \mathbf{u} und θ wird noch das Hesselberg-Mittel $\hat{\psi}$ (mit $\psi \in \{\mathbf{u}, \theta\}$) eingeführt:

$$\hat{\psi} := \frac{\overline{\rho \psi}}{\bar{\rho}},$$

mit der Zerlegung $\psi = \hat{\psi} + \psi''$, und der Forderung $\widehat{\psi''} = 0$. Diese Reynolds-Hesselberg-Mittelung wird auch für COSMO (vgl. [33]) durchgeführt. Das Gleichungssystem (2.1) bis (2.3) hat nach der Mittelung in der Impuls- und potentiellen Temperaturgleichung einen zusätzlichen Term:

$$\begin{aligned} \frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \hat{\mathbf{u}}) &= 0 \\ \frac{\partial \bar{\rho} \hat{\mathbf{u}}}{\partial t} + \nabla \cdot (\bar{\rho} \hat{\mathbf{u}} \otimes \hat{\mathbf{u}} + \bar{p} I_3 + \tau) &= -\bar{\rho} g \mathbf{k}, \\ \frac{\partial \bar{\rho} \hat{\theta}}{\partial t} + \nabla \cdot (\bar{\rho} \hat{\theta} \hat{\mathbf{u}} + \chi) &= 0, \end{aligned}$$

wobei

$$\begin{aligned} \tau &= \frac{\overline{\rho \mathbf{u}'' \otimes \rho \mathbf{u}''}}{\bar{\rho}}, \\ \chi &= \frac{\overline{\rho \theta'' \rho \mathbf{u}''}}{\bar{\rho}}. \end{aligned}$$

Da in dieser Arbeit die Dynamik im Vordergrund steht und keine stark ausgearbeiteten Turbulenzmodelle untersucht werden sollen, wird für die Turbulenzschließung folgender

einfacher Ansatz gemacht:

$$\tau := -\bar{\rho}\mu_m (\nabla\hat{\mathbf{u}} + (\nabla\hat{\mathbf{u}})^\top), \quad (2.4)$$

$$\chi := -\bar{\rho}\mu_h \nabla\hat{\theta}, \quad (2.5)$$

mit konstanter turbulenter kinematischer Viskosität μ_m für den Impuls und μ_h für die potentielle Temperatur. Dieses Modell ist ähnlich zu System 2 aus [42] aber mit symmetrischem Reynolds'schem Spannungstensor. Es ist also rotationsinvariant, und vom Gradienten des Windfelds abhängig und somit ähnlich zum molekularen Spannungstensor eines Newton'schen Fluids. Die Viskositätskonstanten μ_m und μ_h sind abhängig von der Problemstellung. Ab jetzt werden für die Variablen die Überstriche $\bar{\psi}$ und Dächer $\hat{\psi}$ unterdrückt und einfach nur ψ für das Reynolds- oder Hesselberg-Mittel geschrieben. Mit obigem einfachen Turbulenzmodell lassen sich die Euler-Gleichungen auf

$$\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{u}) = 0, \quad (2.6)$$

$$\frac{\partial\rho\mathbf{u}}{\partial t} + \nabla \cdot (\rho\mathbf{u} \otimes \mathbf{u} + pI_3 - \rho\mu_m (\nabla\mathbf{u} + (\nabla\mathbf{u})^\top)) = -\rho g\mathbf{k}, \quad (2.7)$$

$$\frac{\partial\rho\theta}{\partial t} + \nabla \cdot (\rho\theta\mathbf{u} - \rho\mu_h \nabla\theta) = 0, \quad (2.8)$$

erweitern. Insbesondere ändert sich die Kontinuitätsgleichung (2.6) durch die Mittelung nicht in der Form. Die Erhöhung der inneren Energie durch die dissipierte kinetische Energie wird, wie in vielen anderen Modellen auch, vernachlässigt, da sie in der freien Atmosphäre eine untergeordnete Rolle spielt und in dieser Arbeit eher kurze Integrationszeiträume betrachtet werden (vgl. [12]).

Für die Feuchte in der Atmosphäre werden der Wasserdampf, das Wolkenwasser und Regenwasser berücksichtigt. Die Partialdichte des Wasserdampfes wird mit ρ_v , des Wolkenwassers mit ρ_c und des Regenwassers mit ρ_r bezeichnet. Die *spezifische Feuchte* oder der *spezifische Wasserdampfgehalt*, d. h. der Anteil an der Gesamtmasse, wird mit $q_v := \frac{\rho_v}{\rho}$ bezeichnet. Der *spezifische Wolkenwassergehalt* q_c bzw. *spezifische Regenwassergehalt* q_r werden analog definiert. Feuchte Luft wird als Gemisch zweier idealer Gase modelliert, d. h. die Partialdrücke lassen sich nach dem Dalton'schen Gesetz addieren, die Gasgleichung hierfür ist:

$$\begin{aligned} p &= p_d + p_v = \rho R_m T, \\ \text{mit } p_d &= \rho_d R_d T, \\ p_v &= \rho_v R_v T, \\ R_m &= R_d q_d + R_v q_v, \end{aligned}$$

wobei $q_d = 1 - (q_v + q_c + q_r)$ der Massenanteil der trockenen Luft an der Gesamtmasse, R_v die spezifische Gaskonstante für Wasserdampf und p_d bzw. p_v die Partialdrücke von trockener Luft bzw. Wasserdampf sind. Das Volumen des Wolken- und Regenwassers wird vernachlässigt, da sei Beitrag zum Gesamtvolumen in der Größenordnung von 10^{-6} ist, und damit unerheblichen ist (vgl. [33]). Die isobare bzw. isochore spezifische Wärmekapazität für feuchte Luft c_{pml} bzw. c_{vml} sind gegeben durch:

$$\begin{aligned} c_{pml} &= q_d c_{pd} + q_v c_{pv} + (q_c + q_r) c_l, \\ c_{vml} &= q_d c_{vd} + q_v c_{vv} + (q_c + q_r) c_l. \end{aligned}$$

2. Bilanzgleichungen des physikalischen Modells

Die Advektions-Diffusionsgleichungen für die Feuchtegrößen sind:

$$\begin{aligned}\frac{\partial \rho_v}{\partial t} + \nabla \cdot (\rho_v \mathbf{u} - \rho \mu_h \nabla q_v) &= s_v, \\ \frac{\partial \rho_c}{\partial t} + \nabla \cdot (\rho_c \mathbf{u} - \rho \mu_h \nabla q_c) &= s_c, \\ \frac{\partial \rho_r}{\partial t} + \nabla \cdot (\rho_r (\mathbf{u} - v_T \mathbf{k})) &= s_r,\end{aligned}$$

mit den Quelltermen s_v , s_c und s_r . Die Diffusionskonstante der Feuchten ist dieselbe wie die von $\rho\theta$, wobei die Diffusion von Regenwasser vernachlässigt wird. Regenwasser unterscheidet sich von Wolkenwasser dadurch, dass es eine rein vertikale Fallgeschwindigkeit v_T hat. Die Fallgeschwindigkeit von Wolkenwassertröpfchen wird vernachlässigt.

In der feuchten Atmosphäre ist θ keine Erhaltungsgröße mehr (vgl. [22]). Es kommt für Gleichung (2.8) auf der rechten Seite ein Quellterm s_θ hinzu. Dieser modelliert die bei der Wasserdampfkondensation freiwerdende oder bei der Verdunstung von flüssigem Wasser aufgenommene Enthalpie. Aber selbst ohne Phasenübergang ist θ keine Erhaltungsgröße mehr. Definiert man jetzt wie in [22]:

$$\tilde{\theta} = T \left(\frac{p_{00}}{p} \right)^{\frac{R_m}{c_{pml}}},$$

so ist die Größe $\tilde{\theta}$ eine Erhaltungsgröße, solange kein Phasenübergang stattfindet [22]. Um die Gasgleichung zu vereinfachen, wird die *feucht-potentielle Temperatur* mit einem Vorfaktor definiert:

$$\theta_m = \frac{R_m}{R_d} T \left(\frac{p_{00}}{p} \right)^{\frac{R_m}{c_{pml}}}.$$

Die Zustandsgleichung ist mit dieser Größe:

$$p = p_{00} \left(\frac{R_d}{p_{00}} \rho \theta_m \right)^{\frac{c_{pml}}{c_{vml}}}. \quad (2.9)$$

Das in dieser Arbeit verwendete physikalische Modell wird somit beschrieben durch die prognostischen Gleichungen:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.10)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p I_3 - \underbrace{\rho \mu_m (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)}_{\boldsymbol{\tau}}) = -\rho g \mathbf{k}, \quad (2.11)$$

$$\frac{\partial \rho \theta_m}{\partial t} + \nabla \cdot (\rho \theta_m \mathbf{u} - \rho \mu_h \nabla \theta_m) = s_\theta, \quad (2.12)$$

$$\frac{\partial \rho_v}{\partial t} + \nabla \cdot (\rho_v \mathbf{u} - \rho \mu_h \nabla q_v) = s_v, \quad (2.13)$$

$$\frac{\partial \rho_c}{\partial t} + \nabla \cdot (\rho_c \mathbf{u} - \rho \mu_h \nabla q_c) = s_c, \quad (2.14)$$

$$\frac{\partial \rho_r}{\partial t} + \nabla \cdot (\rho_r (\mathbf{u} - v_T \mathbf{k})) = s_r \quad (2.15)$$

$$\begin{array}{lll}
c_{pd} = 1005,00 \frac{\text{J}}{\text{kg K}} & c_{pv} = 1848,80 \frac{\text{J}}{\text{kg K}} & c_l = 4194,17 \frac{\text{J}}{\text{kg K}} \\
c_{vd} = 717,95 \frac{\text{J}}{\text{kg K}} & c_{vv} = 1387,29 \frac{\text{J}}{\text{kg K}} & \\
R_d = 287,05 \frac{\text{J}}{\text{kg K}} & R_v = 461,51 \frac{\text{J}}{\text{kg K}} & g = 9,80665 \frac{\text{m}}{\text{s}^2}
\end{array}$$

Tabelle 2.1.: Gaskonstanten, spezifische Wärmekapazitäten von trockener Luft und Wasserdampf, spezifische Wärmekapazität von Wasser, sowie Erdbeschleunigung (Werte aus COSMO-Quellcode)

und der Zustandsgleichung (2.9). Die Werte der Konstanten R_d , R_v , c_{pd} , c_{vd} , c_{pv} , c_{vv} , c_l sowie die Erdbeschleunigung g können in Tabelle 2.1 nachgeschlagen werden und die Berechnung der Quellterme und der Fallgeschwindigkeit wird in Abschnitt 2.1.1 beschrieben.

In vielen Modellen, unter anderem auch in COSMO, findet man die Näherung $c_{pml} \approx c_{pd}$ (und daraus folgende) in Teilen der Dynamik und in den Parametrisierungen. In dieser Arbeit wird auf diese Näherung verzichtet, es wird immer mit dem feuchten c_{pml} gerechnet.

2.1.1. Kessler-Schema und Sättigungsadjustierung

Da in dieser Arbeit die Dynamik im Vordergrund steht, werden die Quellterme für die Phasenumwandlung des Wassers und die feucht-potentielle Temperaturänderung durch ein einfaches Kessler-Schema modelliert, welches folgende Prozesse berücksichtigt [32]:

S_v Verdampfung von Wolkenwasser zu Wasserdampf,

S_c Kondensation von Wasserdampf zu Wolkenwasser,

S_{au} Autokonversion, dies ist die initiale Regentropfenbildung aus Wolkenwassertropfen,

S_{ac} Regentropfenwachstum innerhalb der Wolke, durch Aufsammeln weiterer Wolkentröpfchen,

S_{ev} Verdunstung von Regentropfen in untersättigten Luftschichten unterhalb der Wolke.

Die Kondensation von Wasserdampf an Regentropfen sowie feste Hydrometeore werden nicht modelliert.

Zuerst wird nur der Phasenübergang zwischen ρ_v und ρ_c sowie die zugehörige Verdunstungs- oder Kondensationsenthalpie, die in s_θ eingeht, betrachtet, der mithilfe der Prozesse S_v und S_c modelliert wird, und die anderen Prozesse werden vorübergehend vernachlässigt. S_v wird als unendlich schneller Prozess approximiert und durch eine Sättigungsadjustierung berechnet. Ist ein Kontrollvolumen mit Wasserdampf übersättigt, so kondensiert so viel Wasserdampf zu Wolkenwasser, bis die Sättigung im Kontrollvolumen erreicht ist. Ist das Kontrollvolumen untersättigt, so verdunstet so lange Wolkenwasser, bis die Sättigung erreicht ist, oder alles Wolkenwasser verdunstet ist. Die Kondensations- bzw. Verdunstungsenthalpie erhöht bzw. erniedrigt die Temperatur im Kontrollvolumen.

Die nachfolgend beschriebene isochore Sättigungsadjustierung wurde von Ulrich Blahak (DWD) entwickelt und für DG-COSMO übernommen. In der operationellen Version von COSMO wird eine ähnliche, isobare Sättigungsadjustierung verwendet (vgl. [32, Abschnitt 5.3]).

2. Bilanzgleichungen des physikalischen Modells

Die Betrachtung bezieht sich auf ein Luftpaket mit der totalen Masse m und dem Volumen V . Der Massenanteil der trockenen Luft sei q_{d0} , der Ausgangszustand hat die spezifische Feuchte q_{v0} , den spezifischen Wolkenwassergehalt q_{c0} und die Temperatur T_{00} . Der Zustand nach der Sättigungsadjustierung sei \tilde{q}_v , \tilde{q}_c und \tilde{T} . Ist das Luftpaket mit Wasserdampf gesättigt, so wird dessen Temperatur mit T_{sat} und der Sättigungsdampfdruck mit p_{sat} bezeichnet. Die spezifische Feuchte bei Sättigung ist:

$$q_{v,\text{sat}} := q_v(T_{\text{sat}}) = \frac{p_{\text{sat}}(T_{\text{sat}})}{R_v T_{\text{sat}} \rho}. \quad (2.16)$$

Aus der Erhaltung der Gesamtwassermasse $q_{v0} + q_{c0}$ während des Phasenübergangs zwischen Wasserdampf und Wolkenwasser ergibt sich:

$$\tilde{q}_v + \tilde{q}_c = q_{v0} + q_{c0} = \text{const}, \quad (2.17)$$

$$c_{vml} = c_{vd} q_{d0} + c_{vv} \tilde{q}_v + c_l \tilde{q}_c. \quad (2.18)$$

Wird (2.17) nach q_c aufgelöst und in (2.18) eingesetzt so gilt:

$$c_{vml} = c_{vd} q_{d0} + (c_{vv} - c_l) \tilde{q}_v + c_l (q_{v0} + q_{c0}). \quad (2.19)$$

Die Sättigungsadjustierung bestimmt nun \tilde{T} und \tilde{q}_v . Als Ausgangspunkt wird der erste Hauptsatz der Thermodynamik verwendet in der Form

$$Q = \frac{dU}{dt} + p_v \frac{dV}{dt}, \quad (2.20)$$

wobei Q die Heizrate und U die innere Energie ist. Für die zeitliche Ableitung der inneren Energie gilt (vgl. [16]):

$$\frac{dU}{dt} = c_{vml} m \frac{dT}{dt}. \quad (2.21)$$

und für die Heizrate:

$$Q = -l_v(T) m \frac{dq_v}{dt}, \quad (2.22)$$

wobei l_v die spezifische Verdunstungs- oder Kondensationsenthalpie² von Wasser ist. Sie ist gegeben durch [16]:

$$l_v(T) = l_{v0} + (c_{pv} - c_l)(T - T_r), \quad (2.23)$$

mit $l_{v0} = 2,501 \frac{\text{MJ}}{\text{kg}}$ und dem Tripelpunkt von Wasser $T_r = 273,16 \text{ K}$. Der zweite Term auf der rechten Seite von (2.20), der die verrichtete Arbeit beschreibt, lässt sich mit der Gasgleichung und der Definition der Dichte umformen zu:

$$p_v \frac{dV}{dt} = -m R_v T \frac{dq_v}{dt}. \quad (2.24)$$

Setzt man (2.21), (2.22) und (2.24) in (2.20) ein und teilt durch m , so erhält man nach Umformung:

$$c_{vml} \frac{dT}{dt} + (l_v - R_v T) \frac{dq_v}{dt} = 0, \quad (2.25)$$

²Hierfür wird auch der Begriff *spezifische latente Wärme* verwendet.

Wird (2.25) nach T abgeleitet, so ergibt sich mit (2.19):

$$\frac{d}{dt} \left((R_v T - l_v) \frac{dq_v}{dT} - (c_{vd} q_{d0} + (c_{vv} - c_l) q_v + c_l (q_{v0} + q_{c0})) \right) = 0.$$

Es ist also für eine Konstante C :

$$(R_v T - l_v) \frac{dq_v}{dT} - (c_{vd} q_{d0} + (c_{vv} - c_l) q_v + c_l (q_{v0} + q_{c0})) = C.$$

Nach Umformung erhält man:

$$\frac{dq_v}{dT} = -\frac{C}{l_v - R_v T} - \frac{c_{vd} q_{d0} + c_l (q_{v0} + q_{c0})}{l_v - R_v T} + \frac{c_{vv} - c_l}{l_v - R_v T} q_v.$$

Interpretiert man die Konstante C physikalisch, so beschreibt sie einen weiteren Prozess, der q_v mit der Temperatur ändert. Die Annahme ist aber, dass bereits alle Prozesse modelliert sind, daher ist der physikalisch sinnvolle Wert $C = 0$. Übrig bleibt:

$$\frac{dq_v}{dT} = -\underbrace{\frac{c_{vd} q_{d0} + c_l (q_{v0} + q_{c0})}{l_v - R_v T}}_{=:A} + \underbrace{\frac{c_{vv} - c_l}{l_v - R_v T}}_{=:B} q_v.$$

Nähert man jetzt T mit T_{00} in A und B , so werden diese Koeffizienten konstant und man erhält die lineare Differentialgleichung:

$$\frac{dq_v}{dT} - B q_v = -A. \quad (2.26)$$

Mit der Anfangsbedingung $q_v(T_{00}) = q_{v0}$ ist die Lösung von (2.26):

$$q_v(T) = \frac{A}{B} + \left(q_{v0} - \frac{A}{B} \right) e^{B(T-T_{00})}. \quad (2.27)$$

Um den Zustand für \tilde{q}_v , \tilde{q}_c und \tilde{T} nach der Sättigungsadjustierung zu berechnen, wird zuerst für $q_v(T)$ in (2.27) der totale Wassergehalt $q_{v0} + q_{c0}$ eingesetzt und nach der freien Variablen T aufgelöst, die Lösung heiße T_w . Aus (2.16) lässt sich $q_{v,\text{sat}}(T_w)$ bestimmen. Der Endzustand ist nun:

$$\tilde{q}_v = \begin{cases} q_{v0} + q_{c0}, & \text{falls } q_{v0} + q_{c0} \leq q_{v,\text{sat}}(T_w), \\ q_{v,\text{sat}}(T_{\text{sat}}) & \text{sonst,} \end{cases} \quad (2.28)$$

$$\tilde{q}_c = q_{v0} + q_{c0} - \tilde{q}_v, \quad (2.29)$$

$$\tilde{T} = \begin{cases} T_w & \text{falls } q_{v0} + q_{c0} \leq q_{v,\text{sat}}(T_w), \\ T_{\text{sat}} & \text{sonst.} \end{cases} \quad (2.30)$$

Die Gleichgewichtstemperatur T_{sat} wird durch Gleichsetzen von (2.27) mit (2.16) berechnet, d. h. man löst folgende implizite Gleichung nach T_{sat} :

$$\frac{A}{B} + \left(q_{v0} - \frac{A}{B} \right) e^{B(T_{\text{sat}}-T_{00})} = \frac{p_{\text{sat}}(T_{\text{sat}})}{R_v T_{\text{sat}} \rho}.$$

Es bieten sich zwei Sichtweisen an, die Sättigungsadjustierung als unendlich schnellen Prozess zu modellieren:

2. Bilanzgleichungen des physikalischen Modells

1. Anstelle der direkten Berechnung der Quellterme werden diese auf Null gesetzt und durch die Sättigungsadjustierung wird eine Nebenbedingung an den Phasenraum gestellt, d.h. es werden nur diejenigen Lösungen der Gleichungen (2.10) bis (2.15) zugelassen, die auch diese Nebenbedingungen erfüllen.
2. Man wählt eine physikalische Zeitskala Δt_p , und berechnet aus den Ausgangswerten und den adjustierten Werten die Quellterme durch:

$$s_v = \rho \frac{\tilde{q}_v - q_{v0}}{\Delta t_p}, \quad s_c = \rho \frac{\tilde{q}_c - q_{c0}}{\Delta t_p}, \quad s_\theta = \rho \frac{\tilde{\theta}_m - \theta_{m0}}{\Delta t_p}, \quad (2.31)$$

wobei θ_{m0} und $\tilde{\theta}_m$ die potentielle Temperatur des Anfangs- bzw. Endzustands der Sättigungsadjustierung bezeichnen. Es werden also nur die Auswirkungen des unendlich schnellen Prozesses auf dieser Zeitskala modelliert.

Die anderen Prozesse des Kessler-Schemas werden jetzt wieder berücksichtigt und mit folgenden Formeln aus [32] berechnet:

$$\begin{aligned} S_{au} &= \tau_{au} q_c, \\ S_{ac} &= c_{ac} q_c \rho_r^{\frac{7}{8}}, \\ S_{ev} &= (\alpha_{ev,1} - \alpha_{ev,2}(T - T_r)) \sqrt{\rho_r} (1 + \beta_{ev} \rho_r^{\frac{3}{16}}) (q_{v,sat} - q_v), \end{aligned}$$

mit den Konstanten (auch aus [32]):

$$\begin{aligned} \tau_{au} &= 0,001 \frac{1}{s}, & c_{ac} &= 1,72 \frac{1}{s}, \\ \alpha_{ev,1} &= 3,86 \cdot 10^{-3} \frac{1}{s}, & \alpha_{ev,2} &= 9,41 \cdot 10^{-5} \frac{1}{Ks}, & \beta_{ev} &= 9,1. \end{aligned}$$

Im Unterschied zum Kessler-Schema in COSMO wurde im Term S_{ev} der Tripelpunkt $T_r = 273,16$ K von Wasser und nicht der Gefrierpunkt $273,15$ K von luftgesättigtem Wasser gewählt. Damit hat die Gleichung (2.23) und das Kessler-Schema den gleichen Bezugspunkt. Die Auswirkung dieser Wahl auf die Ergebnisse in Kapitel 5 ist jedoch vernachlässigbar.

Da die Sättigungsadjustierung als unendlich schnell modelliert wird, muss auch während der Prozesse im Kessler-Schema, insbesondere auch für deren Endzustand, die Bedingungen (2.28) bis (2.30) erfüllt sein. Dies motiviert zuerst aus den Prozessen S_{au} , S_{ac} und S_{ev} für eine physikalische Zeitskala Δt_p der Zustand q_{v0} , q_{c0} und T_{00} zu berechnen und anschließend die Sättigungsadjustierung darauf anzuwenden. Ist der Zustand vor dem Kessler-Schema q_v , q_c , q_r und T , dann werden mit den Tendenzen S_{au} , S_{ac} und S_{ev} berechnet:

$$q_{v0} = q_v + \Delta t_p S_{ev}, \quad (2.32)$$

$$q_{c0} = q_c - \Delta t_p (S_{au} + S_{ac}), \quad (2.33)$$

$$\tilde{q}_r = q_r + \Delta t_p (S_{au} + S_{ac}), \quad (2.34)$$

$$T_{00} = T - \Delta t_p \frac{l_v}{c_{pml}} S_{ev}. \quad (2.35)$$

Nach Anwendung der Sättigungsadjustierung auf q_{v0} , q_{c0} und T_{00} werden die Tendenzen s_v , s_c , s_r , s_θ analog zu (2.31) berechnet zu:

$$s_v = \rho \frac{\tilde{q}_v - q_v}{\Delta t_p}, \quad s_c = \rho \frac{\tilde{q}_c - q_c}{\Delta t_p}, \quad s_r = \rho \frac{\tilde{q}_r - q_r}{\Delta t_p} = \rho (S_{au} + S_{ac}), \quad s_\theta = \rho \frac{\tilde{\theta}_m - \theta_m}{\Delta t_p}, \quad (2.36)$$

wobei θ_m und $\tilde{\theta}_m$ die potentielle Temperatur des Anfangs- bzw. Endzustands des Kessler-Schemas einschließlich Sättigungsadjustierung bezeichnen.

Um das physikalische Modell vollständig zu definieren, bleibt noch die Bestimmung zweier Größen. Der Sättigungsdampfdruck p_{sat} wird in dieser Arbeit wie auch im COSMO-Modell mithilfe der Murray-Formel [57] berechnet:

$$p_{\text{sat}}(T) := 610,78 \cdot e^{\frac{17,2693882 \cdot (T - 273,16 \text{ K})}{(T - 35,86 \text{ K})}} \text{ Pa.}$$

Die Fallgeschwindigkeit v_T hängt nur von ρ_r ab und wird berechnet durch [32]:

$$v_T = 12,63 \cdot \sqrt[8]{\rho_r} \frac{\text{m}}{\text{s}}.$$

2.2. Referenzzustand

Der Zustand einer trockenen Atmosphäre, die in Ruhe und hydrostatisch balancierte ist, löst das Gleichungssystem (2.10) bis (2.15). Um die hydrostatische Balance nicht durch numerische Fehler zu stören, wird ein hydrostatisch balancierter trockener *Referenzzustand* eingeführt, der in Ruhe ist. Die Variablen ρ , $\rho\theta$ und p werden als Summe aus dem Referenzzustand und der Abweichung von diesem definiert:

$$\begin{aligned} \rho(t, x, y, z) &= \rho_0(z) + \rho'(t, x, y, z), \\ (\rho\theta)(t, x, y, z) &= (\rho\theta)_0(z) + (\rho\theta)'(t, x, y, z), \\ p(t, x, y, z) &= p_0(z) + p'(t, x, y, z). \end{aligned}$$

Dabei bezeichnet ψ_0 jeweils den Referenzzustand der Variablen ψ und ψ' die Abweichung hiervon. Zieht man den Referenzzustand von (2.10) bis (2.12) ab, so ergibt sich:

$$\frac{\partial \rho'}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.37)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p' I_3 - \rho \mu_m (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)) = -\rho' g \mathbf{k}, \quad (2.38)$$

$$\frac{\partial (\rho \theta_m)'}{\partial t} + \nabla \cdot (\rho \theta_m \mathbf{u} - \rho \mu_h \nabla \theta_m) = s_\theta. \quad (2.39)$$

Die Gleichungen (2.13) bis (2.15) bleiben unverändert. Die Zustandsgleichung (2.9) wird zu:

$$p' = p_{00} \left(\frac{R_d}{p_{00}} \rho \theta_m \right)^{\frac{c_{pml}}{c_{vml}}} - p_0.$$

Der in COSMO implementierte Referenzzustand ergibt sich durch die Vorgabe des Temperaturprofils:

$$T_0(z) = T_{\text{SL}} + (T_{00} - T_{\text{SL}}) e^{\frac{z}{H_{\text{SC}}}} \quad (2.40)$$

mit $T_{\text{SL}} := 213,15 \text{ K}$, $T_{00} := 288,15 \text{ K}$ und $H_{\text{SC}} := 10 \text{ km}$ und der hydrostatischen Balance. Welche Rolle die Wahl des Referenzzustands für die DG-Verfahren hat, wird in Abschnitt 4.2 untersucht.

2.3. Bilanzgleichungen in geländefolgenden Koordinaten

Das Wettergeschehen hängt stark von der Orographie ab [82], es ist daher in einer realitätsnahen Simulation unerlässlich diese zu modellieren. In operationellen NWV-Systemen wird die Orographie üblicherweise durch *geländefolgende Koordinaten* modelliert. Diese transformieren das Gelände auf ein kartesisches Koordinatensystem und liefern einen stetigen Gradienten des Randes, wodurch sich die Randbedingungen vereinfachen. Insbesondere bei Finite-Differenzenverfahren entfällt die Schwierigkeit Randbedingungen für nicht kartesische Gitter zu stellen [38]. Geländefolgende Koordinaten werden von allen NWV-Systemen des Deutschen Wetterdienstes, d. h. GME, COSMO, und das in Kooperation mit dem MPI entwickelte ICON, sowie auch von Modellen anderer Dienste wie etwa WRF verwendet. Prinzipiell haben DG-Verfahren keine größeren Schwierigkeiten mit unstrukturierten Gittern, aber um bessere Vergleichbarkeit mit COSMO zu erreichen und um die vorhandenen Einleseroutinen verwenden zu können, verwendet diese Arbeit ebenfalls geländefolgende Koordinaten. Auch entfällt damit die Notwendigkeit für einen Gittergenerator.

In geländefolgenden Koordinaten sind die freien Variablen x und y rechtwinklige horizontale Koordinaten, ζ ist eine Höhenkoordinate, wobei $\zeta = 0$ als Erdoberfläche definiert ist. Mit steigender Höhe folgt ζ der Orographie schwächer, am Modelloberrand stimmt ζ mit der z -Koordinate des kartesischen Koordinatensystems überein. Die ζ -Koordinatenachse steht also im allgemeinen nicht rechtwinklig auf die x - und y -Koordinatenachse.

In dieser Arbeit werden die Koordinaten nach Gal-Chen und Somerville verwendet [38]: Sei H die Höhe des Modellgebiets und die Höhe der Orographie sei durch die Funktion $z_s(x, y)$ gegeben, dann lässt sich die Transformation von den kartesischen Koordinaten x, y, z auf die geländefolgenden Koordinaten x, y, ζ definieren durch:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \mapsto \begin{pmatrix} x \\ y \\ \zeta = \frac{H(z - z_s(x, y))}{H - z_s(x, y)} \end{pmatrix}.$$

Abbildung 2.1 veranschaulicht das x, y, ζ -Koordinatensystem anhand eines idealisierten Bergs. Um diese Transformation anwenden zu können, muss $z_s(x, y)$ zwei Mal stetig differenzierbar sein [38], die Höhenfunktion der Orographie ist also gegebenenfalls entsprechend zu glätten. Um die Bilanzgleichungen in geländefolgenden Koordinaten zu formulieren, werden metrische Terme benötigt, die durch die Jacobi-Matrix

$$A = \frac{\partial(x, y, \zeta)}{\partial(x, y, z)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{\partial z_s}{\partial x} \frac{\zeta - H}{H - z_s} & \frac{\partial z_s}{\partial y} \frac{\zeta - H}{H - z_s} & \frac{H}{H - z_s} \end{pmatrix}$$

gegeben sind. Die folgenden Rechnungen werden im Tensorkalkül durchgeführt, daher soll das geländefolgende Koordinatentripel x, y, ζ auch mit $\tilde{x}^1, \tilde{x}^2, \tilde{x}^3$ bezeichnet werden und die kartesischen Koordinaten x, y, z sollen mit x^1, x^2, x^3 bezeichnet werden. In dieser Arbeit werden die geländefolgenden Koordinaten nur für den Fall $\mu_m = \mu_h = 0 \frac{\text{m}^2}{\text{s}}$ behandelt, für die Transformation der Diffusionsterme sei auf [8] verwiesen. Die Gleichungen (2.37) bis (2.39) und (2.13) bis (2.15) lassen sich für ein allgemeines Koordinatensystem (vgl. [50] oder

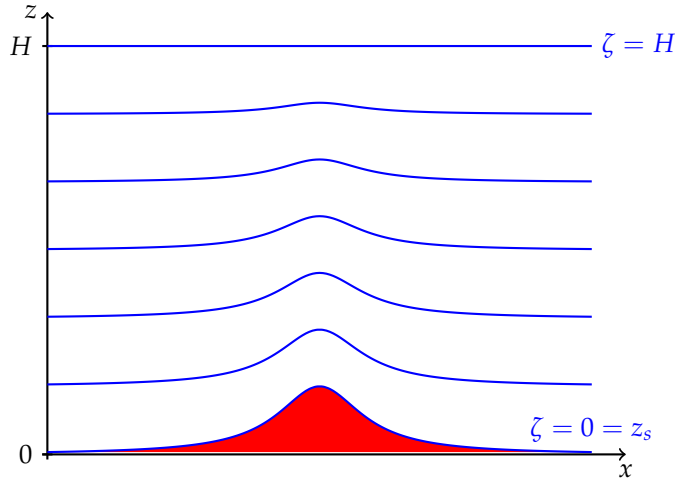


Abbildung 2.1.: geländefolgende Koordinaten nach Gal-Chen und Somerville dargestellt im kartesischen Koordinatensystem. Rot ist das Gelände eingezeichnet, blau die Isolinien der ζ -Koordinate. Man sieht $\zeta = 0$ ist identisch zur Geländeoberfläche, $\zeta = H$ ist identisch zu $z = H$. Der Berg hat das Profil *Versiera di Agnesi* (vgl. Abschnitt 4.4).

Anhang A.2) schreiben als:

$$\frac{\partial \tilde{\rho}^i}{\partial t} + \frac{\partial \tilde{C}^j}{\partial \tilde{x}^j} + \tilde{\Gamma}_{kj}^j \tilde{C}^k = 0, \quad \tilde{C}^j = \tilde{\rho} u^j, \quad (2.41)$$

$$\frac{\partial \tilde{\rho} u^i}{\partial t} + \frac{\partial \tilde{M}^{ij}}{\partial \tilde{x}^j} + \tilde{\Gamma}_{kj}^j \tilde{M}^{ik} + \tilde{\Gamma}_{jk}^i \tilde{M}^{kj} = \tilde{Q}^i, \quad \tilde{M}^{ij} = \tilde{\rho} u^i u^j + \tilde{p}' \delta_{ij}, \quad (2.42)$$

$$\frac{\partial \tilde{\rho} \theta^i}{\partial t} + \frac{\partial \tilde{T}^j}{\partial \tilde{x}^j} + \tilde{\Gamma}_{kj}^j \tilde{T}^k = \tilde{s}_\theta, \quad \tilde{T}^j = \tilde{\rho} u^j \theta, \quad (2.43)$$

$$\frac{\partial \tilde{\rho}_a}{\partial t} + \frac{\partial \tilde{W}_a^j}{\partial \tilde{x}^j} + \tilde{\Gamma}_{kj}^j \tilde{W}_a^k = \tilde{s}_a, \quad \tilde{W}_a^j = \tilde{\rho}_a u^j, \quad a \in \{v, c, r\} \quad (2.44)$$

wobei $\tilde{\Gamma}_{jk}^i$ Christoffel-Symbole der 2. Art sind, $\tilde{Q}^i = -\delta_{i3} \tilde{\rho}' g$, \tilde{s}_θ und \tilde{s}_a sind entsprechend Abschnitt 2.1.1 definiert. Die Christoffel-Symbole erscheinen als zusätzliche Quellterme. Um die Gleichungen (2.41) bis (2.44) auf Erhaltungsform zu bringen, muss das Auftreten von Christoffel-Symbolen vermieden werden. Hierzu werden die prognostischen Variablen ρ' , ρu , ρv , ρw , $\rho \theta'$, ρv , ρc , ρr in kartesischen Koordinaten formuliert, und nur die freien Variablen x, y, ζ in geländefolgenden Koordinaten. Diese Koordinatenwahl wird als *starke Erhaltungsform* bezeichnet (engl. „strong conservation form“ vgl. [50]). Für die Herleitung der prognostischen Variablen in sphärischen Koordinaten und den freien Variablen in geländefolgenden Koordinaten sei auf [74] verwiesen. Dies sind auch die von COSMO verwendeten Koordinaten, d. h. wenn die Krümmung der Erde vernachlässigt wird, verwendet COSMO die Koordinaten in denselben Systemen wie diese Arbeit. Jedoch sind die Bilanzgleichungen in COSMO nicht in Erhaltungsform. Eine Herleitung der Gleichungen aus COSMO in diesen gemischten Koordinaten findet man in [8].

Nun zur Herleitung der starken Erhaltungsform. Für $\tilde{\Gamma}_{kj}^k$ und mit der Wurzel aus der

2. Bilanzgleichungen des physikalischen Modells

Gram'schen Determinanten $\sqrt{\bar{G}} = \sqrt{\det g_{ij}}$, wobei $g_{ij} := A^T A$ der metrische Fundamentaltensor ist, gilt nach [50]:

$$\tilde{\Gamma}_{kj}^k = \frac{1}{\sqrt{\bar{G}}} \frac{\partial \sqrt{\bar{G}}}{\partial \tilde{x}^j}. \quad (2.45)$$

Setzt man (2.45) in (2.41) bis (2.44) ein und multipliziert mit $\sqrt{\bar{G}}$, so bleibt:

$$\frac{\partial \sqrt{\bar{G}} \tilde{\rho}'}{\partial t} + \frac{\partial \sqrt{\bar{G}} \tilde{C}^j}{\partial \tilde{x}^j} = 0 \quad (2.46)$$

$$\frac{\partial \sqrt{\bar{G}} \tilde{\rho} u^i}{\partial t} + \frac{\partial \sqrt{\bar{G}} \tilde{M}^{ij}}{\partial \tilde{x}^j} + \sqrt{\bar{G}} \tilde{\Gamma}_{jk}^i \tilde{M}^{kj} = \sqrt{\bar{G}} \tilde{Q}^i, \quad (2.47)$$

$$\frac{\partial \sqrt{\bar{G}} \tilde{\rho} \theta'}{\partial t} + \frac{\partial \sqrt{\bar{G}} \tilde{T}^j}{\partial \tilde{x}^j} = \sqrt{\bar{G}} \tilde{s}_\theta, \quad (2.48)$$

$$\frac{\partial \sqrt{\bar{G}} \tilde{\rho}_a}{\partial t} + \frac{\partial \sqrt{\bar{G}} \tilde{W}_a^j}{\partial \tilde{x}^j} = \sqrt{\bar{G}} \tilde{s}_a, \quad a \in \{v, c, r\} \quad (2.49)$$

wobei folgende Umformung verwendet wurde:

$$\frac{\partial \sqrt{\bar{G}} \tilde{F}^{ij}}{\partial \tilde{x}^j} = \sqrt{\bar{G}} \frac{\partial \tilde{F}^{ij}}{\partial \tilde{x}^j} + \frac{\partial \sqrt{\bar{G}}}{\partial \tilde{x}^k} \tilde{F}^{ik}.$$

Die skalaren Gleichungen sind somit für neue prognostische Variablen wieder in Erhaltungssform. Für die Impulsgleichung ist mehr Aufwand nötig. Hierzu werden zunächst in (2.47) die prognostischen Variablen in kartesische transformiert mit:

$$\tilde{\rho} u^i = \frac{\partial \tilde{x}^i}{\partial x^k} \rho u^k, \quad \tilde{M}^{ij} = \frac{\partial \tilde{x}^i}{\partial x^k} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl}, \quad \tilde{Q}^i = \frac{\partial \tilde{x}^i}{\partial x^k} Q^k, \quad (2.50)$$

es ergibt sich daraus:

$$\frac{\partial \sqrt{\bar{G}} \frac{\partial \tilde{x}^i}{\partial x^k} \rho u^k}{\partial t} + \frac{\partial \sqrt{\bar{G}} \frac{\partial \tilde{x}^i}{\partial x^k} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl}}{\partial \tilde{x}^j} + \sqrt{\bar{G}} \tilde{\Gamma}_{jk}^i \frac{\partial \tilde{x}^k}{\partial x^m} \frac{\partial \tilde{x}^j}{\partial x^l} M^{ml} = \sqrt{\bar{G}} \frac{\partial \tilde{x}^i}{\partial x^k} Q^k. \quad (2.51)$$

Der zweite Term von (2.51) lässt sich wie folgt umformen:

$$\frac{\partial \sqrt{\bar{G}} \frac{\partial \tilde{x}^i}{\partial x^k} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl}}{\partial \tilde{x}^j} = \frac{\partial \tilde{x}^i}{\partial x^k} \frac{\partial \sqrt{\bar{G}} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl}}{\partial \tilde{x}^j} + \sqrt{\bar{G}} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl} \frac{\partial}{\partial \tilde{x}^j} \frac{\partial \tilde{x}^i}{\partial x^k} \quad (2.52)$$

$$= \frac{\partial \tilde{x}^i}{\partial x^k} \frac{\partial \sqrt{\bar{G}} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl}}{\partial \tilde{x}^j} + \sqrt{\bar{G}} \frac{\partial \tilde{x}^j}{\partial x^l} M^{kl} \frac{\partial x^m}{\partial \tilde{x}^j} \frac{\partial}{\partial x^m} \frac{\partial \tilde{x}^i}{\partial x^k}. \quad (2.53)$$

Der nächste Schritt benötigt jetzt die Transformationseigenschaft der Christoffel-Symbole zwischen den beiden Koordinatensystemen, die gegeben ist durch (vgl. [50]):

$$\Gamma_{jk}^i = \tilde{\Gamma}_{pq}^s \frac{\partial x^i}{\partial \tilde{x}^s} \frac{\partial \tilde{x}^p}{\partial x^j} \frac{\partial \tilde{x}^q}{\partial x^k} + \frac{\partial x^i}{\partial \tilde{x}^s} \frac{\partial}{\partial x^j} \frac{\partial \tilde{x}^s}{\partial x^k}. \quad (2.54)$$

Für das kartesische Koordinatensystem verschwinden die Christoffel-Symbole, und durch Umstellen von (2.54) und nach Multiplikation mit $\frac{\partial \tilde{x}^s}{\partial x^i}$ ergibt sich:

$$\frac{\partial}{\partial x^j} \frac{\partial \tilde{x}^s}{\partial x^k} = -\tilde{\Gamma}_{pq}^s \frac{\partial \tilde{x}^p}{\partial x^j} \frac{\partial \tilde{x}^q}{\partial x^k}. \quad (2.55)$$

Wird jetzt (2.55) in (2.53) eingesetzt so lässt sich der dritten Term von (2.51) eliminieren. Nach Multiplikation mit $\frac{\partial x^k}{\partial \tilde{x}^i}$ bleibt für die Impulsgleichung:

$$\frac{\partial \sqrt{\tilde{G}} \rho u^i}{\partial t} + \frac{\partial \sqrt{\tilde{G}} \frac{\partial \tilde{x}^j}{\partial x^l} M^{il}}{\partial \tilde{x}^j} = \sqrt{\tilde{G}} Q^k.$$

Damit ist auch die Impulsgleichung wieder in Erhaltungsform. Werden in den skalaren Bilanzgleichungen die prognostischen Variablen noch analog zu (2.50) in kartesische Koordinaten transformiert, so sind die Bilanzgleichungen in der starken Erhaltungsform gegeben durch:

$$\begin{aligned} \frac{\partial \sqrt{\tilde{G}} \rho'}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho u \\ \rho v \\ \frac{\partial \zeta}{\partial x} \rho u + \frac{\partial \zeta}{\partial y} \rho v + \frac{\partial \zeta}{\partial z} \rho w \end{pmatrix} &= 0, \\ \frac{\partial \sqrt{\tilde{G}} \rho u}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho u u + p' \\ \rho u v \\ \frac{\partial \zeta}{\partial x} (\rho u u + p') + \frac{\partial \zeta}{\partial y} \rho u v + \frac{\partial \zeta}{\partial z} \rho u w \end{pmatrix} &= 0, \\ \frac{\partial \sqrt{\tilde{G}} \rho v}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho u v \\ \rho v v + p' \\ \frac{\partial \zeta}{\partial x} \rho u v + \frac{\partial \zeta}{\partial y} (\rho v v + p') + \frac{\partial \zeta}{\partial z} \rho v w \end{pmatrix} &= 0, \\ \frac{\partial \sqrt{\tilde{G}} \rho w}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho u w \\ \rho v w \\ \frac{\partial \zeta}{\partial x} \rho u w + \frac{\partial \zeta}{\partial y} \rho v w + \frac{\partial \zeta}{\partial z} (\rho w w + p') \end{pmatrix} &= -\sqrt{\tilde{G}} \rho' g, \\ \frac{\partial \sqrt{\tilde{G}} (\rho \theta)'}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho u \theta \\ \rho v \theta \\ \frac{\partial \zeta}{\partial x} \rho u \theta + \frac{\partial \zeta}{\partial y} \rho v \theta + \frac{\partial \zeta}{\partial z} \rho w \theta \end{pmatrix} &= \sqrt{\tilde{G}} s_\theta, \\ \frac{\partial \sqrt{\tilde{G}} \rho_a}{\partial t} + \nabla \cdot \sqrt{\tilde{G}} \begin{pmatrix} \rho_a u \\ \rho_a v \\ \frac{\partial \zeta}{\partial x} \rho_a u + \frac{\partial \zeta}{\partial y} \rho_a v + \frac{\partial \zeta}{\partial z} \rho_a w \end{pmatrix} &= \sqrt{\tilde{G}} s_a, \quad a \in \{v, c, r\}. \end{aligned} \tag{2.56}$$

Für Simulationen mit Orographie werden damit die prognostischen Variablen $\sqrt{\tilde{G}} \rho'$, $\sqrt{\tilde{G}} \rho u$, $\sqrt{\tilde{G}} \rho v$, $\sqrt{\tilde{G}} \rho w$, $\sqrt{\tilde{G}} (\rho \theta)'$, $\sqrt{\tilde{G}} \rho_v$, $\sqrt{\tilde{G}} \rho_c$ und $\sqrt{\tilde{G}} \rho_r$ verwendet.

In diesem Kapitel wurden die voll-kompressiblen feuchten Euler-Gleichungen für eine Atmosphäre über einer glatten Orographie mit einem einfachen Turbulenzmodell und einem Kessler-Schema für warmen Niederschlag eingeführt. Das nächste Kapitel widmet sich der numerischen Behandlung dieser Gleichungen.

3. Diskontinuierliche Galerkin-Verfahren

In diesem Kapitel wird die numerische Diskretisierung zur Lösung der Atmosphärendynamikbilanzgleichungen (2.56) aus dem vorangegangenen Kapitel beschrieben. Die räumliche Diskretisierung erfolgt unabhängig von der zeitlichen Diskretisierung und verwendet auch ein anderes Verfahren. Dieses Vorgehen ist als Linienmethode (engl. „method of lines“) bekannt. Für die räumliche Diskretisierung wird ein *diskontinuierliches elementbasiertes Galerkin¹-Verfahren* (DG-Verfahren) verwendet. Die DG-Verfahren sollen hier für folgende Advektionsgleichung des Variablenvektors

$$\mathbf{q}(t, \mathbf{x}) := \begin{pmatrix} \rho' \\ \rho \mathbf{u} \\ (\rho \theta)' \\ \rho_v \\ \rho_c \\ \rho_r \end{pmatrix} (t, \mathbf{x})$$

auf dem D -dimensionalen beschränkten Gebiet Ω , wobei $D = 2$ oder $D = 3$, mit wohldefinierten Anfangs- und Randbedingungen skizziert werden:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{q}) = \mathbf{S}(\mathbf{q}). \quad (3.1)$$

In diesem Abschnitt wird Ω als polygonal berandet betrachtet. Die Orographie wird durch die Gleichungen in geländefolgenden Koordinaten und die Randbedingungen berücksichtigt. Das Gebiet Ω wird in, *Elemente* genannte, Teilgebiete $\Omega_1, \dots, \Omega_N$ zerlegt, deren Inneres paarweise elementfremd ist. Benachbarte Teilgebiete haben gemeinsame Randkanten bzw. -flächen. Die Menge aller Elemente heißt *Gitter*. Die Komponenten q_j des Variablenvektors \mathbf{q} sind Funktionen aus dem Raum $V = L^2(\Omega, \mathbb{R})$, also die Menge der *quadratintegriblen Funktionen*. Der Raum V heißt *Ansatz- und Testraum*, da mit seinen Funktionen ein Lösungsansatz erstellt wird und sie auch als Testfunktionen (s. unten) dienen. Betrachtet man ein beliebiges aber festes Element Ω_i und sei \mathcal{B}_i eine Basis von $V|_{\Omega_i}$, so lässt sich auf Ω_i für \mathbf{q} folgende Reihe ansetzen:

$$\mathbf{q}(t, \mathbf{x})|_{\Omega_i} = \mathbf{q}_i = \sum_{\phi \in \mathcal{B}_i} \mathbf{q}_{i,\phi}(t) \phi(\mathbf{x}), \quad (3.2)$$

mit geeigneten Koeffizienten $\mathbf{q}_{i,\phi}$.

Beim Galerkin-Ansatz werden Polynome als Basis gewählt, die Bilanzgleichung (3.1) wird über jedem Gebiet Ω_i mit einer Testfunktion ϕ multipliziert und über Ω_i integriert. Mit partieller Integration entsteht schließlich die *schwache Formulierung*²:

$$\int_{\Omega_i} \frac{\partial \mathbf{q}_i}{\partial t} \phi \, d\Omega + \int_{\Gamma_i} \mathbf{F}(\mathbf{q}_i) \cdot \mathbf{n} \phi \, d\Gamma - \int_{\Omega_i} \mathbf{F}(\mathbf{q}_i) \nabla \phi \, d\Omega = \int_{\Omega_i} \mathbf{S}(\mathbf{q}_i) \phi \, d\Omega. \quad (3.3)$$

¹nach Boris Grigorjewitsch Galerkin (*1871 in Polozk, †1945 in Leningrad)

²Im Gegensatz zur *starken Formulierung* ist die schwache Formulierung ohne weitere Umformung lokal erhaltend [39, Abschnitt 2.2.2].

3. Diskontinuierliche Galerkin-Verfahren

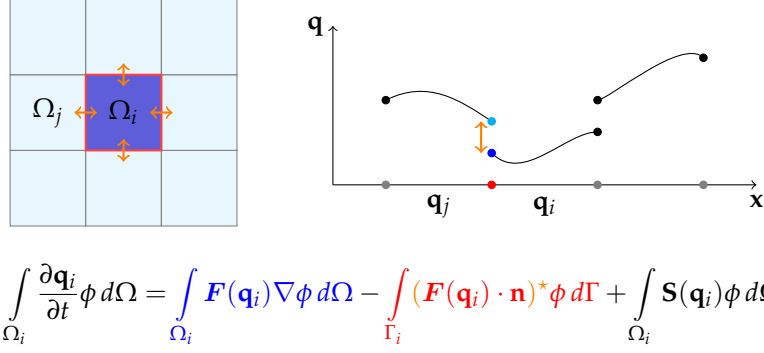


Abbildung 3.1.: Die Tendenz der prognostischen Variablen berechnet sich beim DG-Verfahren durch das Volumenintegral (blau eingefärbt), Randintegral (rot eingefärbt) über dem numerischen Fluss (orange dargestellt) zwischen dem internen und dem externen Element und dem Quellterm.

Hierbei ist \mathbf{n} der nach außen zeigende Einheitsvektor orthogonal zu $\Gamma_i := \partial\Omega_i$.

Zwischen den Elementen wird keine Stetigkeit der Lösung gefordert (daher der Name „diskontinuierliches Galerkin-Verfahren“), d. h. das zweite Integral in (3.3) ist nicht wohldefiniert, da es im allgemeinen einen Sprung der Lösung zwischen den Elementen gibt. Die Eindeutigkeit wird durch die Verwendung eines *numerischen Flusses* $(\mathbf{F}(\mathbf{q}) \cdot \mathbf{n})^*$ anstelle des Flusses $\mathbf{F}(\mathbf{q}) \cdot \mathbf{n}$ hergestellt, man erhält also:

$$\int_{\Omega_i} \frac{\partial \mathbf{q}_i}{\partial t} \phi \, d\Omega + \int_{\Gamma_i} (\mathbf{F}(\mathbf{q}_i) \cdot \mathbf{n})^* \phi \, d\Gamma - \int_{\Omega_i} \mathbf{F}(\mathbf{q}_i) \nabla \phi \, d\Omega = \int_{\Omega_i} \mathbf{S}(\mathbf{q}_i) \phi \, d\Omega. \quad (3.4)$$

Nun sucht man ein \mathbf{q} , das diese Gleichung für jede Basisfunktion ϕ aus \mathcal{B}_i auf jedem Element Ω_i erfüllt.

Die Informationsausbreitung zwischen den Elementen erfolgt also nur über den Fluss über die Ränder, wie bei Finite-Volumen-Verfahren. Haben die Elemente Ω^{int} und Ω^{ext} eine gemeinsame Kante mit einem Randpunkt \mathbf{x}_0 , so ist auf dem Element Ω^{int} der numerische Fluss $(\mathbf{F}(\mathbf{q}^{\text{int}}) \cdot \mathbf{n})^*(\mathbf{x}_0)$ definiert als Funktion von $\mathbf{q}^{\text{int}}(\mathbf{x}_0)$ und $\mathbf{q}^{\text{ext}}(\mathbf{x}_0)$, man kann also $(\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{int}}, \mathbf{q}^{\text{ext}})$ schreiben. Folgende Bedingungen muss der numerische Fluss erfüllen [27]:

$$(\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{int}}, \mathbf{q}^{\text{int}}) = \mathbf{F}(\mathbf{q}^{\text{int}}) \cdot \mathbf{n}, \quad (3.5)$$

$$(\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{int}}, \mathbf{q}^{\text{ext}}) = -(\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{ext}}, \mathbf{q}^{\text{int}}), \quad (3.6)$$

$$\mathbf{q}' \leq \mathbf{q}'' \Rightarrow (\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}', \mathbf{q}^{\text{ext}}) \leq (\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}'', \mathbf{q}^{\text{ext}}), \quad (3.7)$$

$$\mathbf{q}' \leq \mathbf{q}'' \Rightarrow (\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{int}}, \mathbf{q}') \geq (\mathbf{F} \cdot \mathbf{n})^*(\mathbf{q}^{\text{int}}, \mathbf{q}''). \quad (3.8)$$

Die Bedingungen (3.5) und (3.6) sichern die Konsistenz mit den kontinuierlichen Gleichungen und die lokale Erhaltung der Größe \mathbf{q} . Die letzten beiden Monotonieeigenschaften sind für die Stabilität wichtig (vgl. [27]).

In Abbildung 3.1 werden die Terme von (3.4) für den zweidimensionalen Fall veranschaulicht. Im folgenden Abschnitt wird beschrieben, wie Diskretisierungen \mathbf{q}_i^h für die Lösung \mathbf{q}_i der Advektionsgleichung (3.1) gefunden werden. Im darauf folgenden Abschnitt wird das Lösungsverfahren auf die Advektions-Diffusionsgleichungen erweitert. Der Rest des Kapitels diskutiert Rand- und Anfangsbedingungen und schließt mit Einzelheiten zur Implementierung.

3.1. Modales diskontinuierliches Galerkin-Verfahren

Die hier verwendete Variante der DG-Verfahren ist ein modales Verfahren in der schwachen Form auf einem quadrilateralen Gitter, das Legendre-Polynome auf einem Referenzelement definiert. Der im Rahmen dieser Arbeit entwickelte Dynamikkern DG-COSMO kann als zwei- oder als dreidimensionales Modell betrieben werden. Hier wird nur das dreidimensionale Modell beschrieben, da sich das zweidimensionale durch Weglassen der y -Richtung ergibt.

Die Berechnung findet in den geländefolgenden (x, y, ζ) -Koordinaten aus Abschnitt 2.3 statt. Das Gebiet Ω wird in ein quadrilaterales konformes Gitter zerlegt. D. h. die Elemente sind Quader, deren Seitenflächen parallel zu den Koordinatenebenen sind und die Eckpunkte zweier benachbarter Elemente liegen aufeinander. Die Elemente werden mit Ω_{ijk} bezeichnet, dabei sind i und j die horizontalen Indizes in x - und y -Richtung und laufen von 1 bis i_{end} und 1 bis j_{end} . Der in ζ -Richtung laufende Index ist k und läuft von 1 bis k_{end} , dabei bezeichnet $k = 1$ die Elemente am Modelloberrand und $k = k_{\text{end}}$ sind die Elemente am Boden³. Die DG-Verfahren sind Ansatzverfahren, die Variablen sind also an allen Punkten eines Elements definiert (vgl. Abschnitt 1.2). Dieses Gitter ist identisch zum COSMO-Gitter, jedoch sind die Variablen in COSMO nicht an allen Punkten definiert. Das Gitter ist horizontal homogen, vertikal ist eine Streckung möglich. Für die später behandelten Testfälle wird die Streckung der Elemente mit der Gebietshöhe H definiert über:

$$\begin{aligned}\Delta\zeta_k &:= \hat{\zeta}_k - \hat{\zeta}_{k+1}, \\ \hat{\zeta}_k &:= H(\beta\eta_k^2 + (1-\beta)\eta_k), \\ \eta_k &:= 1 - \frac{k-1}{k_{\text{end}}}, \quad k = 1, \dots, k_{\text{end}} + 1,\end{aligned}\tag{3.9}$$

mit dem Streckungsparameter $\beta \in [0, 1]$. Die Elemente Ω_{ijk} haben das Abmaß $\Delta x \times \Delta y \times \Delta\zeta_k$. Auf den Elementen werden lokale Koordinaten ξ eingeführt: Es sei $E = [-1, 1]^D$ der Einheitswürfel und die Transformation auf Ω_{ijk} erfolgt durch:

$$\begin{aligned}\text{Tr}_{ijk} : E &\rightarrow \Omega_{ijk} \\ \xi &\mapsto \mathbf{x} = \mathbf{J}_k \xi + \mathbf{m}_{ijk} \quad \text{mit} \quad \mathbf{J}_k := \frac{1}{2} \begin{pmatrix} \Delta x & 0 & 0 \\ 0 & \Delta y & 0 \\ 0 & 0 & \Delta\zeta_k \end{pmatrix},\end{aligned}$$

wobei \mathbf{m}_{ijk} der Mittelpunkt von Ω_{ijk} ist. Durch die lokalen Koordinaten kann in jedem Element dieselbe Basis verwendet werden. Ist $\phi(\xi)$ eine auf E definierte Basisfunktion, so ist $\phi(\text{Tr}_{ijk}^{-1}(\mathbf{x}))$ die Basisfunktion auf Ω_{ijk} . Für $\text{Tr}_{ijk}^{-1}(\mathbf{x})$ schreibt man auch $\xi(\mathbf{x})$ mit unterdrückten Indizes, da sie aus dem Zusammenhang eindeutig rekonstruiert werden können.

Für die Basis eines Elements Ω_{ijk} des diskretisierten Ansatz- und Testraums $V_h \subset V|_{\Omega_{ijk}}$ bieten sich zwei kanonische Definitionen an, die auch beide in der Literatur Verwendung finden. Für ein gegebenes κ und das l -te normierte Legendre-Polynom \tilde{P}_l (vgl. Anhang A.3) ist

$$\begin{aligned}\mathcal{B}_{mb} &:= \left\{ \phi(\xi) \mid \phi(\xi) = \prod_{i=1}^D \tilde{P}_{l_i}(\xi_i), \sum_{i=1}^D l_i \leq \kappa \right\} && \text{die Minimalbasis und} \\ \mathcal{B}_{vt} &:= \left\{ \phi(\xi) \mid \phi(\xi) = \prod_{i=1}^D \tilde{P}_{l_i}(\xi_i), l_i \leq \kappa \right\} && \text{die volle Tensorproduktbasis.}\end{aligned}$$

³Diese Konvention ist bei meteorologischen Modellen üblich.

3. Diskontinuierliche Galerkin-Verfahren

Der diskretisierte Ansatz- und Testraum ist also ein endlichdimensionaler Funktionenraum, dessen Basis elementweise aus Produkten von normierten Legendre-Polynomen besteht. Die Basis \mathcal{B}_{mb} hat die minimale Anzahl von Polynomen, um ein Verfahren der formalen Ordnung $\kappa + 1$ zu erhalten. In Abschnitt 4.1.3 wird die Minimalbasis mit dem vollen Tensorprodukt verglichen. Die Mächtigkeit $d := \dim V_h$ der Minimalbasis für D physikalische Raumdimensionen lässt sich als kombinatorisches Problem formulieren: Es werden κ Variablen aus der Menge $\{1, x_1, \dots, x_D\}$ ausgewählt, wobei die Reihenfolge beliebig ist und jede Variable mehrfach ausgewählt werden darf. Die Kombinatorik liefert hierfür (vgl. [21, Gleichung (16.5)]) die Lösung:

$$d = \binom{D + \kappa}{\kappa} = \frac{(\kappa + D)!}{D! \kappa!}. \quad (3.10)$$

Die volle Tensorproduktbasis hat die Dimension

$$d = (\kappa + 1)^D. \quad (3.11)$$

Für die folgende Betrachtung wird für die Basisfunktionen eine Aufzählung ϕ_1, \dots, ϕ_d festgelegt. Die Gleichung (3.4) kann für ein Element Ω_{ijk} und die l -te Basisfunktion ϕ_l wie folgt geschrieben werden:

$$\begin{aligned} \int_{\Omega_{ijk}} \frac{\partial \mathbf{q}_{ijk}}{\partial t} \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Omega + \int_{\Gamma_{ijk}} (\mathbf{F}(\mathbf{q}_{ijk}) \cdot \mathbf{n})^* \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Gamma - \int_{\Omega_{ijk}} \mathbf{F}(\mathbf{q}_{ijk}) \nabla \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Omega \\ = \int_{\Omega_{ijk}} \mathbf{S}(\mathbf{q}_{ijk}) \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Omega, \end{aligned}$$

mit $\Gamma_{ijk} := \partial\Omega_{ijk}$. Transformiert man jetzt auf lokale Koordinaten so ist mit $A := \partial E$

$$\int_E \frac{\partial \mathbf{q}_{ijk}}{\partial t} \phi_l J_k dE + \int_A (\mathbf{F}(\mathbf{q}_{ijk}) \cdot \mathbf{n})^* \phi_l J_{\partial,k} dA - \int_E \mathbf{F}(\mathbf{q}_{ijk}) \mathbf{J}_k^{-1} \nabla \phi_l J_k dE = \int_E \mathbf{S}(\mathbf{q}_{ijk}) \phi_l J_k dE, \quad (3.12)$$

wobei $J_k = |\det \mathbf{J}_k|$ die *Jakobi-Determinante* der Koordinatentransformation von E auf das Element Ω_{ijk} ist und $J_{\partial,k} = |\det \mathbf{J}_{\partial,k}|$ die *Jakobi-Determinante* der Koordinatentransformation von A auf Γ_{ijk} . Setzt man für \mathbf{q}_{ijk} die Diskretisierung

$$\mathbf{q}_{ijk}(t, \mathbf{x}) \approx \mathbf{q}_{ijk}^h := \sum_{r=1}^d \mathbf{q}_{ijk,r}(t) \phi_r(\boldsymbol{\xi}(\mathbf{x})) \quad (3.13)$$

in (3.12) ein, so erhält man nach kurzer Rechnung:

$$\sum_{r=1}^d \frac{\partial \mathbf{q}_{ijk,r}}{\partial t} \int_E \phi_l \phi_r dE = \int_E \mathbf{S}(\mathbf{q}_{ijk}^h) \phi_l dE - \frac{J_{\partial,k}}{J_k} \int_A (\mathbf{F}(\mathbf{q}_{ijk}^h) \cdot \mathbf{n})^* \phi_l dA + \int_E \mathbf{F}(\mathbf{q}_{ijk}^h) \mathbf{J}_k^{-1} \nabla \phi_l dE. \quad (3.14)$$

Ein wesentlicher Vorteil des modalen Ansatzes ist die Orthogonalität der Basisfunktionen, d. h.

$$\int_E \phi_l \phi_r dE = \delta_{lr},$$

damit entfällt die Notwendigkeit ein (wenn auch kleines) gekoppeltes Gleichungssystem zu lösen, da (3.14) sich vereinfachen lässt zu:

$$\frac{\partial \mathbf{q}_{ijk,l}}{\partial t} = \int_E \mathbf{S}(\mathbf{q}_{ijk}^h) \phi_l dE - \frac{J_{\partial,k}}{J_k} \int_A (\mathbf{F}(\mathbf{q}_{ijk}^h) \cdot \mathbf{n})^* \phi_l dA + \int_E \mathbf{F}(\mathbf{q}_{ijk}^h) \mathbf{J}_k^{-1} \nabla \phi_l dE. \quad (3.15)$$

Die Wahl des numerischen Flusses beeinflusst die Eigenschaften des DG-Verfahrens. Prinzipiell kann jeder numerische Fluss, der auch für Finite-Volumen-Verfahren verwendet wird auch für DG-Verfahren verwendet werden. In [69] werden mehrere numerische Flüsse miteinander verglichen auf Effizienz und Genauigkeit. Als numerischer Fluss wird in dieser Arbeit der *Rusanov-Fluss*, auch als lokaler *Lax-Friedrichs-Fluss* bezeichnet, verwendet. Für zwei benachbarte Elemente Ω^{ext} und Ω^{int} berechnet sich der Rusanov-Fluss für einen Punkt \mathbf{x} auf der gemeinsamen Randkante durch:

$$\left(\mathbf{F}(\mathbf{q}^{\text{int}}(\mathbf{x})) \cdot \mathbf{n}^{\text{int}}(\mathbf{x}) \right)^* = \frac{1}{2} \left((\mathbf{F}(\mathbf{q}^{\text{int}}(\mathbf{x})) + \mathbf{F}(\mathbf{q}^{\text{ext}}(\mathbf{x}))) \cdot \mathbf{n}^{\text{int}}(\mathbf{x}) - \lambda (\mathbf{q}^{\text{ext}}(\mathbf{x}) - \mathbf{q}^{\text{int}}(\mathbf{x})) \right).$$

Der Term $-\lambda (\mathbf{q}^{\text{ext}}(\mathbf{x}) - \mathbf{q}^{\text{int}}(\mathbf{x}))$ sorgt für zusätzliche *numerische Viskosität*, wodurch das Verfahren stabilisiert wird. Hierbei wird für λ der Eigenwert von \mathbf{F} mit dem größten Betrag verwendet. Für die Euler-Gleichungen ergibt sich:

$$\lambda = |\mathbf{u} \cdot \mathbf{n}| + \sqrt{\frac{c_{pml}}{c_{vml}} R_m T} = |\mathbf{u} \cdot \mathbf{n}| + \sqrt{\frac{c_{pml}}{c_{vml}} \frac{p}{\rho}}.$$

Der Rusanov-Fluss ist sehr diffusiv, auf der anderen Seite ist er aber sehr einfach zu implementieren, und mit steigender Verfahrensordnung nimmt die Bedeutung des numerischen Flusses ab [27]. So ist bei DG-Verfahren, wo eine hohe Ordnung einfach zu erreichen ist, im Vergleich zu Finite-Volumen-Verfahren die Wahl des numerischen Flusses nicht so entscheidend.

Die Integrale werden durch Quadraturformeln genähert. In dieser Arbeit werden Tensorprodukte aus *Legendre-Gauß-Lobatto-Quadraturformeln*⁴ (LGL-Formeln) verwendet. Für eine LGL-Formel mit n Punkten für das eindimensionale Intervall $[-1, 1]$ sind die Quadraturpunkte gegeben durch: ξ_1 ist -1 , ξ_2, \dots, ξ_{n-1} sind die Nullstellen von $\frac{d}{d\xi} P_{n-2}(\xi)$ in aufsteigender Reihenfolge und ξ_n ist 1 , dabei ist P_{n-2} das $(n-2)$ -te Legendre-Polynom. Die zugehörigen Gewichte w_i lassen sich berechnen durch:

$$w_{1,n} = \frac{2}{n(n-1)}, \quad w_i = \frac{2}{n(n-1)P_{n-1}(\xi_i)^2} \quad \text{für } i = 2, \dots, n-1.$$

In Abbildung 3.2 werden die LGL-Punkte für $n = 5$ und $n = 6$ graphisch dargestellt. Die LGL-Formeln integrieren Polynome bis einschließlich Grad $2n - 3$ exakt [89]. Dies ist optimal unter der Nebenbedingung, dass auf dem Rand Quadraturpunkte liegen. Durch diese Nebenbedingung können im dreidimensionalen (zweidimensionalen) Raum die Quadraturpunkte auf den Randflächen (-linien) auch für das Volumenintegral (Flächenintegral) wiederverwendet werden, wodurch Rechenzeit eingespart werden kann. In den Oberflächenintegralen treten Polynome der Ordnung 2κ und in den Volumenintegralen der Ordnung $2\kappa - 1$ auf, der kleinste Wert für n , um exakt integrieren zu können, ist also

$$n = \kappa + 2. \quad (3.16)$$

⁴Die Definition der LGL-Formeln, Fehlerterme, eine Tabelle bis $n = 6$ und weiterführende Referenzen können etwa in [89] nachgeschlagen werden.

3. Diskontinuierliche Galerkin-Verfahren

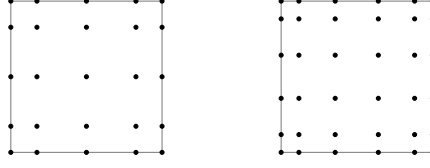


Abbildung 3.2.: Anordnung zweidimensionaler Legendre-Gauß-Lobatto-Quadraturpunkte für $n = 5$ (links) und $n = 6$ (rechts). Die Quadraturpunkte häufen sich in Randnähe.

3.2. Lokales DG für Viskosität

In diesem Abschnitt wird beschrieben, wie die Diffusionsterme der Bilanzgleichungen behandelt werden. Dieser Teil ist nur für kartesische Koordinaten implementiert, d. h. bei Verwendung von Orographie wird der Gradient approximiert durch die Ableitung der physikalischen Größe, die Änderung durch die Christoffel-Symbole wird weggelassen. Für die Gleichungen (2.10) bis (2.15) ist:

$$\begin{aligned} (\nabla\psi)_k &\approx \frac{\partial\psi}{\partial x_k}, & \nabla \cdot \mathbf{v} &= \frac{\partial v^j}{\partial x_j} + \Gamma_{kj}^j V^k, \\ (\nabla\mathbf{v}^i)_k &\approx \frac{\partial v^i}{\partial x_k}, & (\nabla \cdot \mathbf{T})^i &= \frac{\partial T^{ij}}{\partial x_j} + \Gamma_{kj}^j T^{ik} + \Gamma_{jk}^i T^{kj}, \end{aligned} \quad \text{aber}$$

für ein Skalar ψ , einen Vektor \mathbf{v} und einen Tensor 2. Stufe \mathbf{T} . Diese Approximation ist als weitere Vereinfachung des Turbulenzmodells zu verstehen. Bei den später betrachteten Testfällen spielt diese Vereinfachung jedoch keine Rolle, da keiner sowohl Orographie als auch das Turbulenzmodell verwendet.

Das in dieser Arbeit verwendete Verfahren für die Diskretisierung der Diffusionsterme ist das *lokale diskontinuierliche Galerkin-Verfahren* (LDG) [28]. Es wird nur für die Impulsgleichung (2.38) vorgestellt, für die potentielle Temperaturgleichung (2.12) und die Gleichungen der Feuchtegrößen (2.13) bis (2.15) läuft die Diskretisierung analog. Die zu Grunde liegende Idee hierbei ist, dass Gleichung (2.38) in ein Gleichungssystem umgeschrieben wird, in dem nur Ableitungen erster Ordnung auftreten:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p' \mathbf{I}_3 - \rho \mu_m (\mathbf{V} + \mathbf{V}^T)) = -\rho' g \mathbf{k}, \quad (3.17)$$

$$\nabla \mathbf{u} = \mathbf{V}, \quad (3.18)$$

mit der Hilfsvariablen \mathbf{V} . Zuerst wird (3.18) mit dem DG-Ansatz diskretisiert. D. h. es wird mit einer Basisfunktion ϕ_l multipliziert und über ein Element Ω_{ijk} integriert. Nach anschließender partieller Integration der linken Seite ist das Ergebnis:

$$\int_{\Gamma_{ijk}} \mathbf{u}_{ijk} \otimes \mathbf{n} \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Gamma - \int_{\Omega_{ijk}} \mathbf{u}_{ijk} \otimes \nabla \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Omega = \int_{\Omega_{ijk}} \mathbf{V}_{ijk} \phi_l(\boldsymbol{\xi}(\mathbf{x})) d\Omega.$$

Wird jetzt auf das Referenzelement E transformiert, der numerische Fluss $(\mathbf{u}_{ijk} \mathbf{n}^T)^*$ anstelle des Flusses $\mathbf{u}_{ijk} \otimes \mathbf{n}$ im Randintegral verwendet und die Hilfsvariable \mathbf{V} mit der Basis entwickelt zu

$$\mathbf{V}_{ijk} = \sum_{r=1}^d \mathbf{V}_{ijk,r}(t) \phi_r(\boldsymbol{\xi}),$$

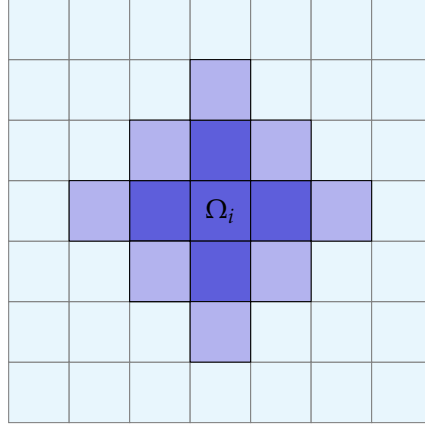


Abbildung 3.3.: Der Diskretisierungsstern für LDG. Die tiefblauen Elemente enthalten die Information für den Advektionsanteil. Die in hellerem blau eingefärbten Elemente werden für die Flussberechnung der Hilfsvariable S benötigt. Die Größe des Diskretisierungssterns ist nur von der Ordnung der Ableitungsoperatoren abhängig, nicht von der Verfahrensordnung.

so bleibt nach kurzer Rechnung:

$$V_{ijk,l} = \frac{|\det J_{\partial,k}|}{|\det J_k|} \int_A (\mathbf{u}_{ijk} \otimes \mathbf{n})^* \phi_l dA - \int_E \mathbf{u}_{ijk} \otimes \nabla \phi_l \mathbf{J}_k^{-1} dE. \quad (3.19)$$

Sind Ω^{int} und Ω^{ext} zwei benachbarte Elemente so wird der numerische Fluss in Gleichung (3.19) für einen Punkt \mathbf{x} aus $\Omega^{\text{int}} \cap \Omega^{\text{ext}}$ definiert durch:

$$\left(\mathbf{u}^{\text{int}}(\mathbf{x}) \otimes \mathbf{n}^{\text{int}}(\mathbf{x}) \right)^* := \frac{1}{2} (\mathbf{u}^{\text{int}}(\mathbf{x}) \otimes \mathbf{n}^{\text{int}}(\mathbf{x}) + \mathbf{u}^{\text{ext}}(\mathbf{x}) \otimes \mathbf{n}^{\text{int}}(\mathbf{x})).$$

Hier kann auf einen zusätzlichen Diffusionsterm verzichtet werden, da der physikalische Diffusionsterm das Verfahren bereits stabilisiert. Nachdem \mathbf{V} ausgerechnet ist, wird es in Gleichung (3.17) eingesetzt und das DG-Verfahren aus Abschnitt 3.1 wird auf die Gleichung (3.17) angewandt. Wenn für alle prognostischen Variablen die Hilfsvariablen für die Viskositätsterme ausgerechnet und eingesetzt sind, ist die Diskretisierung des viskosen Falls formal auch durch Gleichung (3.15) beschrieben. Die zeitliche Diskretisierung behandelt folglich diese Gleichung.

Der Diskretisierungsstern für LDG ist in Abbildung 3.3 dargestellt. Die benötigte Information ist in den direkten Nachbarelementen und deren Nachbarelementen enthalten, also immer noch recht lokal. LDG ist sehr einfach zu implementieren, insbesondere wenn schon ein DG-Rahmenwerk besteht, muss wenig vom bestehenden Code geändert werden. Jedoch müssen für jede Gleichung mit einem Ableitungsoperator zweiter Ordnung D Gleichungen im skalaren Fall und D^2 Gleichungen im vektorwertigen Fall zusätzlich gelöst werden. Für die trockenen dreidimensionalen Euler-Gleichungen mit Diffusionstermen sind dies bereits 12 Gleichungen. Der Rechenaufwand für LDG ist daher sehr hoch.

3.3. Zeitintegration

Die räumlich diskretisierten Gleichungen (3.15) bilden ein gekoppeltes System gewöhnlicher Differentialgleichungen in der Zeit, das sich schreiben lässt als:

$$\frac{d\mathbf{q}^h}{dt} = \mathcal{D}(\mathbf{q}^h) + \mathcal{P}(\mathbf{q}^h), \quad \text{mit } \mathbf{q}^h(t=0) = \mathbf{q}^{h,\text{init}}$$

wobei der Operator \mathcal{P} die Terme der physikalischen Parametrisierungen sind, d. h. die Quellterme s_v, s_c, s_r und s_θ . Der Operator \mathcal{D} besteht aus den Termen der Dynamik der räumlichen Diskretisierung. Die Parametrisierungen und die Dynamik werden mit verschiedenen Zeitschritten integriert. Es sei Δt der Zeitschritt der Dynamik und $\Delta\tau = \sigma \Delta t$ für eine positive ganze Zahl σ der Zeitschritt der Parametrisierungen.

Es sollen hier zwei verschiedene Aufteilungsstrategien zwischen den Parametrisierungen und der Dynamik betrachtet werden (die gleichen wie in [93]). Der Zeitpunkt $n \Delta\tau$ wird durch τ^n bezeichnet und es wird angenommen, dass die Lösung \mathbf{q}^h zur Zeit $t = \tau^n$ bekannt ist und für das Intervall $]\tau^n, \tau^{n+1}]$ berechnet werden soll.

Operatoraufteilung. Bei der *Operatoraufteilung* (auch *Zeitaufteilung* genannt) werden die Prozesse \mathcal{P} und \mathcal{D} nacheinander abgearbeitet, es rechnet also ein Prozess mit dem Ergebnis des anderen Prozesses weiter. Für eine Hilfsvariable \mathbf{q}^* wird

$$\frac{d\mathbf{q}^*}{dt} = \mathcal{P}(\mathbf{q}^*), \quad \text{mit } \mathbf{q}^*(t = \tau^n) = \mathbf{q}^h(t = \tau^n)$$

um einen $\Delta\tau$ -Zeitschritt integriert. Anschließend wird

$$\frac{d\mathbf{q}^h}{dt} = \mathcal{D}(\mathbf{q}^h), \quad \text{mit } \mathbf{q}^h(t = \tau^n) = \mathbf{q}^*(t = \tau^{n+1})$$

um σ -viele Δt -Zeitschritte integriert.

Prozessaufteilung. Bei der *Prozessaufteilung* beginnen beide Prozesse \mathcal{P} und \mathcal{D} beim gleichen Anfangswert. Für die Hilfsvariable \mathbf{q}^* wird

$$\frac{d\mathbf{q}^*}{dt} = \mathcal{P}(\mathbf{q}^*), \quad \text{mit } \mathbf{q}^*(t = \tau^n) = \mathbf{q}^h(t = \tau^n)$$

um einen $\Delta\tau$ -Zeitschritt integriert. Die Tendenz $\mathcal{P}(\mathbf{q}^*)$ wird für den Prozess \mathcal{D} als konstanter Quellterm auf dem Intervall $]\tau^n, \tau^{n+1}]$ betrachtet, während

$$\frac{d\mathbf{q}^h}{dt} = \mathcal{D}(\mathbf{q}^h) + \mathcal{P}(\mathbf{q}^*)$$

um σ -viele Δt -Zeitschritte integriert wird.

Durch die Aufteilung können die Prozesse \mathcal{D} und \mathcal{P} einzeln und mit unterschiedlichen Verfahren integriert werden. In der vorliegenden Arbeit werden *m-stufige explizite Runge-Kutta-Verfahren* (kurz: RK-Verfahren) für die Dynamik verwendet. Diese werden häufig zur Integration gewöhnlicher Differentialgleichungen eingesetzt. Im Folgenden werden RK-Verfahren in der *Shu-Osher-Darstellung* [75] eingeführt. Ist für $t = 0$ der Zustand \mathbf{q}^{init} vorgegeben, so wird die Diskretisierung $\mathbf{q}^{n+1} := \mathbf{q}^h(t = (n+1)\Delta t)$ berechnet durch:

$$\begin{aligned} \mathbf{q}^{(1)} &:= \mathbf{q}^n, \quad \mathbf{q}^0 := \mathbf{q}^{\text{init}}, \\ \mathbf{q}^{(r+1)} &:= \sum_{s=1}^r (\alpha_{rs} \mathbf{q}^{(s)} + \beta_{rs} \Delta t \mathcal{D}(\mathbf{q}^{(s)})), \quad \text{für } r = 1, \dots, m, \\ \mathbf{q}^{n+1} &:= \mathbf{q}^{(m+1)}. \end{aligned}$$

$$\begin{array}{ll}
m = 2 : & \alpha = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}, & \beta = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \\
m = 3 : & \alpha = \begin{pmatrix} 1 & 0 & 0 \\ \frac{3}{4} & \frac{1}{4} & 0 \\ \frac{1}{3} & 0 & \frac{2}{3} \end{pmatrix}, & \beta = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{2}{3} \end{pmatrix}, \\
m = 4 : & \alpha = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, & \beta = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{pmatrix}
\end{array}$$

Tabelle 3.1.: Koeffizienten der in DG-COSMO implementierten Runge-Kutta-Verfahren.

Die Wahl der Koeffizienten α_{rs} und β_{rs} bestimmt das RK-Verfahren, dabei muss $\alpha_{rs} \geq 0$ für alle r und s und $\sum_{s=1}^r \alpha_{rs} = 1$ für alle r gelten. In dieser Arbeit werden RK-Verfahren der Stufe 2 bis 4 verwendet, deren Konsistenzordnung ihrer Stufe entspricht. Die Koeffizienten sind in Tabelle 3.1 aufgeführt. Für den Fall $m = 2$ und $m = 3$ haben diese Verfahren die Eigenschaft der *Starken-Stabilitätsbewahrung* (engl. *Strong-Stability-Preserving* (SSP)) (vgl. [75]), d. h. sie vermeiden Über- oder Unterschwingen und damit Oszillationen. Der Fall $m = 4$ ist das *klassische Runge-Kutta-Verfahren*. In [45] werden RK-Verfahren mit der SSP-Eigenschaft für beliebige Ordnung beschrieben, jedoch müssen ab $m = 4$ negative Koeffizienten oder zusätzliche Stufen in Kauf genommen werden.

Ein Verfahren, das DG-Verfahren für die räumliche und RK-Verfahren für die zeitliche Diskretisierung verwendet, wird *RK-DG-Verfahren* genannt. Ist ein eindimensionales RK-DG-Verfahren der räumlichen Ordnung $\kappa + 1$ für den Fall $\kappa \leq 3$ gegeben mit einer Elementbreite Δx , so ist für eine stabile Integration mit dem Zeitschritt Δt folgende Bedingung, die oft *Courant-Friedrichs-Lewy-Bedingung* oder Kurz CFL-Bedingung genannt wird, notwendig

$$(\|\mathbf{u}\| + \lambda) \frac{\Delta t}{\Delta x} \leq \frac{1}{2\kappa + 1}. \quad (3.20)$$

Ein Beweis für $\kappa = 1$ und $\kappa = 2$ findet sich in [24] für den linearen und in [29] auch für den nicht-linearen Fall, wenn die Lösung glatt ist. Für $\kappa = 3$ ist der Beweis laut [29] ähnlich. Für $\kappa > 3$ ist kein Beweis bekannt. Aber es soll dennoch (3.20) auch für $\kappa > 3$ und auch für den mehrdimensionalen Fall unter Verwendung der Minimalbasis angenommen bzw. verwendet werden. Diese Annahme ist durch die erfolgreichen numerischen Simulationen im Nachhinein gerechtfertigt und wird auch in der Literatur so verwendet [29, 55, 71]. Für das volle Tensorprodukt als Basis muss die CFL-Zahl $(\|\mathbf{u}\| + \lambda) \frac{\Delta t}{\Delta x}$ etwas kleiner gewählt werden. Dies lässt sich damit begründen, dass im vollen Tensorprodukt auch Polynome enthalten sind, deren totaler Grad größer als κ ist. Experimentell ergibt sich, dass die CFL-Zahl beim vollen Tensorprodukt etwa bis zu 85% des Werts aus (3.20) annehmen kann. Um eine höhere Ordnung im Raum zu verwenden, sind kleinere CFL-Zahlen notwendig, wenn zeitlich nur ein RK-Verfahren dritter oder vierter Ordnung verwendet wird. Werte für $\kappa \leq 8$ können aus [27, Tabelle 2.2] entnommen werden.

Nun werden die physikalischen Parametrisierungen betrachtet. Der numerische Zeitschritt $\Delta \tau$ wird auf die physikalische Zeitskala Δt_p gesetzt, und die Parametrisierungen werden mit dem expliziten Euler-Verfahren integriert, d. h. hier fällt die numerische Diskretisierung mit der physikalischen Modellierung direkt zusammen.

3. Diskontinuierliche Galerkin-Verfahren

Dies führt bei der Operatoraufteilung dazu, dass die Werte aus dem Kessler-Schema und der Sättigungsadjustierung direkt verwendet werden können, d. h. es wird

$$\mathbf{q}^*(t = \tau^{n+1}) := \mathbf{q}(t = \tau^n) + \Delta\tau \mathcal{P}(\mathbf{q}(t = \tau^n)) = \begin{pmatrix} \rho' \\ \rho \mathbf{u} \\ \rho \tilde{\theta}_m - (\rho\theta)_0 \\ \rho \tilde{q}_v \\ \rho \tilde{q}_c \\ \rho \tilde{q}_r \end{pmatrix} (t = \tau^n)$$

gesetzt, mit $\tilde{\theta}_m$, \tilde{q}_v , \tilde{q}_c wie in (2.28) bis (2.30) und \tilde{q}_r wie in (2.34). Dies entspricht der Sichtweise 1 der Sättigungsadjustierung. Das Kessler-Schema und die Sättigungsadjustierung berechnen \mathbf{q}^* jedoch im physikalischen Raum, für die Integration der Dynamik wird die modale Darstellung (3.2) benötigt. Die adjustierten Variablen müssen daher in den diskretisierten Ansatz- und Testraum projiziert werden. Diese Projektion wird durch (3.26) in Abschnitt 3.5 beschrieben.

Bei der Prozessaufteilung werden die Tendenzen durch die Gleichungen (2.36) bestimmt:

$$\mathcal{P}(\mathbf{q}^*(t = \tau^{n+1})) := \rho \begin{pmatrix} 0 \\ \mathbf{0} \\ \frac{\tilde{\theta}_m - \theta_m}{\Delta\tau} \\ \frac{\tilde{q}_v - q_v}{\Delta\tau} \\ \frac{\tilde{q}_c - q_c}{\Delta\tau} \\ \frac{\tilde{q}_r - q_r}{\Delta\tau} \end{pmatrix} (t = \tau^n).$$

Dies entspricht der Sichtweise 2 der Sättigungsadjustierung. Hier ist eine Projektion in den diskretisierten Ansatz- und Testraum nicht notwendig, da mit den Quellterme direkt im physikalischen Raum weiter gerechnet werden kann.

Das hier vorgestellte DG-Verfahren ist nicht positiv definit, d. h. es können auch negative Werte für die Variablen q_v , q_c und q_r auftreten, obwohl es hierfür keine sinnvolle physikalische Interpretation gibt. Daher werden die negativen Werte für q_v , q_c und q_r im Kessler-Schema und der Sättigungsadjustierung auf Null gesetzt. Hierdurch wird künstlich Masse erzeugt, d. h. die Massenerhaltung wird für eine feuchte Atmosphäre mit dem aktuellen Verfahren verletzt. Hier kann eine Begrenzung des Flusses und eine positiv-definite Projektion in den diskretisierten Ansatz- und Testraum Abhilfe schaffen. Dies ist bisher aber nicht in DG-COSMO implementiert.

In COSMO wird die Sättigungsadjustierung und das Kessler-Schema durch Operatoraufteilung an die Dynamik gekoppelt. COSMO verwendet an dieser Stelle einen gemeinsamen Zeitschritt für die Dynamik und die Physik. Es werden also wie in Abschnitt 2.1.1 beschrieben zuerst das Kessler-Schema und dann die Sättigungsadjustierung nach jedem Dynamikzeitschritt aufgerufen.

3.4. Randbedingungen

Bei DG-Verfahren können Randbedingungen über zwei unterschiedliche Wege umgesetzt werden: 1. Die Randbedingungen werden in die Flussterme der Gleichung eingesetzt und der resultierende Fluss wird im DG-Verfahren auf den Rändern implementiert; 2. Das Gitter wird an jedem Rand um eine weitere Reihe von Elementen, *Geisterzellen* genannt, erweitert und die Werte für die prognostischen Variablen werden in den Geisterzellen so gesetzt,

dass die Flüsse über die Randkanten des Gebiets die Randbedingungen erfüllen. Im Finite-Differenzen-Kern von COSMO werden Geisterzellen für die Randbedingungen verwendet. Daher verwendet DG-COSMO ebenfalls Geisterzellen. Auch kann hierdurch die in COSMO vorhandene Parallelisierung des Gitters für den DG-Kern übernommen werden.

Im folgenden wird die Umsetzung von üblichen Randbedingungen beschrieben, die später auch in den Testfällen benötigt werden.

periodische Randbedingungen Sind periodische Randbedingungen in eine Richtung gefordert, und ist Ω_1 das erste und Ω_N das letzte Element, so werden die Geisterzellen Ω_0 und Ω_{N+1} vor bzw. hinter das erste bzw. letzte Element gestellt, die Näherung des Variablenvektors \mathbf{q}_0 und \mathbf{q}_{N+1} auf den Elementen Ω_0 und Ω_{N+1} werden wie folgt belegt:

$$\mathbf{q}_0 := \mathbf{q}_N, \quad \mathbf{q}_{N+1} := \mathbf{q}_1.$$

Schlupfrandbedingungen Die *Schlupfrandbedingungen* (englisch *free-slip*) entlang eines Randstücks $R \subseteq \partial\Omega$ fordern, dass die Geschwindigkeit normal zu R verschwindet und die Geschwindigkeit tangential zu R unverändert bleibt. Ist R (lokal) definiert durch

$$R = \{\mathbf{x} \mid r(\mathbf{x}) = 0\},$$

für eine Funktion r , so ist für einen Punkt \mathbf{x}_0 aus R der Normaleneinheitsvektor $\mathbf{n}(\mathbf{x}_0)$ nach [21, Tabelle 3.28] gegeben durch die beiden Bedingungen

$$\frac{n_x - x_0}{\frac{\partial r}{\partial x}|_{\mathbf{x}_0}} = \frac{n_y - y_0}{\frac{\partial r}{\partial y}|_{\mathbf{x}_0}} = \frac{n_z - z_0}{\frac{\partial r}{\partial z}|_{\mathbf{x}_0}}, \quad \|\mathbf{n}(\mathbf{x}_0)\| = 1,$$

wobei r so gewählt sein soll, dass \mathbf{n} in die Richtung des Inneren von Ω zeigt. Für die Schlupfrandbedingungen wird also gefordert

$$\mathbf{u}(\mathbf{x}_0) \cdot \mathbf{n}(\mathbf{x}_0) = 0. \quad (3.21)$$

Ist Ω_i das Randelement, zu dem \mathbf{x}_0 gehört, und sind \mathbf{t}_1 und \mathbf{t}_2 Einheitsvektoren, die die Tangentialebene in \mathbf{x}_0 an R aufspannen, so lässt sich $\mathbf{u}_i(\mathbf{x}_0)$ zerlegen in

$$\mathbf{u}_i(\mathbf{x}_0) = (\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{n}(\mathbf{x}_0)) \mathbf{n}(\mathbf{x}_0) + (\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{t}_1(\mathbf{x}_0)) \mathbf{t}_1(\mathbf{x}_0) + (\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{t}_2(\mathbf{x}_0)) \mathbf{t}_2(\mathbf{x}_0). \quad (3.22)$$

Ist Ω_g die an Ω_i anschließende Geisterzelle, so kann die Forderung (3.21) erfüllt werden, indem der Windvektor \mathbf{u}_g durch Spiegelung von \mathbf{u}_i an der Tangentialebene durch \mathbf{x}_0 definiert wird (vgl. [86]), es wird also

$$\mathbf{u}_g(\mathbf{x}_0) := -(\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{n}(\mathbf{x}_0)) \mathbf{n}(\mathbf{x}_0) + (\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{t}_1(\mathbf{x}_0)) \mathbf{t}_1(\mathbf{x}_0) + (\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{t}_2(\mathbf{x}_0)) \mathbf{t}_2(\mathbf{x}_0) \quad (3.23)$$

gesetzt. Der resultierende Normalwind $(\mathbf{u}_i(\mathbf{x}_0) + \mathbf{u}_g(\mathbf{x}_0)) \cdot \mathbf{n}(\mathbf{x}_0)$ verschwindet also auf dem Randpunkt \mathbf{x}_0 . Setzt man (3.22) in (3.23) ein, so erhält man

$$\mathbf{u}_g(\mathbf{x}_0) = \mathbf{u}_i(\mathbf{x}_0) - 2(\mathbf{u}_i(\mathbf{x}_0) \cdot \mathbf{n}(\mathbf{x}_0)) \mathbf{n}(\mathbf{x}_0).$$

Diese Form des Windvektors der Geisterzelle lässt sich einfacher implementieren und benötigt weniger Rechenaufwand. Die zusätzlichen Randbedingungen für den viskosen Fall stammen aus [85]. Hier wird anstelle der Haftbedingung $\mathbf{u}(\mathbf{x}_0) = \mathbf{0}$ ebenfalls eine Schlupfbedingung für die Ableitungen auf R gefordert:

$$\frac{\partial \mathbf{u}}{\partial \mathbf{n}} = 0 \quad \text{und} \quad \frac{\partial \theta}{\partial \mathbf{n}} = 0.$$

3. Diskontinuierliche Galerkin-Verfahren

Sie werden analog zur Bedingung $\mathbf{u} \cdot \mathbf{n} = 0$ umgesetzt. Die Werte der anderen prognostischen Variablen werden unverändert in die Geisterzellen übernommen. Da in den Testfällen in Kapitel 4 und 5 die Grenzschichten nicht studiert werden, wird der Einfachheit halber die Schlupfbedingung für die Ableitungen auch für andere Simulationen übernommen, wenn mit Viskosität gerechnet wird.

offene Randbedingungen Offene Randbedingungen werden wie in der Meteorologie üblich mit einer *Schwammschicht* (auch *Rayleigh-Dämpfung* genannt) modelliert, die den prognostischen Variablenvektor gegen einen vorgegebenen Zustand $\mathbf{q}_{\text{relax}}$ relaxiert

$$\mathbf{q}^* = (1 - \delta(\mathbf{x})) \mathbf{q} + \delta(\mathbf{x}) \mathbf{q}_{\text{relax}}. \quad (3.24)$$

Hierdurch sollen Reflexionen an den Rändern unterdrückt werden. Es wird angenommen, dass sich die Schwammschicht am oberen Modellrand befindet. Die *Blendfunktion* δ , d. h. die Funktion mit der die prognostischen Werte in die relaxierten Werte übergeführt werden, wird aus COSMO übernommen

$$\delta(z) = \frac{1 - \cos\left(\pi \frac{z - H_{rd}}{z_m}\right)}{2n_\delta \Delta t}, \quad (3.25)$$

dabei ist z_m die Dicke der Schwammschicht, H_{rd} die Höhe, bei der die Schwammschicht beginnt, und der Dämpfungsparameter n_δ ist die Anzahl der Zeitschritte über die sich die Relaxation erstreckt. In dieser Arbeit wird eine explizite Relaxation nach jedem Runge-Kutta-Zeitschritt ausgeführt. Damit kann die Relaxation unabhängig von dem RK-DG-Verfahren implementiert werden. Diese Relaxation wird auf alle Moden innerhalb der Schwammschicht angewandt. Für die Höhe z wird der Elementmittelpunkt eingesetzt. Der Dämpfungsparameter n_δ wird für die Testfälle experimentell ermittelt und so eingestellt, dass keine signifikanten Reflexionen auftreten.

3.5. Anfangsbedingungen

Die Anfangsbedingung $\mathbf{q}_{ijk}^{\text{init}}$ für ein Element Ω_{ijk} liegt für den physikalischen Raum vor. Sie muss daher zuerst in den diskretisierten Ansatz- und Testraum V_h abgebildet werden, damit sie für das modale DG-Verfahren verwendet werden kann. Hierfür soll der diskrete Wert $\mathbf{q}_{ijk}^{h,\text{init}}$ die Gleichung

$$\int_{\Omega_{ijk}} \mathbf{q}_{ijk}^{\text{init}} \phi \, d\Omega = \int_{\Omega_{ijk}} \mathbf{q}_{ijk}^{h,\text{init}} \phi \, d\Omega$$

für alle ϕ aus \mathcal{B} erfüllen. Nach Einsetzen von (3.13) und kurzer Rechnung folgt

$$\mathbf{q}_{ijk,l}^{h,\text{init}} = \int_E \mathbf{q}_{ijk}^{\text{init}} \phi_l \, dE. \quad (3.26)$$

Damit sind die Moden für die prognostischen Variablen zum Zeitpunkt $t = 0$ bestimmt.

Für idealisierte Testfälle wird meistens eine hydrostatisch balancierte und horizontal homogene Hintergrundatmosphäre vorausgesetzt. Die hydrostatische Balance soll auch für die diskretisierten Gleichungen gelten (bzw. der Fehler sollte möglichst klein sein). Für die trockenen Testfälle, die in dieser Arbeit behandelt werden, kann die hydrostatische Balance

analytisch gelöst werden, damit kann der Anfangszustand mit der Projektion (3.26) einfach bestimmt werden. Ist jedoch eine feuchte Atmosphäre über ein potentielles Temperaturprofil $\theta_{m,0}(z)$ und ein Profil für die relative Feuchte $f_0(z)$ gegeben, so muss ein Gleichungssystem mit einer nicht-linearen Differentialgleichung gelöst werden. Im Rest dieses Abschnitts wird beschrieben, wie für diesen feuchten Fall die numerische Näherung der hydrostatischen Balance an den Quadraturpunkten des DG-Verfahrens berechnet wird. Von den Feuchtegrößen muss nur der Wasserdampf berücksichtigt werden, da eine Kondensation mit $f_0(z) \leq 1$ ausgeschlossen ist. Aufgrund der horizontalen Homogenität lässt sich das Problem auf die z -Richtung reduzieren. Die Höhe des Simulationsgebiets wird mit H bezeichnet. Ausgangspunkt ist die Gleichung der hydrostatischen Balance und die Gasgleichung

$$\frac{\partial p}{\partial z}(z) = -\rho(z)g, \quad (3.27)$$

$$p(z) = \rho(z)R_m(z)T(z). \quad (3.28)$$

Es wird (3.28) in (3.27) eingesetzt, somit entsteht die gewöhnliche Differentialgleichung

$$\frac{\partial p}{\partial z} = -\frac{g}{R_m(z)T(z)}p(z). \quad (3.29)$$

Diese Gleichung kann durch eine Fixpunktiteration gelöst werden. Hierfür ist ein Temperaturprofil $T^{(0)}(z)$ für den Start der Iteration notwendig, dann wird folgender Algorithmus abgearbeitet, wobei die Funktionen an den Quadraturpunkten des DG-Gitters ausgewertet werden.

1. $n \leftarrow 0$, $q_v^{(0)}(z) = 0$, $R_m^{(0)}(z) = R_d$, $c_{pml}^{(0)}(z) = c_{pd}$
2. Integriere $p^{(n)}(z)$ aus der Gleichung (3.29) an den Quadraturpunkten der Gitterelemente von $z = 0$ bis $z = H$ mit $T^{(n)}(z)$ (Einzelheiten zu diesem Schritt folgen weiter unten).
3. $T^{(n)}(z) = \theta_{m,0}(z) \left(\frac{p^{(n)}(z)}{p_{00}} \right)^{\frac{R_m^{(n)}(z)}{c_{pml}^{(n)}(z)}}$
4. $q_v^{(n)}(z) = \frac{\rho_v^{(n)}(z)}{\rho^{(n)}(z)}$ mit $\rho^{(n)}(z) = \frac{p^{(n)}(z)}{R_m T^{(n)}(z)}$ und $\rho_v^{(n)}(z) = f_0(z) \frac{p_{\text{sat}}^{(n)}(T(z))}{R_v T^{(n)}(z)}$
5. $R_m^{(n)}(z) = R_d + (R_v - R_d)q_v^{(n)}(z)$, $c_{pml}^{(n)}(z) = c_{pd} + (c_{pv} - c_{pd})q_v^{(n)}(z)$
6. $n \leftarrow n + 1$
7. Wenn das Abbruchkriterium nicht erfüllt ist, wiederhole ab Schritt 2.

Das Abbruchkriterium kann eine feste Schranke für n oder für $|T^{(n)}(z) - T^{(n-1)}(z)|$ sein. Die Profile für das letzte berechnete n werden dann als Näherung für die hydrostatisch balancierte Atmosphäre benutzt. Diese Profile müssen noch mit (3.26) in den diskretisierten Ansatz- und Testraum projiziert werden, hierdurch entsteht ein Fehler in der Balancierung, dessen Größe mit steigender Verfahrensordnung abnimmt.

Die Berechnung von Schritt 2 des obigen Algorithmus wird wie folgt durchgeführt. Gleichung (3.29) wird durch den Galerkin-Ansatz integriert, d. h. Multiplikation mit einer Basisfunktion ϕ_l und Integration in z -Richtung über ein Element Ω_k . Anschließend wird partiell

3. Diskontinuierliche Galerkin-Verfahren

integriert und auf das Referenzintervall $[-1, 1]$ transformiert. Nach kurzer Rechnung erhält man folgende Gleichung

$$\left[p_k(\zeta) \phi_l(\zeta) \right]_{\zeta=-1}^{\zeta=1} - \int_{-1}^1 \left(\frac{\partial \phi_l}{\partial z}(\zeta) - \frac{g}{R_m T_k(\zeta)} \phi_l(\zeta) J_k \right) p_k(\zeta) d\zeta = 0,$$

wobei $J_k = \frac{\Delta z_k}{2}$. Nähert man das Integral durch eine LGL-Formel mit den n Quadraturpunkten ζ_1, \dots, ζ_n und den Gewichten w_1, \dots, w_n , so ist

$$p_{k,n} \phi_l(\zeta_n) - p_{k,1} \phi_l(\zeta_1) - \sum_{r=1}^n w_r \left(\frac{\partial \phi_l}{\partial z}(\zeta_r) - \frac{g}{R_m T_k(\zeta_r)} \phi_l(\zeta_r) J_k \right) p_{k,r} = 0, \quad (3.30)$$

mit $p_{k,r} = p_k(\zeta_r)$. Da hier ein eindimensionales Problem gelöst wird, gibt es $\kappa + 1$ Basisfunktionen für den Polynomgrad κ . Für alle $k = 1, \dots, k_{\text{end}}$ und alle $l = 1, \dots, \kappa + 1$ hat das System (3.30) also $(\kappa + 1) k_{\text{end}}$ Gleichungen mit $n k_{\text{end}}$ Unbekannten. Aus (3.16) folgt, dass nur eine Vorgabe je Element notwendig ist, um die $p_{k,1}, \dots, p_{k,n}$ zu bestimmen. Wird nun der Bodendruck $p_{k_{\text{end}},1}$ (für gewöhnlich mit $p_{k_{\text{end}},1} = p_{00}$) vorgegeben und wird $p_{k,1} = p_{k+1,n}$ für $k < k_{\text{end}}$ gefordert, so kann das Gleichungssystem (3.30) für alle $k = 1, \dots, k_{\text{end}}$ und alle $l = 1, \dots, \kappa + 1$ eindeutig gelöst werden. Der Algorithmus zur Bestimmung der numerisch hydrostatisch balancierten Atmosphäre in den Quadraturpunkten des DG-Gitters ist nun komplett beschrieben.

3.6. Einzelheiten zur Implementierung

In diesem Kapitel sind die Bilanzgleichungen einschließlich Rand- und Anfangsbedingungen diskretisiert. Bevor die numerischen Ergebnisse präsentiert werden, soll noch auf ein paar Einzelheiten zur Implementierung eingegangen werden. Eine besonders gut optimierte Implementierung ist kein Projektziel gewesen, da dieser Schritt erst dann zufriedenstellend bearbeitet werden kann, wenn klar ist, wie der zu implementierende Algorithmus im Einzelnen aussieht. Daher ist bei DG-COSMO noch Optimierungspotential vorhanden, sodass die in den folgenden Kapiteln ermittelten Laufzeiten noch verkürzt werden können.

Der neue dynamische Kern ist wie COSMO auch in Fortran 90/95 implementiert. Dabei basiert alles auf einer Eigenentwicklung. Auf bestehende Bibliotheken, die bei der Numerik unterstützen würden ist, wie in COSMO auch, verzichtet worden. Lediglich die Infrastruktur, die COSMO bereitstellt, wurde verwendet. Der Großteil davon wird für die Ein- und Ausgabe benötigt. Da bei Finite-Differenzenverfahren jeweils nur ein Freiheitsgrad je Element und prognostischer Variablen gespeichert wird, erschienen die Datenstrukturen der prognostischen Variablen in COSMO ungeeignet und es wurden neue Datenstrukturen für die DG-Variablen eingeführt, die je Element alle Koeffizienten der Ansatzpolynome speichern.

Zu Beginn dieser Arbeit war beim Deutschen Wetterdienst ein Großrechner des Typs NEC SX9 verfügbar. Dieser Rechner hat Vektorprozessoren, die dafür ausgelegt sind auf vielen Daten dieselben Rechnungen durchzuführen, wodurch eine gute *schwache Skalierbarkeit* erreicht wird. Hierfür ist es jedoch notwendig, dass die vom Prozessor vorgegebene Vektorlänge (bei der SX9 ist diese 256 doppelgenaue Fließkommazahlen) gut ausgenutzt wird. Kann dies nicht erreicht werden bzw. kann der Code nicht für die Vektoreinheit kompiliert werden, so ist die Laufzeit des Programms erheblich länger (bis zu Faktor 1000). Die meisten Rechnungen in dieser Arbeit wurden mit den Polynomgraden $\kappa = 2$ und $\kappa = 3$ durchgeführt, was selbst für den dreidimensionalen Fall mit der Minimalbasis nur 20 Freiheitsgrade je Element bzw. 25 Quadraturpunkte je Oberflächenintegral ergibt. Die benutzbare Vektorlänge ist

ohne das Zusammenfassen von ineinander geschachtelten Schleifen also erheblich kürzer als die maximale. Darüber hinaus ist die Speicherverwaltung der SX9 auch nicht dafür optimiert, häufig auf dieselben Daten zuzugreifen, was jedoch ein wesentlicher Grundgedanke der DG-Verfahren ist. Die Vektorarchitektur ist daher für DG-Verfahren nicht gut geeignet. Auf Großrechnern, die mit rechenstarken Skalareinheiten und großem Cache ausgestattet sind, können die DG-Verfahren gut implementiert werden, und hier kommen ihre Vorteile auch zum Tragen. Der Großrechner beim EZMW basiert auf Skalarprozessoren. Als abzusehen war, dass der Ende 2013 neu beschaffte Großrechner⁵ des DWD auch Skalarprozessoren verwenden wird, wurde DG-COSMO für diese Architektur umgeschrieben und auf den beiden Skalarprozessor-Großrechnern verwendet.

Für die Parallelisierung eines Programms über mehrere Knoten hinweg müssen über das Netzwerk Daten ausgetauscht werden. Als Standardbibliothek hierfür hat sich die *Message Passing Interface* (MPI) etabliert. Die durch COSMO bereitgestellte Infrastruktur verwendet MPI und ist für dreidimensionale Datenfelder ausgelegt. Das Berechnungsgebiet kann hierdurch in der Horizontalen auf mehrere Prozesse aufgeteilt werden, so dass parallel auf den Daten gerechnet werden kann. In einer ersten Version von DG-COSMO wurde diese Infrastruktur benutzt, indem der *Halo* des Variablenvektors für jede Basisfunktion und jede prognostische Variable zwischen den Prozessen übertragen wurde. Um eine bessere Skalierbarkeit zu erreichen, wurde die MPI-Kommunikation auf die Datenstruktur des DG-Kerns erweitert. Jetzt werden die benötigten Teile des fünfdimensionalen Feldes⁶ für den prognostischen Variablenvektor in einer MPI-Kommunikation zwischen den Prozessen übertragen. Nach dieser Erweiterung konnte für den Baldauf/Brdar 2013-Testfall (siehe Abschnitt 4.1) eine Beschleunigung von etwa 21 % (bei 4 Prozessen auf einer Intel i7-Workstation) erreicht werden. Im Gegensatz zu COSMO ist DG-COSMO zusätzlich mithilfe von OpenMP parallelisiert. Durch OpenMP kann ein Prozess mehrere CPUs verwenden, welche sich die Berechnungen teilen und über gemeinsamen genutzten Speicher des Prozesses miteinander kommunizieren. DG-COSMOs Berechnungsgebiet wird horizontal mit MPI in einzelne Teilstücke zerlegt, wie auch bei COSMO. Zusätzlich wird DG-COSMOs Berechnungsgebiet vertikal mithilfe von OpenMP aufgeteilt. Diese hybride MPI/OpenMP-Parallelisierung ermöglicht das Gebiet feiner zu zerlegen, wodurch die vorhandenen CPUs besser genutzt werden können.

Eine weitere durchgeführte Optimierung, die den Code nochmal um etwa 15 % bis 20 % beschleunigt, betrifft die Ausnutzung des Caches. Hierfür wurde die Schleife, die über die Quadraturpunkte läuft, die für die Berechnung der Integrale in Gleichung (3.15) benötigt werden, in Teilstücke unterteilt, die so klein sind, dass jedes Teilstück in den Cache passt. Auf jedem dieser Teilstücke werden dann alle Berechnungen für die Volumen- und Oberflächenintegrale durchgeführt, dadurch kann möglichst lange auf den Daten im Cache gerechnet werden.

Da DG-COSMO das gleiche Berechnungsgitter wie COSMO verwendet, d. h. die gleiche Abmessung und die gleiche Anzahl an Elementen je Richtung, und auch die gleichen Ausgaberroutinen wie COSMO verwendet, kann über diese nur ein Punkt je Element ausgegeben werden. Hierfür wird der Elementmittelpunkt gewählt. Hierdurch wird die Auflösung von DG-COSMO und damit die durch das Verfahren ermöglichte Genauigkeit stark reduziert. Dies wird deutlich in den Testfällen 4.1 und 5.1. Um alle berechnete Information der DG-Lösung für die Ausgabe nutzen zu können, wurde nachträglich eine Ausgaberroutine geschrieben, die die COSMO-Ausgaberroutine umgeht. Diese verwendet ein Ausgabegitter das vom Berechnungsgitter entkoppelt ist und so mehrere Ausgabepunkte je Berechnungselement haben kann. Hierfür stehen zwei Möglichkeiten zur Verfügung.

⁵Der Großrechner ist eine Cray XC30. Jeder Knoten hat 2 Prozessoren des Typs Intel Xeon E5-2670v2 mit je 10 CPUs.

⁶eine Dimension für den Freiheitsgrad in einem Element, eine für die prognostische Variable und drei Dimensionen für die räumlichen Richtungen

3. Diskontinuierliche Galerkin-Verfahren

1. Es werden für jedes Element in jeder Richtung n äquidistante Punkte ausgegeben, hierbei liegen die äußeren Punkte einen halben Abstand von der Elementgrenze entfernt. Die Anzahl n lässt sich konfigurieren.
2. Es werden $i_a \times j_a \times k_a$ äquidistante Punkte für das gesamte Berechnungsgitter ausgegeben. Für einen Ausgabepunkt \mathbf{x} wird ein Gitterelement Ω_{ijk} gesucht, in dem sich \mathbf{x} befindet. Die Moden des ersten gefundenen Elements werden dann zur Bestimmung des Ausgabewertes verwendet. Auf eine eventuelle Randlage auf dem Element eines Ausgabepunktes wird keine Rücksicht genommen. Die Werte i_a, j_a, k_a lassen sich frei konfigurieren.

Eine Alternative wäre anstelle eines feineren Ausgabegitters die Polynomkoeffizienten direkt auszugeben. In diesem Fall müssten die Polynomwerkzeuge die verwendete Basis kennen oder es müsste ein zusätzliches Werkzeug entwickelt werden, das die Polynome auf einem Ausgabegitter auswertet. Dieser Weg würde die Anwendung von DG-COSMO verkomplizieren, daher wurde er nicht eingeschlagen.

4. Numerische Ergebnisse für die trockene Atmosphäre

In diesem Kapitel wird der dynamische Kern von DG-COSMO anhand von verschiedenen idealisierten trockenen Testfällen validiert. Ein Teil dieser Ergebnisse, sind in die eingereichte Veröffentlichung [74] eingeflossen.

Die zweidimensionalen Testfälle bezeichnen, wie in der Meteorologie üblich, die horizontale Variable mit x und die vertikale Variable mit z . Die Finite-Differenzenverfahren haben je Gitterpunkt nur einen Freiheitsgrad je prognostischer Variablen, die DG-Verfahren haben je Gitterpunkt d Freiheitsgrade (vgl. Gleichungen (3.10) und (3.11)) je prognostischer Variablen. Um die Verfahren miteinander vergleichen zu können wird die *formale Auflösung* eingeführt. Die formale Auflösung Δx_{for} eines FD-Verfahrens ist gleich der Gittermaschenweite Δx , die formale Auflösung eines DG-Verfahrens entlang der x -Richtung ist

$$\Delta x_{\text{for}} := \frac{\Delta x}{\sqrt[d]{d}}.$$

Für die anderen Richtungen wird die formale Auflösung analog definiert. Die formale Auflösung eines DG-Verfahrens entspricht also der Auflösung eines FD-Verfahrens, das genauso viele Freiheitsgrade wie das DG-Verfahren hat, und diese Freiheitsgrade je Variable äquidistant über die Elemente des DG-Gitters verteilt. Die Verfahren können anstelle der formalen Auflösung also auch über die Anzahl der Freiheitsgrade verglichen werden. Hier soll jedoch explizit darauf hingewiesen werden, dass im Allgemeinen weder die formale Auflösung noch die Anzahl der Freiheitsgrade für den FD- und den DG-Löser exakt gleich gewählt werden kann, da die Freiheitsgrade nur ganzzahlig je Richtung definiert werden können. Daher gibt es in diesem und im nächsten Kapitel bei den Vergleichen kleine Unterschiede dieser beiden Größen, auch wenn der Vergleich mit „gleicher Anzahl an Freiheitsgraden“ oder bei „gleicher formaler Auflösung“ durchgeführt wird. Dies gilt auch für verschiedene Ordnungen des DG-Lösers. Mit der obigen Definition der formalen Auflösung deutet sich schon an, dass der Auflösungs begriff nicht unbedingt an ein Gitter gebunden werden muss. Auch ist er verschieden von der Anzahl der Freiheitsgrade. Denn über unterschiedliche Ordnungen können bei gleicher Anzahl an Freiheitsgraden unterschiedlich feine Strukturen aufgelöst werden (vgl. Testfall in Abschnitt 5.1 und die Abbildung 5.2). Es ist also erstrebenswert, einen Auflösungs begriff zu definieren, der unabhängig von den Parametern des Verfahrens bestimmt werden kann. Die spektrale Auflösung¹, wie sie etwa in [77] verwendet wird, wäre wohl ein geeigneter Kandidat. Diese Untersuchung geht aber über diese Arbeit hinaus, daher begnügt sich diese Arbeit mit obiger Definition der formalen Auflösung.

Aus vorherigem Abschnitt folgt auch, dass die Genauigkeit des DG-Verfahrens nicht hinreichend ausgeschöpft wird, wenn nur ein Punkt je Gitterelement ausgegeben wird. Dieser Unterschied zwischen der Ausgabe auf der Gittermaschenweite und der formalen Auflösung wird im Abschnitt 4.1.3 explizit behandelt. Um alle Information der Lösung des DG-Verfahrens zu nutzen, ist es also notwendig das Ausgabegitter und das Berechnungsgitter zu entkoppeln.

¹Betrachtet man für eine mesoskalige Simulation das Spektrum der kinetischen Energie, so folgt sie der beobachtbaren Funktion $k^{-\frac{5}{3}}$, wobei k die Wellenzahl ist, bis zur Dissipationsgrenze des Modells. Die kleinste Wellenzahl, die noch mit vorheriger Funktion übereinstimmt wird dann als spektrale Auflösung definiert.

4. Numerische Ergebnisse für die trockene Atmosphäre

Um zwei Lösungen ψ_a und ψ_b für eine Variable ψ miteinander vergleichen zu können, wird die \mathcal{L}_h -Norm der Differenz, definiert durch:

$$\mathcal{L}_h(\psi_a, \psi_b) := \sqrt[3]{\frac{1}{\mu(\Omega)} \int_{\Omega} \|\psi_a(\mathbf{x}) - \psi_b(\mathbf{x})\|^h d\Omega}, \quad (4.1)$$

verwendet, wobei $\mu(\Omega)$ das Volumen von Ω ist. Wenn eine (linearisierte) Referenzlösung ψ_0 vorhanden ist, so wird für eine numerische Lösung ψ durch $\mathcal{L}_h(\psi, \psi_0)$ der \mathcal{L}_h -Fehler von ψ bestimmt. Ist keine Referenzlösung vorhanden, so können mit der \mathcal{L}_h -Norm zwei Lösungen auf Ähnlichkeit miteinander verglichen werden, d. h. es wird der Unterschied zwischen diesen Lösungen gemessen, ohne einer der beiden Lösungen eine größere Bedeutung zu geben als der anderen.

Die \mathcal{L}_h -Norm wird in der Nachbearbeitung der Ausgabedaten bestimmt. Sind die Elemente des Ausgabegitters durch $\Omega_1, \dots, \Omega_N$ aufgezählt und deren Elementmittelpunkte durch $\mathbf{x}_1, \dots, \mathbf{x}_N$, so wird die Diskretisierung L_h der \mathcal{L}_h -Norm berechnet durch:

$$L_h(\psi_a, \psi_b) := \sqrt[3]{\frac{1}{\mu(\Omega)} \sum_{i=1}^N \|\psi_a(\mathbf{x}_i) - \psi_b(\mathbf{x}_i)\|^h \mu(\Omega_i)}. \quad (4.2)$$

Ist keine Referenzlösung vorhanden wird die DG-COSMO-Lösung mit der COSMO-Lösung verglichen oder mit Ergebnissen aus der Literatur.

Die *Laufzeit* bezeichnet die in der Realität verstrichene Zeit, die benötigt wird, um den Testfall zu simulieren. Die Zeitspanne die zwischen Modellstartzeitpunkt und Modellendzeitpunkt vergeht, wird in dieser Arbeit als *Simulationszeit* bezeichnet. Wenn nichts anderes erwähnt wird, so wird als Referenzatmosphäre in DG-COSMO die Hintergrundatmosphäre des Testfalls gewählt, die Minimalbasis \mathcal{B}_{mb} verwendet und ohne die Viskositätsterme in den Gleichungen (2.11) bis (2.14) gerechnet. Es wird das RK-Verfahren gewählt, dass die Ordnung $\min(\kappa + 1, 4)$ hat, also die gleiche Ordnung wie das DG-Verfahren, aber höchstens vierte Ordnung. Die angegebenen Laufzeiten sind das arithmetische Mittel der Laufzeiten dreier Wiederholungen der betroffenen Simulation. Dabei werden die Laufzeit für die Initialisierungsphase und die Ausgabe nicht berücksichtigt. Mit Freiheitsgrad ist im folgenden immer Freiheitsgrad je prognostischer Variable gemeint. Der Referenzbodendruck ist:

$$p_{00} = 1000 \text{ hPa.}$$

In COSMO wird immer die durch (2.40) festgelegt COSMO-Referenzatmosphäre gewählt.

4.1. Schwerewellentestfall von Baldauf/Brdar 2013

Ein Standardtestfall, mit dem die Korrektheit eines Dynamikkerns überprüft werden kann, ist der Trägheitsschwerewellentestfall von Skamarock und Klemp [79]. Hier wird eine Wellenausbreitung in einem zweidimensionalen Kanal simuliert und mit einer Boussinesq-Approximation verglichen. Die voll-kompressiblen Euler-Gleichungen konvergieren jedoch nicht gegen diese Boussinesq-Lösung. Daher hat Michael Baldauf diesen Testfall derart verändert, dass für die linearisierten Euler-Gleichungen eine analytische Lösung² berechnet werden kann [9]. Durch diese Änderung ist der Testfall geeignet, um einen Gleichungslöser auf Konvergenz zu prüfen. In diesem Abschnitt wird DG-COSMO anhand dieses Testfalls validiert.

²Eine Referenzimplementierung dieser Lösung wurde mir von Michael Baldauf zur Verfügung gestellt.

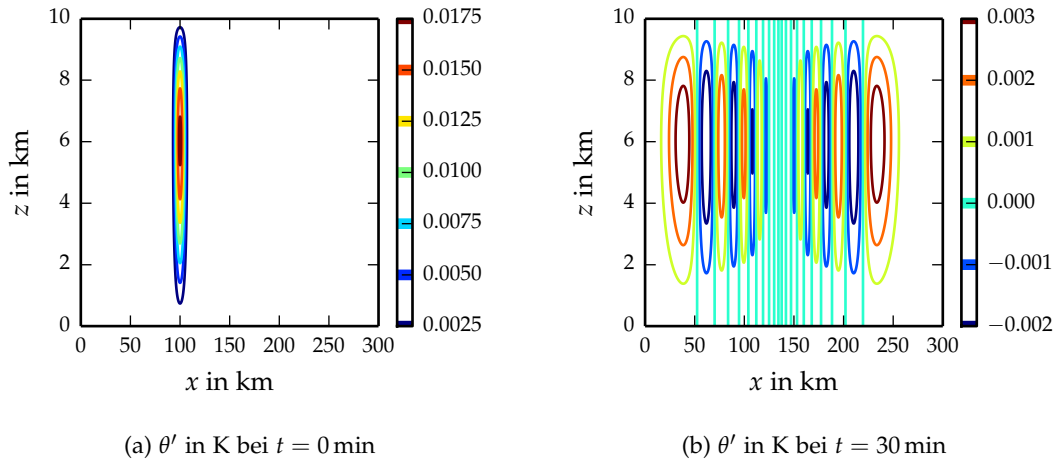


Abbildung 4.1.: Konturlinien von θ' des Testfalls Baldauf/Brdar 2013 [9]. Abbildung (a) zeigt den Anfangszustand, Abbildung (b) den Endzustand nach 30 min. Die Farbskala der Konturlinien wurde für beide Bilder ausgeschöpft.

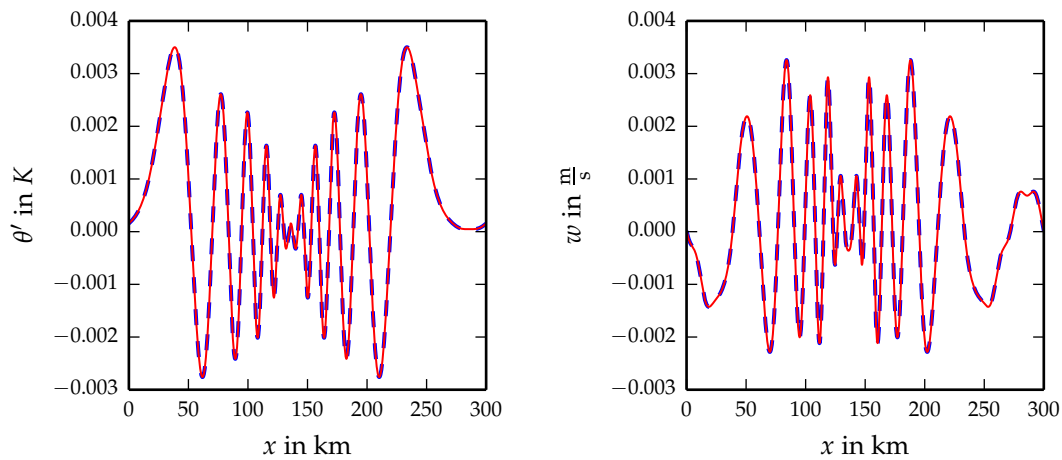


Abbildung 4.2.: Vergleich von θ' und w der analytischen linearisierten Lösung (blau gestrichelt) mit der DG-Lösung (rot durchgezogen) bei $z = 5$ km nach $t = 30$ min. DG-COSMO wurde für $\kappa = 3$ mit der Gittermaschenweite $\Delta x = 500$ m und $\Delta z = 250$ m gerechnet.

4. Numerische Ergebnisse für die trockene Atmosphäre

Der Kanal hat die Länge L und die Höhe H . Es wird eine isotherme, hydrostatisch balancierte Hintergrundatmosphäre mit der Temperatur T_{00} vorgegeben. Für den oberen und unteren Rand werden Schlupfbedingungen verwendet. Die horizontalen Randbedingungen sind periodisch. Die Temperatur, das Druck- und Dichteprofil der Hintergrundatmosphäre sind

$$T_0(z) = T_{00}, \quad p_0(z) = p_{00}e^{-\delta z}, \quad \rho_0(z) = \rho_{00}e^{-\delta z}, \quad p_{00} = \frac{p_{00}}{T_{00}R_d}, \quad \delta = \frac{g}{R_d T_{00}}.$$

Als Anfangszustand ist eine Temperaturstörung der Amplitude ΔT mit dem Zentrum an der Stelle $(x_c, \frac{H}{2})$ definiert. Darüber hinaus wird eine rein horizontale Strömung \mathbf{u} festgelegt. Die Profile für die Temperaturstörung sind

$$\begin{aligned} T'(\mathbf{x}) &= e^{\frac{1}{2}\delta z} T_b(\mathbf{x}), & \text{mit } T_b(\mathbf{x}) &= \Delta T e^{-\frac{(x-x_c)^2}{d^2}} \sin\left(\pi \frac{z}{H}\right), \\ \rho'(\mathbf{x}) &= e^{-\frac{1}{2}\delta z} \rho_b(\mathbf{x}), & \text{mit } \rho_b(\mathbf{x}) &= -\rho_{00} \frac{T_b(\mathbf{x})}{T_{00}}. \end{aligned}$$

Die obigen Testfallparameter werden belegt mit

$$\begin{aligned} L &= 300 \text{ km}, & x_c &= 100 \text{ km}, & T_{00} &= 250 \text{ K}, \\ H &= 10 \text{ km}, & d &= 5 \text{ km}, & \mathbf{u} &= \begin{pmatrix} 20 \frac{\text{m}}{\text{s}} \\ 0 \frac{\text{m}}{\text{s}} \end{pmatrix}. \end{aligned}$$

Diese Störung der balancierten Anfangsbedingung führt durch die Auftriebskraft zu einer Oszillation um die balancierte Ruhelage, wodurch Schwerewellen entstehen die vom Horizontalwind räumlich ausgebreitet werden. Wenn ΔT sehr klein gewählt wird, etwa $\Delta T = 0,01 \text{ K}$, ist die Strömung nahezu linear. In diesem Fall kann die linearisierte analytische Lösung als Referenzlösung verwendet werden. Der Anfangszustand und die Lösung nach $t_{\text{end}} = 30 \text{ min}$ ist für θ' in Abbildung 4.1 dargestellt. Abbildung 4.2 zeigt den Vergleich zwischen numerischer und linearisierter analytischer Lösung für den Horizontalschnitt in 5 km Höhe nach 30 min für das DG-Verfahren vierter Ordnung für θ' und w . Man sieht eine sehr gute Übereinstimmung zwischen der DG-Lösung und der linearisierten analytischen Lösung. Eine Phasenverschiebung lässt sich nicht feststellen, die Amplitude ist bei der gewählten Auflösung und Ordnung praktisch identisch.

4.1.1. Konvergenzanalyse

In [9] haben Baldauf und Brdar COSMO und das in der Einleitung erwähnte DG-Verfahren von Brdar, das auf der Softwarebibliothek DUNE aufbaut, miteinander auf die Konvergenzeigenschaften verglichen. Der DUNE-Löser verwendet für diesen Testfall ein DG-Verfahren zweiter Ordnung mit der vollen Tensorproduktbasis und ein SSP-RK-Verfahren ebenfalls von zweiter Ordnung. Hier soll die Konvergenz von DG-COSMO validiert werden. Zum einen kann dadurch beurteilt werden, ob das Verfahren auch tatsächlich das richtige Problem löst. Zum anderen können durch eine Konvergenzanalyse viele, auch kleinere, Implementierungsfehler gefunden werden. Alle Simulationen für diesen Testfall verwenden ein horizontal und vertikal äquidistantes Gitter. In Tabelle 4.1 ist für $\kappa = 1$ der Fehler zum Zeitpunkt t_{end} für θ' und w für verschiedene Gittermaschenweiten dargestellt. Der Fehler wurde an 6000×400 Punkten bestimmt, die je Richtung äquidistant über das Berechnungsgebiet verteilt sind. Dies ist das gleiche Vorgehen wie für den DUNE-Löser in [9]. Man sieht, dass die *experimentelle Konvergenzordnung (EOC)* über das zu erwartende Optimum ansteigt, sowohl für θ' als auch

Gitter (Δx)	4000 m	2000 m	1000 m	500 m	250 m
Freiheitsgrade	1 125	4 500	18 000	72 000	288 000
L_2 -Fehler	9,78e-4	5,25e-4	1,38e-4	2,08e-5	2,83e-6
EOC	-	0,90	1,93	2,72	2,88
L_∞ -Fehler	3,30e-3	1,85e-3	5,74e-4	9,00e-5	1,27e-5
EOC	-	0,83	1,69	2,67	2,82

(a) θ'

Gitter (Δx)	4000 m	2000 m	1000 m	500 m	250 m
Freiheitsgrade	1 125	4 500	18 000	72 000	288 000
L_2 -Fehler	9,52e-4	5,84e-4	1,67e-4	2,59e-5	3,61e-6
EOC	-	0,70	1,81	2,69	2,84
L_∞ -Fehler	3,21e-3	2,36e-3	8,31e-4	1,36e-4	1,98e-5
EOC	-	0,44	1,50	2,61	2,78

(b) w

Tabelle 4.1.: Fehler und EOC des Baldauf/Brdar 2013-Testfall [9] mit $\kappa = 1$ nach 30 min für verschiedene Gittermaschenweiten. Dabei ist die linearisierte Lösung die Referenzlösung und $\Delta z = \frac{1}{2}\Delta x$.

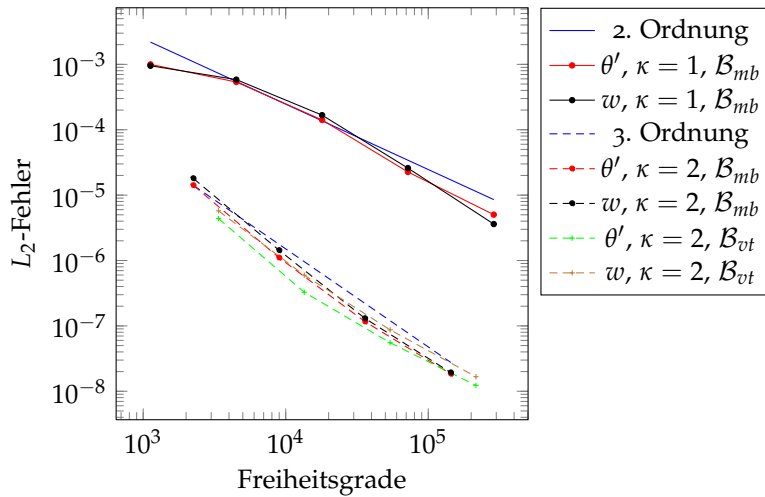


Abbildung 4.3.: Konvergenzanalyse von DG-COSMO. Für $\kappa = 1$ wurde $\Delta T = 0,01$ K verwendet und $\Delta T = 0,001$ K für $\kappa = 2$. Für $\kappa = 2$ ist der Fehler sowohl für \mathcal{B}_{mb} als auch für \mathcal{B}_{vt} eingezeichnet.

4. Numerische Ergebnisse für die trockene Atmosphäre

für w . Abbildung 4.3 visualisiert die experimentelle Konvergenzordnung und die theoretische erwartete Ordnung. Neben $\kappa = 1$ ist auch $\kappa = 2$ für die Minimalbasis und das volle Tensorprodukt eingezeichnet. Für $\kappa = 2$ wurde als Amplitude $\Delta T = 0,001$ K verwendet, da mit $\Delta T = 0,01$ K die EOC bei etwa $\Delta x = 500$ m stark einbricht. Hier ist die Annahme, dass die linearisierte Lösung als Referenzlösung verwendet werden kann, wohl nicht mehr gültig. Aus Tabelle 4.1 und Abbildung 4.3 lässt sich schließen, dass DG-COSMO für idealisierte Fälle die formale Konvergenzordnung erreicht. Der Finite-Differenzen-Kern von COSMO, der formal von zweiter Ordnung ist, hat für diesen Testfall eine Konvergenzordnung von etwa eins (vgl. [9]).

4.1.2. Erhaltungseigenschaften

DG-Verfahren sind lokal erhaltend, da zwischen den Elementen nur über die Flüsse die prognostischen Variablen geändert werden. In diesem Abschnitt soll geprüft werden, wie gut das DG-Verfahren die prognostischen Größen global erhält. Aufgrund der periodischen Randbedingungen geht keine der prognostischen Größen, insbesondere Impuls oder potentielle Temperatur verloren, daher eignet sich die Schwerewellenausbreitung gut für diese Überprüfung. Hierfür wird sie über 40 h auf einem Gitter mit 300×20 Elementen für $\kappa = 3$ mit dem Zeitschritt $\Delta t = 0,15$ s gerechnet. Die totale Masse lässt sich in die beiden Summanden $\int_{\Omega} \rho_0 d\Omega$ und $\rho'_{\Sigma} := \int_{\Omega} \rho' d\Omega$ aufteilen. Der Anteil des Referenzzustand ist per Konstruktion konstant. Die Abweichung von $\rho'_{\Sigma}(t = 0)$ zum Zeitpunkt t sei $\Delta\rho'_{\Sigma}$ und der Mittelwert dieser Abweichung sei

$$\overline{\Delta\rho'_{\Sigma}} := \frac{1}{40\text{h}} \int_{0\text{h}}^{40\text{h}} \Delta\rho'_{\Sigma}(t) dt.$$

Für die prognostischen Variablen ρu , ρw und $(\rho\theta)'$ sei $\Delta\rho u_{\Sigma}$, $\Delta\rho w_{\Sigma}$, $\Delta(\rho\theta)'_{\Sigma}$, sowie $\overline{\Delta\rho u_{\Sigma}}$, $\overline{\Delta\rho w_{\Sigma}}$ und $\overline{\Delta(\rho\theta)'_{\Sigma}}$ analog definiert. Zusätzlich soll der Betrag der gesamten Impulsdichte $\rho\mathbf{u}$ betrachtet werden:

$$\|\rho\mathbf{u}\|_{\Sigma} := \int_{\Omega} \|\rho\mathbf{u}\| d\Omega.$$

Und $\Delta\|\rho\mathbf{u}\|_{\Sigma}$ sei die Abweichung zum Zeitpunkt t vom Anfangszustand $\|\rho\mathbf{u}\|_{\Sigma}(t = 0)$.

In Tabelle 4.2 sind die maximalen Abweichungen vom Startzeitpunkt und die Mittelwerte über die 40 h dieser Abweichungen aufgelistet. Tabelle 4.2 zeigt zusätzlich das Intervall, in dem diese Abweichungen nach einer Einschwingzeit um den Mittelwert pendeln. Diese

ψ	$\max \ \Delta\psi\ $	$\overline{\Delta\psi}$	$(\Delta\psi - \overline{\Delta\psi})(t > 0,1\text{h})$
ρ'_{Σ}	1,53e-14	1,32e-14	[-3,79e-15; 2,14e-15]
$(\rho\theta)'_{\Sigma}$	6,76e-12	-5,97e-12	[-7,93e-13; 6,82e-13]
ρu_{Σ}	1,60e-10	4,71e-11	[-6,17e-11; 7,66e-11]
ρw_{Σ}	2,21e-1	-3,30e-3	[-2,17e-1 ; 2,17e-1]
$\ \rho\mathbf{u}\ _{\Sigma}$	9,01e-5	4,89e-5	[-2,84e-5 ; 3,18e-5]

Tabelle 4.2.: globale Erhaltung der prognostischen Variablen.

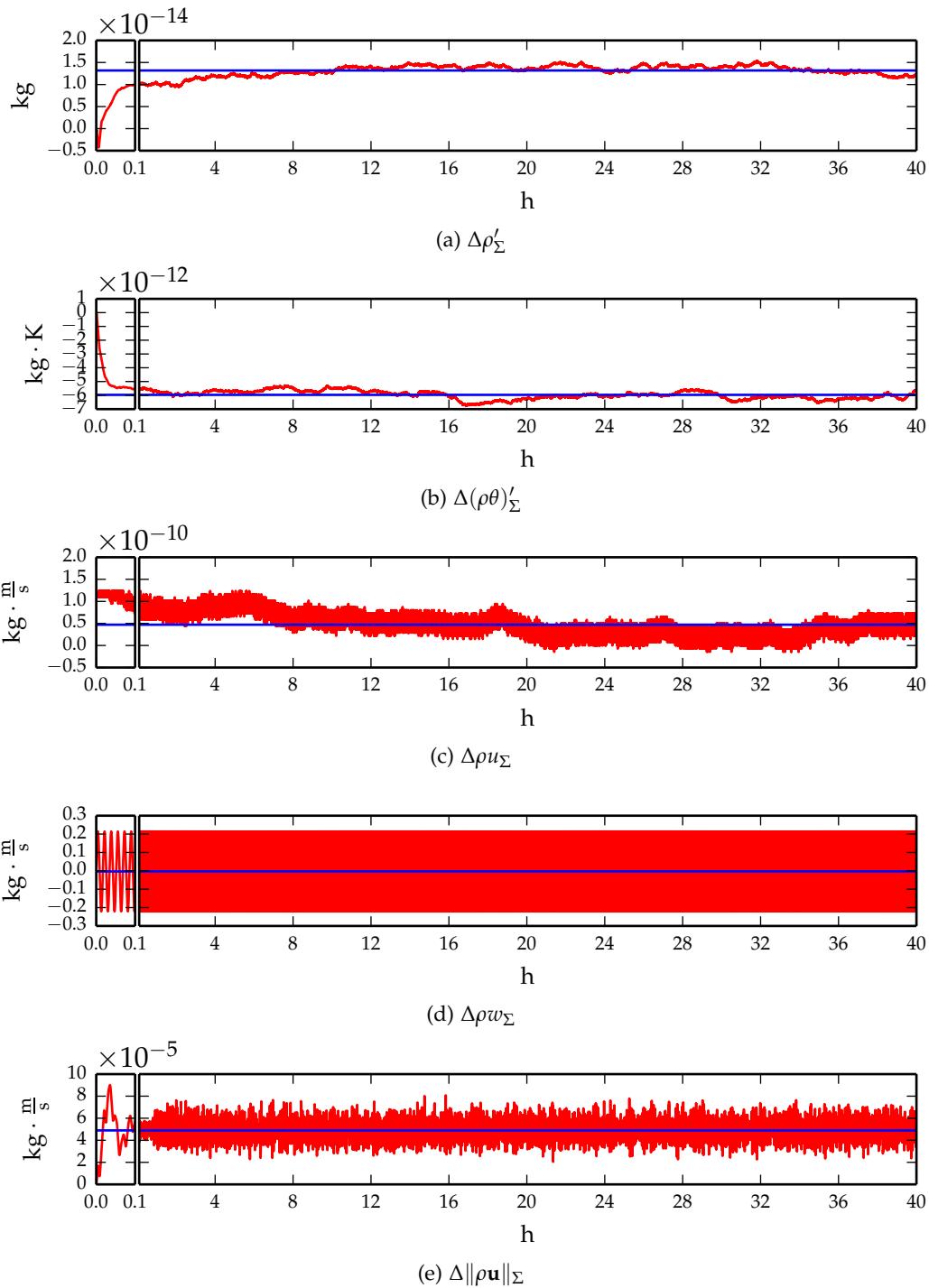


Abbildung 4.4.: Die Abbildungen zeigen die Änderung (a) der Gesamtmasse, (b) des Produktes aus Gesamtmasse und potentieller Temperatur, (c) des Horizontal-, (d) des Vertikalimpulses und (e) des Absolutbetrags des Gesamtimpulses. Die Zeitreihe der Abweichung ist rot und der zeitliche Mittelwert über die 40 h dieser Abweichungen blau. Die ersten 0,1 h sind aufgeweitet dargestellt, um die Einschwingphase erkennen zu können.

Einschwingzeit beträgt etwa 0,1 h. Für ρ'_Σ ist das Intervall $[-3,88e-15; 2,05e-15]$, das ist praktisch Maschinengenauigkeit. Die Abweichung der Werte für $(\rho\theta)'_\Sigma$ sind in der Größenordnung 10^{-13} . Für den Horizontalimpuls ist die Größenordnung 10^{-11} und es treten deutlich stärkere Fluktuationen auf, als für die Masse oder das Produkt aus Masse und potentieller Temperatur. Der Vertikalimpuls wird nicht direkt numerisch erhalten. Dies wird vermutlich durch den Quellterm, dessen Diskretisierung nicht mit dem Flussterm balanciert ist und durch die Randbedingungen verursacht. Die Abweichung pendelt periodisch im Intervall $[-2,17e-1; 2,17e-1]$. Diese Abweichung ist 5 Zehnerpotenzen kleiner als der Gesamtimpuls, es ist also immer noch ein kleiner Wert. Für den Gesamtimpuls bewegt sich die Abweichung vom Mittelwert auf dem Niveau von 10^{-5} , dies ist ein akzeptabler Wert. In Abbildung 4.4 ist die Zeitreihe von ρ'_Σ , ρu_Σ , ρw_Σ , $(\rho\theta)'_\Sigma$ und $\|\rho\mathbf{u}\|_\Sigma$ dargestellt. Eine Tendenz zur Zunahme oder Abnahme einer dieser Größen ist nicht erkennbar. Aufgrund der erfolgreichen Konvergenzanalyse und der Erhaltungseigenschaften kann davon ausgegangen werden, dass das DG-Verfahren korrekt implementiert ist.

4.1.3. Vergleich des vollen Tensorprodukts mit der Minimalbasis

Es bietet sich bei modalen DG-Verfahren sowohl die Verwendung der Minimalbasis als auch des vollen Tensorprodukts an. Welcher der beiden Basen der Vorzug gegeben werden soll, ist von vornherein nicht klar. Auf dem gleichem Berechnungsgitter kann die Minimalbasis mit einem größeren Zeitschritt verwendet werden, das volle Tensorprodukt bietet hingegen eine höhere formale Auflösung. Neben einer unterschiedlichen Genauigkeit sind also auch Unterschiede in der Laufzeit zu erwarten. In diesem Abschnitt werden anhand der Schwerewellenausbreitung im Kanal die beiden Basen für zwei Fälle miteinander verglichen.

Im ersten Testfall wird die gleiche formale Auflösung für beide Basen bei verschiedenen Ordnungen gewählt, d. h. sie haben auch die gleiche Anzahl an Freiheitsgraden. Dabei ist die Anzahl der Freiheitsgrade so hoch gewählt, dass die formale Auflösung der verschiedenen Simulationen sich um weniger als 6 % unterscheiden. Tabelle 4.3 zeigt für beide Basen die L_2 -Fehler, die L_∞ -Fehler und die Laufzeiten für $\kappa = 1, \dots, 6$, sowie die relativen Unterschiede zueinander. Für die Bestimmung der L_H -Fehler sind die Lösungen jeweils auf einem Ausgabegitter mit 6000×400 Elementen berechnet, d. h. der Fehler wird bei allen Lösungen an denselben Stellen bestimmt. Die Lösungen der Minimalbasis haben bei allen Ordnungen den kleineren Fehler als die Lösungen des vollen Tensorprodukts. Mit steigender Ordnung wird der Unterschied zwischen den beiden Basen größer. Auch ist zu beobachten, dass der Fall mit der feinsten Berechnungsgittermaschenweite, das ist $\kappa = 1$, den absolut kleinsten Fehler bei beiden Basen erzeugt. Die hohe Ordnung bringt bei vorgegebener formaler Auflösung für diesen linearen Testfall also keinen Vorteil. Die Laufzeit ist beim vollen Tensorprodukt, für niedrige Ordnung ($\kappa \leq 4$) geringer, obwohl ein kleinerer Zeitschritt als bei der Minimalbasis gewählt werden muss. Ab $\kappa = 5$ kehrt sich der Laufzeitvorteil ins Gegenteil um, und die Minimalbasis ist effizienter als das volle Tensorprodukt. Vermutlich entsteht der Laufzeitvorteil der Minimalbasis bei höherer Ordnung dadurch, dass die Daten beim vollen Tensorprodukt bis $\kappa = 3$ in eine Cachezeile³ passen. Ab $\kappa = 4$ ist diese Grenze überschritten. Bei der Minimalbasis ist diese Grenze erst später überschritten. Hierdurch ergibt sich bei höheren Polynomgraden eine bessere Cacheeffizienz der Minimalbasis.

Im zweiten Test wird hingegen die gleiche Gittermaschenweite gewählt, und nur der Mittelpunkt je Gitterelement ausgegeben. Bei der Entwicklung von DG-COSMO hatte dies den Vorteil, dass direkt die Ausgaberroutinen von COSMO unverändert verwendet werden können. Tabelle 4.4 zeigt für $\kappa = 3$ mit 300×20 Elementen die L_2 -Fehler, L_∞ -Fehler und

³Der Cache lädt oder schreibt immer einen ganzen Speicherblock, „Cachezeile“ genannt, auf einmal.

Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	3	600×40	$288,7 \text{ m} \times 144,3 \text{ m}$	0,200 s	$3,84e-6$	$1,12e-5$	282 s
\mathcal{B}_{vt}	4	520×35	$288,5 \text{ m} \times 142,9 \text{ m}$	0,180 s	$3,88e-6$	$1,26e-5$	237 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			+1,1 %		-1,0 %	-11 %	+19 %
(a) $\kappa = 1$							
Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	6	424×28	$288,8 \text{ m} \times 145,8 \text{ m}$	0,200 s	$4,71e-6$	$1,54e-5$	373 s
\mathcal{B}_{vt}	9	346×23	$289,0 \text{ m} \times 144,9 \text{ m}$	0,180 s	$5,77e-6$	$1,89e-5$	318 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			+0,5 %		-18 %	-18 %	+17 %
(b) $\kappa = 2$							
Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	10	329×22	$288,4 \text{ m} \times 143,7 \text{ m}$	0,180 s	$6,07e-6$	$1,98e-5$	550 s
\mathcal{B}_{vt}	16	260×17	$288,5 \text{ m} \times 147,1 \text{ m}$	0,160 s	$7,69e-6$	$2,51e-5$	450 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			-2,3 %		-21 %	-21 %	+22 %
(c) $\kappa = 3$							
Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	15	268×18	$289,0 \text{ m} \times 143,4 \text{ m}$	0,150 s	$7,46e-6$	$2,44e-5$	672 s
\mathcal{B}_{vt}	25	208×14	$288,5 \text{ m} \times 142,9 \text{ m}$	0,125 s	$9,63e-6$	$3,14e-5$	652 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			+0,6 %		-23 %	-22 %	+3,1 %
(d) $\kappa = 4$							
Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	21	226×15	$289,7 \text{ m} \times 145,5 \text{ m}$	0,125 s	$8,86e-6$	$2,89e-5$	900 s
\mathcal{B}_{vt}	36	173×12	$289,0 \text{ m} \times 138,9 \text{ m}$	0,100 s	$1,16e-5$	$3,78e-5$	923 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			+5,0 %		-24 %	24 %	-2,6 %
(e) $\kappa = 5$							
Basis	d	Gitter	$\Delta x_{\text{for}} \times \Delta z_{\text{for}}$	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	28	196×13	$289,3 \text{ m} \times 145,4 \text{ m}$	0,100 s	$1,02e-5$	$3,33e-5$	1212 s
\mathcal{B}_{vt}	49	148×10	$289,6 \text{ m} \times 142,8 \text{ m}$	0,080 s	$1,36e-5$	$4,42e-5$	1299 s
$\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$			+1,6 %		-25 %	-25 %	-6,7 %
(f) $\kappa = 6$							

Tabelle 4.3.: Vergleich volles Tensorprodukt und Minimalbasis bei gleicher formaler Auflösung, d. h. ähnliche Anzahl an Freiheitsgraden für $\kappa = 1, \dots, 6$. Die Laufzeit bezieht sich auf 4 CPUs der Cray XC30.

4. Numerische Ergebnisse für die trockene Atmosphäre

Laufzeiten. Man sieht, der Fehler sinkt in diesen Ausgabepunkten durch die zusätzlichen Freiheitsgrade des vollen Tensorprodukts nur moderat (weniger als 10 %), die Laufzeit erhöht sich hingegen erheblich (um 38 %).

Basis	Freiheitsgrade	Δt	L_2 -Fehler	L_∞ -Fehler	Laufzeit
\mathcal{B}_{mb}	60 000	0,20 s	3,38e-9	1,01e-8	118,1 s
\mathcal{B}_{vt}	96 000	0,14 s	3,16e-9	9,24e-9	189,8 s
„ $\frac{\mathcal{B}_{mb}}{\mathcal{B}_{vt}}$ “	-38 %		+7 %	+9 %	+38 %

Tabelle 4.4.: Vergleich volles Tensorprodukt und Minimalbasis auf gleichem Gitter, bei einem Ausgabepunkt je Element. Die Laufzeit bezieht sich auf einen Knoten der Cray XC30.

Ist man in der Ausgabe auf die Gittermaschenweite festgelegt, so lohnt sich die Minimalbasis, da die Vorzüge der höheren formalen Auflösung nicht ausgeschöpft werden können. Ebenso ist die Minimalbasis vorzuziehen, wenn die Genauigkeit deutlich wichtiger als die Laufzeit ist. Wenn jedoch das Ausgabegitter vom Berechnungsgitter entkoppelt wird, so kann mit der bestehenden Implementierung mit dem vollen Tensorprodukt ein effizienteres Verfahren bei gleicher formaler Auflösung realisiert werden, wenn die Verfahrensordnung nicht zu hoch ist. Für reale Anwendungen ist der größere Fehler des vollen Tensorprodukts vermutlich eher unbedeutend.

4.2. Ruhende hydrostatische Atmosphäre und Wohlbalanciertheitsproblem

Wird für ein DG-Verfahren eine niedrige Ordnung gewählt und als Referenzatmosphäre eine von der Hintergrundatmosphäre abweichende, so entstehen signifikante Fehler. Dieses Phänomen ist bekannt als *Wohlbalanciertheitsproblem*⁴. Eine Ursache ist, dass die Integrale für den Quellterm und den Flussterm in (3.15) mit verschiedenen numerischen Verfahren gelöst werden. Dadurch müssen sich die Rundungsfehler nicht gegenseitig aufheben, wodurch ein numerisch nicht balanciertes Verfahren entsteht. Bei DG-Verfahren für atmosphärische Strömungen gibt es darüberhinaus noch einen zusätzlichen Grund für ein numerisch nicht wohlbalanciertes Verfahren. Hierfür soll folgendes Problem betrachtet werden: Mit dem COSMO-Referenzzustand wird die vorherige Schwerenwellenausbreitung für $\kappa = 1$ mit einer Gittermaschenweite von $\Delta x = 500$ m, $\Delta z = 250$ m und einem Zeitschritt $\Delta t = 0,2$ s gelöst. Abbildung 4.5 zeigt die numerisch induzierte Abweichung θ' und w nach *einem* einzigen Zeitschritt für einen vertikalen Schnitt bei $x = 275$ km von $z = 1$ km bis $z = 9$ km. Man sieht für θ' eine Abweichung von etwa $1 \cdot 10^{-6}$ K und für w eine von etwa $1 \cdot 10^{-4} \frac{m}{s}$ jeweils über die ganze Höhe. Hier müsste die Atmosphäre noch in Ruhe sein. Der Fehler tritt also nicht nur am Rand auf, sondern im gesamten Gebiet.

Der Grund für diese Störung liegt in der vertikalen Komponente des Rusanov-Flusses und dem θ -Profil. Betrachtet man den Fluss zwischen zwei benachbarten Elementen Ω_i und Ω_j , wobei $e = \Omega_i \cap \Omega_j$ und Ω_i über Ω_j sein soll, dann ist der Rusanov-Fluss von Ω_i nach Ω_j folgender Term:

$$\frac{1}{2}(\rho\theta_i w_i|_e + \rho\theta_j w_j|_e) - \frac{1}{2}\lambda(\rho\theta'_j|_e - \rho\theta'_i|_e). \quad (4.3)$$

⁴Das Wohlbalanciertheitsproblem tritt auch bei anderen Diskretisierungsverfahren auf.

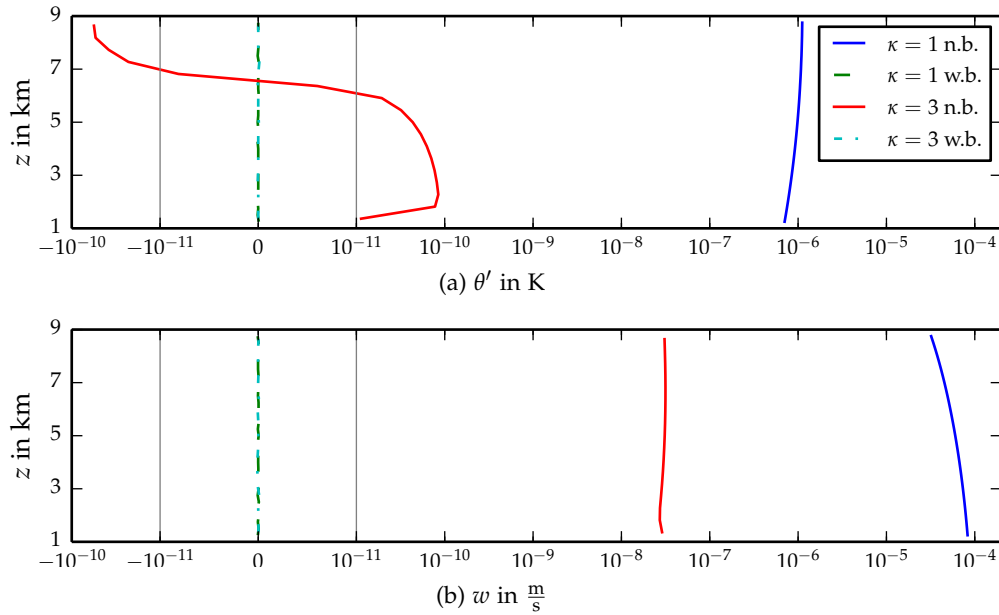


Abbildung 4.5.: Vertikalschnitt durch Baldauf/Brdar 2013 [9] bei $x = 275$ km. Dargestellt werden die nicht-balancierten Lösungen für $\kappa = 1$ (blau) und $\kappa = 3$ (rot) nach einem Zeitschritt, sowie die wohl-balancierten Lösungen für $\kappa = 1$ (grün) und $\kappa = 3$ (türkis). In (a) wird θ' und in (b) wird w dargestellt. Die x -Achsen sind zwischen -10^{-11} und 10^{-11} linear skaliert, ansonsten logarithmisch.

Für die Hintergrundatmosphäre als Referenzatmosphäre verschwindet der zweite Term von (4.3), für die COSMO-Referenzatmosphäre modelliert der erste Term von (4.3) nach wie vor eine sinnvolle Diskretisierung des $\rho\theta$ -Flusses, aber der zweite Term, der zur Stabilisierung des DG-Verfahrens notwendig ist, erzeugt in diesem Fall eine Störung in der Lösung. In Abbildung 4.5 ist auch die nicht-balancierte Lösung für $\kappa = 3$ mit der gleichen formalen Auflösung eingezeichnet. Der Fehler in w geht um etwa drei Zehnerpotenzen zurück, der Fehler in θ' sogar um etwa vier. Die Schwerewellenausbreitung kann mit $\kappa = 3$ und dem COSMO-Referenzzustand auch über längere Zeiträume ohne signifikante Störungen simuliert werden. Dieses Wohlbalanciertheitsproblem wird also mit steigender Ordnung des DG-Verfahrens unbedeutender. Für die Flachwassergleichungen sind wohlbalancierte DG-Verfahren bekannt [94, 65]. Für die Euler-Gleichung ist von Botta et. al. [17] ein FV-Verfahren entwickelt, dass von zweiter Ordnung wohlbalanciert ist. Es setzt jedoch ein bekanntes Profil der potentiellen Temperatur voraus. Wenn dieses bekannt ist, dann kann auch der Referenzzustand aus Abschnitt 2.2 verwendet werden, somit entsteht durch das Botta et. al.-Verfahren kein substantieller Vorteil.

4.3. Fallende Kälteblase nach Straka et. al. 1993

Nachdem im Testfall des vorstehenden Abschnitts DG-COSMOs Konvergenz- und Erhaltungseigenschaften für ein lineares Regime gezeigt wurden, wird in diesem Abschnitt die nicht-lineare Dynamik und die konstante Viskosität validiert. Hierfür wird die zweidimensionale fallende Kälteblase aus [85] verwendet, die auch ausführlich etwa in [63] und [42] behandelt wurde. Dieser Testfall wurde speziell für den Vergleich von verschiedenen numeri-

4. Numerische Ergebnisse für die trockene Atmosphäre

schen Strömungslösern entworfen. Er soll also neben der Validierung von DG-COSMO auch für den Vergleich zwischen DG-COSMO und COSMO herangezogen werden. Dieser Testfall ist mit der einfachen Laplace-Diffusion definiert, d. h. in Gleichung (2.11) wird für τ

$$\tau_L := \rho \mu_m \nabla \mathbf{u}$$

gesetzt. Das Simulationsgebiet ist $L = 51,2$ km lang und $H = 6,4$ km hoch. Die Hintergrundatmosphäre ist hydrostatisch balanciert und das Temperaturprofil ergibt sich aus der Integration des trocken-adiabatischen Temperaturgradienten [16] zu $T_0(z) = T_{00} - \frac{g}{c_p} z$. Für diese isentrope Atmosphäre wird $\theta_0(z) = T_{00} = 300$ K gewählt. Für das Druckprofil folgt $p_0(z) = p_{00} \left(\frac{T_0(z)}{T_{00}} \right)^{\frac{c_p}{R_d}}$. Es werden Schlupfrandbedingungen verwendet, d. h. an den vertikalen Rändern gilt

$$w = \frac{\partial u}{\partial z} = \frac{\partial \theta}{\partial z} = 0,$$

und an den horizontalen Rändern

$$u = \frac{\partial w}{\partial x} = \frac{\partial \theta}{\partial x} = 0.$$

Um eine auskonvergierte Lösung auf einem Gitter mit der formalen Auflösung von 25 m in x - und z -Richtung zu erhalten, wird eine konstante kinematische Viskosität von $\mu_m = \mu_h = 75 \frac{\text{m}^2}{\text{s}}$ verwendet. Die Anfangsbedingungen sind durch folgende Temperaturstörung gegeben:

$$\begin{aligned} T_{\text{init}} &= T_0 + \Delta T, \\ \text{mit } \Delta T &= \begin{cases} 0 \text{ K} & \text{für } R > 1, \\ -15 \text{ K} \frac{1 + \cos(\pi R)}{2} & \text{für } R \leq 1, \end{cases} \\ R &= \sqrt{\left(\frac{x - x_c}{x_r} \right)^2 + \left(\frac{z - z_c}{z_r} \right)^2}, \\ x_c &= 25,6 \text{ km}, \quad x_r = 4 \text{ km}, \quad z_c = 3 \text{ km}, \quad z_r = 2 \text{ km}. \end{aligned}$$

Nach Simulationsbeginn fällt die kalte Blase abwärts, bis sie auf den Boden trifft, und verteilt sich dann symmetrisch in beide horizontale Richtungen, dabei bildet sie etwa alle fünf Minuten einen Kelvin-Helmholtz-Rotor aus.

Die Simulation wird auf einem horizontal und vertikal äquidistanten Berechnungsgitter mit 320×40 Elementen durchgeführt. Als formale Auflösung ergibt sich $\Delta x_{\text{for}} = \Delta z_{\text{for}} = 50,57$ m. Integriert wird mit $\kappa = 3$ und $\Delta t = 0,05$ s. In Abbildung 4.6 wird θ' (a) zum Anfangszeitpunkt $t = 0$ min und (b) zum Endzeitpunkt $t_{\text{end}} = 15$ min dargestellt. In der rechten Hälfte von Abbildung (b) ist die Referenzlösung REFC aus [85] eingezeichnet. Die formale Auflösung von REFC beträgt 25 m in beiden Richtungen. Die Konturlinien der DG-COSMO-Lösung stimmen sehr gut mit der Referenz überein. Um die Qualität der Lösung von DG-COSMO weiter zu beurteilen, auch im Vergleich zu COSMO, werden in Tabelle 4.5 die Extrema der rechten Gebietshälfte für u , w und θ' der beiden Modelle der REFC-Lösung gegenübergestellt. Für COSMO wird hierfür in beiden Richtungen eine formale Auflösung von 50 m gewählt. In der Tabelle 4.5 ist auch die relative Abweichung beider Dynamikkerne von REFC notiert. DG-COSMO ist bei praktisch gleicher formaler Auflösung näher an den Werten von REFC. Der relative Fehler bei DG-COSMO für die Extrema von u_{max} , u_{min} , w_{max} , w_{min} und θ'_{max} ist grob gesprochen nur halb so groß als bei COSMO. Bei θ'_{min} ist der relative Fehler von

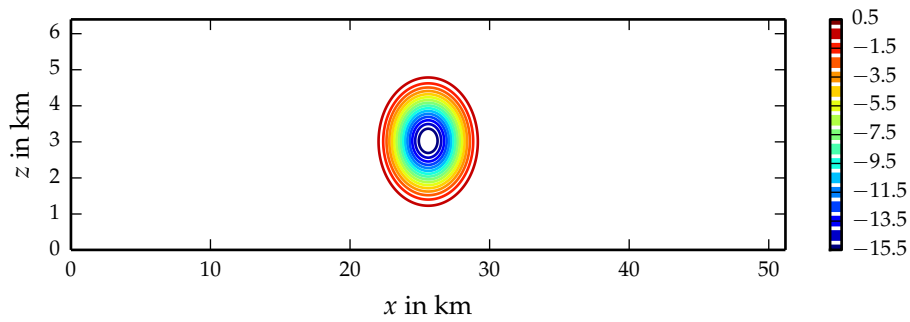
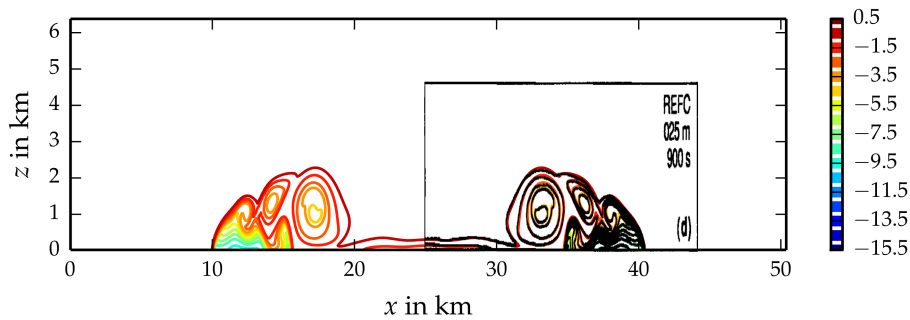
(a) θ' in K bei $t = 0$ min(b) θ' in K bei $t = 15$ min

Abbildung 4.6.: (a) zeigt θ' des Anfangszustands der fallende Kälteblase nach [85]. In (b) wird θ' nach 15 min für $\kappa = 3$ gezeigt (bunte durchgezogene Linien). Die rechte Seite des Bildes ist überlagert mit der Referenzlösung REFC aus [85] (schwarze durchgezogene Linien im schwarzen Rahmen).

DG-COSMO mehr als eine Zehnerpotenz kleiner als der relative Fehler von COSMO. Um schließlich das symmetrische Turbulenzmodell aus Gleichung (2.4) zu validieren, wird der Testfall nochmal mit diesem gerechnet. Es werden nur kleine, aber signifikante Abweichungen erwartet (vgl. [42]). In Tabelle 4.5 sind diese Werte ebenfalls eingetragen. Die Extrema weichen nur geringfügig von der Lösung mit der Laplace-Diffusion ab. In Abbildung 4.7 ist ein Differenzenplot für θ' der beiden Lösungen mit den unterschiedlichen Diffusionstermen dargestellt. Man sieht, dass die Wirbel bis auf Abweichungen von wenigen Hundertstel Kelvin an den Rändern der Wirbel in Bodennähe nahezu gleich sind. Wird die Front der Wirbel in der rechten Bildhälfte definiert als die äußerste x -Stelle der -1 K-Konturlinie in der Höhe $z = 164$ m, so ist die Front der Wirbel für die Diffusion τ_L ($x = 40\,995$ m) einen Meter weiter außen als für die Diffusion τ ($x = 40\,994$ m). Die Front ist in [42] und [85] nicht für eine bestimmte Höhe definiert. Die beiden DG-Lösungen liegen in größerer Höhe aber praktisch aufeinander, daher wird hier die Front in Bodennähe betrachtet. Wie in [42] argumentiert, sollte die Lösung mit der Diffusion τ die bessere sein, da die verwendete Diffusion näher an der physikalischen Diffusion ist.

Zum Abschluss dieses Testfalls soll gezeigt werden, dass das LDG-Schema mit konstanter kinematischer Viskosität tatsächlich gegen die hochaufgelöste Lösung konvergiert. Da keine

4. Numerische Ergebnisse für die trockene Atmosphäre

	DG-COSMO mit τ_L		DG-COSMO mit τ		COSMO		REFC
u_{\max}	36,15 $\frac{\text{m}}{\text{s}}$	-0,8 %	36,17 $\frac{\text{m}}{\text{s}}$	-0,8 %	35,54 $\frac{\text{m}}{\text{s}}$	-2,5 %	36,46 $\frac{\text{m}}{\text{s}}$
u_{\min}	-15,34 $\frac{\text{m}}{\text{s}}$	+1,0 %	-15,35 $\frac{\text{m}}{\text{s}}$	+1,0 %	-15,52 $\frac{\text{m}}{\text{s}}$	+2,2 %	-15,19 $\frac{\text{m}}{\text{s}}$
w_{\max}	13,08 $\frac{\text{m}}{\text{s}}$	+1,2 %	13,09 $\frac{\text{m}}{\text{s}}$	+1,2 %	13,28 $\frac{\text{m}}{\text{s}}$	+2,7 %	12,93 $\frac{\text{m}}{\text{s}}$
w_{\min}	-16,04 $\frac{\text{m}}{\text{s}}$	+0,6 %	-16,05 $\frac{\text{m}}{\text{s}}$	+0,6 %	-16,15 $\frac{\text{m}}{\text{s}}$	+1,2 %	-15,95 $\frac{\text{m}}{\text{s}}$
θ'_{\max}	0,14 K	-	0,14 K	-	0,23 K	-	0,00 K
θ'_{\min}	-9,79 K	+0,2 %	-9,79 K	+0,2 %	-9,46 K	-3,2 %	-9,77 K

Tabelle 4.5.: Vergleich der Extrema für den Straka et. al. 93-Testfall nach 15 min für die rechte Gehäufthälfte. Gezeigt werden DG-COSMO mit der Laplace-Diffusion τ_L , dem symmetrischen Turbulenzmodell τ aus (2.4), sowie COSMO und als Referenz REFC. Die Prozentangaben sind die relative Abweichung von REFC.

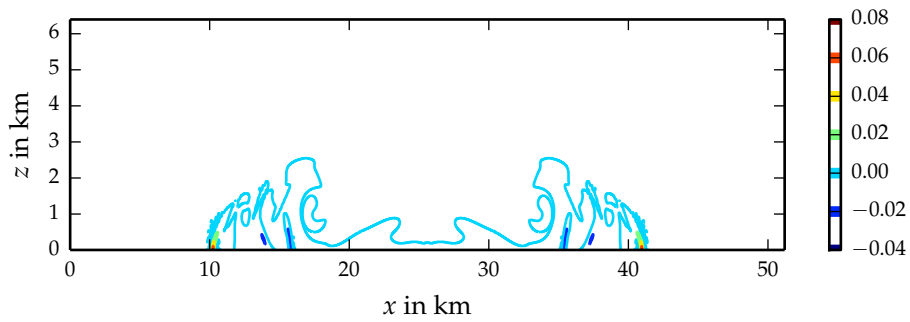


Abbildung 4.7.: Differenzenplot zwischen den DG-Lösungen mit den beiden unterschiedlichen Diffusionstermen des Straka-Testfalls. Es wird $\theta'_{\tau_L} - \theta'_{\tau}$ dargestellt.

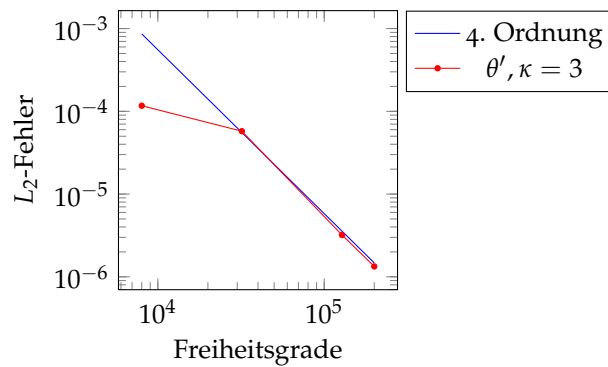


Abbildung 4.8.: Konvergenzanalyse für den Straka et. al. 93-Testfall nach 15 min für DG-COSMO mit $\kappa = 3$ für θ' .

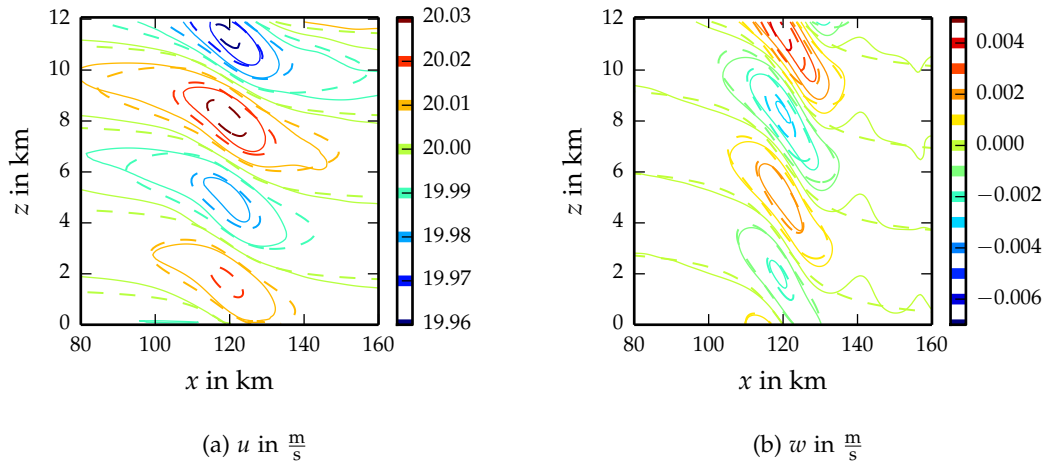


Abbildung 4.9.: lineare Bergüberströmung. Abbildung (a) zeigt die Horizontal- und Abbildung (b) die Vertikalgeschwindigkeit. Die durchgezogenen Linien zeigen die Lösung von DG-COSMO, die gestrichelten Linien die linearisierte analytische Lösung.

numerischen Daten für die REFC-Lösung vorliegen, wird eine eigene Referenzlösung mit $\kappa = 3$ und einer formalen Auflösung von 10,1 m in beide Richtungen berechnet. Die Fehlerentwicklung für $\kappa = 3$ ist in Abbildung 4.8 dargestellt. Man sieht, dass DG-COSMO die formale Konvergenzordnung erreicht.

4.4. Bergüberströmungen

In diesem Abschnitt werden die geländefolgenden Koordinaten von DG-COSMO mit einer zweidimensionalen Überströmung eines idealisierten Bergs validiert. Bergüberströmungen einschließlich der Herleitung von (linearisierten) analytischen Lösungen für Spezialfälle werden ausführlich in [82] oder [35] diskutiert. Das Simulationsgebiet ist $L = 240$ km lang und $H = 42$ km hoch. Der Berg befindet sich in der Mitte des Simulationsgebiets, d. h. $x_c = 120$ km. Für die geländefolgenden Koordinaten wird ein Höhenprofil z_s und dessen Ableitung benötigt. Hier soll das in der Literatur unter dem Namen *Versiera di Agnesi*⁵ bekannte Profil verwendet werden

$$z_s(x) = \frac{h_c}{1 + \left(\frac{x-x_c}{a_c}\right)^2}, \quad \frac{\partial z_s}{\partial x} = -\frac{2h_c a_c^2 (x-x_c)}{(a_c^2 + (x-x_c)^2)^2}$$

mit der Berghöhe h_c und der Halbwertsbreite a_c . Abbildung 2.1 stellt das Höhenprofil graphisch dar. Die horizontalen Randbedingungen sind periodisch. Die untere vertikale Randbedingung ist eine Schlupfbedingung, die obere ist offen, wobei die Schwammschicht sich über die oberen 12 km erstreckt. Die Atmosphäre ist isotherm und hydrostatisch balanciert.

⁵In der Meteorologie ist dieses Profil auch unter dem Namen „Witch of Agnesi“ bekannt. Der Ausdruck „Witch“ basiert jedoch auf einem Übersetzungsfehler, mit „Versiera“ ist in etwa „die sich Wendende“ gemeint.

4. Numerische Ergebnisse für die trockene Atmosphäre

Modell	DG-COSMO	DG-COSMO	COSMO	
Basis, Gitter	$\mathcal{B}_{mb}, 240 \times 100$	$\mathcal{B}_{vt}, 196 \times 82$	588 × 245	
Freiheitsgrade	144 000	144 648	144 060	
Δt	0,0250 s	0,0225 s	10,0 s	0,0250 s, $t_{\text{end}} = 10$ h
L_2 -Fehler von u'	2,95e−3	2,88e−3	2,60e−3	4,39e−3
L_2 -Fehler von w	2,31e−4	3,35e−4	1,56e−4	4,22e−4
Laufzeit	15 160 s	12 618 s	226 s	65 051 s

Tabelle 4.6.: L_2 -Fehler für DG-COSMO mit \mathcal{B}_{mb} , DG-COSMO mit \mathcal{B}_{vt} , COSMO für die lineare hydrostatische Bergüberströmung bei gleicher Anzahl von Freiheitsgraden. Die linearisierte analytische Lösung aus [35] ist die Referenz. Die Laufzeit bezieht sich auf 3 Knoten der Cray XC30 und wurde bei allen drei Modellen mit einer Simulationszeit von $t_{\text{end}} = 10$ h ermittelt.

Die Profile für Temperatur, Druck und potentielle Temperatur sind gegeben durch:

$$\begin{aligned}
 T_0(x, z) &= T_{00} = 250 \text{ K}, & \theta_0(x, z) &= T_{00} e^{\frac{g}{c_p T_{00}} z}, \\
 p_0(x, z) &= p_{00} e^{-\frac{g}{R_d T_{00}} z}, & \rho_0(x, z) &= \rho_{00} e^{-\frac{g}{R_d T_{00}} z}, \quad \rho_{00} = \frac{p_{00}}{R_d T_{00}}.
 \end{aligned}$$

Die Brunt-Väisälä-Frequenz ist in diesem Testfall konstant mit

$$N = \frac{g}{\sqrt{c_{pd} T}} = 0,019564 \frac{1}{\text{s}}.$$

Der Anfangszustand besteht aus einem rein horizontalen Wind mit $u_0 = 20 \frac{\text{m}}{\text{s}}$. Die Simulationszeit beträgt $t_{\text{end}} = 20$ h. Am Simulationende hat sich eine stationäre innere Schwerewelle ausgebildet. Für $\frac{N h_c}{u_0} \ll 1$ ist die Welle nahe einer linearen und für $\frac{N a}{u_0} \gg 1$ nahe an einer hydrostatischen Lösung (vgl. [35]). Mit $h_c = 1$ m und $a_c = 10$ km sind beide Bedingungen erfüllt. Die numerische Lösung kann also mit einer linearisierten analytischen Lösung verglichen werden. Für den hier definierten Testfall findet man diese linearisierte analytische Lösung etwa in [35].

Die Simulation wird auf dem Berechnungsgitter 240×100 Elemente mit $\beta = 0,9$ für die Streckungsfunktion (3.9) und $\kappa = 2$ mit einem Zeitschritt von $\Delta t = 0,025$ s durchgeführt. Der Dämpfungsparameter der Schwammschicht (3.25) ist $n_\delta = 4000$. Die Wahl dieses Parameters kann die Lösung stark beeinflussen. Wird dieser Wert zu klein gewählt, so kommt es zu Reflexionen am unteren Rand der Schwammschicht, wird er zu groß gewählt gibt es Reflexionen am oberen Rand. Abbildung 4.9 zeigt die unteren 12 km der Lösung für DG-COSMO und die linearisierte Lösung aus [35]. In Tabelle 4.6 ist der L_2 -Fehler von DG-COSMO für obige Simulation mit der Minimalbasis, für eine Simulation mit vollem Tensorprodukt als Basis und von COSMO dargestellt. COSMO wurde mit den Zeitschritten $\Delta t_{\text{COSMO}} = 10$ s und $\Delta t_{\text{COSMO}} = 0,025$ s gerechnet. Für $\Delta t_{\text{COSMO}} = 0,025$ s wurde aufgrund der hohen Laufzeiten nur bis $t_{\text{end}} = 10$ h integriert. Die Laufzeiten sind für die Vergleichbarkeit deshalb auch bei DG-COSMO für die Simulationszeit $t_{\text{end}} = 10$ h angegeben. Alle drei Simulationen sind bei gleicher formaler Auflösung gerechnet. Der L_2 -Fehler wird über dem Gebietsausschnitt $[80 \text{ km}, 160 \text{ km}] \times [0 \text{ km}, 12 \text{ km}]$ berechnet und bezieht sich auf die Abweichung von der Hintergrundströmung, d. h. der L_2 -Fehler wird für $u' := u - u_0$ und für w bestimmt. Dabei hat

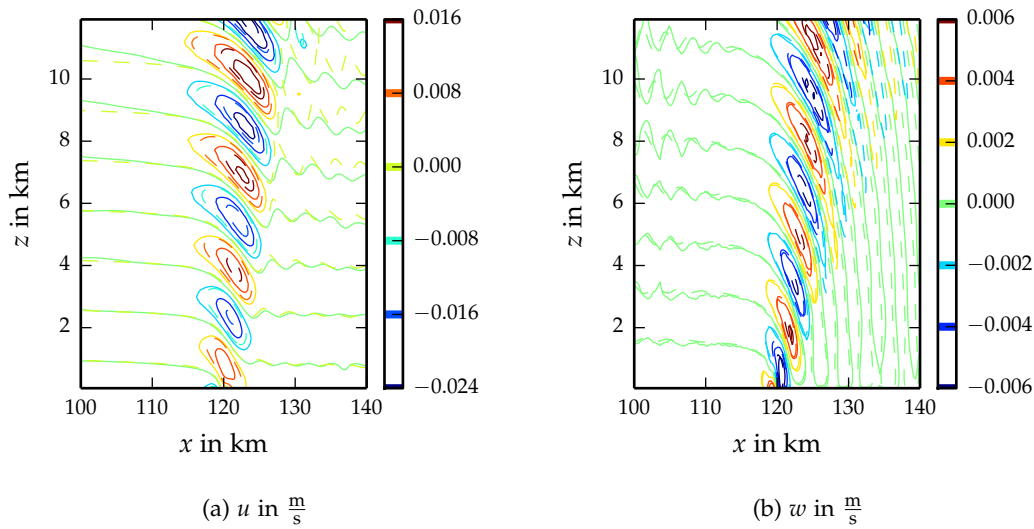


Abbildung 4.10.: lineare nicht-hydrostatische Bergüberströmung. Abbildung (a) zeigt die Horizontal- und Abbildung (b) die Vertikalgeschwindigkeit. Die durchgezogenen Linien zeigen die Lösung von DG-COSMO, die gestrichelten Linien die linearisierte analytische Lösung aus [5].

das Ausgabegitter für die beiden DG-Simulationen 960×400 Elemente, das Ausschnittsgebiet, auf dem der L_2 -Fehler gerechnet wird, hat 320×115 Elemente. Bei COSMO ist das Ausgabegitter gleich dem Berechnungsgitter. Der L_2 -Fehler wird also auf einem anderen Gitter bestimmt, das Ausschnittsgebiet hat 196×124 Elemente. Durch die verschiedenen Gitter ändert sich der L_2 -Fehler um etwa 1%. Die beiden DG-COSMO-Simulationen und die COSMO Simulation mit $\Delta t_{\text{COSMO}} = 10$ s haben einen vergleichbaren L_2 -Fehler für u' . Für w ist der Fehler dieser COSMO-Lösung signifikant kleiner als die beiden DG-COSMO-Lösungen. Das volle Tensorprodukt hat in w den größeren L_2 -Fehler als die Minimalbasis. Für den kleinen Zeitschritt hat die COSMO-Lösung einen deutlich größeren L_2 -Fehler in u' und w als die anderen Simulationen, hier ist der stationäre Zustand wohl noch nicht ganz erreicht. Die Simulation mit dem vollen Tensorprodukt ist hingegen etwa 20% schneller als die Simulation mit der Minimalbasis. DG-COSMO ist erheblich schneller als COSMO wenn ein vergleichbarer Zeitschritt verwendet wird. Wird für DG-COSMO ein vertikal impliziter Löser entwickelt, so kann der Zeitschritt fast genau so groß wie in COSMO gewählt werden. Auch wird erwartet, dass ein vertikal impliziter Löser besser als ein semi-impliziter Löser skaliert (vgl. Diskussion in Abschnitt 6.2). Aus diesen beiden Argumenten lässt sich die begründete Hoffnung ableiten, dass DG-COSMO auch mit einem größeren Zeitschritt (durch eine vertikal implizite Integration) immer noch effizient ist.

Als zweiter Testfall für die Orographie wird eine lineare nicht-hydrostatische Bergüberströmung betrachtet. Hierfür dient obiger Testfall als Grundlage und wird wie folgt verändert: der Berg hat eine Halbwertsbreite von $a = 1$ km, der Hintergrundwind ist $u_{\text{init}} = 10 \frac{\text{m}}{\text{s}}$ und die Simulationszeit beträgt $t_{\text{end}} = 10$ h. In Abbildung 4.10 sind die unteren 12 km des Horizontal- und Vertikalwinds dargestellt. Die linearisierte analytische Lösung ist aus [5]⁶. Aus Tabelle 4.7 kann entnommen werden, dass bei diesem Testfall die Minimalbasis den kleineren Fehler

⁶Für den hier gerechneten Fall wurde von Michael Baldauf einen Datensatz mit der linearisierten analytischen Lösung bereitgestellt.

4. Numerische Ergebnisse für die trockene Atmosphäre

Modell	DG-COSMO	DG-COSMO
Basis, Gitter	$\mathcal{B}_{mb}, 240 \times 100$	$\mathcal{B}_{vt}, 196 \times 82$
Freiheitsgrade	144 000	144 648
Δt	0,0250 s	0,0225 s
L_2 -Fehler von u'	$9,94e-4$	$2,88e-3$
L_2 -Fehler von w	$6,67e-4$	$2,19e-3$

Tabelle 4.7.: L_2 -Fehler der lineare nicht-hydrostatische Bergüberströmung für die Minimalbasis und für das volle Tensorprodukt.

als das volle Tensorprodukt liefert. Dieser Testfall belegt, wie auch die Schwerewellenausbreitung im Kanal, dass die Minimalbasis in der Genauigkeit besser abschneidet, das volle Tensorprodukt jedoch die geringere Laufzeit hat.

In [40] wurde dieser Testfall mit stetigen Galerkin-Verfahren für Bilanzgleichungen in der Advektionsform gerechnet und für geländefolgende Koordinaten und kartesische Koordinaten miteinander verglichen. Die geländefolgenden Koordinaten haben einen größeren L_2 -Fehler als die Rechnung mit kartesischen Koordinaten.

4.5. Berg in ruhender Atmosphäre

Ein weiterer Testfall für atmosphärische Modelle ist die Simulation des Ruhezustands. Die Anfangsatmosphäre ist hydrostatisch balanciert, ungestört und in Ruhe, weicht jedoch von der Referenzatmosphäre ab. Im Simulationsgebiet ist ein (steiler) Berg platziert. Obwohl die physikalische Atmosphäre unverändert bleiben sollte, haben die numerischen Strömungslöser Schwierigkeiten dies wiederzugeben. Der hier vorgestellte Test lehnt sich an die Simulation in [95] an. Ich betrachte Berge der Form einer Gauß'schen Glockenkurve mit der Höhenfunktion

$$z_s(x) = h e^{-\frac{(x-x_c)^2}{a^2}}.$$

Dabei sollen die beiden Fälle a) $h = 4000$ m und b) $h = 7000$ m betrachtet werden. Als Halbwertsbreite wird $a = 3000$ m verwendet. Die maximale Steigung beträgt somit a) $48,8^\circ$ und b) $63,4^\circ$. In [95] wird die Halbwertsbreite 2000 m gewählt, aber COSMO läuft bei so steilen Bergen nicht mehr stabil, daher wird in dieser Betrachtung ein flacherer Berg gewählt. Damit läuft COSMO wenigstens für $h = 4000$ m stabil, für $h = 7000$ m ist es immer noch instabil. Das Simulationsgebiet ist $L = 35$ km breit und $H = 40$ km hoch, wobei die oberen 15 km eine Schwammschicht sind. Der Zeitschritt ist $\Delta t = 0,01$ s, sowohl für COSMO als auch für DG-COSMO. Als formale horizontale Auslösung wird 300 m gewählt. In der Vertikalen wird ein gestrecktes Gitter mit 100 Freiheitsgraden gewählt und $\beta = 0,95$ für die Streckungsfunktion (3.9). Folglich werden 11700 Freiheitsgrade verwendet. DG-COSMO simuliert beide Höhen mit $\kappa = 1, 2, 3$. Abbildung 4.11 zeigt die numerisch induzierte Vertikalgeschwindigkeit nach $t_{\text{end}} = 6$ h für die vier Simulationen. Tabelle 4.8 zeigt für COSMO- und die verschiedenen DG-COSMO-Läufe die verwendeten Gitter und die Extrema der Vertikalgeschwindigkeit nach t_{end} . Für $h = 4000$ m ist der numerisch induzierte Vertikalwind von DG-COSMO mit $\kappa = 1$ etwa eine Zehnerpotenz größer als bei COSMO. Bei $\kappa = 2$ sind die Fehler in der gleichen Größenordnung, bei $\kappa = 3$ ist DG-COSMOs Fehler eine Zehnerpotenz kleiner als COSMOs Fehler. Für $h = 7000$ m geht bei DG-COSMO w_{min} mit der Erhöhung der Ordnung jeweils um etwa eine Zehnerpotenz zurück, w_{max} geht deutlich langsamer zurück, bei $\kappa = 2$ ist w_{max} sogar größer als bei $\kappa = 1$.

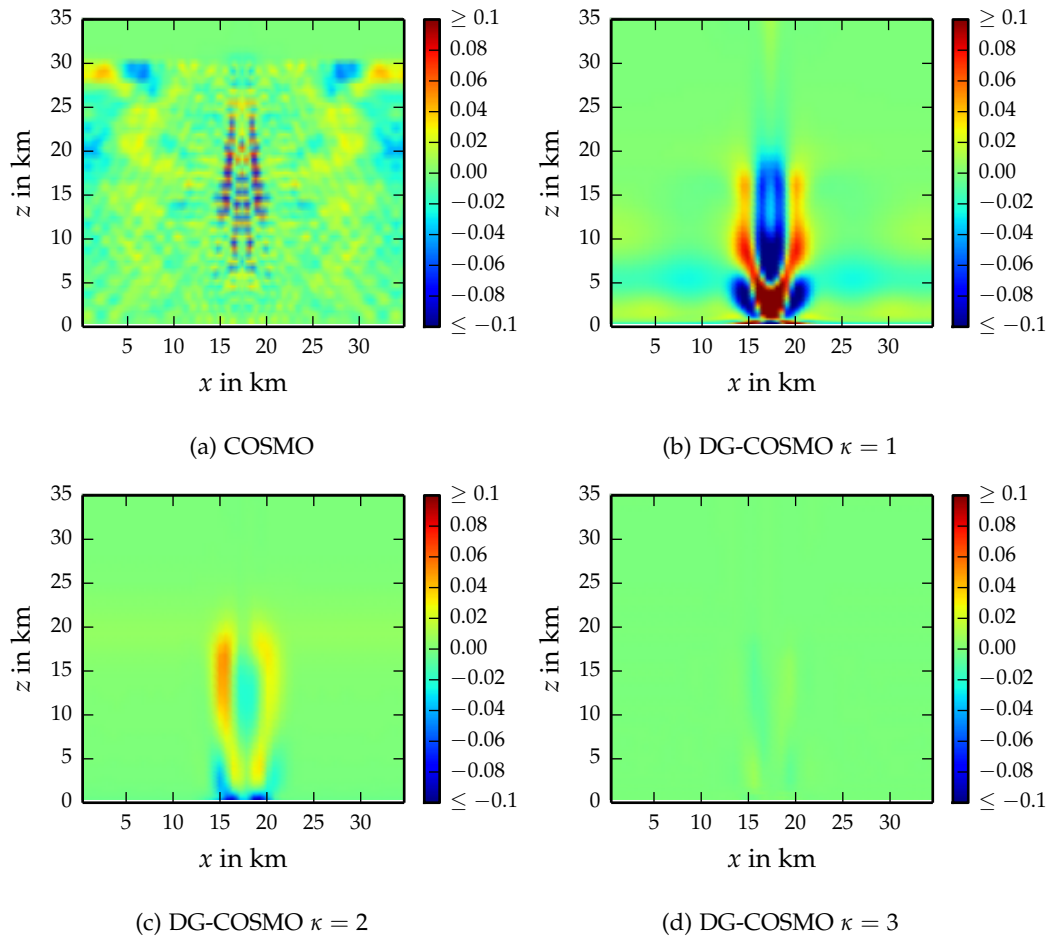


Abbildung 4.11.: Atmosphäre in Ruhe für $h = 4000$ m. Gezeigt wird w in $\frac{\text{m}}{\text{s}}$ für COSMO und DG-COSMO. Der Farbbereich ist auf die COSMO-Lösung ausgerichtet, für (b) (siehe Tabelle 4.8) treten Werte um eine Größenordnung größer auf. Je Gitterelement ist nur der Mittelpunkt dargestellt.

Modell	Gitter	$h = 4000$ m		$h = 7000$ m	
		w_{\min}	w_{\max}	w_{\min}	w_{\max}
COSMO	117×100	$-8,6e-2$	$7,2e-2$	instabil	
DG-COSMO $\kappa = 1$	67×58	$-7,1e-1$	$3,5e-1$	$-8,5e-1$	$2,1e-1$
DG-COSMO $\kappa = 2$	48×41	$-9,5e-2$	$5,1e-2$	$-8,5e-2$	$3,2e-1$
DG-COSMO $\kappa = 3$	37×32	$-7,7e-3$	$9,0e-3$	$-7,4e-3$	$1,9e-2$

Tabelle 4.8.: Extrema der numerischen Störung in der Vertikalgeschwindigkeit. Verglichen wird COSMO mit DG-COSMO für verschiedene Polynomgrade κ , bei gleicher Anzahl an Freiheitsgraden für zwei verschiedene Höhen.

4. Numerische Ergebnisse für die trockene Atmosphäre

In [74], dessen Hauptautor der Autor dieser Arbeit ist, wird dieser Testfall noch zusätzlich mit dem DUNE-Löser von Brdar (vgl. Abschnitt 1.1) mit einem kartesischen Gitter verglichen. Ansonsten ist für die trockene Atmosphäre, wie bereits in der Einleitung erwähnt, der DUNE-Löser bis auf die unterschiedlichen Koordinaten mit DG-COSMO vergleichbar. Der DUNE-Löser erreicht auf dem kartesischen Gitter bereits bei $\kappa = 1$ einen mit COSMO vergleichbaren L_2 -Fehler, bei $\kappa = 2$ ist der Fehler bereits eine Zehnerpotenz, bei $\kappa = 3$ zwei Zehnerpotenzen kleiner als COSMO und eine Zehnerpotenz kleiner als DG-COSMO. Das im letzten Abschnitt erwähnte Ergebnis aus [40], wo mit stetigen Galerkin-Verfahren eine Bergüberströmung sowohl mit geländefolgenden Koordinaten als auch mit kartesischen Koordinaten simuliert wurde lässt sich also, durch den Vergleich zwischen DG-COSMO und dem DUNE-Löser, auf die ruhende Atmosphäre übertragen.

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

In diesem Kapitel soll das Modell der feuchten Atmosphäre mit warmem Niederschlag, d. h. ohne Eisphase, evaluiert werden. Darüber hinaus wird untersucht, welchen Einfluss die Ordnung des Verfahrens, die Größe der Zeitskala für das Kessler-Schema und die Sättigungsadjustierung und die beiden in Abschnitt 3.3 diskutierten Aufteilungsstrategien der räumlichen Diskretisierung auf das Simulationsergebnis haben. Die Aufteilungsstrategien werden auch in der Laufzeit verglichen.

Als zweiter Teil wird in diesem Kapitel eine Skalierbarkeitsanalyse von DG-COSMO für einen trockenen und einen feuchten Testfall durchgeführt. Dieser Analyse wird die Skalierbarkeit von COSMO gegenübergestellt und es wird für den trockenen Fall gezeigt, ab welcher CPU-Anzahl das DG-Verfahren weniger Laufzeit benötigt als das FD-Verfahren.

5.1. Feuchtkonvektion nach Weisman und Klemp 1982

Im folgenden Testfall wird eine feuchte Konvektionszelle betrachtet, die während des Aufstiegs ausregnet. Der Simulationsaufbau basiert auf dem Testfall aus [87]. Mit diesem Testfall soll untersucht werden, welchen Einfluss das Verhältnis σ der Zeitschritte der Parametrisierungen $\Delta\tau$ und der Dynamik Δt (vgl. Abschnitt 3.3), die Verfahrensordnung und die Aufteilungsstrategie auf die Lösung haben. Auch wirkt sich eine Änderung der formalen Auflösung stark auf die Lösung aus (vgl. [37]), diese wird hier jedoch konstant gewählt.

Das dreidimensionale Simulationsgebiet erstreckt sich über $L = 50$ km in beiden horizontalen Richtungen und über $H = 22$ km in der vertikalen. Am Oberrand und an den horizontalen Rändern werden offene Randbedingungen verwendet. Die vertikale Schwammschicht ist 7 km dick, die horizontalen jeweils 10 km. Der Dämpfungsfaktor n_δ aus Gleichung (3.25) ist 2000. Am unteren Rand werden Schlupfbedingungen gefordert. Die Hintergrundatmosphäre ist hydrostatisch balanciert und durch ein vertikales Profil der potentiellen Temperatur θ und der relativen Feuchte f gegeben:

$$\theta(z) = \begin{cases} \theta_{00} + (\theta_{tr} - \theta_{00}) \left(\frac{z}{z_{tr}}\right)^{\frac{5}{4}}, & z \leq z_{tr}, \\ \theta_{tr} e^{\frac{g}{c_p d T_{tr}}(z - z_{tr})}, & z > z_{tr}, \end{cases}$$

$$f(z) = \begin{cases} 1 - \frac{3}{4} \left(\frac{z}{z_{tr}}\right)^{\frac{5}{4}}, & z \leq z_{tr}, \\ \frac{1}{4}, & z > z_{tr}, \end{cases}$$

mit der Troposphärenhöhe $z_{tr} = 12$ km, der potentiellen Temperatur am Boden $\theta_{00} = 300$ K, und am Oberrand der Troposphäre: $\theta_{tr} = 338$ K und $T_{tr} = 213$ K. Der Wasserdampfgehalt der Hintergrundatmosphäre ist auf ein Maximum von 0,014 begrenzt. Hierdurch wird eine gut durchmischte Grenzschicht modelliert. Der Anfangszustand des Testfalls ist die Hintergrundatmosphäre mit folgender Temperaturstörung:

$$\Delta T = \begin{cases} d_T \cdot \cos\left(\frac{\pi}{2}L\right)^2, & R \leq 1, \\ 0, & R > 1, \end{cases}$$

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

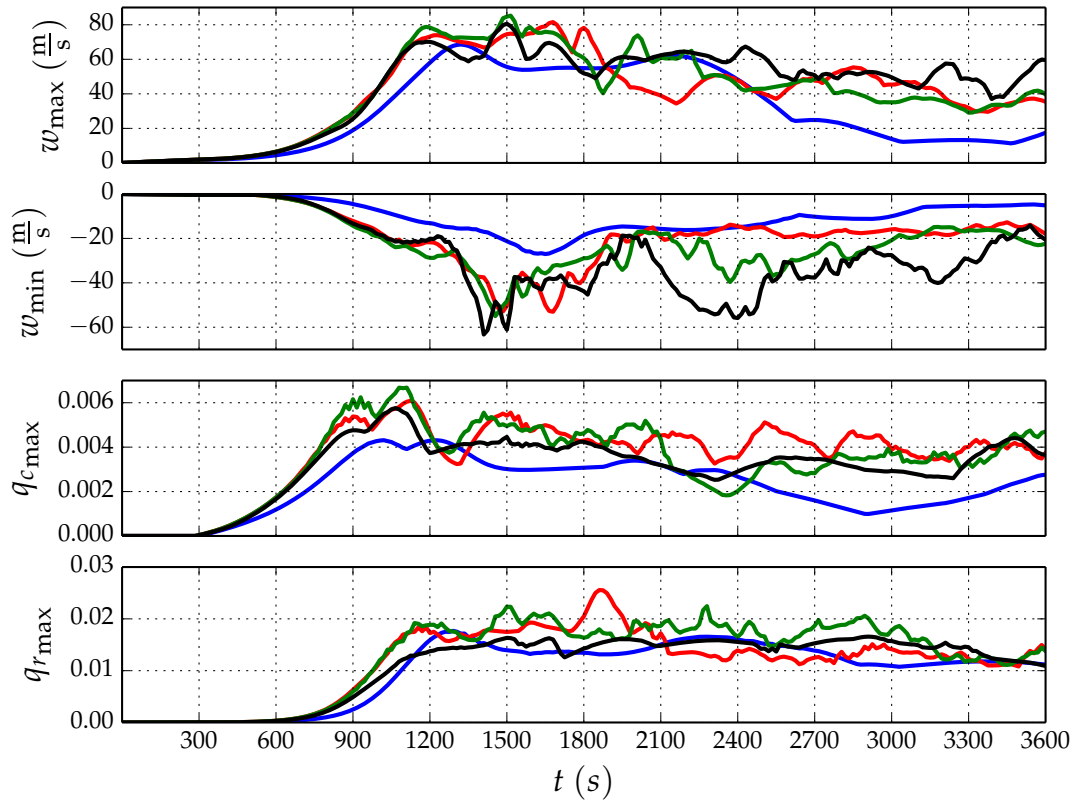


Abbildung 5.1.: Vergleich von DG-COSMO $\kappa = 1$ (blau), DG-COSMO $\kappa = 2$ (rot), DG-COSMO $\kappa = 3$ (grün) und COSMO (schwarz) der Entwicklung des Minimums und Maximums der Vertikalgeschwindigkeit und des Maximums des Wolkenwassergehalts.

mit $R = \sqrt{\left(\frac{x-x_c}{r_x}\right)^2 + \left(\frac{y-y_c}{r_y}\right)^2 + \left(\frac{z-z_c}{r_z}\right)^2}$, $d_T = 2 \text{ K}$, $r_x = r_y = 10 \text{ km}$ und $r_z = 1400 \text{ m}$. Diese Wärmeblase liegt auf dem Boden und ist in der horizontalen Mitte des Simulationsgebiets positioniert, das heißt $z_c = 1400 \text{ m}$, $x_c = 25 \text{ km}$ und $y_c = 25 \text{ km}$. Die Simulationszeit beträgt $t_{\text{end}} = 1 \text{ h}$.

Nach dem Start der Simulation beginnt die Blase aufgrund von Feuchtkonvektion aufzusteigen und eine Wolke zu bilden. Die freigesetzte Enthalpie beschleunigt den Aufstieg, bis die Blase den oberen Rand der Troposphäre erreicht. Dies ist nach etwa 20 min der Fall. Es bildet sich schon während des Aufstiegs Regenwasser, das zum Teil niederschlägt und zum Teil vom Auftrieb mit nach oben gerissen wird. Im weiteren Verlauf löst sich die anfängliche Wolkenstruktur auf und es entstehen neue Wolken. Dieser Testfall ist sehr sensitiv, es ist also zu erwarten, dass die Lösungen von COSMO und DG-COSMO sich in ihrer zeitlichen Entwicklung stark unterscheiden, aber wenigstens in der Aufstiegsphase sollten beide Lösungen ähnlich genug sein, um die DG-COSMO-Lösung auf Plausibilität zu überprüfen durch einen Vergleich mit der COSMO-Lösung.

Damit nicht allzu viele Feinstrukturen auftreten, wird eine konstante kinematische Viskosität von $\mu_m = \mu_h = 50 \frac{\text{m}^2}{\text{s}}$ verwendet, um noch realitätsnahe Simulationen zu erhalten,

wird der Wert nicht höher gewählt. Die Simulation des Testfalls mit DG-COSMO wird für drei Ordnungen gerechnet: 1. Für den Polynomgrad $\kappa = 1$ auf $68 \times 68 \times 54$ Elementen; 2. $\kappa = 2$ auf $50 \times 50 \times 40$ Elementen und 3. $\kappa = 3$ auf $40 \times 40 \times 32$ Elementen; jede hat also etwa eine Million Freiheitsgrade und die gleiche formale Auflösung. Der Dynamikzeitschritt ist $\Delta t = 0,025$ s für alle drei Ordnungen, der Zeitschritt der Parametrisierung $\Delta \tau = 120 \cdot \Delta t = 3$ s. Zum Vergleich wird die COSMO-Lösung mit der gleichen formalen Auflösung gerechnet, es ergeben sich hierfür $108 \times 108 \times 86$ Elemente. COSMO verwendet für die Dynamik, das Kessler-Schema und die Sättigungsadjustierung den gleichen Zeitschritt, er beträgt $\Delta \tau_{\text{COSMO}} = 3$ s.

Abbildung 5.2 vergleicht die vertikale Geschwindigkeit von DG-COSMO mit $\kappa = 1, 2, 3$ nach 10 min, 15 min und 20 min, was in etwa dem Anfang, der Mitte und dem Ende der Aufstiegsphase der Anfangsstörung entspricht. Zur Plausibilitätsprüfung ist die COSMO-Lösung ebenfalls eingezeichnet. Abbildung 5.3 zeigt den Wolkenwassergehalt und Abbildung 5.4 den Regenwassergehalt für die gleichen Zeitpunkte. In diesen drei Abbildungen sieht man deutlich, dass der Fall $\kappa = 1$ viel zu diffusiv ist. Für $\kappa = 2$ ist die Form für alle drei Größen sehr ähnlich der Lösung aus COSMO. Nach 20 min tritt jedoch ein größeres Maximum des Wolkenwassers auf. Dieses findet man in der mittleren Wolkenhöhe. Für q_r findet sich ebenfalls ein größeres Maximum, jedoch in der oberen Wolke. Für $\kappa = 3$ ist die Übereinstimmung in q_r des unteren Teils der großen Rotoren auf der linken und rechten Seite der Wolke größer, aber der obere Teil der Wolke scheint schneller aufzusteigen und zusätzliche Turbulenz im Bezug auf die COSMO-Lösung zu erzeugen. In der unteren Atmosphäre tritt mit höherer Ordnung eine „Nebenwolke“ auf.

Um die Plausibilität der Ergebnisse weiter zu prüfen, werden in Abbildung 5.1 die Zeitreihen der räumlich maximalen und minimalen Vertikalgeschwindigkeit, sowie der räumlich maximale Wert für q_c und q_r , dargestellt. Die Kurven für $\kappa = 1$ verlaufen deutlich glatter und sind für w auch flacher als für die anderen Ordnungen, hier zeigt sich wieder die zu starke Diffusivität. Die w -Extrema der DG-COSMO-Lösungen für $\kappa = 2, 3$ stimmen gut mit der COSMO-Lösung in den ersten 20 min bis 30 min überein, danach zeigt sich das Auseinanderlaufen der Lösungen auch in den Extrema von w stärker. Für die Feuchte zeigt sich ein anderes Bild. Sowohl für q_c als auch für q_r liefert DG-COSMO viel größere Maxima, dies gilt bereits für die ersten 20 min. Der Verlauf der Maxima von DG-COSMO ist nicht so glatt wie bei COSMO, daher kann man schließen, dass DG-COSMO eine geringere Diffusivität als COSMO hat.

Nun soll die Abhängigkeit von $\Delta \tau$ betrachtet werden. Hierzu wurde die Simulation für $\kappa = 2$ und $\kappa = 3$ für die Werte $\Delta \tau = 30$ s, 3 s, 0,5 s gerechnet und der Zeitpunkt 20 min als Vergleichszeitpunkt ausgewählt. Für $\Delta \tau = \Delta t = 0,025$ s entstehen so hohe Windgeschwindigkeiten, dass die CFL-Bedingung (3.20) für $\kappa = 2$ nach etwa 23 min und für $\kappa = 3$ schon nach 17 min verletzt ist. Daher soll dieser Fall nicht mit in den Vergleich aufgenommen werden. Auf den folgenden Seiten werden die Ergebnisse systematisch wie folgt angeordnet: w wird in Abbildung 5.5 für $\kappa = 2$ und in Abbildung 5.6 für $\kappa = 3$ dargestellt. In Abbildung 5.7 werden die unterschiedlichen Einflüsse auf w verglichen. Die Abbildungen 5.8 und 5.9 stellen die Ergebnisse für $\kappa = 2$ und $\kappa = 3$ von q_c dar und die Abbildungen 5.10 und 5.11 stellen die Ergebnisse für $\kappa = 2$ und $\kappa = 3$ von q_r dar. Es ist deutlich zu sehen, dass für $\kappa = 2$ mit kleiner werdendem $\Delta \tau$ zusätzliche Feinstrukturen auftreten (vgl. Abbildung 5.5 für w , Abbildung 5.8 für q_c und Abbildung 5.10 für q_r), dabei unterscheiden sich die Fälle $\Delta \tau = 3$ s, und $\Delta \tau = 0,5$ s überwiegend im unteren Teil der Atmosphäre. Um bestimmen zu können, ob die Abhängigkeit von der Zeitskala mit steigender Verfahrensordnung (bei gleicher formaler Auflösung) zunimmt, wird mit den Ergebnissen für $\kappa = 3$ verglichen, sie sind für w in der Abbildung 5.6, für q_c in der Abbildung 5.9 und für q_r in der Abbildung 5.11 dargestellt. Hier ist der Fall $\Delta \tau = 30$ s deutlich turbulenter als für $\kappa = 2$. Um zu beurteilen, ob der Unterschied mit kleiner

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

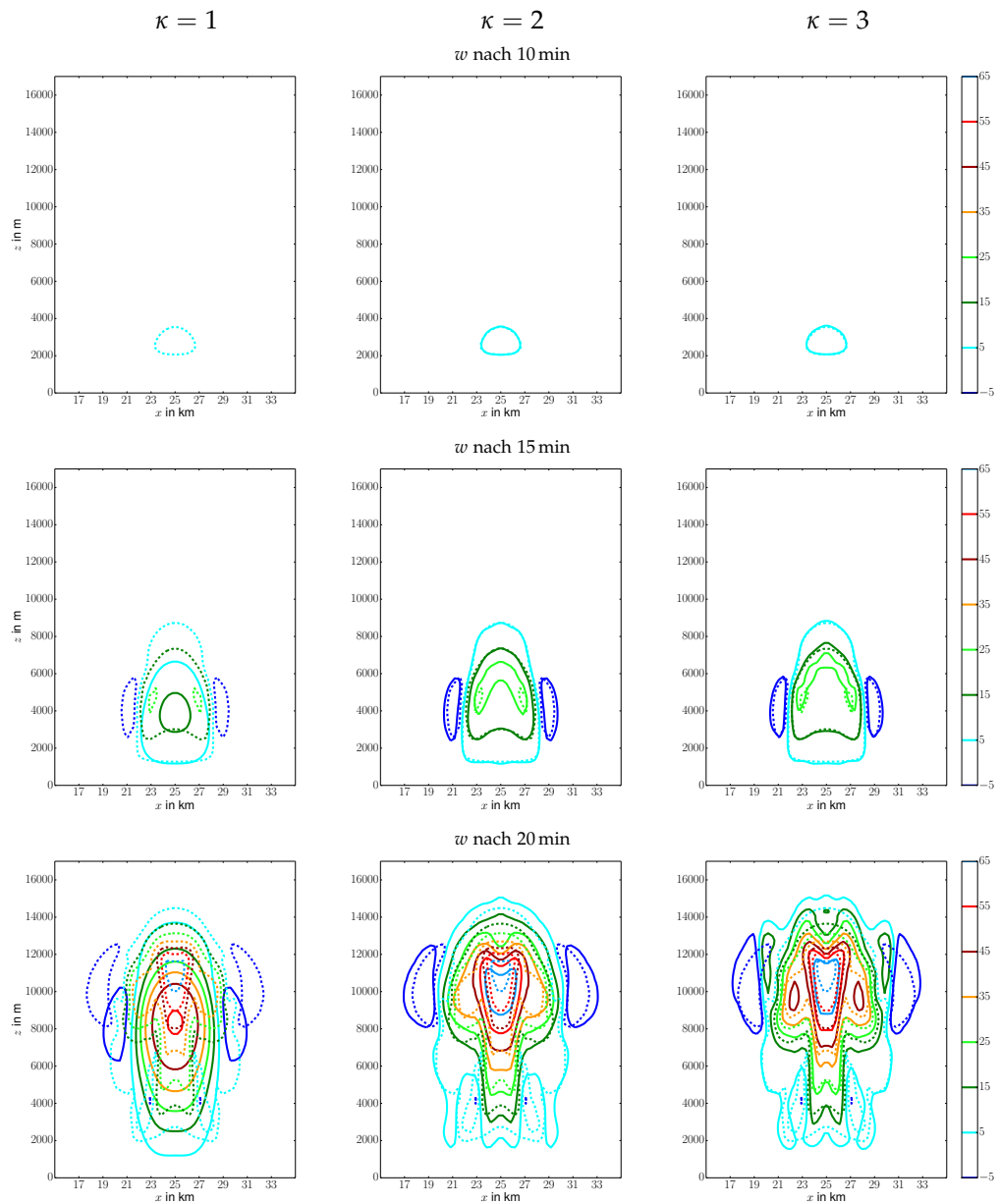


Abbildung 5.2.: Vertikaler Schnitt von w durch die Mitte der y -Richtung für den Anfang, die Mitte und das Ende der Aufstiegsphase. Von links nach rechts wird die DG-COSMO-Lösung gezeigt für $\kappa = 1$, $\kappa = 2$ und $\kappa = 3$ mit durchgezogenen Linien. Die Lösung von COSMO ist durch gestrichelte Linien dargestellt.

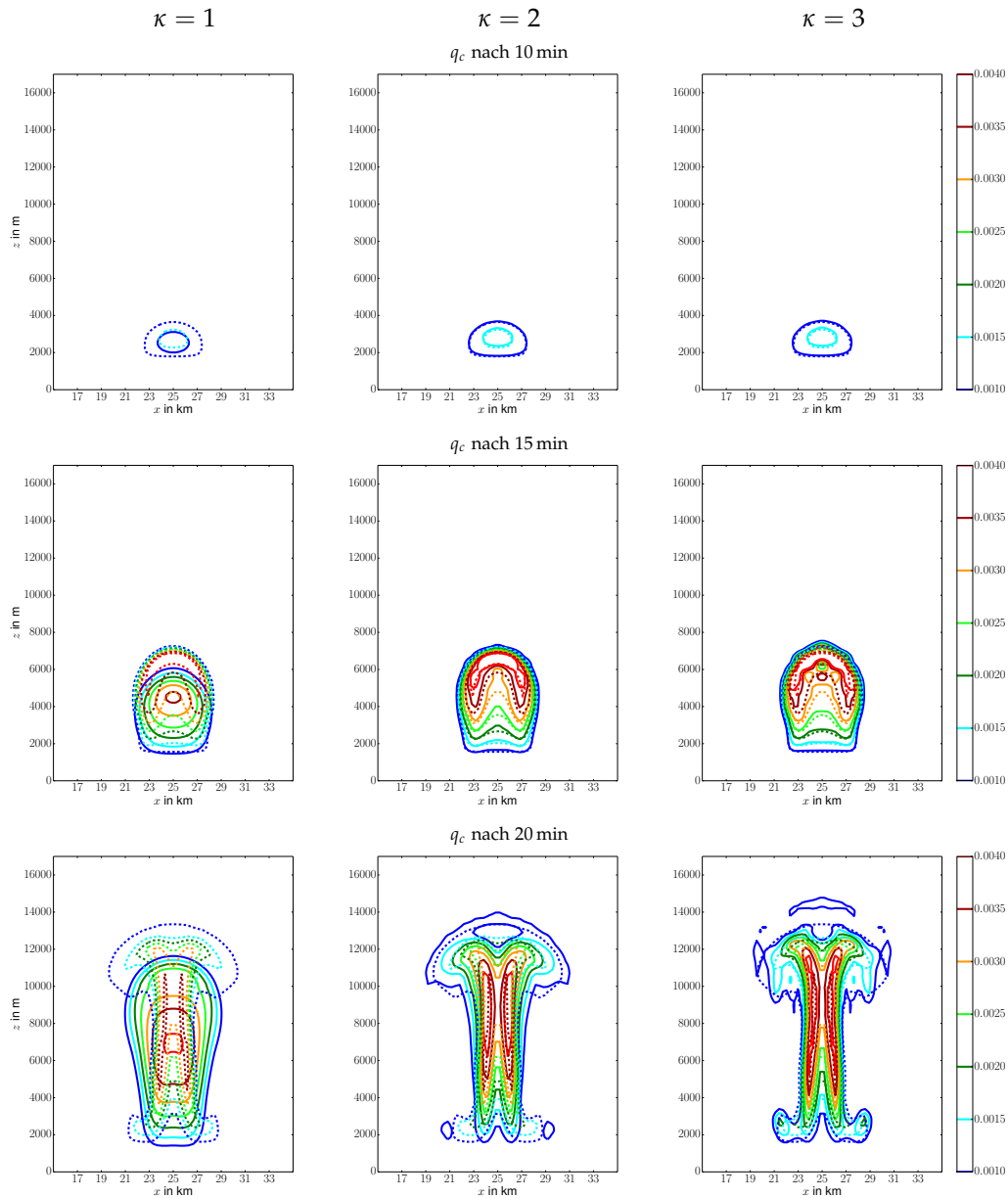


Abbildung 5.3.: Vertikaler Schnitt von q_c durch die Mitte der y -Richtung für den Anfang, die Mitte und das Ende der Aufstiegsphase. Von links nach rechts wird die DG-COSMO-Lösung gezeigt für $\kappa = 1$, $\kappa = 2$ und $\kappa = 3$ mit durchgezogenen Linien. Die Lösung von COSMO ist durch gestrichelte Linien dargestellt.

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

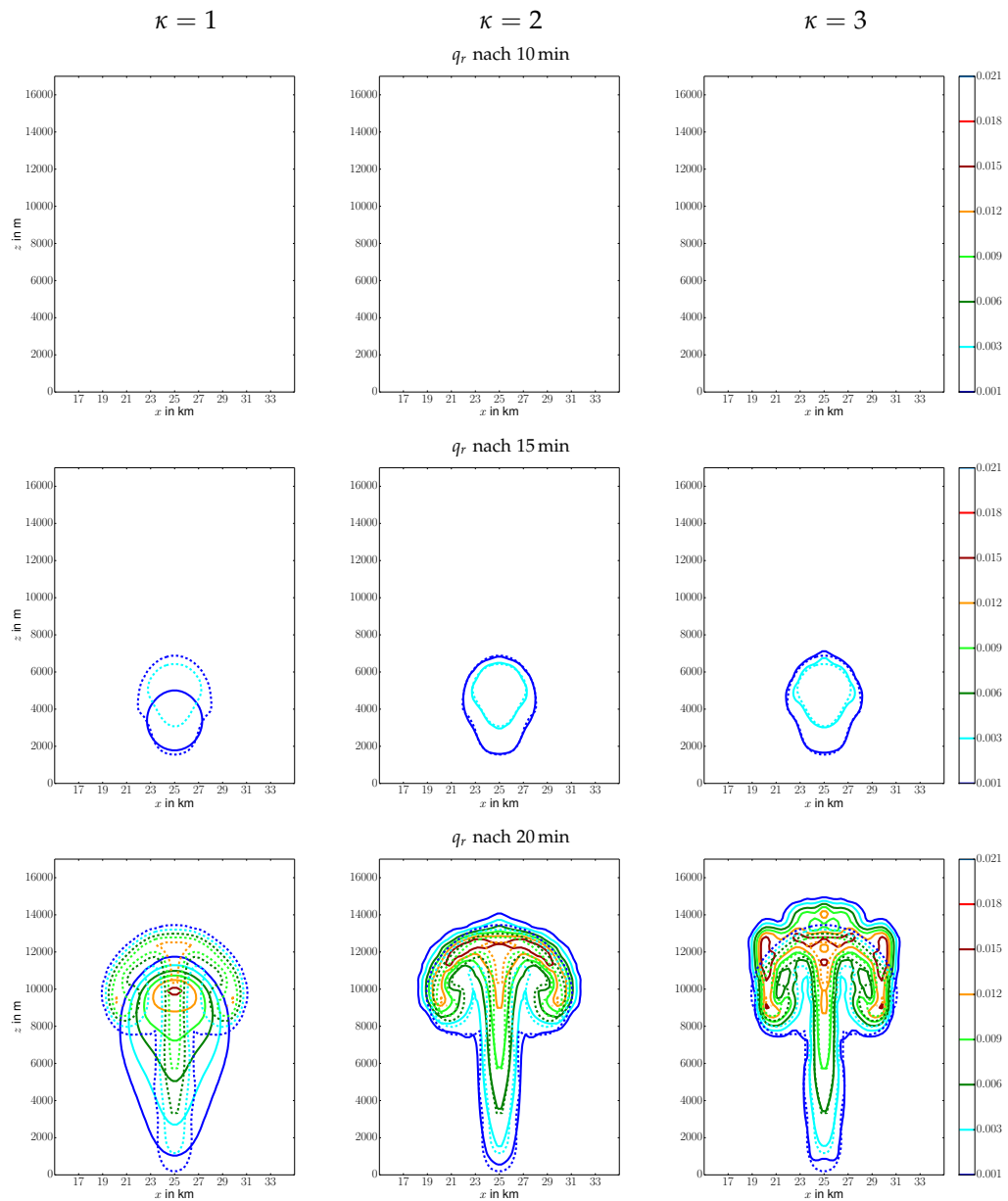


Abbildung 5.4.: Vertikaler Schnitt von q_r durch die Mitte der y -Richtung für den Anfang, die Mitte und das Ende der Aufstiegsphase. Von links nach rechts wird die DG-COSMO-Lösung gezeigt für $\kappa = 1$, $\kappa = 2$ und $\kappa = 3$ mit durchgezogenen Linien. Die Lösung von COSMO ist durch gestrichelte Linien dargestellt.

werdendem $\Delta\tau$ größer oder kleiner wird, sind die Bilder alleine nicht mehr aussagekräftig genug.

In den bereits erwähnten Abbildungen 5.5 bis 5.11 werden die Simulationsergebnisse sowohl für die Prozessaufteilung als auch für die Operatoraufteilung dargestellt. Sie ermöglichen also auch den Einfluss der Aufteilungsstrategie zu ermitteln.

Um den Einfluss der Zeitskala der Parametrisierungen im Zusammenhang mit dem Einfluss der Ordnung und der Aufteilungsstrategie betrachten zu können, wird zwischen den unterschiedlichen Konfigurationen der Abbildungen 5.5 und 5.6 die L_2 -Norm berechnet, die Ergebnisse sind in Abbildung 5.7 veranschaulicht. In Teilbild (a) ist für die Zeitskala die L_2 -Norm der Differenz von $\Delta\tau = 0,5\text{ s}$ und $\Delta\tau = 3\text{ s}$ dargestellt. In dieser Abbildung sind die verschiedenen Konfigurationen in den Eckpunkten eines Würfels mit folgender Systematik angeordnet:

- in der vorderen Ebene ist $\Delta\tau = 0,5\text{ s}$, in der hinteren $\Delta\tau = 3\text{ s}$,
- in der oberen Ebene ist $\kappa = 2$, in der unteren $\kappa = 3$,
- in der linken Ebene ist die Prozess-, in der rechten die Operatoraufteilung.

Die L_2 -Norm ist auf den Pfeilen zwischen den Konfigurationen angegeben. Beim kleineren der beiden Zeitschritte ist das Ergebnis kaum abhängig von der Wahl der Aufteilungsstrategie (vgl. blaue Pfeile in der vorderen Ebene). Der Unterschied für den größeren Zeitschritt ist signifikant größer (blaue Pfeile in der hinteren Ebene), aber immer noch eher klein. Auch der Unterschied zwischen den beiden Zeitschritten ist bei den Ordnung $\kappa = 2$ und $\kappa = 3$ als auch bei den Aufteilungsstrategien noch nicht besonders groß (grüne Pfeile zwischen der vorderen und hinteren Ebene). Den stärksten Einfluss hat die Wahl der Ordnung, die L_2 -Norm beim Wechsel der Ordnung (rote Pfeile zwischen der oberen und unteren Ebene) ist etwa doppelt so groß als die L_2 -Norm beim Wechsel der Zeitskala zwischen $\Delta\tau = 0,5\text{ s}$ und $\Delta\tau = 3\text{ s}$.

Da in DG-COSMO der Vektor der Fallgeschwindigkeit durch die Dynamik diskretisiert wird und die Dynamik mit einem anderen Zeitschritt als die Sättigungsadjustierung und das Kessler-Schema läuft, kann für diese Parametrisierungen ein größerer Zeitschritt gewählt werden als in COSMO. Wählt man etwa $\Delta\tau = 30\text{ s}$, das ist ein Zeitschritt mit dem der COSMO-Dynamikkern bei der hier verwendeten formalen Auflösung nicht mehr stabil läuft, so ergibt sich ein Ergebnis, das vom vorherigen abweicht. Dieses ist in Teilbild (b) der Abbildung 5.7 dargestellt. Der Wechsel der Aufteilungsstrategie hat jetzt für $\Delta\tau = 30\text{ s}$ (blaue Pfeile in der hinteren Ebene) einen mindestens so großen Einfluss wie der Wechsel der Ordnung (rote Pfeile in der hinteren Ebene). Aber während die Differenz des Wechsel zwischen $\kappa = 2$ und $\kappa = 3$ für die Operatoraufteilung bei $\Delta\tau = 30\text{ s}$ (roter Pfeil rechts hinten) praktisch gleich ist als der entsprechende Wechsel in Teilbild (a), so ist die Differenz des Ordnungswechsel für die Prozessaufteilung (roter Pfeil links hinten) kleiner geworden. Von den drei Variationsmöglichkeiten dominiert jetzt der Wechsel in der Zeitskala (grüne Pfeile zwischen der vorderen und hinteren Ebene) sowohl die Wahl der Aufteilungsstrategie (blauen Pfeile), also auch die Wahl der Ordnung (rote Pfeile). Dabei ist die Prozessaufteilung empfindlicher gegenüber der Wahl von $\Delta\tau$ als die Operatoraufteilung.

In Tabelle 5.1 sieht man die Laufzeit aufgliedert nach Dynamik, Parametrisierung und der Kommunikation für beide Strategien. Aufgrund der geringen Unterschiede bei den Parametrisierungen liegt aber nahe, dass die Operatoraufteilung mit $\Delta\tau = 3\text{ s}$ oder größeren Werten durch die zusätzliche Rückprojektion keinen signifikanten Laufzeitnachteil hat.

Vergleicht man die Laufzeiten der beiden Aufteilungsstrategien miteinander, so benötigt für $\kappa = 2$ und $\Delta\tau = 3\text{ s}$ die Prozessaufteilung für die Simulation 5818 s, die Operatoraufteilung benötigt nur 5122 s. Das entspricht einer Beschleunigung von 13,6%. Auch für $\kappa = 3$ ergibt sich ebenfalls eine Beschleunigung von 13,6% für Operatoraufteilung (11 213 s) im Bezug auf die Prozessaufteilung (12 744 s). Dieses Ergebnis ist auf den ersten Blick entgegen der Intuition, da die Operatoraufteilung die prognostischen Variablen, nachdem sie durch die

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

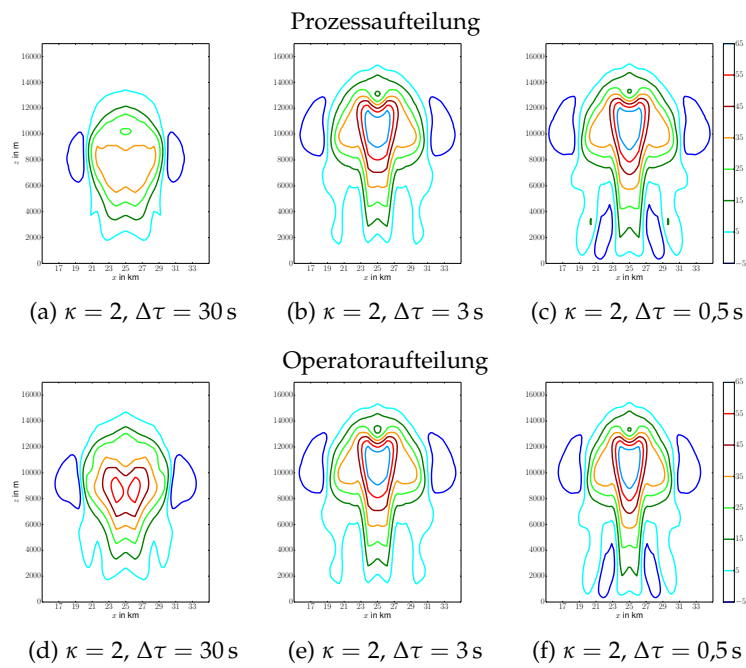


Abbildung 5.5.: Vergleich von w mit $\kappa = 2$ für mehrere Zeitskalen und Aufteilungsstrategien.

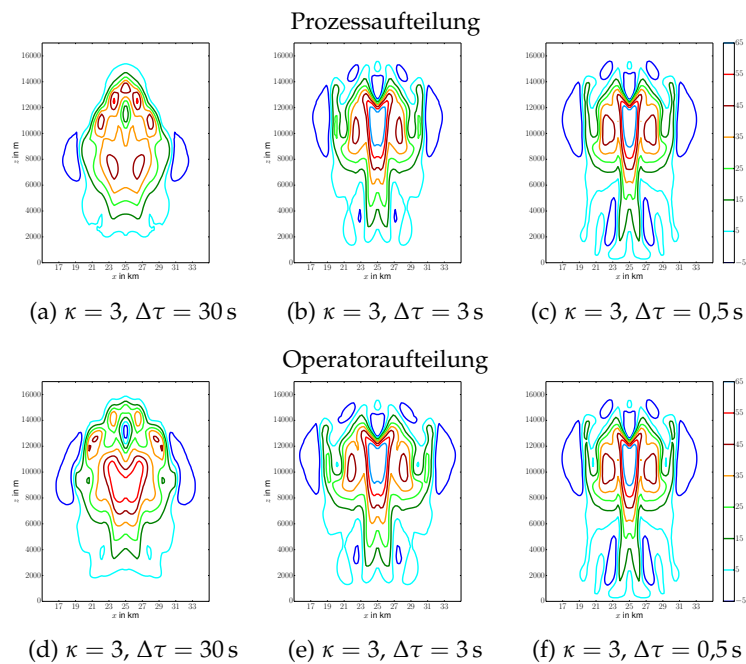
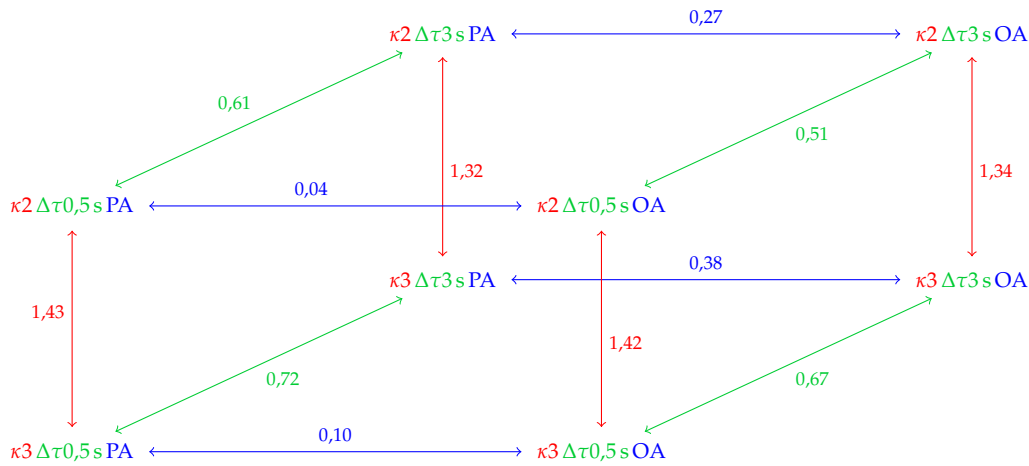
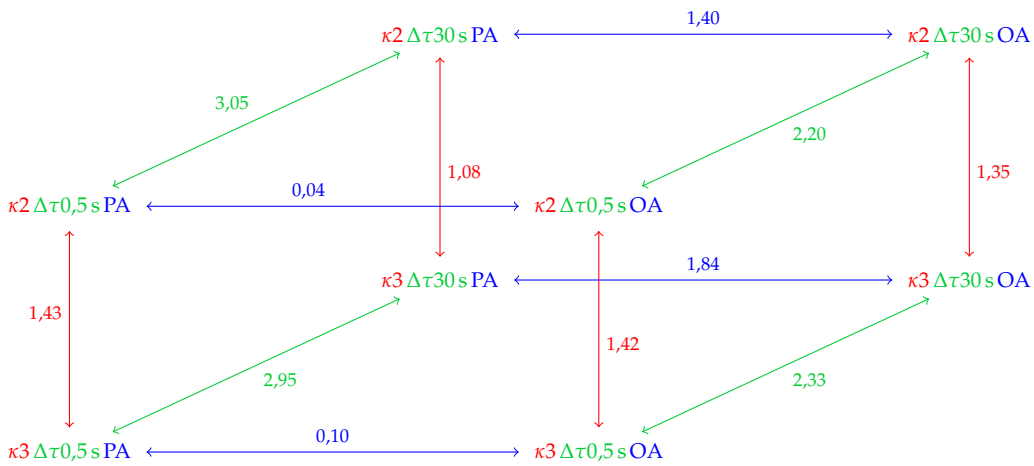


Abbildung 5.6.: Vergleich von w mit $\kappa = 3$ für mehrere Zeitskalen und Aufteilungsstrategien.



(a) L_2 -Norm der Differenz der Konfigurationen mit $\Delta\tau = 0,5\text{s}$ und $\Delta\tau = 3\text{s}$.



(b) L_2 -Norm der Differenz der Konfigurationen mit $\Delta\tau = 0,5\text{s}$ und $\Delta\tau = 30\text{s}$.

Änderung	L_2 -Norm
$\kappa = 2 \leftrightarrow \kappa = 3$	1,32 ... 1,43
$\Delta\tau = 0,5\text{s} \leftrightarrow \Delta\tau = 3\text{s}$	0,51 ... 0,72
PA \leftrightarrow OA	0,04 ... 0,38

(c) Bereich der L_2 -Norm aus Teilbild (a).

Änderung	L_2 -Norm
$\kappa = 2 \leftrightarrow \kappa = 3$	1,08 ... 1,43
$\Delta\tau = 0,5\text{s} \leftrightarrow \Delta\tau = 30\text{s}$	2,20 ... 3,05
PA \leftrightarrow OA	0,04 ... 1,84

(d) Bereich der L_2 -Norm aus Teilbild (b).

Abbildung 5.7.: Direkter Vergleich des Einflusses der Zeitskala der Parametrisierung, der Ordnung und der Aufteilungsstrategie. Die Konfiguration $\kappa m \Delta\tau n s XX$ bezeichnet die m -te Ordnung, mit $\Delta\tau = n$, $XX = PA$ steht für Prozessaufteilung und $XX = OA$ für Operatoraufteilung. Die Zahlen auf den doppelspitzigen Pfeilen sind die L_2 -Normen der Differenzen der beiden Konfigurationen an den Pfeilenden für w , dabei ist in rot der Wechsel der Ordnung, in blau der Wechsel der Aufteilungsstrategie und in grün der Wechsel der Zeitskala notiert. Teilbild (a) vergleicht $\Delta\tau = 0,5\text{s}$ mit $\Delta\tau = 3\text{s}$ und Teilbild (b) $\Delta\tau = 0,5\text{s}$ mit $\Delta\tau = 30\text{s}$. Die Tabellen (c) bzw. (d) zeigen die Intervalle in denen sich die L_2 -Norm für Bild (a) bzw. Bild (b) ändert.

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

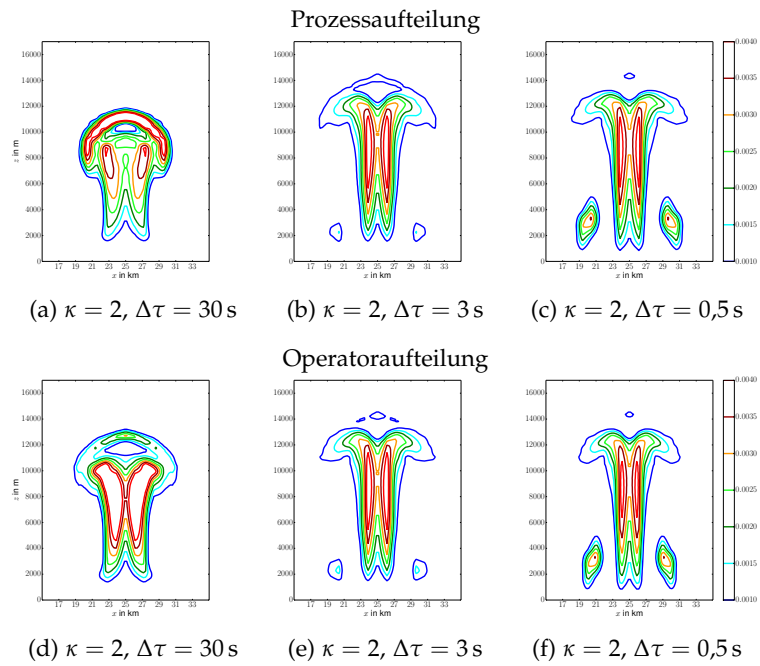


Abbildung 5.8.: Vergleich von q_c mit $\kappa = 2$ für mehrere Zeitskalen und Aufteilungsstrategien.

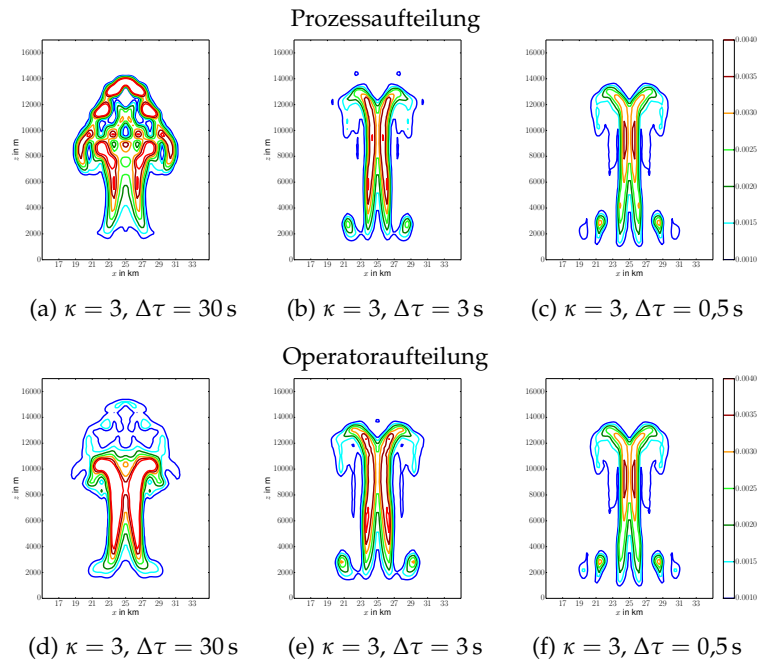


Abbildung 5.9.: Vergleich von q_c mit $\kappa = 3$ für mehrere Zeitskalen und Aufteilungsstrategien.

5.1. Feuchtkonvektion nach Weisman und Klemp 1982

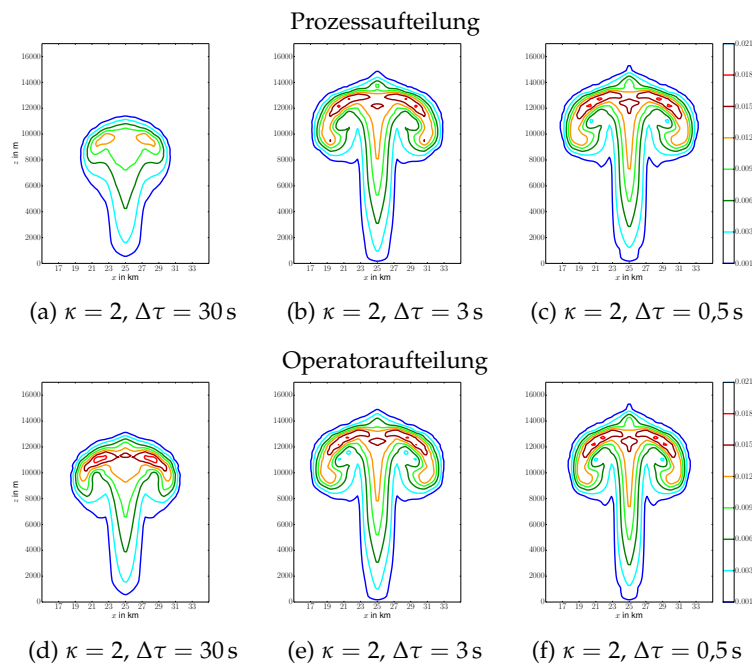


Abbildung 5.10.: Vergleich von q_r mit $\kappa = 2$ für mehrere Zeitskalen und Aufteilungsstrategien.

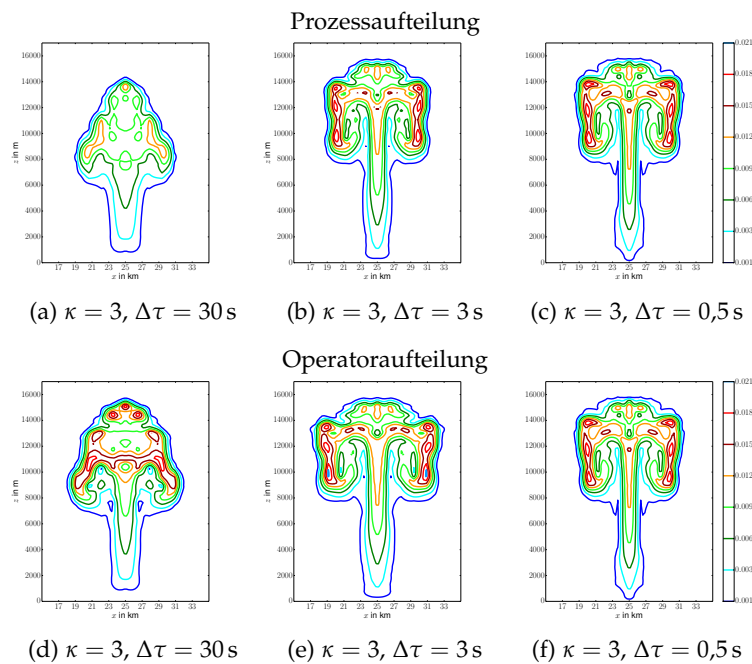


Abbildung 5.11.: Vergleich von q_r mit $\kappa = 3$ für mehrere Zeitskalen und Aufteilungsstrategien.

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

	Prozess-	Operatoraufteilung
Dynamik	5032 s	4413 s
Parametrisierung	4,58 s	4,92 s
Kommunikation	781 s	703 s
gesamte Laufzeit	5818 s	5122 s

Tabelle 5.1.: Laufzeit der Prozessaufteilung der Operatoraufteilung aufgegliedert in einzelne Implementierungsteile. Laufzeiten beziehen sich auf $\kappa = 2$ und $\sigma = 120$.

physikalischen Parametrisierungen modifiziert wurden, zurück in den Ansatzraum projizieren muss, d. h. es wird ein zusätzlicher Rechenschritt benötigt. Jedoch stellt sich heraus, dass durch die zusätzlichen Quellterme bei der Prozessaufteilung, die im dynamischen Kern benötigt werden, die Cache-Effizienz soweit zurückgeht, dass der Vorteil durch den eingesparten Rechenschritt überkompensiert wird. Ob dieser Laufzeitnachteil für den aktuellen Großrechner durch eine Änderung der Implementierung beseitigt werden kann, wird im Rahmen dieser Arbeit nicht untersucht.

Intuitiv müsste die Prozessaufteilung das bessere Ergebnis liefern, da sie eine feinere Rückkopplung zwischen Dynamik und Parametrisierungen liefert. Ob diese Intuition auch zutrifft, wird in dieser Arbeit nicht untersucht. Aber auch unter dieser Annahme ist nicht ersichtlich, ob der quantitative Unterschied zwischen den beiden Aufteilungsstrategien ausreichend groß ist, um durch eine laufzeitoptimierte Prozessaufteilung einen realen Mehrwert zu erhalten. Aus den oben gewonnenen Ergebnissen lässt sich der Schluss ziehen, dass die Operatoraufteilung der Prozessaufteilung aufgrund der höheren Effizienz, vorgezogen werden kann.

5.2. Skalierbarkeit

Auch wenn die DG-COSMO-Implementierung prototypisch ist und noch Optimierungspotential bietet, soll ihr Laufzeitverhalten untersucht und mit COSMO verglichen werden. In diesem Abschnitt wird die starke Skalierbarkeit von DG-COSMO mit COSMO verglichen. Hierfür wird der dreidimensionale Testfall aus dem vorherigen Abschnitt verwendet. Da bei der Implementierung der Viskositätsterme bisher kein besonderes Augenmerk auf die Effizienz gelegt wurde, wird hier ohne die Viskositätsterme gerechnet. In Abschnitt 6.2 wird beschrieben, wie die Viskositätsterme effizienter implementiert werden können. Hier steht die Skalierbarkeit im Vordergrund, durch den Verzicht auf die Viskositätsterme ändert sich diese nicht. Nur die Anzahl an CPUs, ab der ein realer Laufzeitvorteil, aufgrund der besseren Skalierbarkeit besteht, ist ohne Viskositätsterme kleiner. Die Skalierbarkeit soll mit und ohne Feuchte betrachtet werden. Dabei laufen beide Modelle mit dem Zeitschritt $\Delta t = 0,05$ s und $16 \cdot 10^6$ Freiheitsgraden. Für COSMO hat das Gitter $400 \times 400 \times 100$ Elemente, für DG-COSMO $100 \times 100 \times 25$ mit $\kappa = 3$ und \mathcal{B}_{vt} als Basis. Für die Zeitintegration wird ein dreistufiges Runge-Kutta-Verfahren genutzt. Für COSMO ist dies das Wicker-Skamarock-Verfahren [6]. Da COSMO ein split-explizites Verfahren verwendet (vgl. Abschnitt 1.2), hat es in der Zeit nur zweite Ordnung [6]. DG-COSMO verwendet das Shu-Osher-Verfahren (vgl. $\kappa = 3$ in Tabelle 3.1) und hat in der Zeit (für die trockene Atmosphäre) dritte Ordnung. DG-COSMO hat also mindestens die gleiche formale Ordnung in der Zeit wie COSMO. Für die räumliche Diskretisierung ist die formale Ordnung von COSMO zweiter Ordnung für die schnellen Wellen und fünfter Ordnung für die langsamen Wellen. Die räumliche

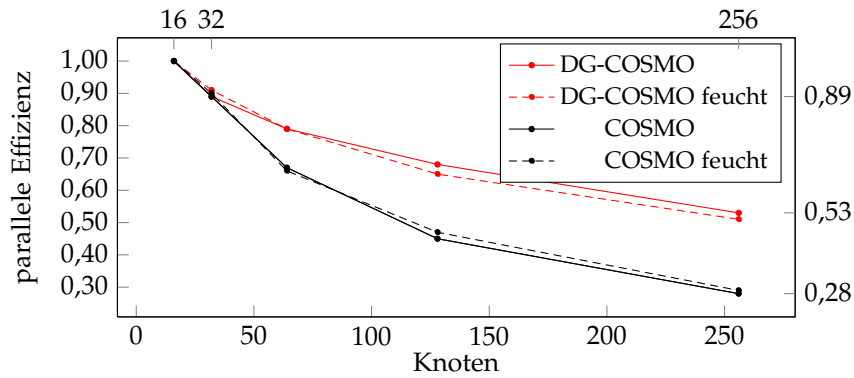


Abbildung 5.12.: parallele Effizienz von DG-COSMO (rot) und COSMO (schwarz).

Diskretisierung von DG-COSMO ist vierter Ordnung. Es wird davon ausgegangen, dass DG-COSMO mit dieser Konfiguration mindestens die gleiche Genauigkeit wie COSMO erreicht. Die Ergebnisse aus Abschnitt 4.3 und Abschnitt 4.4 legen diesen Schluss nahe. Als Referenz für die Skalierbarkeit dienen 16 Knoten der, beim DWD verwendeten, Cray XC30 (320 CPUs). Abbildung 5.12 zeigt für 16 bis 256 Knoten die Skalierbarkeit von DG-COSMO und COSMO. Gemessen wurde die Laufzeit wie in Kapitel 4 ohne Initialisierungsphase und ohne Ausgabe.

Während beide Modelle bei 32 Knoten noch etwa 89% parallele Effizienz haben, so hat COSMO bei 256 Knoten nur noch eine parallele Effizienz von 28%, DG-COSMO hingegen noch einen akzeptablen Wert von 53%. DG-COSMO skaliert also, wie erwartet, deutlich besser als COSMO. Eine mögliche Erklärung für die geringe parallele Effizienz von COSMO ist: COSMO lief beim DWD die vergangenen Jahre auf einem Großrechner mit Vektorprozessoren (NEC SX9) und wurde natürlich auch auf diesen optimiert. Die Hardware dieses Großrechners unterscheidet sich stark vom aktuellen Großrechner mit Skalarprozessoren (Cray XC30). Insbesondere haben die einzelnen CPUs der SX9 eine höhere Rechenleistung als der die CPUs der XC30, wodurch für den operationellen Betrieb für COSMO (in der Konfiguration COSMO-DE) nur 68 CPUs verwendet wurden. Skalierbarkeit auf sehr vielen CPUs (hier bis 5120 getestet) hat also bisher eine untergeordnete Rolle gespielt. Da DG-COSMO Parallelisierungsroutinen verwendet, die nur auf seine Datenstrukturen angepasst wurden (vgl. Abschnitt 3.6), in ihrer Struktur aber nicht geändert wurden, lässt sich die Skalierbarkeit mit COSMO direkt miteinander vergleichen. Auch soll hier nochmal betont werden, dass hier starke Skalierbarkeit untersucht wird. Für schwache Skalierbarkeit (vgl. Anhang A.4) werden für beide Modell eine höhere parallele Effizienz erwartet.

Die bessere Skalierbarkeit von DG-COSMO schlägt sich in der Laufzeit nieder. Die Beschleunigung von DG-COSMO bezogen auf COSMO wird dargestellt in Abbildung 5.13. Die Laufzeiten sind für die trockene Simulation und für die feuchte Simulation unabhängig voneinander auf die Laufzeit von COSMO normiert, es kann also aus Abbildung 5.13 nicht auf den Laufzeitunterschied zwischen der trockenen und der feuchten DG-COSMO-Simulation geschlossen werden. Die trockene Simulation ist etwa 2,6 Mal schneller als die feuchte, in der drei zusätzliche prognostische Gleichungen gelöst werden müssen. Der Code von DG-COSMO ist überwiegend für die trockene Dynamik optimiert. Der Autor erwartet, dass die Cache-Ausnutzung für die feuchte Simulation noch verbessert werden kann.

Für den trockenen Testfall ist auf 16 Knoten COSMO noch 1,7 Mal schneller als DG-COSMO, aber ab etwa 211 Knoten (4220 CPUs) ist DG-COSMO schneller. Bei 256 Knoten

5. Numerische Ergebnisse für die feuchte Atmosphäre und Skalierbarkeit

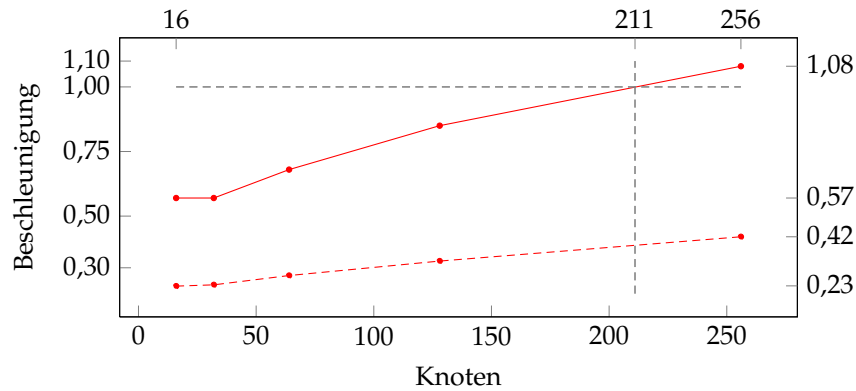


Abbildung 5.13.: Beschleunigung der trockenen DG-COSMO-Simulation (rot durchgezogen) und feuchten DG-COSMO-Simulation (rot gestrichelt) bezogen auf COSMO. Für die trockene Simulation hat DG-COSMO eine geringere Laufzeit als COSMO ab etwa 211 Knoten (4220 CPUs).

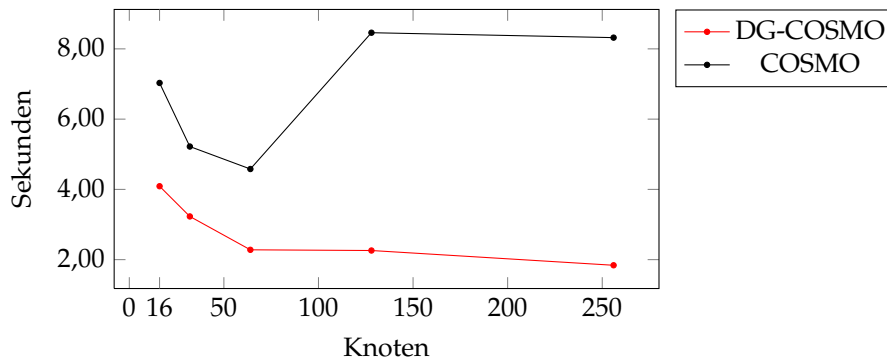


Abbildung 5.14.: durchschnittliche Kommunikationszeit der trockenen Simulation je Prozess für DG-COSMO und COSMO.

(5120 CPUs) sind es schon 8 %. Für 300 Knoten nimmt die Laufzeit von COSMO wieder zu. Der Vergleich aufgrund der fehlenden Optimierung von COSMO für diese hohe Anzahl an Knoten erscheint nicht mehr sinnvoll, daher wird nur der Bereich betrachtet, in dem beide Modelle noch schneller werden. In der Abbildung 5.13 sieht man auch, dass beim feuchten Testfall mit der aktuellen DG-COSMO-Implementierung bisher kein realer Laufzeitgewinn erzielt werden kann. Das Ergebnis der Skalierbarkeit der trockenen Simulation lässt also den Schluss zu, dass für meteorologisch relevante Simulationen DG-Verfahren in der Laufzeit durchaus konkurrenzfähig im Vergleich mit Finite-Differenzenverfahren sind, wenn die Zeitschrittbeschränkung auf ein ähnliches Niveau gehoben werden kann. Wie dies erreicht werden kann wird im Abschnitt 6.2 dieser Arbeit diskutiert.

Jetzt wird wieder die trockene Simulation betrachtet und der Grund, warum DG-COSMO besser skaliert als COSMO, wird weiter diskutiert. Der Anteil der Laufzeit, der für die Kommunikationen zwischen den einzelnen Prozessen benötigt wird, ist in Abbildung 5.14 dargestellt. Für DG-COSMO ist die Kommunikationszeit für 16 Knoten bei etwa 4 s und nimmt bis auf etwa 2 s ab. Auf diesem Wert stagniert sie dann bis 256 Knoten. Bei COSMO ergibt sich ein anderes Bild. Die Kommunikationszeit für 16 Knoten ist etwa 7 s. Sie nimmt

für 64 Knoten bis auf etwa 4,5 s ab, steigt aber auf etwa 8,5 s wieder an, und stagniert auf diesem Wert. Die Simulationen mit COSMO auf vielen Knoten benötigen also mehr Zeit für die Kommunikation als für wenig Knoten. Dies behindert eine gute Skalierbarkeit. Auch soll erwähnt werden, dass COSMO auch bei den kleinen Zeitschritten einen vertikal impliziten Löser verwendet. Dieser ist aber nur für größere Zeitschritte für eine stabile Integration notwendig. COSMO könnte für diesen Vergleich also durchaus noch optimiert werden.

Es lässt sich jedoch für DG-COSMO (und auch für COSMO) noch eine weitaus bessere Skalierbarkeit erreichen. Hierfür muss die Kommunikationsstruktur des COSMO-Codes jedoch grundlegend geändert werden. Momentan laufen die Berechnungen und die Kommunikation synchron, d. h. nach jedem Berechnungsschritt werden die Daten des Halos ausgetauscht. Während dieser Zeit laufen keine Berechnungen. Wird dies auf asynchrone Kommunikation geändert, d. h. es wird die Kommunikation mit der Berechnung zeitlich verschachtelt, so kann eine deutlich bessere Skalierbarkeit erreicht werden. Im DG-Kontext kann etwa das Volumenintegral in Gleichung (3.15) parallel zum Austauschen des Halos berechnet werden. In [54] wurde neben der auch von DG-COSMO verwendeten synchronen Kommunikation auch die beschriebene asynchrone Kommunikation mithilfe der DUNE-Bibliothek implementiert. Der Testfall in [54] ist eine dreidimensionale Version der fallenden Kälteblase nach Straka et. al. [85] (vgl. Abschnitt 4.3). Während mit synchroner Kommunikation die Skalierbarkeit vergleichbar zu DG-COSMO ausfällt, kann mit asynchroner Kommunikation die parallele Effizienz¹ bei 4096 CPUs auf 91 % erhöht werden und selbst bei 65 536 CPUs wird noch 72 % erreicht.

Für zweidimensionale Simulationen verschiebt sich der Laufzeitunterschied stark zugunsten von DG-COSMO. Wenn etwa die Schwerewellenausbreitung aus Abschnitt 4.1 mit 240 000 Freiheitsgraden und einem Zeitschritt von 0,2 s auf 8 Knoten gerechnet wird, so beträgt für DG-COSMO mit $\kappa = 3$ die Laufzeit 18,3 s für COSMO hingegen 177,5 s, das entspricht einer Beschleunigung von DG-COSMO gegenüber COSMO um den Faktor 9,7. Hierbei ist aber zu berücksichtigen, dass COSMO für dreidimensionale Simulationen optimiert ist.

Im operationellen Betrieb hat COSMO für das COSMO-DE-Gebiet $461 \times 421 \times 50$ Elemente, mit der formalen Auflösung von etwa 2,8 km. Dies sind 9 704 050, also etwa 61 % der Freiheitsgrade der vorherigen dreidimensionalen Skalierbarkeitsanalyse. Bei gleicher Anzahl der vertikalen Flächen, könnte man mit den $16 \cdot 10^6$ Elemente etwa eine formale Auflösung von 2,1 km realisieren. D. h. wenn COSMO-DE das nächste mal auf ein feineres Gitter umgestellt wird, zeigt obiger Test, dass DG-COSMO konkurrenzfähig ist, unter der Annahme, dass DG-COSMO mit einem ähnlichen Zeitschritt integriert werden kann (vgl. Abschnitt 6.2).

¹In [54] wurden 512 CPUs als Referenz für die Skalierbarkeitsanalyse verwendet.

6. Zusammenfassung und Ausblick

In diesem Kapitel werden die wesentlichen Ergebnisse dieser Arbeit zusammengefasst und es wird ein Ausblick auf mögliche Verbesserungen und Fortsetzungen gegeben.

6.1. Zusammenfassung

In dieser Arbeit wurde ein neuer Dynamikkern „DG-COSMO“ in das COSMO-Modell integriert, der eine hohe Verfahrensordnung ermöglicht und somit eine potentiell höhere Genauigkeit liefert als der bestehende, auf Finite-Differenzenverfahren basierende, Dynamikkern. Für den neuen Dynamikkern wurden diskontinuierliche Galerkin-Verfahren als Grundlage genommen. Die physikalischen Erhaltungssätze für die prognostischen Variablen werden daher auch lokal erfüllt. Durch die gleiche Kommunikationsinfrastruktur, Gitterstruktur und Behandlung der Orographie wurde ein Löser erstellt, der gute Vergleichbarkeit mit dem Finite-Differenzen-Dynamikkern in COSMO ermöglicht.

Die hier vorgestellte DG-Variante hat eine diagonale Massenmatrix und verwendet exakte Quadraturformeln. Die modale Basis ist hierarchisch, daher kann der Dynamikkern künftig einfach um p -Adaptivität erweitert werden. Auch ist für die Anwendung eines Filters die modale Form vorteilhaft [47]. In dieser Arbeit wurde jedoch keine Notwendigkeit für einen Filter gesehen. Es wurde die schwache Form verwendet, da sie die lokale Erhaltungseigenschaften direkt garantiert (vgl. [42]) während bei der starken Form noch eine zusätzliche Transformation angewendet werden muss (vgl. [39]) und kein Effizienzvorteil der starken Form erkennbar ist [42].

Anhand von trockenen und feuchten idealisierten Testfällen sind im Wesentlichen folgende Ergebnisse entstanden:

- DG-COSMO konvergiert von hoher Ordnung. Dies wurde bis zur vierten Ordnung auch für nicht-lineare Fälle gezeigt. Gegenüber dem Finite-Differenzen-Dynamikkern von COSMO ist dies eine deutliche Verbesserung. Der FD-Kern hat eine Konvergenzordnung von etwa eins. [9].
- Der Rusanov-Fluss verursacht eine zusätzliche numerische Störung der hydrostatischen Balance, die in der Größe über die Ordnung kontrolliert werden kann.
- Bei gleicher Anzahl an Freiheitsgraden erhält man mit der vollen Tensorproduktbasis, für einen Polynomgrad $\kappa \leq 4$ ein effizienteres Verfahren als mit der Minimalbasis.
- Die Masse und das Produkt aus Dichte und potentieller Temperatur wird nach einer Einschwingphase (fast) bis auf Maschinengenauigkeit erhalten. Der horizontal Impuls wird ebenfalls sehr gut erhalten (Größenordnung 10^{-11}), für den vertikalen Impuls gibt es aufgrund des Quellterms und der Randbedingungen keine exakte Erhaltung; er oszilliert um den Mittelwert. Jedoch ist auch für den Impuls kein Trend erkennbar.
- DG-COSMO läuft im Vergleich zu COSMO auch bei deutlich steileren Bergen noch stabil, obwohl es die Orographie genauso wie COSMO durch geländefolgende Koordinaten modelliert.

- Der Unterschied zwischen der Operatoraufteilung und der Prozessaufteilung nimmt wie erwartet mit größer werdendem Parametrisierungszeitschritt zu, jedoch dominiert bei kleinem Zeitschritt die Verfahrensordnung das Ergebnis. Auch ist das Ergebnis sensitiver gegenüber der Wahl des Zeitschritts als gegenüber der Wahl der Aufteilungsstrategie. Der Einfluss des Dynamikkerns tritt also bei großem Parametrisierungszeitschritt in den Hintergrund.
- Die aktuelle Implementierung von DG-COSMO hat eine deutlich bessere parallele Skalierbarkeit als COSMO. Bereits auf dem beim DWD momentan verfügbaren Großrechner der Typs Cray XC30 lässt sich zeigen, dass für die trockene Atmosphäre DG-COSMO (bei gleichem Zeitschritt) eine geringere Laufzeit als COSMO hat, wenn viele (> 4220) CPUs verwendet werden. An dieser Stelle soll aber nochmal betont werden, dass hier nicht nur die numerischen Verfahren, sondern vor allem auch die Implementierungen gegeneinander verglichen werden. Eventuelle Optimierungen der Implementierungen können dieses Ergebnis noch verändern.

6.2. Ausblick

Beim in dieser Arbeit entwickelten Dynamikkern wurde darauf geachtet, dass das Verfahren einfach bleibt, da noch nicht alle Eigenschaften, die für die operationelle Vorhersage notwendig sind, vorhanden sind. Sobald dies nachgeholt ist, kann abgeschätzt werden, welche effizienteren oder genaueren Verfahren direkt implementiert werden können, und wo effizientere Verfahren noch angepasst oder neu entwickelt werden müssen. Im folgenden sollen ein paar mögliche Verbesserungen diskutiert werden.

Zuerst kann LDG in der *primalen Form* implementiert werden [67]. In dieser Form kann auf die Berechnung der Hilfsvariablen V in Gleichung (3.17) verzichtet werden. Weiter können effizientere Verfahren wie etwa CDG [67] oder CDG2 [20] verwendet werden, die auch einen kompakten Differenzenstern besitzen. Hierbei muss aber geprüft werden, ob auf CDG(2) auch das Turbulenzmodell übertragen werden kann.

Die Quadraturformeln können weiter verbessert werden, indem die Anzahl und die Lage der LGL-Punkte weiter optimiert wird. Ein erster Schritt ist die mehrdimensionalen LGL-Punkte nicht über ein Tensorprodukt zu definieren, sondern über eine Rekursion, die den Punktabstand ebenfalls vom „LGL-Typ“ wählt [39]. Als zweiter Schritt kann die Lage der Quadraturpunkte über eine elektrostatische Optimierung [48] weiter verbessert werden. In Verbindung hiermit kann auch eine nodale Quadratur für das modale DG-Verfahren verwendet werden (vgl. [39]).

Ein quadraturfreies DG-Verfahren [61, 2] kann die Effizienz weiter steigern. Hierbei werden die Flüsse auch als Linearkombinationen der Basispolynome interpretiert. Dadurch können die Quadraturformeln auf Produkte der Basisfunktionen beschränkt werden, wodurch sie im Voraus berechnet werden können. Während der räumlichen Integration ist nur noch eine Matrix-Vektor-Multiplikation nötig. Dieses Verfahren benötigt aber Filter, da hierbei Aliasing-Effekte auftreten können, die zu nicht-linearen Instabilitäten führen [47].

Die Implementierung kann auch noch für einen speziellen Polynomgrad optimiert werden, dadurch verliert der DG-Löser jedoch einen Großteil seiner Flexibilität. Ein Mittelweg ist hier die automatische Codegenerierung für den Polynomgrad der für die gewünschte Rechnung verwendet werden soll. Dies wird von Klöfkorn in [54] für die DUNE-Bibliothek umgesetzt. Auf einem kartesischen Gitter kann hierdurch für $\kappa = 4$ der Code um knapp 50% beschleunigt werden, auf einem unstrukturierten Gitter sogar um mehr als 50%. Diese Beschleunigung wird dadurch erreicht, dass die tatsächlich erreichte Anzahl an Fließkommaoperationen (FLOPS) näher an die theoretisch erreichbare (FLOPS(peak)) rückt. So beträgt das Verhältnis

FLOPS zu FLOPS(peak) ohne Codegenerierung etwa 16 %, mit Codegenerierung etwa 24 % [54, Tabelle 4.3]. Es kann allerdings nicht erwartet werden, dass für DG-COSMO ähnliche hohe Beschleunigungen durch automatische Codegenerierung erreicht werden.

Wie schon in Abschnitt 5.2 erwähnt ist durch asynchrone Kommunikation besonders bei einer hohen CPU-Anzahl noch eine Effizienzsteigerung möglich. Auch durch eine weitere Reduktion der Daten die über den Halo getauscht werden, kann die Skalierbarkeit noch weiter gesteigert werden. Momentan werden alle Moden im Halo zwischen den benachbarten CPUs getauscht. Dies kann auf die Flüsse an den Randpunkten der Elemente reduziert werden.

Eine der größten Beschleunigungen wird wohl durch die Verwendung eines HEVI- (horizontal explizit, vertikal implizit) Verfahrens erreicht. Hierdurch wird der Zeitschritt nicht mehr durch die geringe Gittermaschenweite in vertikaler Richtung beschränkt, sondern nur noch durch die horizontale Gittermaschenweite. Auf der anderen Seite bleibt ein HEVI-Verfahren immer noch gut parallelisierbar, da bei einer horizontalen Gebietszerlegung kein Gleichungssystem gelöst werden muss, das über eine vertikale Säule hinausgeht. In [18] wurde mit einem semi-impliziten DG-Löser für eine Bergüberströmung vergleichbar zu Abschnitt 4.4 auf einem vertikal äquidistanten Gitter eine Beschleunigung von etwa 2,4 erreicht im Bezug auf den voll-expliziten Löser. Es wird erwartet, dass auf einem vertikal gestreckten Gitter eine größere Beschleunigung erreicht werden kann, da in diesem Fall die Zeitschrittbegrenzung in der Vertikalen größer ist (dies wird durch [51] gestützt, wo durch ein semi-implizites DG-Verfahren für ein anisotropes Gitter eine Beschleunigung von bis zu 128 erreicht wurde, allerdings nur im eindimensionalen). Da mit einem HEVI-Verfahren die größte Einschränkung des Zeitschritts durch das Gitter beseitigt wird und aufgrund der guten Skalierbarkeit wird erwartet, dass mit ihnen ein noch effizienteres Verfahren auf massiv parallelen Systemen realisiert werden kann als mit den semi-impliziten Verfahren. Die Entwicklung eines HEVI-Verfahrens für DG steht jedoch noch aus.

Da sich mit DG-Verfahren recht einfach Verfahren entwickeln lassen, die h - und p -adaptiv sind, kann DG-COSMO um diese Eigenschaften erweitert werden (vgl. Referenzen in der Einleitung). Dies ermöglicht feinere Strukturen lokal aufzulösen, bei relativ geringem Rechenaufwand. Das große Problem hierbei ist ein Kriterium zu finden anhand dessen man das Gitter verfeinert oder vergrößert bzw. die Ordnung erhöht oder erniedrigt. Um ein solches Kriterium auf reale Anwendbarkeit zu testen, muss aber zuerst ein geeigneter Dynamikkern existieren. DG-COSMO könnte also auch, mit nicht all zu hohem Aufwand, künftig hierzu einen Beitrag leisten. Als ersten Schritt für ein real anwendbares Kriterium könnten bestimmte meteorologische Objekte besser aufgelöst werden. Bei Globalmodellen wären etwa tropische Wirbelstürme (vgl. hierzu das OMEGA-Modell [3]) oder der Tagesgang, um Konvektion besser aufzulösen, denkbar.

Neben der Effizienzsteigerung und der offensichtlichen Erweiterung um die fehlenden physikalischen Parametrisierungen ist für den implementierten Teil noch folgendes zu berücksichtigen: Momentan ist das DG-Verfahren für die feuchte Atmosphäre nicht positiv definit. Es kann negative Wassermasse entstehen, die bisher im Kessler-Schema und der Sättigungsadjustierung einfach auf Null abgeschnitten wird. Hierdurch wird zusätzliche Masse erzeugt. Um ein durchgängig positiv definites Verfahren zu erhalten ist ein angepasstes DG-Verfahren nötig, etwa durch einen Flussbegrenzer oder eine positiv definite Korrektur der Polynomkoeffizienten [96]. Für die Prozessaufteilung muss die Wirkung des Quellterms begrenzt werden. Ist dieser negativ, so darf er nur dann die Phasenumwandlung modellieren, wenn auch noch Wasserdampf bzw. Flüssigwasser vorhanden ist. Genauso können bei der Operatoraufteilung durch die Rückprojektion, der durch die Parametrisierung aufdatierten Werte in den Ansatz- und Testraum, Unterschwinger entstehen. Hier müsste eine positiv definite Projektion entwickelt werden, oder die Polynomkoeffizienten müssten nach der

6. Zusammenfassung und Ausblick

Projektion korrigiert werden.

In Abschnitt 5.2 wurde gezeigt das bei gleichem Zeitschritt DG-COSMO ab etwa 4220 CPUs eine geringere Laufzeit als COSMO hat. Wird in DG-COSMO vertikal implizit gelöst, genauso wie in COSMO, so hängt der Zeitschritt nur noch von der horizontalen Gittermaschenweite ab. Wird für $\kappa = 3$ in der Horizontalen die formale Auflösung gleich zur formalen Auflösung von COSMO-DE (2,78 km) gewählt, das entspricht einer DG-Gittermaschenweite von 7,6 km für die Minimalbasis und 11,2 km für das volle Tensorprodukt, so kann der Zeitschritt auf etwa 3,2 s für die Minimalbasis aufgeweitet werden, beim vollen Tensorprodukt können etwa 3,6 s verwendet werden, dieser ist nur noch etwa 14 % kleiner als der schnelle COSMO-Zeitschritt $4\frac{1}{6}$ s. Es lässt sich also erwarten, dass DG-COSMO mit einem vertikal impliziten Löser (und eventuell einer Aufteilung der Zeitintegration in langsame und schnelle Wellen, wie sie von COSMO verwendet wird (vgl. Einleitung)) und aufgrund der besseren Skalierbarkeit ein konkurrenzfähiges Vorhersagesystem, mit zusätzlichen physikalischen Eigenschaften, ist.

A. Mathematischer Anhang

A.1. Begriffe und Notation

Experimentelle Konvergenzordnung Oft mit EOC (engl. „experimental order of convergence“) abgekürzt. Die experimentelle Konvergenzordnung wird durch eine Serie von Simulationen mit verschiedenen Gittermaschenweiten ermittelt. Sind zwei Gittermaschenweiten h_1 und h_2 mit $h_1 > h_2$ gegeben und die zugehörigen Fehler (bezogen auf eine Referenzlösung) F_1 und F_2 gemessen, so kann die EOC berechnet werden durch:

$$EOC_{h_1, h_2} := \frac{\ln \frac{F_1}{F_2}}{\ln \frac{h_1}{h_2}}.$$

Für mehrdimensionale Gitter bezieht sich h auf die Diagonale durch die Elemente.

Freiheitsgrade Mit *Freiheitsgrade* werden die Unbekannten bezeichnet, die je prognostischer Größe durch lösen der Gleichungssysteme zu bestimmen sind. Bei Finiten-Differenzenverfahren sind das die Werte an den Gitterpunkten, bei Ansatzverfahren, zu denen die DG-Verfahren gehören, sind es die Koeffizienten der Ansatzfunktionen, es gibt folglich bei DG-Verfahren für $\kappa > 0$ mehrere Freiheitsgrade je Gitterelement.

Halo Wird ein Berechnungsgebiet in mehrere Teilstücke zerlegt, die von verschiedenen Prozessen bearbeitet werden, so nennt man den Bereich, der Teilstücke deren Daten während der Berechnung zwischen den Prozessen ausgetauscht werden müssen, *Halo*.

Raum $L^2(\Omega, \mathbb{R})$ der quadratintegralen Funktionen Menge alle Funktionen mit Definitionsbereich Ω und Bildbereich \mathbb{R} bei denen das Integral über das punktweise Quadrat der Funktion endlich ist, also:

$$L^2(\Omega, \mathbb{R}) := \left\{ \psi \mid \psi : \Omega \rightarrow \mathbb{R}, \int_{\Omega} \psi^2 d\Omega < \infty \right\}.$$

A.2. Allgemeine Koordinaten

Sind zwei Koordinatensysteme mit den Grundvektoren $\tilde{g}_1, \tilde{g}_2, \tilde{g}_3$ und g_1, g_2, g_3 auf derselben Untermannigfaltigkeit gegeben, so transformieren sich die Koordinaten eines Vektors $\mathbf{u} = u^k g_k = \tilde{u}^i \tilde{g}_i$ (unter der Annahme der Einstein'schen Summenkonvention) durch

$$\tilde{u}^i = \frac{\partial \tilde{x}^i}{\partial x^k} u^k \quad \text{bzw.} \quad u^i = \frac{\partial x^k}{\partial \tilde{x}^i} \tilde{u}^k.$$

Der *metrische Fundamentaltensor* ist definiert durch

$$g_{ik} = g_i \cdot g_k$$

und transformiert sich wie folgt:

$$\tilde{g}_{ik} = \frac{\partial \tilde{x}^i}{\partial x^n} \frac{\partial \tilde{x}^k}{\partial x^m} g_{nm}.$$

Die dem metrischen Fundamentaltensor zugeordnete *Gram'sche Determinante* $\sqrt{G} := \sqrt{\det g_{ik}}$ beschreibt die Volumenverzerrung bei der Transformation von kartesischen Koordinaten auf die Untermannigfaltigkeit (vgl. [56]). Wird die Divergenz eines Vektor- u^i und eines Tensorfelds T^{ik} zweiter Stufe in allgemeinen Koordinaten formuliert so lauten sie

$$\begin{aligned}(\nabla \cdot \mathbf{u}^i)_i &= \frac{\partial u^i}{\partial x_i} + \Gamma_{ki}^i u^k, \\(\nabla \cdot \mathbf{T}^{ik})_i &= \frac{\partial T^{ik}}{\partial x_i} + \Gamma_{im}^i T^{mk} + \Gamma_{im}^k T^{im},\end{aligned}$$

mit den *Christoffel-Symbolen* zweiter Art

$$\Gamma_{jk}^i = \frac{1}{2} g_{ik}^{-1} \left(\frac{\partial g_{ik}}{\partial x^i} + \frac{\partial g_{jk}}{\partial x^j} - \frac{\partial g_{ij}}{\partial x^k} \right).$$

Die Christoffel-Symbole beschreiben die Änderung des Vektor- oder Tensorfelds, die nicht durch die Strömung selbst, sondern aufgrund der Änderung der Mannigfaltigkeit besteht. Einzelheiten kann man in ein [50] oder einem Buch über Tensoranalysis (etwa [60, Abschnitt 1.2]) nachlesen.

A.3. Legendre-Polynome

Die Legendre-Polynome sind Polynome, die an vielen Stellen in der Mathematik auftauchen, und viele interessante Eigenschaften besitzen. Hier sind nur die für diese Arbeit notwendigen beschrieben. Eine konzeptionell besonders einfache Definition ist folgende: Wendet man auf die Polynome $1, x, x^2, x^3, \dots$ über dem Intervall $[-1,1]$ das *Gram-Schmidt'sche Orthonormalisierungsverfahren* [21] bezüglich des Skalarprodukts

$$\langle f, g \rangle = \int_{-1}^1 f(x)g(x) dx$$

an, und normiert die Polynome so, dass sie an der Stelle $x = 1$ den Funktionswert 1 haben, so sind die resultierenden Polynome die *Legendre-Polynome*. Sie werden mit $P_1(x), P_2(x), P_3(x), \dots$ bezeichnet. Die normierten Legendre-Polynome werden mit $\tilde{P}_k(x)$ bezeichnet. Sie sind bezüglich $\|P_k(x)\| := \sqrt{\langle P_k(x), P_k(x) \rangle}$ normiert und lassen sich berechnen durch:

$$\tilde{P}_k(x) = \frac{P_k(x)}{\|P_k\|} = \frac{2}{2k+1} P_k(x).$$

Die normierten Legendre-Polynome bilden also nach Konstruktion ein Orthonormalsystem, es gilt insbesondere

$$\begin{aligned}\|\tilde{P}_k\| &= 1, \\ \langle \tilde{P}_k, \tilde{P}_l \rangle &= \int_{-1}^1 \tilde{P}_k(x) \tilde{P}_l(x) dx = \delta_{kl}.\end{aligned}$$

Das Polynom $P_k(x)$ (und damit auch $\tilde{P}_k(x)$) hat n Nullstellen auf $[-1,1]$. Für die Implementierung ist folgende rekursive Formel [23] für die Legendre-Polynome und deren Ableitung

hilfreich:

$$P_k^{(l)}(x) = \frac{(2k-1)x P_{k-1}^{(l)}(x) - (k-1+l) P_{k-2}^{(l)}(x)}{(k-l)},$$

mit geeigneten Anfangsbedingungen, sowie die Eigenschaft:

$$P_k(-x) = (-1)^k P_k(x).$$

Diese und weitere Eigenschaften und andere Zugänge zu den Legendre-Polynomen können in Standardwerken wie [21, 70, 88] nachgelesen werden.

A.4. Skalierbarkeit, Effizienz

In dieser Arbeit wird eine einzelne Arithmetikeinheit eines Rechners *CPU* (engl. „Central Processing Unit“) genannt. Im folgenden werden grundlegende Begriffe, für die Bewertung wie gut ein Algorithmus parallel abgearbeitet wird und wie die Laufzeiten zweier verschiedener Algorithmen miteinander verglichen werden können, eingeführt.

Ist $L_P(n)$ die Laufzeit einer Implementierung eines Algorithmuses P (kurz Programm), die auf n CPUs ausgeführt wird, dann ist die *Beschleunigung*¹ $S(P, R, n, r)$ eines Programms P auf n CPUs im Vergleich zu einem Programm R auf r CPUs definiert durch:

$$S(P, R, n, r) := \frac{L_P(n)}{L_R(r)}.$$

Mit dieser Definition kann man zwei unterschiedliche Programme miteinander vergleichen, oder ein Programm mit unterschiedlicher Anzahl an CPUs. Für $S(P, P, n \cdot r, r)$, also der Beschleunigung eines Programms auf der n -fachen Anzahl an CPUs, ist die optimal zu erwartende Beschleunigung n . In der Realität wird dies oft nicht erreicht, da es Anteile des Programms gibt, die sich nicht parallelisieren lassen. Es gibt jedoch auch Fälle, wo die Beschleunigung größer als n ist. Dies geschieht zum Beispiel dann, wenn durch die Aufteilung auf mehrere CPUs alle Daten die von einer CPU bearbeitet werden in ihren Cache passen, für den Referenzfall die CPUs jedoch Daten aus dem Hauptspeicher nachladen müssen.

Die *parallele Effizienz*² $E_{P,r}$ eines Programms P misst das Verhältnis der genutzten Rechenzeit zur zurverfügungstehenden Rechenzeit und ist definiert durch:

$$E_{P,r}(n) := \frac{S(P, P, n \cdot r, r)}{n},$$

für einen Referenzwert r . Das zu erwartende Optimum ist $E_{P,r}(n) = 1,0$. In modernen Rechnern sind mehrere CPUs auf einem *Prozessor* verbaut. In Großrechnern sind ein oder mehrere Prozessoren zu *Knoten* zusammengefasst. Die Bussysteme oder das Netzwerk zwischen den verschiedenen Stufen haben unterschiedliche Bandbreiten und Latenzzeiten. Die Überschreitung einer solchen Stufe macht sich im Laufzeitverhalten eines parallelen Programms deutlich bemerkbar. Um in erster Linie die Skalierbarkeit des Algorithmuses und nicht der Hardware zu testen wird der Referenzwert r so gewählt, dass er alle CPUs eines Knoten umfasst, bei der Cray XC30, die beim Deutschen Wetterdienst verfügbar ist, ist somit $r = 20$. Hierdurch wird immer für die Skalierbarkeit auch die Kommunikation auf der größten Stufe berücksichtigt.

¹englisch: speedup. Auch im Deutschen werden hier oft die englischen Begriffe verwendet, deshalb sollen sie hier kurz erwähnt werden.

²englisch: parallel efficiency

A. Mathematischer Anhang

Mit *Skalierbarkeit* bezeichnet man, wie sich die Effizienz eines Programms mit steigender CPU-Anzahl entwickelt. Wünschenswert ist, dass die Effizienz auch bei sehr vielen CPUs noch nahe an 1 ist. Bei der *starken Skalierbarkeit* wird die Anzahl der Freiheitsgrade konstant gehalten und die CPU-Anzahl erhöht, d. h. die auf einer CPU berechneten Teilgebiete werden immer kleiner. Darüber hinaus gibt es noch die *schwache Skalierbarkeit*. Bei ihr wird die Anzahl der Freiheitsgrade je CPU konstant gehalten, d. h. mit steigender CPU-Anzahl wächst auch die Problemstellung. Die schwache Skalierbarkeit wird in der vorliegenden Arbeit nicht untersucht.

Literatur

- [1] Dimitrios Alevras. "Simulating Tsunamis in the Indian Ocean with real Bathymetry by using a High-Order Triangular Discontinuous Galerkin Oceanic Shallow Water Model". Diss. Naval Postgraduate School, März 2009.
- [2] Harold L. Atkins und Chi-Wang Shu. "Quadrature-Free Implementation Of The Discontinuous Galerkin Method For Hyperbolic Equations". In: *AIAA Journal* 36 (1996), S. 775–782.
- [3] David P. Bacoan, Nash'at N. Ahmad, Zafer Boybeyi, Thomas J. Dunn, Mary S. Hall, Pius C. S. Lee, R. Ananthakrishna Sarma, Mark D. Turner, Kenneth T. Waight III, Steve H. Young und John W. Zack. "A Dynamically Adapting Weather and Dispersion Model: The Operational Multiscale Environment Model with Grid Adaptivity (OMEGA)". In: *Monthly Weather Review* 128 (2000), S. 2044–2076.
- [4] M. Baldauf, O. Fuhrer, M. J. Kurowski, G. de Morsier, M. Müllner, Z. P. Piotrowski, B. Rosa, P. L. Vitagliano, D. Wójcik und M. Ziemiański. *The COSMO Priority Project 'Conservative Dynamical Core' Final Report*. COSMO-Newsletter (Nr. 23). Oktober 2013.
- [5] Michael Baldauf. *A linear solution for flow over mountains and its comparison with the COSMO model*. COSMO-Newsletter (Nr. 9). Dezember 2008.
- [6] Michael Baldauf. "Linear Stability Analysis of Runge–Kutta-Based Partial Time-Splitting Schemes for the Euler Equations". In: *Monthly Weather Review* 138.12 (2010), S. 4475–4496. DOI: 10.1175/2010MWR3355.1.
- [7] Michael Baldauf. "Stability analysis for linear discretisations of the advection equation with Runge–Kutta time integration". In: *Journal of Computational Physics* 227 (2008), S. 6638–6659. DOI: 10.1016/j.jcp.2008.03.025.
- [8] Michael Baldauf. *The Coordinate Transformations of the 3-Dimensional Turbulent Diffusion in LMK*. COSMO-Newsletter (Nr. 5). April 2005.
- [9] Michael Baldauf und Slavko Brdar. "An analytic solution for linear gravity waves in a channel as a test for numerical models using the non-hydrostatic, compressible Euler equations". In: *Quarterly Journal of the Royal Meteorological Society* 139 (677 2013), S. 1977–1989. DOI: 10.1002/qj.2105.
- [10] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger und O. Sander. "A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part II: Implementation and Tests in DUNE". In: *Computing* 82.2–3 (2008), S. 121–138.
- [11] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, M. Ohlberger und O. Sander. "A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part I: Abstract Framework". In: *Computing* 82.2–3 (2008), S. 103–119.
- [12] Erich Becker. "Frictional heating in global climate models." In: *Monthly weather review* 131.3 (2003), S. 508–520.
- [13] Sébastien Blaise und Amik St-Cyr. "A Dynamic Hp-Adaptive Discontinuous Galerkin Method for Shallow-Water Flows on the Sphere with Application to a Global Tsunami Simulation". In: *Monthly Weather Review* 140.3 (2011), S. 978–996. DOI: 10.1175/MWR-D-11-00038.1.
- [14] Markus Blatt und Peter Bastian. "On the Generic Parallelisation of Iterative Solvers for the Finite Element Method". In: *International Journal of Computational Science and Engineering* 4.1 (2008), S. 56–69. DOI: 10.1504/IJCSE.2008.021112.
- [15] Markus Blatt und Peter Bastian. "The Iterative Solver Template Library". In: *Applied Parallel Computing. State of the Art in Scientific Computing*. Hrsg. von Bo Kågström, Erik Elmroth, Jack Dongarra und Jerzy Waśniewski. Bd. 4699. Lecture Notes in Computer Science. Springer, 2007, S. 666–675.

- [16] Craig F. Bohren und Bruce A. Albrecht. *Atmospheric Thermodynamics*. Oxford University Press, 1998.
- [17] N. Botta, R. Klein, S. Langenberg und S. Lützenkirchen. "Well balanced finite volume methods for nearly hydrostatic flows". In: *Journal of Computational Physics* 196 (2 Mai 2004), S. 539–565. DOI: 10.1016/j.jcp.2003.11.008.
- [18] Slavko Brdar. "High Order Locally Adaptive Discontinuous Galerkin Approach for Atmospheric Simulations". Diss. Albert-Ludwigs-Universität Freiburg, 2012.
- [19] Slavko Brdar, Michael Baldauf, Andreas Dedner und Robert Klöfkor. "Comparison of dynamical cores for NWP models: comparison of COSMO and Dune". In: *Theoretical and Computational Fluid Dynamics* (2012), S. 1–20. DOI: 10.1007/s00162-012-0264-z.
- [20] Slavko Brdar, Andreas Dedner und Robert Klöfkor. "Compact and stable Discontinuous Galerkin methods for convection-diffusion problems". In: *SIAM Journal on Scientific Computing* 34.1 (2012), A263–A282.
- [21] Ilja N. Bronstein, Konstantin A. Semendjajew, Gerhard Musiol und Heiner Mühling, Hrsg. *Taschenbuch der Mathematik*. 5. Aufl. Harri Deutsch, 2001.
- [22] George H. Bryan und J. Michael Fritsch. "A Benchmark Simulation for Moist Nonhydrostatic Numerical Models". In: *Monthly Weather Review* 130.12 (2002), S. 2917–2928. DOI: 10.1175/1520-0493(2002)130<2917:ABSFMN>2.0.CO;2.
- [23] W. Bundke. "Eine neue Rekursionsformel für die Legendre-Polynome und ihre Ableitungen für maschinentechnische Anwendungen". German. In: *Computing* 1 (4 1966), S. 309–315. DOI: 10.1007/BF02345484.
- [24] Guy Chavent und Bernardo Cockburn. *Consistance et stabilité des schémas LRG pour les lois de conservation scalaires*. Techn. Ber. 710. INRIA - Institut National de Recherche en Informatique et en Automatique, Juli 1987.
- [25] Bernardo Cockburn. "Discontinuous Galerkin methods". In: *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik* 83 (2003), S. 731–754. DOI: 10.1002/zamm.200310088.
- [26] Bernardo Cockburn, George E. Karniadakis und Chi-Wang Shu, Hrsg. *Discontinuous Galerkin Methods*. Lecture Notes in Computational Science and Engineering. New York, Berlin, Heidelberg: Springer, 2000.
- [27] Bernardo Cockburn und Chi-Wang Shu. *Runge-Kutta Discontinuous Galerkin Methods for Convection-dominated Problems*. Techn. Ber. ICASE NASA Langley Research Center, Nov. 2000.
- [28] Bernardo Cockburn und Chi-Wang Shu. "The Local Discontinuous Galerkin method for time-dependent convection-diffusion systems". In: *SIAM Journal on Numerical Analysis* 35.6 (Dez. 1998), S. 2440–2463.
- [29] Bernardo Cockburn und Chi-Wang Shu. "TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II - General Framework". In: *Mathematics of Computation* 52.186 (Apr. 1989), S. 411–435.
- [30] A. Dedner, R. Klöfkor, M. Nolte und M. Ohlberger. "A Generic Interface for Parallel and Adaptive Scientific Computing: Abstraction Principles and the DUNE-FEM Module". In: *Computing* 90.3–4 (2010), S. 165–196.
- [31] A. Dedner, R. Klöfkor, M. Nolte und M. Ohlberger. "DUNE-FEM. A general purpose discretization toolbox for parallel and adaptive scientific computing". In: *Advances in DUNE. Proceedings of the DUNE User Meeting, held 6.-8.10.2010, in Stuttgart, Germany*. Hrsg. von A. Dedner, B. Flemisch und R. Klöfkor. Springer, Mai 2012, S. 17–31. DOI: 10.1007/978-3-642-28589-9_2.
- [32] Günther Doms, Jochen Förstner, E. Heise, H.-J. Herzog, Dimitri Mironov, Matthias Raschendorfer, Thorsten Reinhard, Bodo Ritter, R. Schrodin, Jan-Peter Schulz und G. Vogel. *A Description of the Nonhydrostatic Regional COSMO-Model, Part II: Physical Parameterization*. Techn. Ber. Consortium for Small-Scale Modelling, Sep. 2011.

- [33] Günther Doms, Ulrich Schättler und Michael Baldauf. *A Description of the Nonhydrostatic Regional COSMO-Model, Part I: Dynamics and Numerics*. Techn. Ber. Consortium for Small-Scale Modelling, Sep. 2011.
- [34] Mark Dubal, Nigel Wood und Andrew Staniforth. "Some numerical properties of approaches to physics–dynamics coupling for NWP". In: *Quarterly Journal of the Royal Meteorological Society* 132.614 (2006), S. 27–42.
- [35] Dale R. Durran und Joseph B. Klemp. "A Compressible Model for the Simulation of Moist Mountain Waves". In: *Monthly Weather Review* 111.12 (1983), S. 2341–2361. DOI: 10.1175/1520-0493(1983)1112.0.CO;2.
- [36] Dieter Etling. *Theoretische Meteorologie - Eine Einführung*. 3. Aufl. Springer, 2008.
- [37] Saša Gaberšek, Francis X. Giraldo und James D. Doyle. "Dry and moist idealized experiments with a two-dimensional spectral element model". In: *Monthly Weather Review* 140 (2012), S. 3163–3182.
- [38] Tzvi Gal-Chen und Richard C. J. Somerville. "On the use of a coordinate transformation for the solution of the Navier-Stokes equations". In: *Journal of Computational Physics* 17.2 (1975), S. 209–228. DOI: 10.1016/0021-9991(75)90037-6.
- [39] Gregor Gassner. "Discontinuous Galerkin Methods for the Unsteady Compressible Navier-Stokes Equations". Diss. Universität Stuttgart, Jan. 2009.
- [40] Sean L. Gibbons. "Impacts of sigma coordinates on the Euler and Navier-Stokes equations using continuous Galerkin methods". Diss. Naval Postgraduate School, März 2009.
- [41] Francis X. Giraldo, Jan S. Hesthaven und Tim Warburton. "Nodal High-Order Discontinuous Galerkin Methods for the Spherical Shallow Water Equations". In: *Journal of Computational Physics* 181.2 (2002), S. 499–525. DOI: 10.1006/jcph.2002.7139.
- [42] Francis X. Giraldo und Marco Restelli. "A study of spectral element and discontinuous Galerkin methods for the Navier-Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases". In: *Journal of Computational Physics* 227.8 (2008), S. 3849–3877. DOI: 10.1016/j.jcp.2007.12.009.
- [43] Francis X. Giraldo und Tim Warburton. "A high-order triangular discontinuous Galerkin oceanic shallow water model". In: *International Journal for numerical methods in fluids* 56 (2008), S. 899–925. DOI: 10.1002/1fd.1562.
- [44] F. X. Giraldo, J. F. Kelly und E. M. Constantinescu. "Implicit-explicit formulations of a three-dimensional nonhydrostatic unified model of the atmosphere (NUMA)". In: *SIAM Journal on Scientific Computing* 35.5 (2013), B1162–B1194.
- [45] Sigal Gottlieb. "On High Order Strong Stability Preserving Runge–Kutta and Multi Step Time Discretizations". In: *Journal of Scientific Computing* 25 (1 2005), S. 105–128. DOI: 10.1007/s10915-004-4635-5.
- [46] David M. Hall und Ramachandran D. Nair. "Discontinuous Galerkin Transport on the Spherical Yin–Yang Overset Mesh". In: *Monthly Weather Review* 141.1 (2013), S. 264–282. DOI: 10.1175/MWR-D-12-00108.1.
- [47] Jan S. Hesthaven und Tim Warburton. *Nodal Discontinuous Galerkin Methods - Algorithms, Analysis, and Applications*. Springer, 2008.
- [48] J. S. Hesthaven. "From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex". In: *SIAM Journal on Numerical Analysis* 35.2 (1998), S. 655–676.
- [49] *IFS Documentaion - Part III: Dynamics and numerical procedures*. Cry40r1. Nov. 2013.
- [50] Bo Hoffmann Jørgensen. *Tensor formulation of the model equations on strong conservation form for an incompressible flow in general coordinates*. Techn. Ber. Risø National Laboratory, Dez. 2003.

- [51] Alex Kanevsky, Mark H. Carpenter, David Gottlieb und Jan S. Hesthaven. "Application of implicit–explicit high order Runge–Kutta methods to discontinuous-Galerkin schemes". In: *Journal of Computational Physics* 225.2 (2007), S. 1753–1781. DOI: <http://dx.doi.org/10.1016/j.jcp.2007.02.021>.
- [52] James F. Kelly und Francis X. Giraldo. "Continuous and Discontinuous Galerkin Methods for a Scalable Three-Dimensional Nonhydrostatic Atmospheric Model: Limited-Area Mode". In: *Journal of Computational Physics* 231.24 (2012), S. 7988–8008. DOI: 10.1016/j.jcp.2012.04.042.
- [53] J. B. Klemp, W. C. Skamarock und J. Dudhia. "Conservative Split-Explicit Time Intergration Methods for the Compressible Nonhydrostatic Equations". In: *Monthly Weather Review* 135.8 (Aug. 2007), S. 2897–2913.
- [54] Robert Klöforn. "Efficient Matrix-Free Implementation of Discontinuous Galerkin Methods for Compressible Flow Problems". In: *Proceedings of the ALGORITMY 2012*. Hrsg. von A. Handlovicova et al. 2012, S. 11–21.
- [55] Robert Klöforn. "Numerics for Evolution Equations — A General Interface Based Design Concept". Diss. Albert-Ludwigs-Universität Freiburg, 2009.
- [56] Konrad Königsberger. *Analysis 2*. 4. Aufl. Springer, 2002.
- [57] Helmut Kraus. *Die Atmosphäre der Erde - Eine Einführung in die Meteorologie*. 3. Aufl. Springer, 2004.
- [58] Norbert Kroll, Heribert Bieler, Herman Deconinck, Vincent Couaillier, Herman van der Ven und Kaare Sørensen, Hrsg. *ADIGMA - A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*. Bd. 113. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Springer, 2010.
- [59] Eric W. Lemmon, Richard T. Jacobsen, Steven G. Penoncello und Daniel G. Friend. "Thermodynamic Properties of Air and Mixtures of Nitrogen, Argon, and Oxygen From 60 to 2000 K at Pressures to 2000 MPa". In: *Journal of Physical and Chemical Reference Data* 29.3 (2000).
- [60] Albrecht Lindner. *Grundkurs Theoretische Physik*. 2. Aufl. Teubner Studienbücher, 1997.
- [61] David P. Lockard und Harold L. Atkins. "Efficient Implementations of the Quadrature-Free Discontinuous Galerkin Method". In: *14th Computational Fluid Dynamics Conference*. Fluid Dynamics and Co-Located Conferences. American Institute of Aeronautics und Astronautics, 1999. DOI: 10.2514/6.1999-3309.
- [62] Andreas Müller. "Untersuchungen zur Genauigkeit adaptiver un stetiger Galerkin Simulationen mit Hilfe von Luftblasen-Testfällen". Diss. Johannes Guttenberg-Universität Mainz, Juli 2012.
- [63] Andreas Müller, Jörn Behrens, Francis X. Giraldo und Volkmar Wirth. "Comparison between adaptive and uniform discontinuous Galerkin simulations in dry 2D bubble experiments". In: *Journal of Computational Physics* 235 (2013), S. 371–393. DOI: 10.1016/j.jcp.2012.10.038.
- [64] Ramachandran D. Nair, Stephen J. Thomas und Richard D. Loft. "A Discontinuous Galerkin Global Shallow Water Model". In: *Monthly Weather Review* 133.4 (2005), S. 877–888.
- [65] Sebastian Noelle, Yulong Xing und Chi-Wang Shu. "Numerical Methods for Balance Laws". In: Bd. 24. quaderni di matematica. Dipartimento di Matematica della Seconda Università di Napoli, 2010. Kap. High-order Well-balanced Schemes.
- [66] F. W. J. Olver, D. W. Lozier, R. F. Boisvert und C. W. Clark, Hrsg. *NIST Handbook of Mathematical Functions*. Print companion to [70]. New York, NY: Cambridge University Press, 2010.
- [67] Jaime Peraire und Per-Olof Persson. "The Compact Discontinuous Galerkin (CDG) Method for Elliptic Problems". In: *SIAM Journal on Scientific Computing* 30.4 (2008), S. 1806–1824.
- [68] Roger A. Pielke. *Mesoscale Meteorological Modeling*. Academic Press, London, 1984.
- [69] Jianxian Qiu, Boo Cheong Khoo und Chi-Wang Shu. "A numerical study for the performance of the Runge–Kutta discontinuous Galerkin method based on different numerical fluxes". In: *Journal of Computational Physics* 212.2 (2006), S. 540–565. DOI: 10.1016/j.jcp.2005.07.011.

- [70] NIST Digital Library of Mathematical Functions. <http://dlmf.nist.gov/>, Release 1.0.5 of 2012-10-01. Online companion to [66].
- [71] Marco Restelli. "Semi-Lagrangian and semi-implicit discontinuous Galerkin Methods for atmospheric modeling applications". Diss. Politecnico di Milano, März 2007.
- [72] Marco Restelli und Francis X. Giraldo. "A Conservative Discontinuous Galerkin Semi-Implicit Formulation for the Navier-Stokes Equations in Nonhydrostatic Mesoscale Modeling". In: *SIAM Journal on Scientific Computing* 31 (3 Mai 2009), S. 2231–2257. DOI: 10.1137/070708470.
- [73] Christop Schraff und Reinhold Hess. *A Description of the Nonhydrostatic Regional COSMO-Model, Part III: Data Assimilation*. Techn. Ber. Consortium for Small-Scale Modelling, Juni 2003.
- [74] Dieter Schuster, Slavko Brdar, Michael Baldauf, Andreas Dedner, Robert Klöforn und Dietmar Kröner. *On discontinuous Galerkin approach for atmospheric flow in the mesoscale with and without moisture*. eingereicht in: *Meteorologische Zeitschrift* (2013).
- [75] Chi-Wang Shu und Stanley Osher. "Efficient Implementation of Essentially Non-oscillatory Shock-Capturing Schemes". In: *Journal of Computational Physics* 77.2 (1988), S. 439–471. DOI: 10.1016/0021-9991(88)90177-5.
- [76] Herbert Sigloch. *Technische Fluidmechanik*. 7. Aufl. Springer, 2009.
- [77] William C. Skamarock. "Evaluating Mesoscale NWP Models Using Kinetic Energy Spectra". In: *Monthly Weather Review* 132.12 (2004), S. 3019–3032. DOI: 10.1175/MWR2830.1.
- [78] William C. Skamarock und Joseph B. Klemp. "A time-split nonhydrostatic atmospheric model for weather research and forecasting applications". In: *Journal of Computational Physics* 227.7 (2008). Predicting weather, climate and extreme events, S. 3465–3485. DOI: 10.1016/j.jcp.2007.01.037.
- [79] William C. Skamarock und Joseph B. Klemp. "Efficiency and Accuracy of the Klemp-Wilhelmson Time-Splitting Technique". In: *Monthly Weather Review* 122.11 (11 1994), S. 2623–2630. DOI: 10.1175/1520-0493(1994)122<2623:EAA0TK>2.0.CO;2.
- [80] William C. Skamarock, Joseph B. Klemp und Jimy Dudhia. "Prototypes for the WRF (Weather Research and Forecasting) Model". In: *18th Conference on Weather Analysis and Forecasting and the 14th Conference on Numerical Weather Prediction and Ninth Conference on Mesoscale Processes*. American Meteorological Society. 2001.
- [81] William C. Skamarock, Joseph B. Klemp, Jimy Dudhia, David O. Gill, Dale M. Barker, Michael G. Duda, Xiang-Yu Huang, Wei Wang und Jordan G. Powers. *A Description of the Advanced Research WRF Version 3*. Techn. Ber. NCAR/TN-475+STR. NCAR, Juni 2008.
- [82] Ronald B. Smith. "The Influence of Mountains on the Atmosphere". In: *Advances in Geophysics* 21 (1979), S. 87–230. DOI: 10.1016/S0065-2687(08)60262-9.
- [83] Andrew Staniforth, Nigel Wood und Jean Côté. "Analysis of the numerics of physics–dynamics coupling". In: *Quarterly Journal of the Royal Meteorological Society* 128.586 (2002), S. 2779–2799.
- [84] Andrew Staniforth, Nigel Wood und Jean Côté. "A Simple Comparison of Four Physics–Dynamics Coupling Schemes". In: *Monthly Weather Review* 130.12 (2002), S. 3129–3135. DOI: 10.1175/1520-0493(2002)130.0.CO;2.
- [85] J. M. Straka, Robert B. Wilhelmson, Louis J. Wicker, John R. Anderson und Kelvin K. Droegemeier. "Numerical solutions of a non-linear density current: A benchmark solution and comparisons". In: *International Journal for Numerical Methods in Fluids* 17.1 (1993), S. 1–22. DOI: 10.1002/flid.1650170103.
- [86] J. J. W. van der Vegt und H. van der Ven. "Slip flow boundary conditions in discontinuous Galerkin discretizations of the Euler equations of gas dynamics". In: *Proceedings of the 5th World Congress on Computational Mechanics (WCCM V), Vienna, Austria*. Hrsg. von H. A. Mang und F. G. Rammenstorfer. Vienna, Austria: Vienna University of Technology, Juli 2002, S. 1–16.
- [87] M. L. Weisman und J. B. Klemp. "The Dependence of Numerically Simulated Convective Storms on Vertical Wind Shear and Buoyancy". In: *Monthly Weather Review* 110.6 (1982), S. 504–520. DOI: 10.1175/1520-0493(1982)110.0.CO;2.

Literatur

- [88] Eric W. Weisstein, Hrsg. „*Legendre Polynomial.*“ *From MathWorld – A Wolfram Web Resource.* <http://mathworld.wolfram.com/LegendrePolynomial.html>. Nachgeschlagen 17. Januar 2013.
- [89] Eric W. Weisstein, Hrsg. „*Lobatto Quadrature.*“ *From MathWorld – A Wolfram Web Resource.* <http://mathworld.wolfram.com/LobattoQuadrature.html>. Nachgeschlagen 10. Januar 2013.
- [90] Deutscher Wetterdienst. *COSMO-EU: Einführung des Runge-Kutta-Dynamik-Kerns, Änderung der Referenzatmosphäre.* Juni 2010.
- [91] David L. Williamson. “The effect of time steps and time-scales on parametrization suites”. In: *Quarterly Journal of the Royal Meteorological Society* 139.671 (2013), S. 548–560.
- [92] David L. Williamson. “The Evolution of Dynamical Cores for Global Atmospheric Models”. In: *Journal of the Meteorological Society of Japan* 85B (2007), S. 241–269.
- [93] David L. Williamson. “Time-Split Versus Process-Split Coupling of Parameterizations and Dynamical Core”. In: *Monthly Weather Review* 130.8 (2002), S. 2024–2041. DOI: 10.1175/1520-0493(2002)1302.0.CO;2.
- [94] Yulong Xing, Xiangxiong Zhang und Chi-Wang Shu. “Positivity preserving high order well balanced discontinuous Galerkin methods for the shallow water equations”. In: *Advances in Water Resources* 33 (2010), S. 1476–1493.
- [95] Günther Zängl. “Extending the Numerical Stability Limit of Terrain-Following Coordinate Models over Steep Slopes”. In: *Monthly Weather Review* 140.11 (2012), S. 3722–3733. DOI: 10.1175/MWR-D-12-00049.1.
- [96] Xiangxiong Zhang und Chi-Wang Shu. “Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments”. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science* 467.2134 (2011), S. 2752–2776.