

**KfK 4980**  
**Februar 1992**

# **Zur Behandlung von offenen Rändern**

**Formulierung und Erweiterungen nach  
Orlanski, Raymond & Kuo, Miller & Thorpe**

**C.-P. Hugelmann, T. Kleiser**  
**Hauptabteilung für Daten- und Informationsverarbeitung**  
**Institut für Meteorologie und Klimaforschung**

**Kernforschungszentrum Karlsruhe**



KERNFORSCHUNGSZENTRUM KARLSRUHE  
Hauptabteilung für Daten- und Informationsverarbeitung  
Institut für Meteorologie und Klimaforschung

KfK 4980

**Zur Behandlung von offenen Rändern**  
Formulierung und Erweiterungen nach  
Orlanski, Raymond & Kuo, Miller & Thorpe

Claus-Peter Hugelmann, Thomas Kleiser

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript gedruckt  
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH  
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

**Zur Behandlung von offenen Rändern**  
Formulierung und Erweiterungen nach  
Orlanski, Raymond & Kuo, Miller & Thorpe

Zusammenfassung:

Verschiedene Formulierungen von Ausstrahlungsbedingungen für offene Ränder wurden an zweidimensionalen Problemen mit mehreren numerischen Verfahren ausgetestet. Die Resultate sind durch Serien von Isolindarstellungen wiedergegeben, um diese Randbedingungen miteinander vergleichen zu können.

**On the Treatment of Open Boundaries**  
Formulation and Extensions following  
Orlanski, Raymond & Kuo, Miller & Thorpe

Abstract:

Several formulations of radiation conditions at (artificial) open boundaries are tested on two-dimensional problems with various numerical schemes. Results are shown by series of contourplots in order to compare these boundary conditions.

## Vorwort

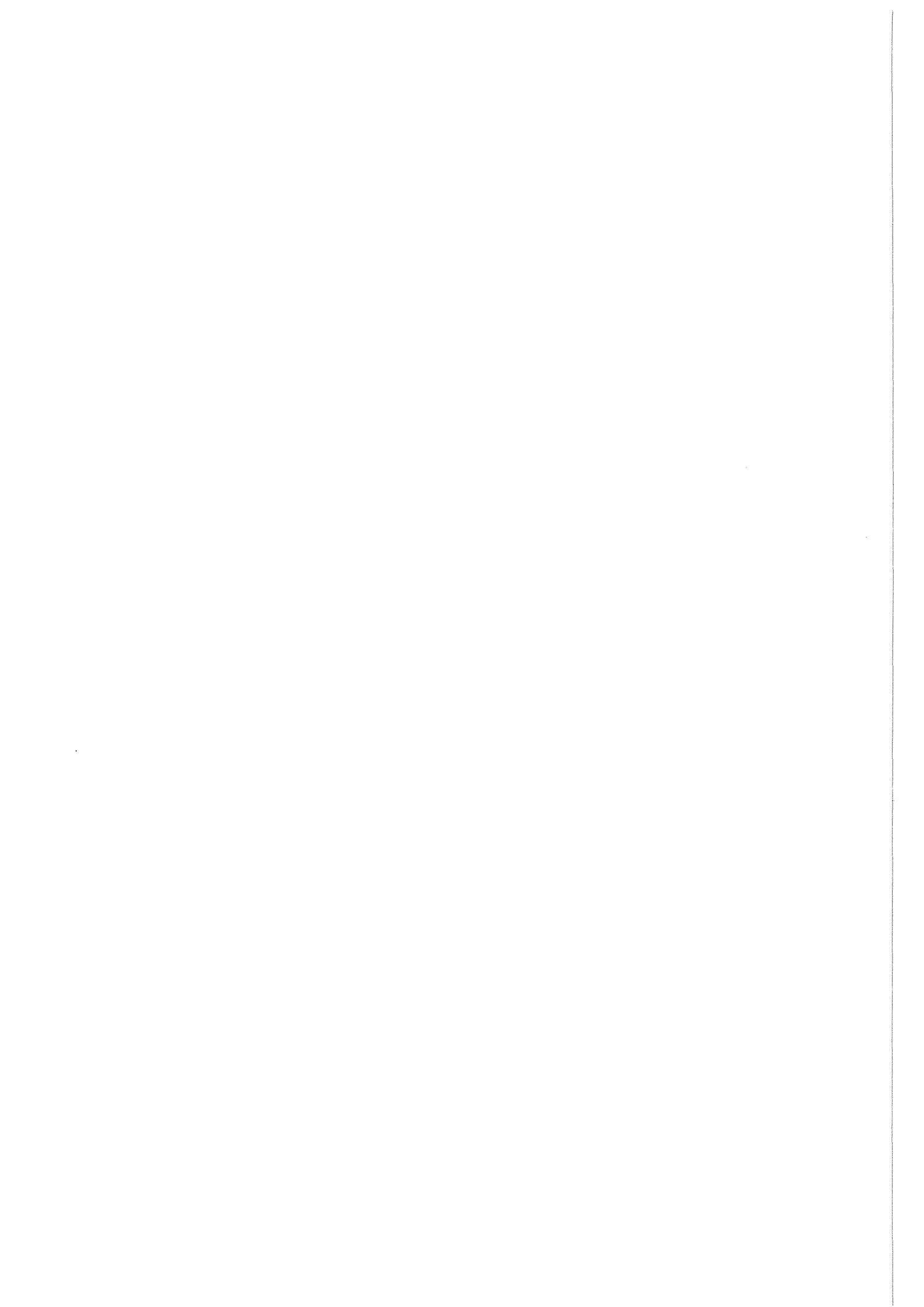
Bei der Simulation komplexer Systeme geht man oft so vor, daß aus dem realen Bereich der Teil *herausgeschnitten* und modelliert wird, in dem die entscheidenden Vorgänge ablaufen. Außerhalb dieses Modellgebiets wird ein homogenes Verhalten der Lösung erwartet, das für die eigentliche Betrachtung uninteressant ist und dessen Berechnung eingespart werden soll. Die *Schnittkanten* dieses Modellgebiets sind daher künstliche Ränder, an denen zur mathematischen Berechnung die Lösung bekannt sein oder zumindest eine Vorschrift zu deren Bestimmung vorliegen muß. Heuristisch sind die Anforderungen an diese *offenen* Ränder klar : Inhomogenitäten, die im Modellgebiet entstehen, sollen sich ungestört über den Rand hinweg ausbreiten und sich im Außengebiet verlieren können. Dagegen müssen Einflüsse von außen auf den Modellbereich verhindert werden.

Eine spezielle Art dieser Randformulierungen, in Anlehnung an die Ausstrahlungsbedingung nach Sommerfeld, wurde von Orlandi und weiteren Autoren für die ganze Klasse hyperbolischer und parabolischer Differentialgleichungen vorgeschlagen. Bestechend daran ist auch, neben diesen universellen Einsatzmöglichkeiten, der relativ geringe Aufwand für die Implementierung.

Als Diskussionbeitrag für den Einsatz solcher Verfahren in einem meteorologischen mesoskaligen Modell haben wir derartige Randwertformulierungen einer Reihe einfacher Tests unterzogen. In diesem Bericht werden die Erfahrungen festgehalten, die wir bei unseren Untersuchungen im Sommer 91 gemacht haben.

Karlsruhe, im Herbst 1991

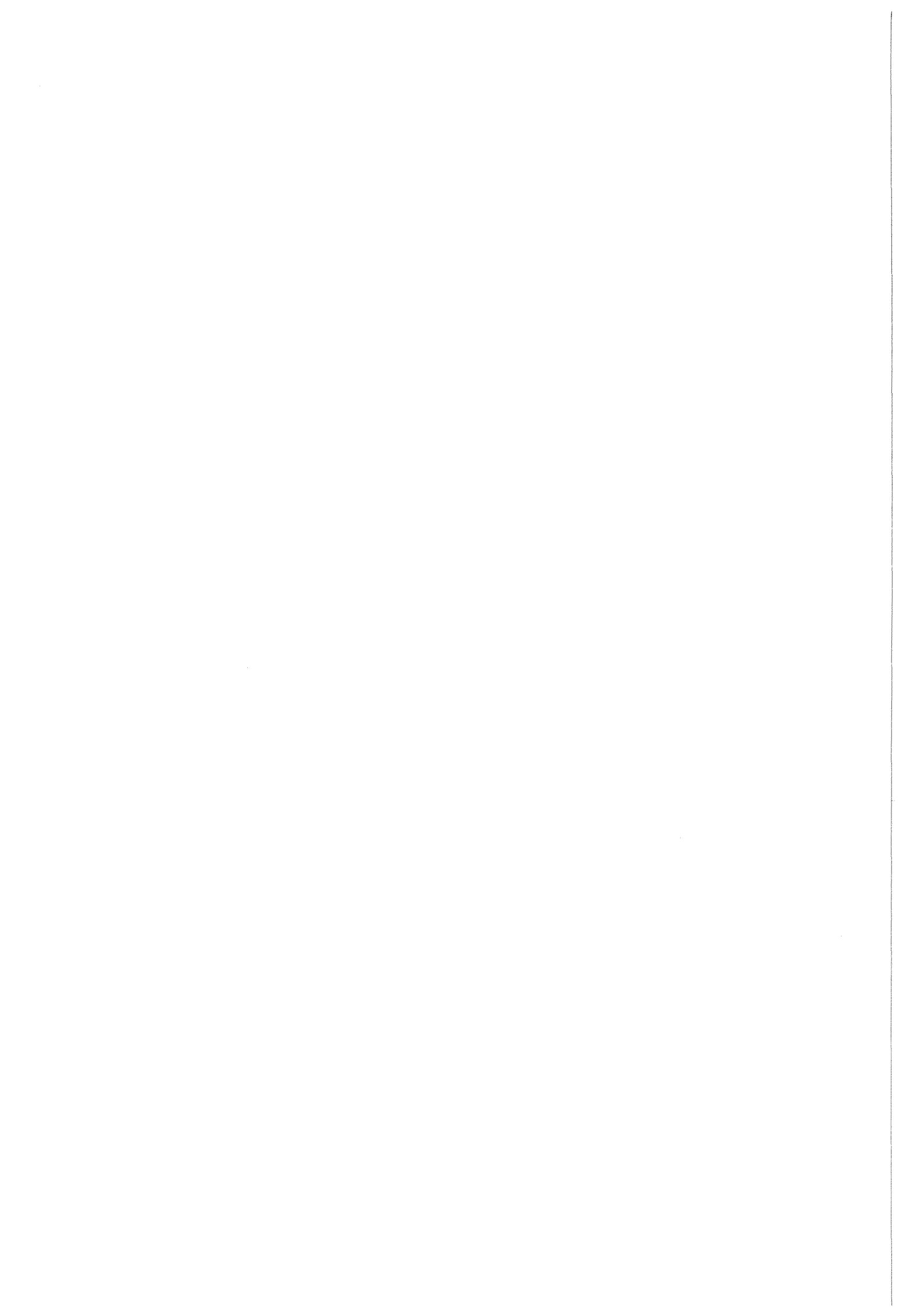
Claus-Peter Hugelmann  
Thomas Kleiser



# Inhaltsverzeichnis

<b>Kapitel 1. Einleitung</b> .....	<b>1</b>
<b>Kapitel 2. Die numerischen Behandlung der Modellgleichungen</b> .....	<b>5</b>
2.1 Die Advektionsgleichung .....	5
2.2 Die Diffusionsgleichung .....	8
<b>Kapitel 3. Die Randwertformulierungen</b> .....	<b>11</b>
3.1 Die Formulierung von Orlandi .....	11
3.2 Die Erweiterung durch Raymond und Kuo .....	12
3.3 Die Modifikation nach Miller und Thorpe .....	14
3.4 Einfache Extrapolationsformeln .....	15
3.5 Programmversionen .....	16
<b>Kapitel 4. Die Beurteilung der Tests</b> .....	<b>19</b>
4.1 Die Ergebnisse des Advektionsproblems .....	20
4.2 Die Ergebnisse des Diffusionsproblems .....	21
4.3 Zusammenfassung und Ausblick .....	23
<b>Anhang A. Beschreibung der Unterprogramme</b> .....	<b>25</b>
A.1 ORLAN1 .....	25
A.2 ORL1D2D .....	28
A.3 ORL2D2D .....	31
A.4 ORL3D3D .....	38
<b>Anhang B. Bilder</b> .....	<b>45</b>
<b>Literatur</b> .....	<b>93</b>





## Abbildungsverzeichnis

Abbildung	1. Entwicklung einer N-Welle	5
Abbildung	2. Diffusionsprozess an einer Kante	8
Abbildung	3. Drei Möglichkeiten der Extrapolation	15
Abbildung	4. Isolinien für die Anfangsdaten	45
Abbildung	5. Isolinien für die Advektion (ohne Randvorgabe) im 8. Schritt	46
Abbildung	6. Isolinien für die Advektion (ohne Randvorgabe) im 12. Schritt	46
Abbildung	7. Isolinien für die Advektion mit ORL1D2D im 12. Schritt	47
Abbildung	8. Isolinien für die Advektion mit ORL2D2D im 8. Schritt	47
Abbildung	9. Isolinien für die Advektion mit ORL2D2D im 12. Schritt	48
Abbildung	10. 3D-Darstellung für die Advektion mit ORL2D2D im 12. Schritt	48
Abbildung	11. Isolinien für die Advektion mit ORL2D2D im 30. Schritt	49
Abbildung	12. Isolinien für die Advektion mit ORL2DAD im 8. Schritt	49
Abbildung	13. 3D-Darstellung für die Advektion mit ORL2DAD im 8. Schritt	50
Abbildung	14. Isolinien für die Advektion mit ORL2DAD im 10. Schritt	50
Abbildung	15. Isolinien für die Advektion mit ORL2DAD im 12. Schritt	51
Abbildung	16. Isolinien für die Advektion mit ORL2DADS im 8. Schritt	51
Abbildung	17. 3D-Darstellung für die Advektion mit ORL2DADS im 8. Schritt	52
Abbildung	18. Isolinien für die Advektion mit ORL2DADS im 12. Schritt	52
Abbildung	19. 3D-Darstellung für die Advektion mit ORL2DADS im 12. Schritt	53
Abbildung	20. Isolinien für die Advektion mit ORL2DADS im 30. Schritt	53
Abbildung	21. Isolinien für die Advektion (ohne Randvorgaben) im 8. Schritt	54
Abbildung	22. Isolinien für die Advektion (ohne Randvorgaben) im 12. Schritt	54
Abbildung	23. Isolinien für die Advektion mit ORL1D2D im 8. Schritt	55
Abbildung	24. Isolinien für die Advektion mit ORL1D2D im 12. Schritt	55
Abbildung	25. Isolinien für die Advektion mit ORL2D2D im 8. Schritt	56
Abbildung	26. Isolinien für die Advektion mit ORL2D2D im 12. Schritt	57
Abbildung	27. Isolinien für die Advektion mit ORL2D2D im 30. Schritt	57
Abbildung	28. Isolinien für die Advektion mit ORL2DAD im 8. Schritt	58
Abbildung	29. Isolinien für die Advektion mit ORL2DAD im 12. Schritt	58
Abbildung	30. 3D-Darstellung für die Advektion mit ORL2DAD im 12. Schritt	59
Abbildung	31. Isolinien für die Advektion mit ORL2DADS im 8. Schritt	60
Abbildung	32. Isolinien für die Advektion mit ORL2DADS im 12. Schritt	60
Abbildung	33. Isolinien für die Advektion mit ORL2DADS im 30. Schritt	61
Abbildung	34. Isolinien für die Diffusion im 10. Schritt	61
Abbildung	35. Isolinien für die Diffusion im 20. Schritt	62
Abbildung	36. Isolinien für die Diffusion im 30. Schritt	63
Abbildung	37. Isolinien für die Diffusion mit $S(0) = S(1)$ im 10. Schritt	63
Abbildung	38. Isolinien für die Diffusion mit $S(0) = S(1)$ im 20. Schritt	64
Abbildung	39. 3D-Netz für die Diffusion mit $S(0) = S(1)$ im 20. Schritt	64
Abbildung	40. Isolinien für die Diffusion mit $S(0) = S(1)$ im 30. Schritt	65
Abbildung	41. Isolinien für die Diffusion mit $S(0) = S(2)$ im 10. Schritt	65
Abbildung	42. Isolinien für die Diffusion mit $S(0) = S(2)$ im 20. Schritt	66
Abbildung	43. Isolinien für die Diffusion mit $S(0) = S(2)$ im 30. Schritt	66
Abbildung	44. Isolinien für die Diffusion mit $S(0) = 2S(1) - S(2)$ im 10. Schritt	67
Abbildung	45. Isolinien für die Diffusion mit $S(0) = 2S(1) - S(2)$ im 20. Schritt	67
Abbildung	46. Isolinien für die Diffusion mit $S(0) = 2S(1) - S(2)$ im 30. Schritt	68
Abbildung	47. Isolinien für die Diffusion mit ORL1D2D im 10. Schritt	68
Abbildung	48. Isolinien für die Diffusion mit ORL1D2D im 20. Schritt	69
Abbildung	49. Isolinien für die Diffusion mit ORL1D2D im 30. Schritt	69

Abbildung 50.	Isolinien für die Diffusion mit ORL2D2D im 10. Schritt	70
Abbildung 51.	Isolinien für die Diffusion mit ORL2D2D im 20. Schritt	70
Abbildung 52.	Isolinien für die Diffusion mit ORL2D2D im 30. Schritt	71
Abbildung 53.	Isolinien für die Diffusion (implizit) im 10. Schritt	71
Abbildung 54.	Isolinien für die Diffusion (implizit) im 20. Schritt	72
Abbildung 55.	Isolinien für die Diffusion (implizit) im 30. Schritt	72
Abbildung 56.	Isolinien für die Diffusion mit SORL2D2D im 10. Schritt	73
Abbildung 57.	Isolinien für die Diffusion mit SORL2D2D im 30. Schritt	73
Abbildung 58.	Isolinien für die Diffusion mit ORL2DAD im 10. Schritt	74
Abbildung 59.	Isolinien für die Diffusion mit ORL2DAD im 20. Schritt	74
Abbildung 60.	Isolinien für die Diffusion mit ORL2DAD im 30. Schritt	75
Abbildung 61.	Isolinien für die Diffusion mit ORL2DADS im 10. Schritt	75
Abbildung 62.	Isolinien für die Diffusion mit ORL2DADS im 20. Schritt	76
Abbildung 63.	Isolinien für die Diffusion mit ORL2DADS im 30. Schritt	76
Abbildung 64.	Isolinien für die Anfangsbedingungen des versetzten Kastens	77
Abbildung 65.	Isolinien für die Diffusion nach 2 Schritten	77
Abbildung 66.	Isolinien für die Diffusion im 10. Schritt	78
Abbildung 67.	Isolinien für die Diffusion im 20. Schritt	78
Abbildung 68.	Isolinien für die Diffusion im 30. Schritt	79
Abbildung 69.	Isolinien für die Diffusion mit $S(0) = S(1)$ im 10. Schritt	79
Abbildung 70.	Isolinien für die Diffusion mit $S(0) = S(1)$ im 20. Schritt	80
Abbildung 71.	Isolinien für die Diffusion mit $S(0) = S(1)$ im 30. Schritt	81
Abbildung 72.	Isolinien für die Diffusion mit $S(0) = S(2)$ im 10. Schritt	81
Abbildung 73.	Isolinien für die Diffusion mit $S(0) = S(2)$ im 20. Schritt	82
Abbildung 74.	Isolinien für die Diffusion mit $S(0) = S(2)$ im 30. Schritt	82
Abbildung 75.	Isolinien für die Diffusion mit $S(0) = 2S(1) - S(2)$ im 10. Schritt	83
Abbildung 76.	Isolinien für die Diffusion mit $S(0) = 2S(1) - S(2)$ im 30. Schritt	84
Abbildung 77.	Isolinien für die Diffusion mit ORL1D2D im 10. Schritt	84
Abbildung 78.	Isolinien für die Diffusion mit ORL1D2D im 30. Schritt	85
Abbildung 79.	Isolinien für die Diffusion mit ORL2D2D im 10. Schritt	85
Abbildung 80.	Isolinien für die Diffusion mit ORL2D2D im 30. Schritt	86
Abbildung 81.	Isolinien für die Diffusion (implizit) im 10. Schritt	87
Abbildung 82.	Isolinien für die Diffusion (implizit) im 30. Schritt	87
Abbildung 83.	Isolinien für die Diffusion mit SORL2D2D im 10. Schritt	88
Abbildung 84.	Isolinien für die Diffusion mit SORL2D2D im 30. Schritt	88
Abbildung 85.	Isolinien für die Diffusion mit ORL2DAD im 10. Schritt	89
Abbildung 86.	Isolinien für die Diffusion mit ORL2DAD im 30. Schritt	90
Abbildung 87.	Isolinien für die Diffusion mit ORL2DADS im 10. Schritt	90
Abbildung 88.	Isolinien für die Diffusion mit ORL2DADS im 30. Schritt	91

# Kapitel 1. Einleitung

Ausgangspunkt unserer Untersuchungen ist das Karlsruher atmosphärische mesoskalige Modell KAMM des meteorologischen Instituts Karlsruhe. Es werden die zugrundeliegenden Navier-Stokes- und Transportgleichungen mit Hilfe von Differenzenverfahren gelöst, um, wie der Name andeutet, physikalische Vorgänge in einem Ausschnitt der Erdatmosphäre simulieren zu können.<sup>1</sup> Der Ausschnitt ist dabei kastenförmig gewählt mit einer vertikalen Erstreckung vom Boden (mit Topographie) bis etwa zur Tropopause (8-10km) und einer Kantenlänge in der Horizontalen bis zu 300km. Während der Boden als natürlicher unterer Rand des Modellgebiets physikalischen Beschreibungen und Vorgaben (wenn auch beliebig komplex) zugänglich ist, so bereitet der obere Rand bereits mehr Schwierigkeiten. Immerhin können dort auch noch physikalische Annahmen, etwa die Geostrophie des Windfeldes, gemacht werden und in das Modell als Randbedingung einfließen. Die seitlichen Ränder des Kastens dagegen sind absolut künstlich, aber angesichts der Beschränktheit der Ressourcen des Rechners auch in dieser Form notwendig. Außer den Modellgleichungen genügen die zu berechnenden Größen dort keinerlei physikalischen Bedingungen, die zu ihrer Bestimmung führen könnten. Hier müssen zusätzliche plausible Annahmen über das Verhalten der Lösung jenseits der Ränder ersatzweise zu den erforderlichen Bedingungen führen.

Diese Situation, illustriert am Modell KAMM, ist wohl typisch für die meisten Modelle der Strömungsmechanik, eine akzeptable Lösung ist daher von allgemeinem Interesse. Entsprechend findet man in der Literatur eine Vielzahl von Vorschlägen. Ein paar Möglichkeiten seien hier, mehr methodisch, angesprochen :

- △ **Vorgabe fester Randwerte.** Diese können etwa durch die Einbettung des Modellgebiets in ein größeres (und gröberes!) Umfeld bestimmt werden. Eine Methode, die dann notwendig erscheint, wenn dem Modellgebiet ein äußerer Zustand aufgeprägt werden soll, wie z.B. in der Meteorologie eine großräumige Wetterlage. Aber auch nur die Vorgabe einer Grundlösung mit bestimmten Eigenschaften, z.B. eine vollentwickelte turbulente Rohrströmung am Einströmrand eines Kanals, erfordert meist umfangreiche zusätzliche Berechnungen bis hin zu kompletten Modell-(vor-)läufen.
- △ **Verschiedene Formen der Extrapolation vom Inneren auf die Ränder.** Viele Methoden bis hin zur aufwendigen Fourieranalyse des Gesamtgebiets sind denkbar. Einfache Extrapolationsformeln können gleich in den Lösungsoperator eingebaut werden. Drei typische Vertreter haben wir in unsere Untersuchungen mit aufgenommen.
- △ **Einbau einer absorbierenden Randschicht.** Eine Methode, wie sie z.B. von Klemp/Durran(1983) für die Behandlung des oberen Randes eines Troposphärenausschnitts vorgeschlagen und in KAMM schon erprobt wurde. Dabei wird die Lösung in einem nicht geringen Teil des Modellgebiets bereits vor dem Rand durch künstlich eingeführte Diffusion soweit modifiziert, daß Stö-

---

<sup>1</sup> Zwar wäre man bei der Modellierung der Troposphäre als ganze, dünne Kugelschicht über dem Erdboden in der glücklichen Lage, horizontal mit periodischen Randbedingungen arbeiten zu können, jedoch gestattet dies, bei dem geforderten Auflösungsvermögen mesoskaliger Modelle, die Kapazität heutiger Supercomputer und selbst Rechner der nächsten Generation(en) noch nicht !

rungen, die zu unerwünschten Reflexionen führen könnten, herausgedämpft werden. Es bedarf einiger zusätzlicher Analysen der Störung zur Bestimmung der richtigen Diffusionskoeffizienten.

- △ **Ableitung von speziellen Differenzgleichungen am Rand aus den vorliegenden Differentialgleichungen.** Faßt man Annahmen über das Verhalten der Lösung außerhalb des Modellgebiets in mathematische Formulierungen, so kann man unter Umständen daraus Differentialgleichungen für die Lösung auf dem Rand gewinnen. Beispielsweise liefert die Potentialtheorie, zu der man durch Stationaritätsbetrachtungen gelangen kann, in gewissen Fällen Lösungen des sogenannten Außenraumproblems. Die Theorie der Pseudodifferentiale liefert Ausdrücke in den Differentiationsoperatoren aus Dispersionsrelationen oder zusätzlichen Annahmen über die Fouriertransformierte der Lösung am Rand. Wir können keinen Überblick über alle denkbaren Möglichkeiten geben. Vielfach kommen diese Methoden auch erst in Kombination mit anderen Arten von Randbedingungen zum Tragen. So liefert erst die Einführung einer Randschicht zwischen dem (z.B. rechteckigen) Modellrand und dem Übergang zum (etwa radialsymmetrischen) Fernfeld im obigen Beispiel das gewünschte Ergebnis. Oder es treten zusätzliche Schwierigkeiten auf, wie im Fall der Pseudodifferentiale, wo man vor der sicher nicht leichten Aufgabe steht, die Pseudodifferentiale in Differenzgleichungen für die Variablen zu "übersetzen"! Dennoch sind die Ergebnisse teilweise bestechend und sollten nicht aus den Augen verloren werden.
- △ **Allgemeine Ausstrahlungsrandbedingungen.** Ausgangspunkt dieses Typs ist eine Bemerkung Sommerfelds(1949) über die Phasen der Strömungsvariablen im Fernfeld einer ausstrahlenden Quelle. Bei dieser Art der Randformulierung wird allein an Hand der Daten unterschieden, ob der Phasenfluß ins Modellgebiet hinein oder aus dem Modellgebiet hinaus gerichtet ist. Im letzteren Fall sollen "Störungen" ohne Reflexion aus dem Modellgebiet hinauswandern können, während andererseits ein Eindringen von Information außerhalb des Modellgebiets ins Modellgebiet verhindert werden soll. Hierzu gehören die Arbeiten von Orlanski(1982), Raymond,Kuo(1984) und Miller,Thorpe(1981)

Unsere Untersuchungen beschränkten sich im wesentlichen auf die zitierten Arbeiten dieser letzten Klasse von Randbedingungen. Der geringe Aufwand und universelle Einsatzmöglichkeiten ließen die Methoden für ein Modell wie "KAMM" bei den meisten Anwendungen als geeignet erscheinen.

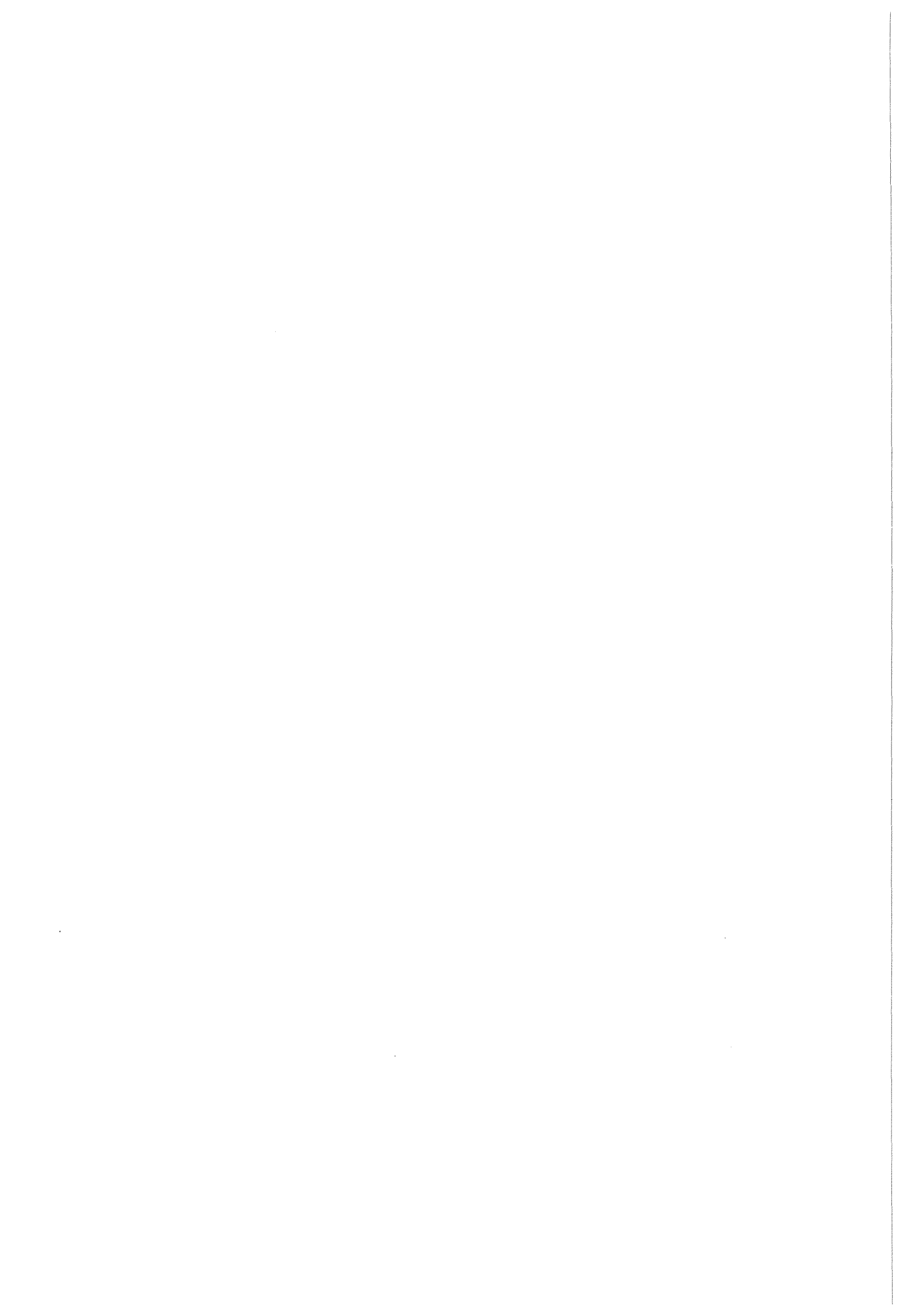
Speziell für Probleme, bei denen Advektion und/oder Wellenbewegungen dominierend sind, hat Orlanski und vor ihm schon Pearson(1974) eine derartige "offene" Randbedingung vorgeschlagen. Dabei hat Orlanski die Hoffnung geäußert, ein Verfahren zu formulieren, das zumindest auf eine weite Klasse hyperbolischer partieller Differentialgleichungen anwendbar sein soll. Er "belegt" dies, wie andere Autoren auch, am Beispiel der Flachwasserwellengleichung mit dem Crank-Nicolson- und dem Leap-Frog- Schema. Für die vorgesehenen Testläufe boten sich uns dafür in natürlicher Weise die hyperbolische Advektionsgleichung und die parabolische Diffusionsgleichung in 2 Dimensionen an. Zum einen sind die beiden Gleichungen charakteristisch für Ausbreitungs- und Evolutionsvorgänge. Zum anderen bilden sie die wichtigsten Bausteine, etwa beim Operatorsplitting, für die Navier-Stokes-Gleichungen (Munz,1991). Schließlich sind für spezielle Fälle auch analytische Lösungen bekannt. Das Verhalten von Störungen am Rand sollte man mindestens in zwei Dimensionen untersuchen, da Störungen, die nicht normal zum Rand laufen, den meisten " Ärger " verursachen, was ja erst zu den ange-

gebenen Erweiterungen geführt hat. Eine Beschränkung auf zwei Dimensionen wiederum ist insofern berechtigt, da die Ergebnisse durchaus übertragbar sind auf drei Dimensionen, dafür aber schneller gewonnen und übersichtlicher dargeboten werden können.

Um zu deutlichen Aussagen zu kommen, wurden Anfangsbedingungen mit steilen Gradienten gewählt. Während dies für die parabolische Diffusionsgleichung sofort zur Glättung führt, bleiben diese im hyperbolischen Fall erhalten oder verstärken sich sogar noch (siehe nichtlinearer Fall). Das stellt besondere Anforderungen an die diskreten Lösungsoperatoren der Modellgleichungen. Wir haben daher besondere Sorgfalt bei der Auswahl der numerischen Verfahren verwendet, die dem Vergleich mit analytischen Lösungen standhalten mußten. Für die Advektion benutzten wir ein eindimensionales Verfahren, das mit Raumsplittung auf die zweidimensionale Advektionsgleichung im linearen und nichtlinearen Fall angewandt wurde. Bei der einfacheren Behandlung der Diffusionsgleichung konnten wir explizite und implizite Verfahren für die volle zweidimensionale Gleichung einsetzen, oder, ebenfalls mit Raumsplittung, explizite eindimensionale Methoden ausprobieren. Ein großer Einfluß auf die Ergebnisse der Randwertformulierungen durch die unterschiedlichen Methoden ließ sich aber nicht erkennen. In Kapitel 2 werden zunächst diese Modellgleichungen für Advektion und Diffusion sowie die zugehörigen Operatoren für die Lösung der diskretisierten Gleichungen vorgestellt. Dabei setzten wir eine gewisse Vertrautheit mit Begriffen aus der numerischen Berechnung strömungsdynamischer Probleme voraus, die etwa bei Roache(1982) beschrieben sind.

In Kapitel 3 werden dann die verschiedenen Ansätze für die Randbelegung behandelt. Hierbei unterscheiden wir noch einmal zwischen einfachen Extrapolationen auf den Rand und den offenen Randbedingungen nach Orlanski, Raymond, Kuo und Miller, Thorpe. Wir beschreiben insbesondere die Umsetzung dieser Ansätze auf die von uns verwendete Diskretisierungsvorschrift. Die genauen Formulierungen verschiedener Programme wurden in Form von Kommentarkarten, die den Fortran-Programmen vorangestellt sind, im Anhang A zusammengefaßt.

Kapitel 4 zeigt dann die Ergebnisse der vorgestellten Randschemata und versucht eine Bewertung bezüglich der Effizienz. Analytische Lösungen dienten als Maßstab für die Güte der Lösungsoperatoren, wurden aber bei der Bewertung lediglich als Orientierungshilfe benutzt, um etwaige Mängel des diskreten Operators nicht dem Randschema aufzubürden. Stattdessen werden die Ergebnisse des Randschemas für ein kleines Gebiet verglichen mit den Ergebnissen eines Gebiets, das die Randwerte als innere Punkte enthält. Die graphischen Darstellungen sind einfachheitshalber im Anhang B zusammengefaßt worden.



# Kapitel 2. Die numerischen Behandlung der Modellgleichungen

## 2.1 Die Advektionsgleichung

Als allgemeine Advektionsgleichung in zwei Dimensionen für eine Größe  $\phi(x, y, t)$  bezeichnen wir die partielle Differentialgleichung

$$(2.1) \quad \frac{\partial \phi}{\partial t} + \frac{\partial f(\phi)}{\partial x} + \frac{\partial g(\phi)}{\partial y} = 0$$

mit stetig differenzierbaren Funktionen  $f, g$ .

Die einfachsten Fälle sind durch die linearen Funktionen  $f(\phi) = u \cdot \phi$  und  $g(\phi) = v \cdot \phi$  gegeben mit konstanten Geschwindigkeitskomponenten  $u$  und  $v$ . Die analytische Lösung der Differentialgleichung (2.1) geht dann aus jeder beliebigen Anfangsbedingung

$$(2.2) \quad \phi(x, y, 0) = \phi_0(x, y)$$

durch Verschieben in Richtung des Geschwindigkeitsvektors  $\vec{v} = (u, v)$  hervor:

$$(2.3) \quad \phi(x, y, t) = \phi_0(x - u \cdot t, y - v \cdot t) .$$

Insbesondere bleiben Unstetigkeiten in den Anfangsbedingungen für alle Zeiten in der Lösung erhalten.

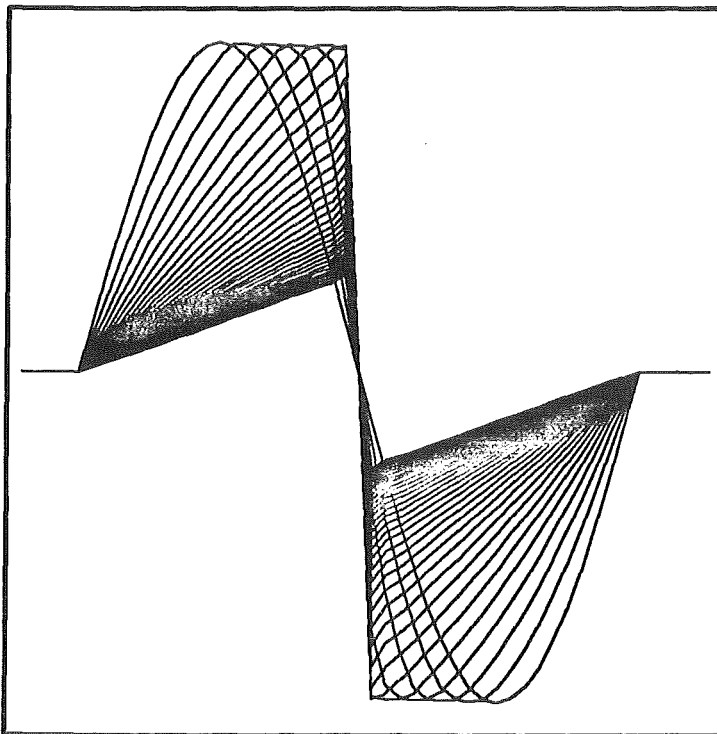


Abbildung 1. Entwicklung einer N-Welle



Im nichtlinearen Fall, etwa für  $f(\phi) = \frac{1}{2}\phi^2$  ist sogar ein Aufsteilen der Lösung im Laufe der Zeit zu erwarten. Abbildung 1 auf Seite 5 zeigt dazu, für den eindimensionalen Fall, die zeitliche Entwicklung bei einer sinusförmigen Anfangsbedingung. Das benutzte numerische Verfahren gibt sehr gut die analytische Lösung, die Entwicklung einer "N-Welle", wieder (vgl. Smoller, 1983). Wir stellen es hier kurz vor.

Die eindimensionale Advektionsgleichung wird dazu in Flußform diskretisiert :

$$(2.4) \quad \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + \frac{f_{i+1/2} - f_{i-1/2}}{\Delta x} = 0$$

Dabei bezeichnet  $\phi_i^n$  die numerische Approximation der gesuchten Funktion  $\phi$  zu einem Zeitpunkt  $t_n$  an der Stelle  $x_i$ . Der Zeitschritt  $\Delta t$  ist durch  $\Delta t = t_{n+1} - t_n$  gegeben, die räumliche Diskretisierung wird zur Vereinfachung als äquidistant angesehen :  $\Delta x = x_{i+1} - x_i$ . Der numerische Fluß  $f_{i+1/2}$  setzt sich zusammen aus einem monotonen Fluß und einem limitierten antidiffusiven Fluß

$$f_{i+1/2} = f_{i+1/2}^{\text{mon}} + f_{i+1/2}^{\text{lim}}$$

entsprechend den Flußkorrekturverfahren, wie sie etwa bei Hugelmann(1988) für den linearen Fall beschrieben sind. Für den monotonen Fluß wählen wir das Schema von Engquist, Osher(1981), ein verallgemeinertes Upwind-Verfahren, das im linearen Fall mit dem allgemein bekannten Verfahren von Courant, Isaacson, Rees(1952) zusammenfällt. Diese Klasse von Verfahren zeichnet sich dadurch aus, daß sie den Einflußbereich der Charakteristiken, längs derer die Lösung konstant bleibt, (längs derer also jegliche Information fließt,) genau berücksichtigt. Das ist in unserem Fall von besonderer Bedeutung. Wenn etwa im linearen Fall  $f(\phi) = u \cdot \phi$  mit  $u > 0$  die Lösung auf dem rechten Rand **vollständig** durch die Werte im Inneren bestimmt sind, so wird dies durch das Verfahren automatisch sichergestellt. Es wählt die mathematisch korrekten Daten für die Bestimmung der Lösung am Rand, künstliche Vorgaben werden ignoriert. Leider sind die monotonen Verfahren von (höchstens) erster Ordnung. Um die numerische Diffusion zu begrenzen, bedarf es daher zusätzlicher Korrekturen.

Der antidiffusive Fluß ergibt sich wieder aus der Differenz eines Verfahrens höherer Ordnung ( hier : Lax, Wendroff(1960)) und dem monotonen Verfahren, muß aber, anders wie im linearen Fall, selbst nochmals aufgespalten werden in einen Anteil nach rechts und einen Anteil nach links :

$$f_{i+1/2}^{\text{ad}} = \alpha_{i+1/2}^+ \Delta f_{i+1/2}^+ - \alpha_{i+1/2}^- \Delta f_{i+1/2}^-$$

mit

$$\Delta f_{i+1/2}^+ = f(\phi_{i+1}) - f_{i+1/2}^{\text{mon}}, \quad \Delta f_{i+1/2}^- = f_{i+1/2}^{\text{mon}} - f(\phi_i)$$

und

$$\alpha_{i+1/2}^+ = \frac{1}{2} \left[ 1 - \frac{\Delta t}{\Delta x} \frac{\Delta f_{i+1/2}^+}{\phi_{i+1} - \phi_i} \right]$$

$$\alpha_{i+1/2}^- = \frac{1}{2} \left[ 1 - \frac{\Delta t}{\Delta x} \frac{\Delta f_{i+1/2}^-}{\phi_{i+1} - \phi_i} \right]$$

Beide Anteile werden nun für sich getrennt mit einer Limiterfunktion  $\Phi(r)$  begrenzt:

$$f_{i+1/2}^{\text{lim}} = \Phi(r_{i+1/2}^+) \cdot \alpha_{i+1/2}^+ \Delta f_{i+1/2}^+ - \Phi(r_{i+1/2}^-) \cdot \alpha_{i+1/2}^- \Delta f_{i+1/2}^-$$

mit den Verhältnissen der Gradienten als Argumente :

$$r_{i+1/2}^+ = \frac{\Delta f_{i-1/2}^+}{\Delta f_{i+1/2}^+}, \quad r_{i+1/2}^- = \frac{\Delta f_{i+3/2}^-}{\Delta f_{i+1/2}^-}$$

Für  $\Phi$  wurde der "Superbee-Limiter" gewählt, so daß das Gesamtverfahren die TVD-Eigenschaft besitzt (Sweby, 1984) (das heißt, die Totalvariation sich nicht erhöht) und in glatten Bereichen von zweiter Ordnung ist. Am Rand wird dann wieder auf das monotone Verfahren zurückgeschaltet. Sind die Charakteristiken aus dem Innern des Gebiets auf den Rand gerichtet, so bewirkt die Vorgabe von Randwerten lediglich eine Anpassung und hat, sofern der Rand nicht von Ausströmen auf Einströmen dreht, für den Prozeß im Innern keine Bedeutung.

Die Stabilität des Verfahrens ist durch eine Verallgemeinerung des Courant-Friedrichs-Lewy-Kriteriums (kurz CFL-Kriterium) gegeben :

$$(2.5) \quad \frac{\Delta t}{\Delta x} \cdot \max_{\phi} |f'(\phi)| \leq \sigma \leq 1$$

Im linearen Fall erhält man daraus für den Zeitschritt

$$(2.5') \quad \Delta t = \sigma \cdot \Delta x \cdot \left[ \max_u |u| \right]^{-1}$$

mit einem *Sicherheitsfaktor*  $0 < \sigma \leq 1$ .

Für die Modellgleichung (2.1) wird das gleiche eindimensionale Schema (2.4) abwechselnd auf die x-Richtung und die y-Richtung angewandt (**Raumsplitting**). Das heißt, daß für **einen** Zeitschritt die beiden Gleichungen

$$\frac{\partial \phi}{\partial t} + \frac{\partial f(\phi)}{\partial x} = 0$$

(2.6.a,b) und

$$\frac{\partial \phi}{\partial t} + \frac{\partial g(\phi)}{\partial y} = 0$$

**nacheinander** gelöst werden. Dabei findet ein Wechsel in der Reihenfolge der Bearbeitung statt, was aber kaum Veränderungen im Lösungsverhalten mit sich bringt. Der Zeitschritt ist für beide diskreten Gleichungen natürlich gleich. Man beachte aber, daß das Kriterium (2.5) zunächst nur für die erste zu lösende Gleichung einen stabilen Zeitschritt liefert. Der Sicherheitsfaktor soll gewährleisten, daß das Verfahren auch im zweiten Schritt, mit den modifizierten Daten aus dem ersten Schritt, noch stabil ist, andernfalls muß Schritt 1 mit verkleinerter Zeitschrittweite wiederholt werden.

## 2.2 Die Diffusionsgleichung

$$(2.7) \quad \frac{\partial \phi}{\partial t} = \frac{\partial}{\partial x} \left( D \cdot \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left( E \cdot \frac{\partial \phi}{\partial y} \right)$$

Hier ist der einfachste Fall durch konstante Diffusionskoeffizienten gegeben :

$$(2.8) \quad D(\phi; x, y) = \text{const} = E(\phi; x, y)$$

Charakteristisch für die Lösungen dieser Differentialgleichung ist das sofortige Ausglätten selbst un stetiger Anfangsbedingungen.

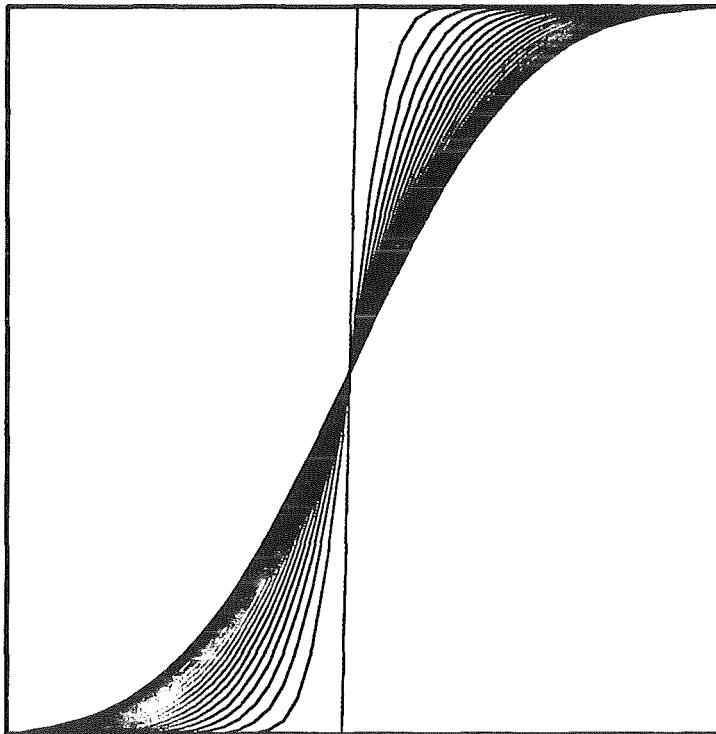


Abbildung 2. Diffusionsprozess an einer Kante

In Abbildung 2 ist dies am Beispiel einer Kante dargestellt.

Für die Diffusion wurden zunächst verschiedene **explizite** Schemata (voll 2D - Raumsplitting - Raumsplitting mit Raumrichtungswechsel) ausprobiert, die aber alle keine nennenswerten Unterschiede aufwiesen. Selbst die Symmetrie war in allen Fällen überraschend gut gewährleistet. Wir beschränken uns daher auf die Darstellung des vollen zweidimensionalen Falls. Dabei werden die Diffusionsterme in (2.7) einfach durch zentrale Differenzen approximiert :

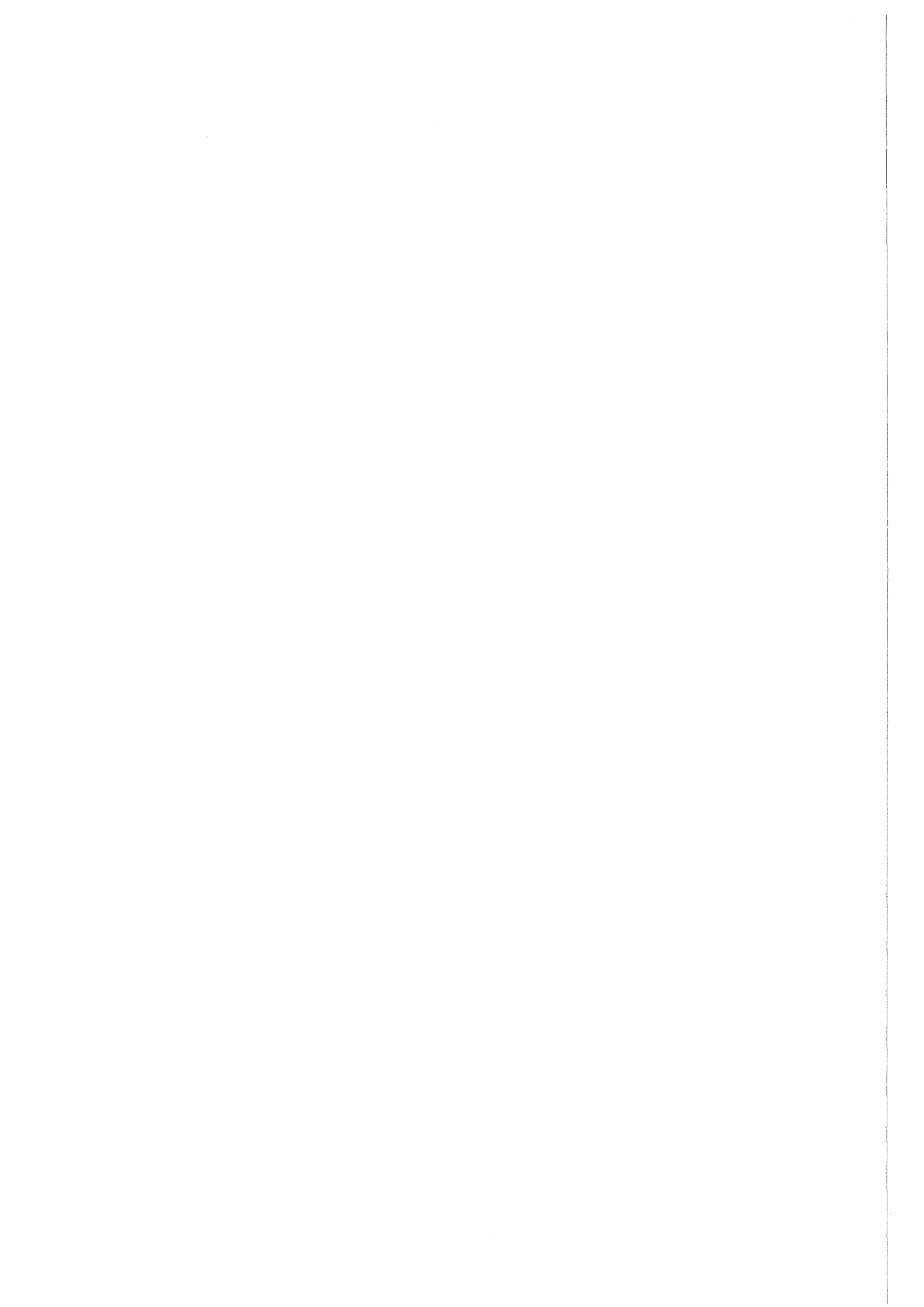
$$(2.9) \quad \frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\Delta t} = \frac{1}{\Delta x^2} [D_{i+1/2,j} \cdot (\phi_{i+1,j}^n - \phi_{i,j}^n) - D_{i-1/2,j} \cdot (\phi_{i,j}^n - \phi_{i-1,j}^n)] \\ + \frac{1}{\Delta y^2} [E_{i,j+1/2} \cdot (\phi_{i,j+1}^n - \phi_{i,j}^n) - E_{i,j-1/2} \cdot (\phi_{i,j}^n - \phi_{i,j-1}^n)]$$

Für alle inneren Punkte kann nach Gleichung (2.9) der Funktionswert zum neuen Zeitpunkt berechnet werden, die Werte am Rand müssen aus einer Randbedingung bestimmt werden. Für die Stabilität des Verfahrens ist wieder eine Begrenzung des Zeitschritts notwendig :

$$(2.10) \quad \Delta t \leq \sigma \frac{1}{2} \cdot \left[ \frac{D_{\max}}{\Delta x^2} + \frac{E_{\max}}{\Delta y^2} \right]^{-1}$$

mit dem Sicherheitsfaktor  $0 < \sigma < 1$  .

Ein **implizites** Schema ergibt sich, wenn auf der rechten Seite von (2.9) anstelle der Funktionswerte  $\phi^n$  die entsprechenden unbekanntenen Werte  $\phi^{n+1}$  zum neuen Zeitpunkt  $t_{n+1}$  eingesetzt werden. Durchlaufen  $i$  und  $j$  in (2.9) dabei die Indizes aller inneren Punkte des Modellgebiets, so ergibt sich ein Gleichungssystem, das bei Vorgabe der Randpunkte zum neuen Zeitpunkt vollständig gelöst werden kann. Eine Einschränkung des Zeitschritts aus Stabilitätsgründen ist im impliziten Fall nicht erforderlich. Wohl ist es aus physikalischen Gründen sinnvoll, den Zeitschritt nicht allzu groß zu wählen, im vorliegenden Fall wurde aber zum besseren Vergleich ein Zeitschritt, wie er sich aus (2.10) ergibt, eingesetzt.



## Kapitel 3. Die Randwertformulierungen

### 3.1 Die Formulierung von Orlandi

Orlandis Idee für die Behandlung des Randes lehnt sich an die allgemeine Ausstrahlungsbedingung nach Sommerfeld an. Danach genügt eine Größe  $\phi$  am Rand, fernab der Quelle, einer einfachen Transportgleichung

$$(3.1) \quad \frac{\partial \phi}{\partial t} + C \frac{\partial \phi}{\partial n_r} = 0 \quad ,$$

wobei  $\phi$  die Variable,  $\partial/\partial n_r$  die Ableitung und  $C$  die Phasengeschwindigkeit normal zum Rand sind. Diese Gleichung, in diskretisierter Form, soll dazu benutzt werden, die Randwerte zu belegen. Zuvor muß aber die noch unbekannte Größe  $C$  bestimmt werden. Pearson(1974) schätzt sie aus einer linearisierten Dispersionsbeziehung. Orlandi nutzt die Beziehung (3.1) nochmals aus und erhält, zunächst rein formal, für die lokale Phasengeschwindigkeit :

$$(3.2) \quad C = - \left( \frac{\partial \phi}{\partial t} \right) \left( \frac{\partial \phi}{\partial n_r} \right)^{-1}$$

Allerdings können diese Gleichung und (3.1) nicht im jeweils gleichen Diskretisierungspunkt am Rand ausgewertet werden. Orlandi wählt dafür die um einen Raumpunkt nach innen und ein Zeitniveau zurück versetzte Stelle. Ein weiteres Problem entsteht bei der Quotientenbildung in (3.2) für verschwindend kleine Nenner. Nun sind unerwünschte einlaufende Wellen, etwa am rechten Rand, durch negative Phasengeschwindigkeiten  $C < 0$  gekennzeichnet und sollen unterdrückt werden. Hier bietet sich eine Begrenzung nach unten durch  $C = 0$  an. Das bedeutet nach (3.1), daß der Randwert unverändert bleibt. Die Phasengeschwindigkeiten auslaufender Wellen sind nicht negativ und auf Grund des hyperbolischen Charakters beschränkt. Das Stabilitätskriterium (2.5') liefert hier als obere Schranke  $C \leq \Delta x/\Delta t$ .

Eine Darstellung der Originalbeziehungen von Orlandi geben wir nur im Anhang A wieder (ORLAN1), da sie auf der Diskretisierung durch ein Leap-Frog-Schema auf einem zeitlich versetzten Gitter beruhen. Wir übertragen vielmehr diese Idee auf ein Upwind-Schema mit Euler-Vorwärts-Differenzen in der Zeit.

Die Diskretisierung von (3.1) in einer zweidimensionalen Formulierung am rechten Rand, d.h. für  $x = x_N$ , lautet dann :

$$(3.3) \quad \phi_{N,j}^{\tau+1} = \phi_{N,j}^{\tau} - \frac{\Delta t}{\Delta x} C \cdot (\phi_{N,j}^{\tau} - \phi_{N-1,j}^{\tau})$$

Die Phasengeschwindigkeit  $C$  ergibt sich aus (3.2)

$$(3.4) \quad \tilde{C} = - \left( \frac{\phi_{N-1,j}^{\tau} - \phi_{N-1,j}^{\tau-1}}{\Delta t} \right) \left( \frac{\phi_{N-1,j}^{\tau-1} - \phi_{N-2,j}^{\tau-1}}{\Delta x} \right)^{-1}$$

mit den Beschränkungen

$$(3.5) \quad c = \begin{cases} 0 & \tilde{C} < 0 \\ \tilde{C} & 0 \leq \tilde{C} \leq \Delta x / \Delta t \\ \Delta x / \Delta t & \tilde{C} > \Delta x / \Delta t \end{cases}$$

Eine Belegung der Ränder kann nun analog dazu auch für die anderen drei Seiten erfolgen, wobei auf das richtige Vorzeichen bei der Begrenzung der Phasengeschwindigkeiten zu achten ist.

Diese Formulierung ist durch das Unterprogramm ORL1D2D realisiert.

### 3.2 Die Erweiterung durch Raymond und Kuo

Die Berechnung von C in der Version nach Orlandi unter Berücksichtigung jeweils nur der Raumrichtung normal zum Rand hin hat nun den Nachteil, daß teilweise unrealistische Phasengeschwindigkeiten prognostiziert werden. Raymond und Kuo (1984) haben deshalb dieses einfache Verfahren in der Weise erweitert, daß am Rand auch eine Advektion in jede beliebige Raumrichtung zulässig ist. Zur Bestimmung des Phasengeschwindigkeitsvektors setzen wir zunächst, analog zu (3.1), an :

$$(3.6) \quad \begin{aligned} \frac{\partial \phi}{\partial t} &= -\hat{C}_x \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial t} &= -\hat{C}_y \frac{\partial \phi}{\partial y} \\ \frac{\partial \phi}{\partial t} &= -\hat{C}_z \frac{\partial \phi}{\partial z} \end{aligned}$$

Dabei sind  $\hat{C}_x$ ,  $\hat{C}_y$ ,  $\hat{C}_z$  die scheinbar auftretenden Phasengeschwindigkeiten in der x-, y-, und z-Koordinate, bezogen relativ zur mittleren Strömung in diesen Raumrichtungen. Man hat  $\hat{C}_\xi = \omega / k_\xi$ , wobei  $\omega$  die Frequenz der Störung und  $k_\xi$  die Wellenzahl in der entsprechenden Raumrichtung  $\xi = x, y$  oder  $z$  ist.

Diese Geschwindigkeiten sind nun aber nicht einfach die Projektionen des Phasengeschwindigkeitsvektors  $\vec{C}$  auf die einzelnen Raumkomponenten, vielmehr ist C gegeben durch

$$\vec{C} = (C_x, C_y, C_z) = \omega / k^2 \vec{k}$$

mit

$$k^2 = [\vec{k}]^2 = k_x^2 + k_y^2 + k_z^2$$

Um die Beziehung zwischen den auftretenden Phasengeschwindigkeiten  $\hat{C}_x$ ,  $\hat{C}_y$ ,  $\hat{C}_z$  und den Komponenten des Phasengeschwindigkeitsvektors  $C_x$ ,  $C_y$ ,  $C_z$  herzuleiten, werden die Gleichungen (3.5) mit Wichtungsfaktoren

$\alpha_x$ ,  $\alpha_y$  und  $\alpha_z$  multipliziert, wobei die Bedingung  $\alpha_x + \alpha_y + \alpha_z = 1$  erfüllt sein muß. Summiert man nun die drei Anteile auf, so ergibt sich

$$(3.7) \quad \frac{\partial \phi}{\partial t} = - \left( \alpha_x \hat{c}_x \frac{\partial \phi}{\partial x} + \alpha_y \hat{c}_y \frac{\partial \phi}{\partial y} + \alpha_z \hat{c}_z \frac{\partial \phi}{\partial z} \right)$$

Andererseits gilt aber

$$(3.8) \quad \frac{\partial \phi}{\partial t} = - \mathbf{c} \cdot \nabla \phi = - \left( c_x \frac{\partial \phi}{\partial x} + c_y \frac{\partial \phi}{\partial y} + c_z \frac{\partial \phi}{\partial z} \right)$$

Ein Vergleich ergibt

$$(3.9) \quad \begin{aligned} \alpha_x &= k_x^2 / k^2 = \left( \frac{\partial \phi}{\partial x} \right)^2 / G \\ \alpha_y &= k_y^2 / k^2 = \left( \frac{\partial \phi}{\partial y} \right)^2 / G \\ \alpha_z &= k_z^2 / k^2 = \left( \frac{\partial \phi}{\partial z} \right)^2 / G \end{aligned}$$

mit

$$(3.10) \quad G = \left( \frac{\partial \phi}{\partial x} \right)^2 + \left( \frac{\partial \phi}{\partial y} \right)^2 + \left( \frac{\partial \phi}{\partial z} \right)^2$$

Im Gegensatz zur einfachen Orlandi-Ausstrahlungsbedingung, die für einen Rand jeweils nur eine Phasengeschwindigkeit zuläßt, wird in dieser Formulierung auch die Ausbreitung von Störungen tangential zum Rand zugelassen. Insofern sind die hier errechneten Phasengeschwindigkeiten für manche Anwendungen deutlich realistischer als im Fall der Orlandi-Formulierung.

Die Diskretisierung, hier wieder zweidimensional für den rechten Rand, wird bei den Ableitungen normal zum Rand wie in (3.3) einseitig gebildet, läßt aber bei der Behandlung der tangentialen Flüsse zwei Möglichkeiten offen.

1.) Mit zentrale Differenzen, wie es in der Originalarbeit von Raymond und Kuo vorgeschlagen wird :

$$(3.11) \quad \begin{aligned} \phi_{N,j}^{\tau+1} &= \phi_{N,j}^{\tau} - \frac{\Delta t}{\Delta x} c_x \cdot (\phi_{N,j}^{\tau} - \phi_{N-1,j}^{\tau}) \\ &\quad - \frac{\Delta t}{2\Delta x} c_y \cdot (\phi_{N,j+1}^{\tau} - \phi_{N,j-1}^{\tau}) \end{aligned}$$

2.) Mit Upwind-Differenzen

$$(3.12) \quad \begin{aligned} \phi_{N,j}^{\tau+1} &= \phi_{N,j}^{\tau} - \frac{\Delta t}{\Delta x} c_x \cdot (\phi_{N,j}^{\tau} - \phi_{N-1,j}^{\tau}) \\ &\quad - \frac{\Delta t}{2\Delta x} \left[ (c_y + |c_y|)(\phi_{N,j}^{\tau} - \phi_{N,j-1}^{\tau}) + (c_y - |c_y|)(\phi_{N,j+1}^{\tau} - \phi_{N,j}^{\tau}) \right] \end{aligned}$$

Die Geschwindigkeiten ergeben sich gemäß (3.7) - (3.10) aus



$$(3.13) \quad \begin{aligned} \tilde{C}_x &= - \left( \frac{\phi_{N-1,j}^\tau - \phi_{N-1,j}^{\tau-1}}{\Delta t} \right) \left( \frac{\phi_{N-1,j}^{\tau-1} - \phi_{N-2,j}^{\tau-1}}{\Delta x} \right) / G \\ \tilde{C}_y &= - \left( \frac{\phi_{N-1,j}^\tau - \phi_{N-1,j}^{\tau-1}}{\Delta t} \right) \left( \frac{\phi_{N-1,j+1}^{\tau-1} - \phi_{N-1,j-1}^{\tau-1}}{2\Delta y} \right) / G \end{aligned}$$

mit

$$(3.14) \quad G = \left( \frac{\phi_{N-1,j}^{\tau-1} - \phi_{N-2,j}^{\tau-1}}{\Delta x} \right)^2 + \left( \frac{\phi_{N-1,j+1}^{\tau-1} - \phi_{N-1,j-1}^{\tau-1}}{2\Delta y} \right)^2$$

und den Beschränkungen

$$(3.15a) \quad C_x = \begin{cases} 0 & \tilde{C}_x < 0 \\ \tilde{C}_x & 0 \leq \tilde{C}_x \leq \Delta x / \Delta t \\ \Delta x / \Delta t & \tilde{C}_x > \Delta x / \Delta t \end{cases}$$

$$(3.15b) \quad C_y = \begin{cases} -\Delta y / \Delta t & \tilde{C}_y < -\Delta y / \Delta t \\ \tilde{C}_y & -\Delta y / \Delta t \leq \tilde{C}_y \leq \Delta y / \Delta t \\ \Delta y / \Delta t & \tilde{C}_y > \Delta y / \Delta t \end{cases}$$

Entsprechend dieser Formulierung können nun auch für die anderen Ränder Phasengeschwindigkeiten  $C_x$  und  $C_y$  berechnet werden. Wir unterscheiden die Programme ORL2D2D, wobei (3.11) benutzt wird, und ORL2DAD mit dem Ansatz (3.12):

### 3.3 Die Modifikation nach Miller und Thorpe

Miller und Thorpe(1981) haben bei der hier verwendeten Art der Diskretisierung eine Berechnung der Phasengeschwindigkeit in der Form

$$(3.16) \quad \tilde{C} = - \left( \frac{\phi_{N-1}^{\tau+1} - \phi_{N-1}^\tau}{\Delta t} \right) \left( \frac{\phi_{N-1}^\tau - \phi_{N-2}^\tau}{\Delta x} \right)^{-1},$$

benutzt, also einen Zeitpunkt später wie in (3.4). Das entspricht der Vorstellung, daß die Phasengeschwindigkeit räumlich überall etwa gleich ist, während sie in der Orlandski-Formulierung mit der maximalen Geschwindigkeit  $u = \Delta x / \Delta t$  sozusagen *mittransportiert* wird. Wir haben auch diesen Ansatz untersucht und die Modifikation der Schemata mit (3.16) anstatt (3.4) ( und entsprechend auch anstatt (3.13) ) jeweils als " Sofortversion " bezeichnet.

### 3.4 Einfache Extrapolationsformeln

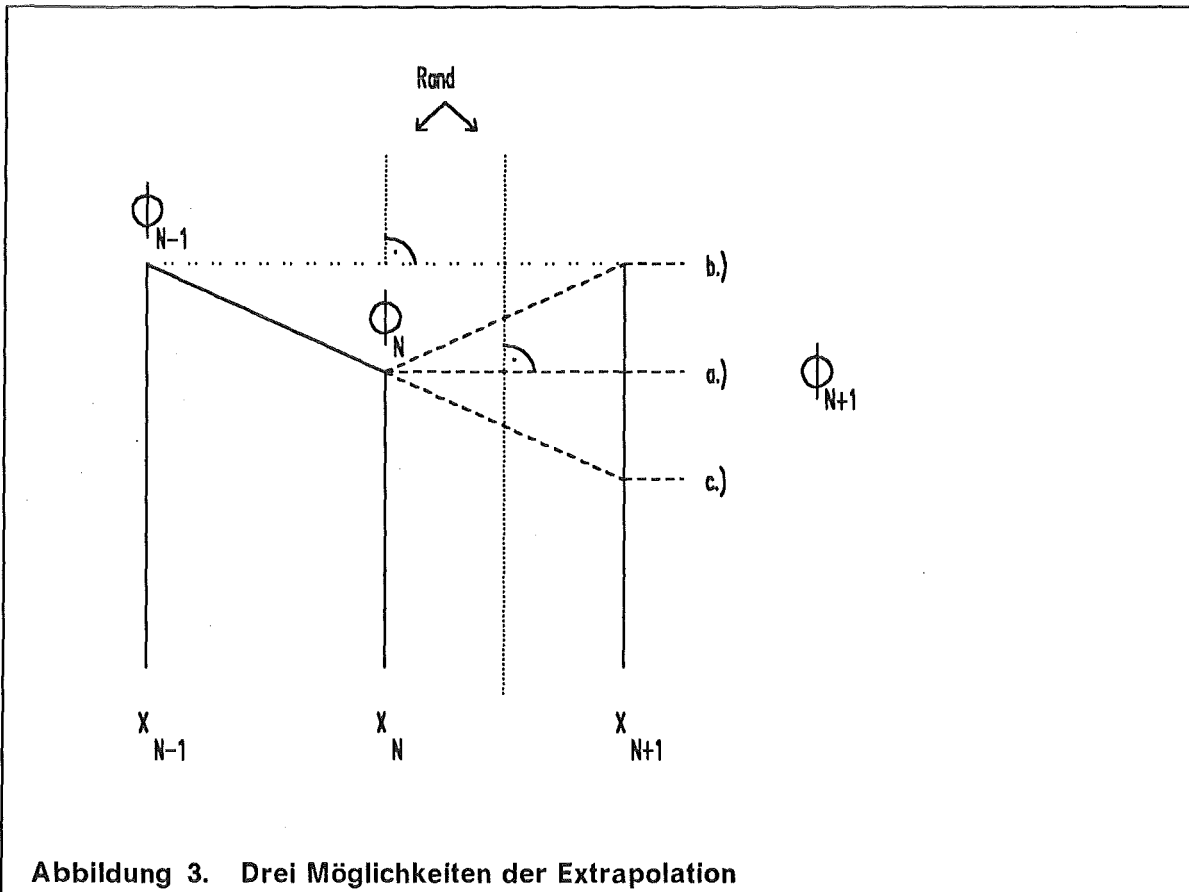
Bei der formalen Umsetzung von Diskretisierungsformeln am Rand, so etwa im ersten Term von (2.9) auf der rechten Seite für  $i = N$  (der Zeitindex wurde weggelassen)

$$\dots = \frac{1}{\Delta x^2} [D_{N+1/2,j} \cdot (\phi_{N+1,j}^n - \phi_{N,j}^n) - D_{N-1/2,j} \cdot (\phi_{N,j}^n - \phi_{N-1,j}^n)] + \dots ,$$

tauchen sogenannte *Pseudopunkte*, hier  $\phi_{N+1,j}$ , auf. Oft werden sie durch einfache Extrapolationsformeln ersetzt und gleich ins Unterprogramm mit eingearbeitet. Wir greifen drei der gebräuchlichsten Formulierungen heraus :

$$(3.17) \quad \begin{array}{l} \text{a.)} \quad \phi_{N+1,j} = \phi_{N,j} \\ \text{b.)} \quad \phi_{N+1,j} = \phi_{N-1,j} \\ \text{c.)} \quad \phi_{N+1,j} = 2 \cdot \phi_{N,j} - \phi_{N-1,j} \end{array}$$

deren eindimensionale geometrische Bedeutung aus der Skizze



ersichtlich wird. Man kann b.) und c.) als Symmetrie- bzw. Schiefsymmetrie-Bedingung für die Funktion  $\phi$  bezüglich des Randes bei  $i = N$  auffassen, Fall a.) spielt dann die gleiche Rolle wie .b), jedoch für einen versetzten Rand bei  $i = N + 1/2$ .

Direkt als Bedingung für die Flüsse (Gradienten) ergibt sich dabei :

$$(3.18) \quad \phi_{N+1,j} - \phi_{N,j} = \begin{cases} 0 & \text{Fall a.)} \\ - (\phi_{N,j} - \phi_{N-1,j}) & \text{Fall b.)} \\ + (\phi_{N,j} - \phi_{N-1,j}) & \text{Fall c.)} \end{cases}$$

Im Eindimensionalen, bei konstantem Diffusionskoeffizienten  $D$ , bleibt der Randwert für c.) damit zeitlich konstant :  $\phi_{R}^{n+1} = \phi_{R}^n$ . Dagegen ergibt sich für einen Zeitschritt mit  $D\Delta t/\Delta x^2 \approx 1/2$  im Fall a.) der maximale Transport von Information  $\phi_{R}^{n+1} \approx \phi_{R-1}^n$  . und für b.) ein gemittelter Wert  $\phi_{R}^{n+1} \approx 1/2(\phi_{R}^n + \phi_{R-1}^n)$  .

Diese einfachen Fälle wurden zum Vergleich mit in unsere Untersuchung einbezogen.

### 3.5 Programmversionen

Basierend auf diesen grundlegenden Herleitungen und Betrachtungen wurden nun Unterprogramme mit der Formulierung der Randbedingung nach Orlanski (erweitert nach Raymond und Kuo bzw. Miller und Thorpe) erstellt, für Modellgebiete verschiedener Dimension und bezüglich unterschiedlicher Diskretisierungen. Für den Test benutzten wir aber nur Versionen mit explizit vorwärts gerichteten zeitlichen Differenzen.

Erstellt wurden folgende Subroutinen:

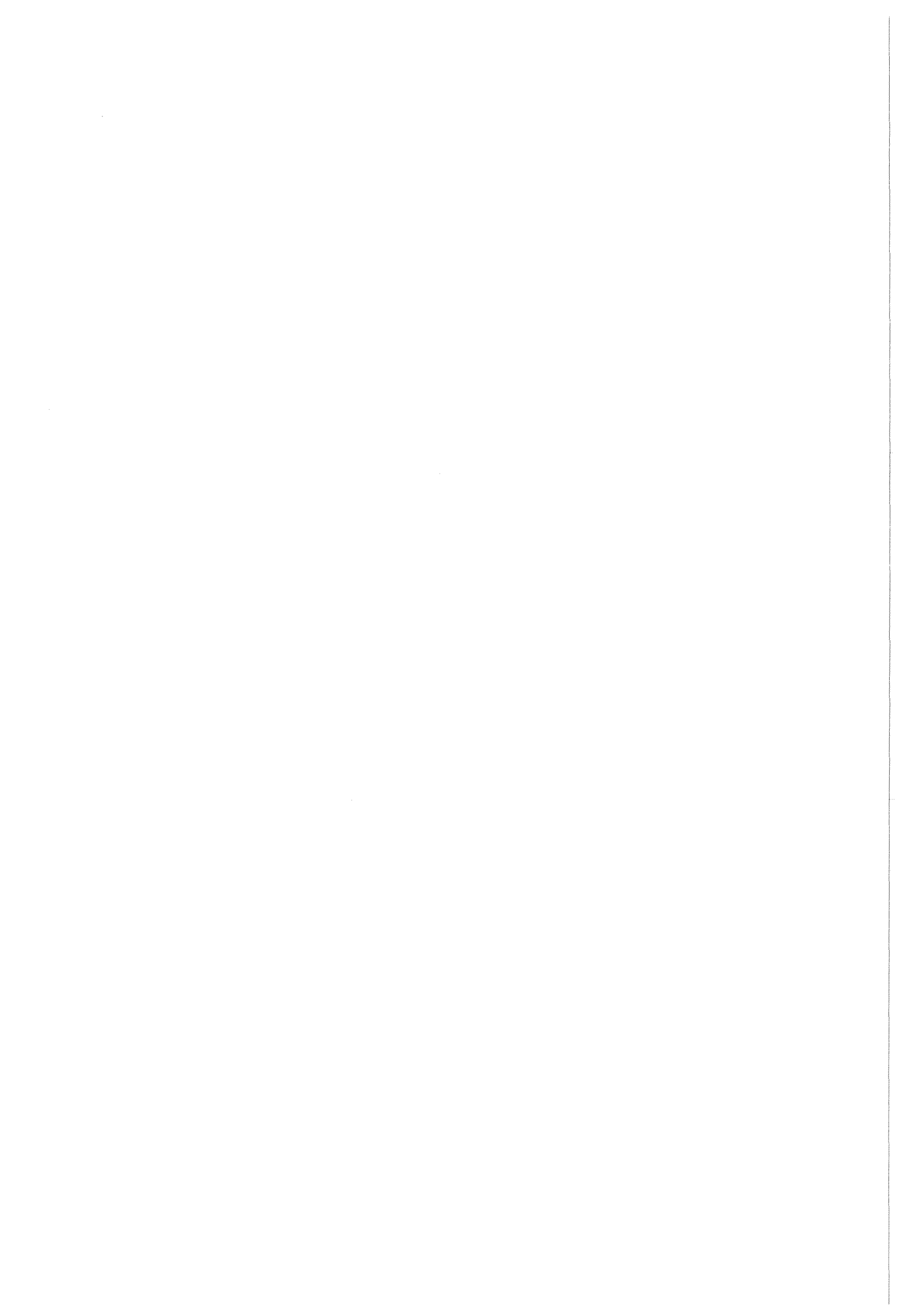
<b>ORLAN1</b>	Original Orlanski, eindimensional, Leap-Frog-Schema
<b>ORL1D2D</b>	2D-Version, jedoch nur normale Flüsse
<b>SORL1D2D</b>	2D-Version, jedoch nur normale Flüsse (Sofort -Version, siehe hierzu Beschreibung in Kapitel 3.3)
<b>ORL2D</b>	Version, mit normalen und tangentialen Flüssen in X-Richtung, und einfachen Randbedingungen in Y-Richtung
<b>ORL2D2D</b>	komplette Version, auch tangential
<b>SORL2D2D</b>	komplette 2D-Version, auch tangentiale Flüsse (Sofort -Version, siehe hierzu Beschreibung in Kapitel 3.3)
<b>ORL2DAD</b>	komplette Version, andere Formulierung der tangentialen Flüssen
<b>ORL2DDS</b>	wie ORL2DAD, aber Sofort-Version

Für die Extrapolationsformeln aus Abschnitt 3.4 liegen keine eigenen Unterprogramme vor, vielmehr wurden diese über die Flüsse (3.18) direkt mit dem Diskretisierungsoperator (2.9) verwendet.

Da diese Unterprogramme auch im mesoskaligen atmosphärischen Strömungsmodell " KAMM " des IMK verwendet werden sollen, existiert auch eine Erweiterung auf drei Raumrichtungen :

<b>ORL3D3D</b>	komplette Version, auch tangential in drei Raumrichtungen, mit variabler vertikaler Raumschrittweite
----------------	--

Die wichtigsten Vertreter, unter anderem auch eine Originalversion nach Orlandi (ORLAN1) mit dem ursprünglichen Leap-Frog-Schema, sind in Anhang A festgehalten. Dort sind einzelne Subroutinen in Form von einleitenden Kommentarkarten ausführlich beschrieben, der Bezug zu diesem Kapitel dokumentiert und die Zugriffsmöglichkeiten erklärt. Die Programme liegen auf geschützten Dateien der beiden Autoren, können aber auf Anfrage zugänglich gemacht werden.



## Kapitel 4. Die Beurteilung der Tests

Es wurden eine ganze Reihe von Testläufen mit verschiedenen Anfangswerten, Operatoren, Modellen gestartet. Die Anfangswerte sollten den Charakter einer Störung haben, daher haben wir mit geometrisch einfachen Konfigurationen und verschiedenen Lagen im Gitter experimentiert. Bei den Operatoren interessierten uns etwaige Unterschiede zwischen expliziten und impliziten Verfahren, zwischen vollen 2D-Diskretisierungen und Raumsplitting. Bei der Advektion wurden auch nichtlineare Modellprobleme untersucht und dafür aufwendige Operatoren eingesetzt. Bei der Diffusion galt unsere Aufmerksamkeit zusätzlich dem Abklingverhalten für lange Simultionszeiten. Schließlich bot sich auch noch das Spiel mit Modellparametern wie Dimensionierungen, Maschenweiten, Diffusionskoeffizienten an. All diese Variationen ließen aber keine prinzipiellen Unterschiede im Verhalten am Rand erkennen. Wir beschränken uns daher bei unserer Beschreibung auf je zwei (typische) Testreihen zu den einfachsten linearen Fällen auf kleinen Gittern mit überschaubaren Lösungen.

Die Rechnungen wurden gleichzeitig auf zwei verschieden großen Gittern durchgeführt. Dabei bildete zu Beginn die Belegung der Daten des kleinen Gitters gerade ein Ausschnitt der Belegung des großen Gitters. Auf beiden Gittern wurde nun mit derselben Diskretisierung operiert, die Daten am Rand des kleinen Gitters wurden aber zusätzlich der Behandlung durch die Randformulierungen unterworfen. Die davon verschonten Daten des größeren Gitters dienten so als Referenz. Es wurde nur solange gerechnet, wie sichergestellt war, daß etwaig auftretende Störungen am Rand des großen Gitters noch nicht den Vergleichsbereich erreichten. Nähere Einzelheiten sind unter den Schlagworten zusammengefaßt oder weiter unten erläutert.

Für die Beurteilung der Programmergebnisse stand uns neben den Druckerausgaben in Form von Tabellen, Druckplots und Druckisoplots auch die graphische Ausgabe in Form von Schnitten, Isolinien und 3D-Darstellungen zur Verfügung. Wir haben uns bemüht, zur Beurteilung jeweils den geeignetsten Typ auszuwählen. In der Regel ist es wenig sinnvoll, quantifizierte Aussagen zu machen, da ja doch alles mehr oder weniger eine Frage der Skalierung ist. Deshalb dokumentieren wir in der Hauptsache mit Isolinienplots den qualitativen Verlauf und ergänzen dort, wo Schwierigkeiten in der Vorstellung auftreten könnten, mit einfachen 3D-Plots. Die Bilder sind in Anhang B zusammengefaßt.

Hier sind listenartig die wichtigsten Punkte zusammengefaßt :

<b>Rechner</b>	Alle unsere Test liefen auf der IBM 3090 des HDI
<b>Gitter</b>	Das große Gitter bestand aus 39 x 39 Punkten, das kleine Gitter aus 11 x 11 Punkten, die Raumschrittweite wurde mit $\Delta x = \Delta y = 1$ festgelegt.
<b>Zeitschritt</b>	Für dessen Festlegung wurde im Fall der Advektion eine zweidimensionale Entsprechung des Stabilitätskriteriums (2.5') gewählt, bei der Diffusion der maximale Wert aus (2.10). Diese obere Schranke für den Zeitschritt wurde noch mit dem Faktor $\sigma = 0.9$ multipliziert, damit Störungen auch mit einer numerischen Dämpfung belegt werden, so daß nicht nur die Anfachung unterbunden, sondern auch das reine Schwingen einer Störung, wie man es bei $\sigma \approx 1$ beobachtet, verhindert wird. Dies gilt so

nicht für die Advektion, wo bei genauer Erfüllung des CFL-Kriteriums natürlich die besten Resultate erzielt werden, zum besseren Vergleich wurde aber auch hier dieser Wert beibehalten. Die Zahl der Zeitschritte wurde auf 30 begrenzt um den Einfluß vom Rand des großen Gitters zurücklaufender Wellen auszuschließen.

<b>Testversionen</b>	Advektion : verschiedene Geschwindigkeiten Diffusion : verschiedene Anfangslagen
<b>Anfangswerte</b>	Das geringe Auflösungsvermögen des gewählten Gitters gestattete nur einfachste Formen. Wir wählten deshalb den Kasten. Für die Diffusion ergaben sich auch damit noch Variationsmöglichkeiten bezüglich der Stellung zum Rand.
<b>Graphiksoftware</b>	Es wurde ausschließlich eigene Software auf GKS-Basis benutzt.
<b>Darstellungen</b>	Bei den Isolinienbildern stellt die linke obere Ecke den Koordinatenursprung (Gitterpunkt (1,1)) dar, die X-Achse geht von oben nach unten, die Y-Achse von links nach rechts. Für die 3D-Darstellungen liegt der diagonale Punkt (11,11) unmittelbar vor dem Betrachter.
<b>Isolinien</b>	Üblicherweise wurden die Isolinienwerte äquidistant zwischen dem minimalen und dem maximalen Wert gewählt. Wenn davon zur Verdeutlichung abgewichen wurde, so ist das im Text unten entsprechend vermerkt.

## 4.1 Die Ergebnisse des Advektionsproblems

Wir führten die beschriebenen Tests mit der Modellgleichung (2.1) und den linearen Funktionen  $f = u \cdot \phi$ ,  $g = v \cdot \phi$  mit zwei verschiedenen konstanten Geschwindigkeitsvektoren  $\vec{v} = (u, v)$  durch. Mit  $\vec{v} = (1,1)$  wurden die Geschwindigkeitskomponenten einmal so gewählt, daß die Anfangskonfiguration genau auf die Ecke zulief, das andere Mal, für  $\vec{v} = (2,1)$ , schräg auf eine Seite.

Zur Diskretisierung wurde das in Kapitel 2.1 beschriebene Raumsplittingverfahren benutzt. Jeweils zu Beginn wurden die Funktionswerte an den 9 Gitterpunkten im Zentrum auf 50 gesetzt, an allen anderen Punkten wurde 5 vorgeschrieben. Diese absoluten Werte spielen aber im folgenden eigentlich keine Rolle mehr. Abbildung 4 auf Seite 45 zeigt das zugehörige Isolinienbild.

Im Vergleich mit dem größeren Gitter gibt es bereits Unterschiede beim Einsatz des Operators allein. Dennoch läuft der vorgegebene Kasten relativ glatt aus dem Bild, wie Abbildung 5 und Abbildung 6 belegen. Man beachte, trotz Raumsplitting, die gute Erhaltung der Symmetrie.

Das beste Ergebnis mit Randbehandlung erzielte, erwartungsgemäß, ORL1D2D. Hier ist ja genau das Upstream-Differencing eingebaut, wie es auch im Advektionsoperator am Rand benutzt wird. Abbildung 7 zeigt dies deutlich, drei Zeitschritte später sind alle Reste des Kastenprofils gänzlich verschwunden. Schwierigkeiten hat dagegen das Verfahren ORL2D2D. Sind die Ergebnisse im 8. Schritt (Abbildung 8) vergleichbar mit Abbildung 5, so öffnen sich danach (Abbildung 9), wohl auf Grund einer zu großen Phasengeschwindigkeit tangential zum Rand, Löcher im Grundniveau, wie man am besten in der 3D-Darstellung (Abbildung 10) erkennt. Diese bleiben, auch wenn das ganze Kastenprofil weg-

transportiert ist, für immer auf dem Rand liegen (Abbildung 9) . Ähnlich, aber noch ausgeprägter, ergeht es dem Verfahren mit anderer Formulierung der tangentialen Flüsse, ORL2DAD. Was recht harmlos beginnt (Abbildung 12 und Abbildung 13), schaukelt sich mächtig auf (Abbildung 14 und Abbildung 15). Die "Sofort"-Version ORL2DADS vermeidet zwar die Lochbildung (Abbildung 16 , Abbildung 17) und schafft es aber nicht, alles abzutransportieren (Abbildung 18 , Abbildung 19). So bleibt ein Rest, etwa 8 Prozent über dem Grundniveau, auf dem Rand liegen.

Auch eine Variante dieser Testreihe, mit den Geschwindigkeitskomponenten  $u = 2, v = 1$ , bestätigt im Grunde die Ergebnisse. Das Profil wird jetzt in einem steileren Winkel über den für den Betrachter unteren Rand geschoben. Dagegen streifen die durch die numerische Dämpfung verschmierten Ausläufer den rechten Rand in einem flachen Winkel.

Für diesen Fall zeigen Abbildung 21 , Abbildung 22 das Verhalten des Advektionsoperators ohne irgendwelche gesetzten Randbedingungen. Das Kastenprofil hat im 8. Schritt bei der gewählten Zeitschrittweite und den vorgegebenen Geschwindigkeitskomponenten gerade zu einem Drittel den Rand überschritten. Auch hier schneidet ORL1D2D wieder gut ab (Abbildung 23). Bemerkenswert ist allerdings das Umlenken der Daten um die rechte untere Ecke herum im 12. Schritt (Abbildung 24). ORL2D2D zeigt wieder die typischen Löcher (Abbildung 25, Abbildung 26, Abbildung 27), ORL2DAD schaukelt sich auf (Abbildung 28 bis Abbildung 30), und ORL2DADS schafft es wieder nicht, alles wegzuräumen (Abbildung 31, Abbildung 32, Abbildung 33).

Insgesamt ergab sich der Eindruck, daß durch zusätzlich gesetzte Randbedingungen das Ergebnis eher verschlechtert wurde.

## **4.2 Die Ergebnisse des Diffusionsproblems**

Die Modellgleichung dieser Testserien ist durch (2.7), (2.8) mit  $D = E = 20$  gegeben. Die explizite Diskretisierung wird dabei in allen Fällen wie in (2.9) nur auf die inneren Punkte des Gebiets angewandt. Der Rand wird dann durch die beschriebenen Unterprogramme gesetzt oder wie in Kapitel 3.4 behandelt. Wir wählen die gleichen Anfangsdaten wie bei der Advektion und benutzen dabei zwei verschiedene Ausgangspositionen für den Kasten : a.) in der Mitte des Gitters zentriert und b.) um eine Maschendiagonale davon verschoben.

Das Ausgangsprofil für a.) ist wieder durch Abbildung 4 gegeben. Die analytische Lösung läßt, soweit es die grobe Auflösung zuläßt, konzentrische Kreise um die zentrale Masche erwarten. Das leistet der diskrete Diffusionsoperator auch ganz schön, wie die Ausschnitte aus dem großen Gitter (Abbildung 34, Abbildung 35, Abbildung 36) belegen.

Wir beginnen bei Abbildung 37 mit der einfachen Extrapolationsformel (3.17.a). Bis zum 10. Schritt, wenn das diffundierende Medium am Rand angelangt ist, sind die Erwartungen einigermaßen erfüllt. Abbildung 38 und Abbildung 40 zeigen aber ein zu starkes Abfließen normal über den Rand. Die 3D-Darstellung (Abbildung 39) verdeutlicht, wie das Medium fast ausschließlich längs der Rücken in den Hauptrichtungen abtransportiert wird.



Ähnlich ist es bei Formel (3.17.b), wenngleich nicht ganz so ausgeprägt. Das verzögerte Verhalten dieser Randbedingung, wie es sich auch schon theoretisch ergibt, ist allerdings nur schwer aus den Bildern (Abbildung 41 bis Abbildung 43) an den etwas höheren Rücken in Richtung der Hauptsymmetrieachsen zu erkennen.

In Abbildung 44 sind für die Extrapolationsformel (3.17.c) die Randbereiche stärker aufgelöst. Das diffundierende Medium hat hier im 10.Schritt gerade den Rand erreicht und man kann gut den Übergang vom ungestörten Innern zum gestörten Rand erkennen. Die Formel überschätzt dagegen den Anteil schräg anlaufender Beiträge und drückt oder zieht daher die Isolinien stärker in die Ecken (Abbildung 45, Abbildung 46).

Die gleichen Schwierigkeiten hat ORL1D2D, wie Abbildung 47 bis Abbildung 49 zeigen. Auch hier wirkt sich die rein normale Randbehandlung wieder negativ aus.

Etwas runder dagegen stellt sich ORL2D2D dar (Abbildung 50). Eine Tendenz zur Ausbeulung ist aber auch hier gegeben, wie man an Abbildung 51 und Abbildung 52 sehen kann. Nur der scharfe Vergleich zwischen Abbildung 49 auf Seite 69 , Abbildung 52 auf Seite 71 und Abbildung 36 auf Seite 63 zeigt, daß ORL2D2D bezüglich der zeitlichen Entwicklung zwar nicht optimal, aber gegenüber ORL1D2D doch eine Spur besser abschneidet.

Dazwischen noch scheint das Ergebnis zu liegen, wenn man die Randbedingung ORL2D2D mit einem **impliziten** Diffusionsoperator kombiniert (Abbildung 55). Allerdings läßt Abbildung 53 im direkten Unterschied zu Abbildung 34 darauf schließen, daß das implizite Verfahren prinzipiell ein langsames (und vermutlich richtigeres) Fortschreiten des Diffusionsprozesses beschreibt.

Die Sofortversion SORL2D2D zeigt ähnliche Resultate (Abbildung 56, Abbildung 57), jedoch werden die vier Eckpunkte zu stark abgesenkt.

Auch ORL2DAD und ORL2DADS liegen noch ganz gut im Rennen. Beide erhalten zunächst die Radialsymmetrie (Abbildung 58 , Abbildung 61), passen sich dann aber doch der eckigen Gittergeometrie mehr an (Abbildung 59 , Abbildung 62), wie die höhere Auflösung der Isolinien am Rand verdeutlicht. Qualitativ unterscheiden sich beide Verfahren später (Abbildung 60 , Abbildung 63) kaum noch von den anderen, es sind lediglich graduelle Unterschiede beim Fluß über den Rand auszumachen.

Allgemein findet man bei den Orlanski-Typ-Verfahren eine Neigung zur Ausrichtung der Isolinien tangential zu den Rändern, die sicher nicht auf den Diskretisierungsoperator zurückzuführen ist.

Die Tests mit Ausgangsbedingung b.) (Abbildung 64) bestätigen diese Einschätzung, unserer Meinung nach, sehr eindrucksvoll. Zur Orientierung geben wir auch hier zunächst die Ausschnitte aus dem großen Gitter in Abbildung 66 bis Abbildung 68 wieder.

Die drei Extrapolationsmethoden (3.17) schneiden nun sehr schlecht ab. Wieder zeigen (3.17.a), (3.17.b) ganz ähnliches Verhalten, die Isolinien werden sekrecht zum Rand gebogen. (Abbildung 69 - Abbildung 74). Bei (3.17.c) ist extrem ausgeprägt, was alle anderen Verfahren auch zeigen : die innere Isolinie ist zusam-

mengesetzt aus zwei Viertelkreisbögen mit verschiedenen Radien und zwei Ellipsenbögen und je nach dem Verhältnis dieser Radien ist das ursprüngliche Zentrum zur Mitte des Gitters hin verschoben. In Abbildung 76 ist das sehr ungünstige Verhältnis deutlich zu erkennen, in Abbildung 75 ist als Ursache dafür ein Stau am Rand auszumachen.

ORL1D2D (Abbildung 78) und ORL2D2D (Abbildung 80) verhalten sich innen gleich, die Radien sind nahezu identisch. Außen jedoch biegen sich die Isolinien bei ORL1D2D langsam auf, stellen sich normal zum Rand.

Die implizite Version mit ORL2D2D (Abbildung 81 , Abbildung 82) unterscheidet sich dagegen wenig vom Original.

Deutlich verbeulter als ORL2D2D ist die zugehörige Sofortversion (Abbildung 83). Auch hier hat SORL2D2D wieder Probleme an den Ecken, siehe Abbildung 84.

Den gleichen Gesamteindruck vermitteln auch ORL2DAD (Abbildung 85 , Abbildung 86) und ORL2DADS (Abbildung 87 , Abbildung 88). Neben den Deformationen treten hier Wellenbewegungen entlang des Randes in Erscheinung, die auf den Isolinienbildern als Einbuchtungen zu erkennen sind.

### **4.3 Zusammenfassung und Ausblick**

Die vorgestellten Verfahren haben sich insoweit als durchaus tauglich erwiesen, da Reflexionen am Rand, die als Störung ins Modellgebiet zurücklaufen, vermieden werden konnten. Schwächen zeigten sich erst bei einer relativen groben Auflösung, wobei größere Unterschiede bei der Behandlung der verschiedenen Gleichungstypen auftraten. Es gibt eben kein Verfahren für alle Fälle.

Unsere Empfehlung lautet daher, zur Lösung der Advektion am Rand ein einfaches Upstream-Schema normal zum Rand zu verwenden, wie es durch ORL1D2D gegeben ist oder bei unserem Operator schon mit enthalten ist. Bei der vorgestellten Behandlung der Diffusionsgleichung sind die tangentialen Flüsse am Rand nicht vernachlässigbar. Hier ist ein Verfahren wie ORL2D2D dringend anzuraten, um einen stärkeren Einfluß des Randes auf das Gebietsinnere zu vermeiden.

Das Raumsplitting-Konzept hat sich bewährt und sollte je nach den gegebenen Möglichkeiten, schon aus Gründen des geringeren Aufwands, eingesetzt werden.

Es liegen keine Zeitmessungen vor, da durch eine Reihe mitlaufender Diagnose-tools und Output-Operationen ein schiefes Bild entstehen würde. Außerdem wurde mehr Wert auf eine einheitliche Behandlung aller Verfahren gelegt als auf einen optimalen Code. Sicher ist jedoch der Zeitaufwand wegen der lokalen Bestimmung des Phasenflusses und dem lokalen Setzen der Daten (nur) proportional zur Anzahl der Randpunkte. Für drei Dimensionen erscheint uns da der Speicherbedarf kritischer. Man benötigt 6x3 ständig belegte 2D-Felder für die Phasenflüsse und selbst in einem Modell wie KAMM mit anderen Randbedingungen am Boden (und oben ?) entspricht dies etwa einem halben zusätzlichen 3D-Feld. Hier könnte die Sofortversion einen schwachen Kompromiß darstellen, weil dazu

nur kurzfristig Hilfsfelder benötigt würden, die in jedem größeren Code zur Verfügung stehen. Offen dabei ist allerdings noch das Problem der Eckenbehandlung.

Wer es genauer haben will, muß doch übergehen zu Randformulierungen, die aus den vorliegenden Differentialgleichungen abgeleitet werden müssen. Hierbei sind vor allem die Methoden zu erwähnen, die sich durch den Formalismus der Pseudodifferentialoperatoren und Fourier-Integral-Operatoren ergeben (Treves,1980). Lindman(1975) leitete eine entsprechende Formulierung aus einer Dispersionsrelation für die Wellengleichung ab und entwickelte ebenso wie Engquist und Osher(1977) mit Hilfe von Taylorentwicklungen daraus brauchbare Randbedingungen. Kröner(1990) zeigte am Beispiel der Euler-Gleichungen die Qualitäten der Methode auch für Systeme. Wagatha(1983) verwies auf mögliche Stabilitätsprobleme bei dieser Vorgehensweise, die aus Entstehung nichtsachgemäß gestellter Probleme resultieren, und entwickelte stabile und automatisierende Algorithmen, teilweise auf der Basis von Least-Square-Approximationen.

# Anhang A. Beschreibung der Unterprogramme

## A.1 ORLAN1

```

  0      R R      L      A A A      N      N      1
  0 0    R R R    L      A A A      N N N    N    1 1
  0 0    R R      L      A A A      N N N    1    1
  0 0    R R R    L      A A A      N N N    1    1
  0      R      R    L L L    A A      N      N      1

```

SUBROUTINE ORLAN1(V,VN,VNN,N,DT,DX,CL,CR)

```

*****
*
* Dieses Programm realisiert die Idee der Ausstrahlungsrandbedingung
* nach Sommerfeld mit dem Leap-Frog-Schema, entsprechend der Arbeit
* -----
* von Orlandi, (Journal of Comp. Phys. 21, 251-269 (1976))
*
* 1 - dimensionale Realisierung
*
*****
*
* Phasengeschwindigkeiten: CL = C am l-inken Rand: normal
*                          : CR = C am r-echten Rand: normal
*
* das Leap-frog-Schema ist zentriert zum Zeitpunkt tn-1
* V(N) ist das skalare Feld zum Zeitpunkt tn-2
* VN(N) ist das skalare Feld zum Zeitpunkt tn-1
* VNN(N) ist das skalare Feld zum Zeitpunkt tn
*****

```

```

REAL      V(N),VN(N),VNN(N)
REAL      DT,DX
REAL      CL,CR

```

lokale Variablen

```

REAL      DTRDX,DXRDT
REAL      EPS
LOGICAL   CSTART
SAVE      CSTART
DATA      CSTART/.TRUE./

```

Die Leap-Frog-Diskretisierung der eindimensionalen Transportgleichung:

$$\frac{dV}{dt} + C \frac{dV}{dx} = 0 \quad \text{zum Zeitpunkt } t^{n-1} \quad \text{am Ort } x_i$$

lautet:

$$\frac{V_i^n - V_i^{n-2}}{t^n - t^{n-2}} + C \frac{V_{i+1}^{n-1} - V_{i-1}^{n-1}}{2 DX} = 0 ,$$

wobei am Rand die zentrale räumliche Diskretisierung durch einseitige Differenzen ersetzt werden. Da in einem Eliasian Grid zu verschiedenen Zeitpunkten die Punkte im Raum auch unterschiedlich sitzen, ist eine (zeitliche) Mittelung erforderlich:

$$\frac{V_1^n - V_1^{n-2}}{DT^n + DT^{n-1}} + C_1^{n-1} \frac{V_2^{n-1} - \frac{1}{2} (V_1^n + V_1^{n-2})}{DX} = 0$$

bzw.

$$\frac{V_N^n - V_N^{n-2}}{DT^n + DT^{n-1}} + C_N^{n-1} \frac{\frac{1}{2} (V_N^n + V_N^{n-2}) - V_{N-1}^{n-1}}{DX} = 0$$

Wir benutzen diese Gleichungen nun zweifach, zum einen, um die Phasengeschwindigkeiten in der Nähe der Ränder zu schätzen:

$$C_2^{n-1} = \left\{ \begin{array}{l} - \frac{DX}{DT^{n-1}} \\ \frac{*n-1}{C_2} \end{array} \right. \text{ für } - \frac{DX}{DT^{n-1}} < \frac{*n-1}{C_2} < 0$$

$$0 < \frac{*n-1}{C_2}$$

mit:

$$\frac{*n-1}{C_2} = - \frac{DX}{DT^{n-1} + DT^n} * \frac{V_2^n - V_2^{n-2}}{V_3^{n-1} - \frac{1}{2} (V_2^n + V_2^{n-2})}$$

und:

$$C_{N-1}^{n-1} = \left\{ \begin{array}{l} \frac{DX}{DT^{n-1}} \\ \frac{*n-1}{C_{N-1}} \end{array} \right. \text{ für } \frac{DX}{DT^{n-1}} > \frac{*n-1}{C_{N-1}} > 0$$

$$0 > \frac{*n-1}{C_{N-1}}$$

mit:

$$C_{N-1}^{n-1} = \frac{DX}{DT^{n-1} + DT^n} * \frac{V_{N-1}^n - V_{N-1}^{n-2}}{2 \left( \frac{1}{V_{N-1}^n + V_{N-1}^{n-2}} \right) - V_{N-2}^{n-1}}$$

zum ändern, um sie im nächsten Zeitschritt (n --> n+1) dann für die Berechnung der Randpunkte zu benutzen: (Zeitschritt DT bleibt konstant)

$$V_1^{n+1} = \frac{1 + C_1^n \frac{DT^n}{DX}}{1 - C_1^n \frac{DT^n}{DX}} V_1^{n-1} - \frac{2 C_1^n \frac{DT^n}{DX}}{1 - C_1^n \frac{DT^n}{DX}} V_2^n$$

mit:  $C_1^n = C_2^{n-1}$  und

$$V_N^{n+1} = \frac{1 - C_N^n \frac{DT^n}{DX}}{1 + C_N^n \frac{DT^n}{DX}} V_N^{n-1} + \frac{2 C_N^n \frac{DT^n}{DX}}{1 + C_N^n \frac{DT^n}{DX}} V_{N-1}^n$$

mit:  $C_N^n = C_{N-1}^{n-1}$

Das vollständige Programm steht auf der Datei IMK965.FCTTOM.FORT unter dem Membernamen ORIAN1 !

## A.2 ORL1D2D

```

O      R R      L      1      D      2      D
O O    R R      L      1 1    D D      2      D D
O  O   R R      L      1 1    D D      2      D D
O  O   R  R    L      1      D D      2      D D
O      R  R    L L L    1      D      2 2 2    D

```

SUBROUTINE ORL2D(SALT,S,IX,IY,DT,DX,DY,CXL,CXR,CYV,CYH)

```

*****
*
* Dieses Programm realisiert die Idee der Ausstrahlungsrandbedingung
* nach Sommerfeld mit dem Explizit - Euler - Schema, in Anlehnung
* an die Arbeit von Miller, M.J. und Thorpe, A. J.
* (Quart. J. R. Met. Soc., 107, 615-628 (1981))
*
*
* Betrachtung erfolgt jeweils nur eindimensional -
* (d.h. normal zum Rand hin) -
* Phasengeschwindigkeiten werden aber
* für beide Raumrichtungen (x,y) gelöst
*
*
*****

```

```

REAL      SALT(IX,IY),S(IX,IY)
REAL      DT,DX,DY
REAL      CXL(IY),CXR(IY)
REAL      CYV(IX),CYH(IX)

```

CXL, CXR, CYV, CYH sind jeweils die Phasengeschwindigkeiten C normal zu den Rändern (L - inks, R - echts, V - orne, H - inten) in der jeweiligen Raumrichtung ( x, y )

SALT ist das skalare Feld zum alten Zeitpunkt,  
S ist das skalare Feld zum neuen Zeitpunkt,

Die Vorwärts-Euler-Diskretisierung einer  
-----  
eindimensionalen Transportgleichung:  
-----

$$\frac{dS}{dT} + C \frac{dS}{dX} = 0 \quad \text{zum Zeitpunkt } t^{n-1} \quad \text{am Ort } x_i$$

lautet:  
-----

$$\frac{S_i^n - S_i^{n-1}}{t^n - t^{n-1}} + C_i \frac{S_{i+1}^{n-1} - S_{i-1}^{n-1}}{2 DX} = 0 \quad ,$$

wobei am Rand die zentrale räumliche Diskretisierung durch einseitige Differenzen ersetzt werden muß.

Wir benutzen diese Gleichung nun zweifach, einmal um die Phasengeschwindigkeiten in der Nähe der Ränder zu schätzen:

linker Rand:

$$\begin{array}{l}
 \left. \begin{array}{l}
 \frac{DX}{DT} \\
 \frac{C}{2}
 \end{array} \right\} = \left\{ \begin{array}{l}
 \frac{DX}{DT} \quad \text{für} \quad \frac{C}{2} < \frac{DX}{DT} \\
 \frac{C}{2} \quad \text{für} \quad \frac{DX}{DT} < \frac{C}{2} < 0 \\
 0 \quad \text{für} \quad 0 < \frac{C}{2}
 \end{array} \right.
 \end{array}$$

mit:

$$\frac{C}{2} = \frac{DX}{DT} \cdot \frac{S_{n-1} - S_{n-2}}{S_{n-2} - S_{n-3}}$$

rechter Rand:

$$\begin{array}{l}
 \left. \begin{array}{l}
 \frac{DX}{DT} \\
 \frac{C}{N-1}
 \end{array} \right\} = \left\{ \begin{array}{l}
 \frac{DX}{DT} \quad \text{für} \quad \frac{C}{N-1} > \frac{DX}{DT} \\
 \frac{C}{N-1} \quad \text{für} \quad \frac{DX}{DT} > \frac{C}{N-1} > 0 \\
 0 \quad \text{für} \quad 0 > \frac{C}{N-1}
 \end{array} \right.
 \end{array}$$

mit:

$$\frac{C}{N-1} = \frac{DX}{DT} \cdot \frac{S_{N-1} - S_{N-2}}{S_{N-2} - S_{N-3}}$$

(analog läuft dies dann auch für den y - Rand!)

zum anderen dann, um sie im nächsten Zeitschritt ( n-1 --> n ) für die Berechnung der Randpunkte zu benutzen:



$$S_1^n = S_1^{n-1} - \frac{DT}{DX} * C_1^n * (S_2^{n-1} - S_1^{n-1})$$

mit:  $C_1^n = C_2^{n-1}$  und

$$S_N^n = S_N^{n-1} - \frac{DT}{DX} * C_N^n * (S_N^{n-1} - S_{N-1}^{n-1})$$

mit:  $C_N^n = C_{N-1}^{n-1}$

Diese Version findet sich auf der Datei IMK965.FCTTOM.FORT unter dem Membernahmen ORL1D2D!

#### Achtung:

-----  
 diese Version ist nicht original Miller und Thorpe!  
 In ihrer Originalversion werden die Phasengeschwindigkeiten berechnet und dann noch im selben Zeitschritt verwendet, hier werden jedoch erst die Ränder belegt, dann die Phasengeschwindigkeiten berechnet und diese dann erst im nächsten Zeitschritt zur Neuberechnung der Ränder verwendet.

Durch einfaches Umstellen des Ablaufs (d.h. erst Phasengeschwindigkeiten nach dem beschriebenen Schema berechnen und dann erst Ränder neu belegen) läßt sich diese Version aber ohne weiteren programmiertechnischen Aufwand in die Miller/Thorpe Originalversion überführen.

Diese Variante steht im Member SORL1D2D auf derselben Datei wie oben angegeben.

### A.3 ORL2D2D

```

  0      R R      L      2      D      2      D
  0 0    R R      L      2      D D      2      D D
  0 0    R R      L      2      D D      2      D D
  0 0    R R      L      2      D D      2      D D
  0      R R      L L L    2 2 2    D      2 2 2    D

```

```

SUBROUTINE ORL2D(SALT,S,IX,IY,DT,DX,DY,CXL,CYL,CXR,CYR,
&                CXV,CYV,CXH,CYH)

```

```

*****
*
* Dieses Programm realisiert die Idee der Ausstrahlungsrandbedingung
* nach Sommerfeld mit dem Explizit - Euler - Schema, in Anlehnung
* an die Arbeit von Miller, M.J. und Thorpe, A. J.
* (Quart. J. R. Met. Soc., 107, 615-628 (1981))
*
*
* Betrachtung erfolgt hier zweidimensional -
* Phasengeschwindigkeiten werden nun sowohl normal als auch
* tangential für beide Raumrichtungen (x,y) gelöst
*
*
*****

```

```

REAL      SALT(IX,IY),S(IX,IY)
REAL      DT,DX,DY

```

CXL,CXR,CYV,CYH sind die normalen Phasengeschwindigkeiten

```

REAL      CXL(IY),CXR(IY)
REAL      CYV(IX),CYH(IX)

```

CYL,CYR,CXV,CYH sind die tangentialen Phasengeschwindigkeiten

```

REAL      CYL(IY),CYR(IY)
REAL      CXV(IX),CXH(IX)

```

L steht für L-inks  
R steht für R-echts  
V steht für V-orne  
H steht für H-inten

SALT ist das skalare Feld zum alten Zeitpunkt,  
S ist das skalare Feld zum neuen Zeitpunkt,

Die Vorwärts-Euler-Diskretisierung einer  
-----  
zweidimensionalen Transportgleichung:  
-----

$$\frac{dS}{dt} + CX \frac{dS}{dX} + CY \frac{dS}{dY} = 0 \quad \text{zum Zeitpunkt } t^{n-1} \quad \text{am Ort } x_{i,j}$$

lautet:

$$\frac{s_{i,j}^n - s_{i,j}^{n-1}}{t^n - t^{n-1}} + c_{x,i,j}^{n-1} \frac{s_{i+1,j}^{n-1} - s_{i-1,j}^{n-1}}{2 \Delta x} + c_{y,i,j}^{n-1} \frac{s_{i,j+1}^{n-1} - s_{i,j-1}^{n-1}}{2 \Delta y} = 0$$

wobei am Rand die zentrale räumliche Diskretisierung in einer Raumrichtung und in den Eckpunkten in beiden Raumrichtungen durch einseitige Differenzen ersetzt werden muß.

Wir benutzen diese Gleichung nun zweifach:

1. Um die Phasengeschwindigkeiten in der Nähe der Ränder zu schätzen:

linker Rand:

$$c_{x,2,j}^{n-1} = \begin{cases} -\frac{\Delta x}{\Delta t} & \text{für } c_{x,2,j}^{*n-1} < -\frac{\Delta x}{\Delta t} \\ c_{x,2,j}^{*n-1} & \text{für } -\frac{\Delta x}{\Delta t} < c_{x,2,j}^{*n-1} < 0 \\ 0 & \text{für } 0 < c_{x,2,j}^{*n-1} \end{cases}$$

mit:

$$c_{x,2,j}^{*n-1} = -\frac{\Delta x}{\Delta t} \cdot \frac{s_{2,j}^{n-1} - s_{2,j}^{n-2}}{s_{3,j}^{n-2} - s_{2,j}^{n-2}}$$

Um die tatsächliche Phasengeschwindigkeit normal zu linken Rand zu erhalten, muß nun noch die Multiplikation mit dem Normierungsfaktor  $1/G$  durchgeführt werden.

$G$  wird folgendermaßen berechnet:

zunächst Aufspaltung in die geometrischen Anteile  $G_X$  und  $G_Y$ :

$$GX = S_{3,J}^{n-2} - S_{2,J}^{n-2}$$

$$GY = S_{2,J+1}^{n-2} - S_{2,J-1}^{n-2}$$

Dann wird gebildet:  $G = GX * GX + GY * GY + EPS$

EPS entspricht einem infinitesimal kleinen Wert und wird verwendet, um G nicht Null werden zu lassen.

Achtung: Auch hier kann am Rand natürlich GY nicht zentriert gebildet werden, muß also einseitig gelöst werden.

$$CXL_{2,J}^{n-1} = CXL_{2,J}^{*n-1} * 1./G$$

Berechnung der tangentialen Phasengeschwindigkeit:

$$CYL_{2,J}^{n-1} = \left\{ \begin{array}{l} \left\{ \begin{array}{l} \frac{DY}{DT}^{n-1} \text{ für } CXL_{2,J}^{*n-1} < \frac{DY}{DT}^{n-1} \\ \frac{DY}{DT}^{n-1} \text{ für } \frac{DY}{DT}^{n-1} < CXL_{2,J}^{*n-1} < \frac{DY}{DT}^{n-1} \\ \frac{DY}{DT}^{n-1} \text{ für } \frac{DY}{DT}^{n-1} < CXL_{2,J}^{*n-1} \end{array} \right. \end{array} \right.$$

Achtung: in y - Richtung ist diese Begrenzung solange zulässig, solange J nicht 1 oder IY ist. In dem Fall muß dann die Beschränkung dergestalt geändert werden, daß nur ein Fluß eines Signals aus dem Modellgebiet hinaus möglich ist.

mit:

$$CXL_{2,J}^{*n-1} = \frac{DY}{DT}^{n-1} * \frac{S_{2,J}^{n-1} - S_{2,J}^{n-2}}{S_{2,J+1}^{n-2} - S_{2,J-1}^{n-2}}$$

Auch hier muß wieder der Normierungsfaktor G berücksichtigt werden:

$$CYL_{2,J}^{*n-1} = CYL_{2,J}^{*n-1} * 1./G$$

rechter Rand:

$$\begin{array}{r}
 \left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \\
 CXR_{N-1,J}^{n-1} = \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right.
 \end{array}
 \begin{array}{l}
 \text{für } \frac{DX}{DT}^{n-1} \\
 \text{für } CXR_{N-1,J}^{*n-1} \\
 \text{für } 0
 \end{array}
 \begin{array}{l}
 \text{für } CXR_{N-1,J}^{*n-1} > \frac{DX}{DT}^{n-1} \\
 \text{für } \frac{DX}{DT}^{n-1} > CXR_{N-1,J}^{*n-1} > 0 \\
 \text{für } 0 > CXR_{N-1,J}^{*n-1}
 \end{array}$$

mit:

$$CXR_{N-1,J}^{*n-1} = \frac{DX}{DT}^{n-1} * \frac{S_{N-1,J}^{n-1} - S_{N-1,J}^{n-2}}{S_{N-1,J}^{n-2} - S_{N-2,J}^{n-2}}$$

Um die tatsächliche Phasengeschwindigkeit normal zu linken Rand zu erhalten, muß auch hier noch die Multiplikation mit dem Normierungsfaktor 1./G durchgeführt werden.

G wird wieder folgendermaßen berechnet:

zunächst Aufspaltung in die geometrischen Anteile GX und GY:

$$GX = S_{N-1,J}^{n-2} - S_{N-2,J}^{n-2}$$

$$GY = S_{N-1,J+1}^{n-2} - S_{N-2,J-1}^{n-2}$$

$$\text{Dann wird gebildet: } G = GX * GX + GY * GY + EPS$$

eps entspricht einem infinitesimal kleinen Wert epsilon und wird verwendet, um G nicht Null werden zu lassen.

Achtung: Auch hier kann am Rand natürlich GY nicht zentriert gebildet werden, muß also einseitig gelöst werden.

$$\text{CXR}_{N-1,j}^{*n-1} = \text{CXR}_{N-2,j}^{*n-1} * 1./G$$

Nun folgt wieder die Berechnung der tangentialen Phasengeschwindigkeit:

$$\text{CXR}_{N-1,j}^{n-1} = \left\{ \begin{array}{l} \left\{ \begin{array}{l} \frac{DY}{DT}^{n-1} \text{ für } \text{CXR}_{N-1,j}^{*n-1} < \frac{DY}{DT}^{n-1} \\ \text{CXR}_{N-1,j}^{*n-1} \text{ für } -\frac{DY}{DT}^{n-1} < \text{CXR}_{N-1,j}^{*n-1} < \frac{DY}{DT}^{n-1} \\ \frac{DY}{DT}^{n-1} \text{ für } \frac{DY}{DT}^{n-1} < \text{CXR}_{N-1,j}^{*n-1} \end{array} \right. \end{array} \right.$$

Achtung: in y - Richtung ist diese Begrenzung solange zulässig, solange j nicht 1 oder IY ist. In dem Fall muß dann die Beschränkung dergestalt geändert werden, daß nur ein Fluß eines Signals aus dem Modellgebiet hinaus möglich ist.

mit:

$$\text{CXR}_{N-1,j}^{*n-1} = \frac{DY}{DT}^{n-1} * \frac{S_{N-1,j}^{n-1} - S_{N-1,j}^{n-2}}{S_{N-1,j+1}^{n-2} - S_{N-1,j-1}^{n-2}}$$

Auch hier muß wieder der Normierungsfaktor G berücksichtigt werden.

$$\text{CXR}_{2,j}^{*n-1} = \text{CXR}_{2,j}^{*n-1} * 1./G$$

analog läuft dies auch für den v - ordenen und h - interen Rand, dies soll hier jedoch nicht mehr explizit aufgeführt werden - das Bildungsgesetz ist analog wie oben aufgeführt, zum Verständnis sei auf den Quelltext des Programms verwiesen.

Wichtig: die Behandlung der Ecken ist auch in den Originalarbeiten nicht näher beschrieben, insofern wird hier der einfachste Weg genommen, d.h. auf die Ecken wird die Advektion mittels der Phasengeschwindigkeiten aus beiden Raumrichtungen (x , y) gebildet, sowohl tangential als auch

normal, wobei die Differenzen jedoch nur einseitig gebildet werden können, und dann wird aus den beiden neuen Eckwerten der Mittelwert gebildet.

2. die Phasengeschwindigkeiten werden nun zum neuen Zeitschritt (n-1->n) für die Berechnung der Randpunkte benutzt;

Belegung der Ränder mit neuen Werten - dargestellt wird die Berechnung für die Kanten

links:  
-----

$$s_{1,j}^n = s_{1,j}^{n-1} - \frac{DT}{DX} * CXL_{1,j}^n * (s_{2,j}^{n-1} - s_{1,j}^{n-1}) -$$

$$\frac{DT}{DY} * CYL_{1,j}^n * (s_{1,j+1}^{n-1} - s_{1,j-1}^{n-1})$$

mit:  $\frac{CXL_{1,j}^n}{-----} = \frac{CXL_{2,j}^{n-1}}{---}$  und

mit:  $\frac{CYL_{1,j}^n}{-----} = \frac{CYL_{2,j}^{n-1}}{---}$

rechts:  
-----

$$s_{N,j}^n = s_{N,j}^{n-1} - \frac{DT}{DX} * CXR_{N,j}^n * (s_{N,j}^{n-1} - s_{N-1,j}^{n-1}) -$$

$$\frac{DT}{DY} * CYR_{N,j}^n * (s_{N,j+1}^{n-1} - s_{N,j-1}^{n-1})$$

mit:  $\frac{CXR_{N,j}^n}{-----} = \frac{CXR_{N-1,j}^{n-1}}{---}$  und:

mit:  $\frac{CYR_{N,j}^n}{-----} = \frac{CYR_{N-1,j}^{n-1}}{---}$

damit ist die Belegung des linken und rechten Randes (bis auf die

Ecken, die hier nicht näher aufgeführt wurden) abgeschlossen.  
Die Behandlung der Vorderseite und der Rückseite verläuft analog, zum Verständnis wird auf den erläuterten Quelltext im Unterprogramm selbst verwiesen.

Das Programm findet sich im Member ORL2D2D auf der Datei IMK965.FCTTOM.FORT!

Auch hier sei wieder darauf verwiesen, daß die oben beschriebene Formulierung nicht der Idee von Miller und Thorpe entspricht, da dort wiederum die berechnete Phasengeschwindigkeit zum selben Zeitpunkt, wie sie gebildet wird, auch zur Neubelegung der Ränder verwendet wird.

Ein Unterprogramm, das dieser Idee entspricht, ist aber auch vorhanden, außerdem läßt sich aus dem oben beschriebenen Unterprogramm durch Umstellung im Ablauf (d.h. zuerst Berechnung der Phasengeschwindigkeiten und dann Neubelegung der Ränder) diese Formulierung leicht erreichen.

Das Unterprogramm mit dieser Formulierung ist als Member SORL2D2D abgespeichert.

Ein Member, das anstelle der zentrierten Differenzen eine Upwindformulierung verwendet (Ansatz (3.12), beschrieben in Kapitel 3.2), existiert unter dem Membernamen ORL2DAD. Die zugehörige " Sofort-"Version findet man unter ORL2DDS





## Dimensionierung der Felder

---

alt-es Feld des S-kalars sowie neues Feld nach Anwendung der  
Orlanski - Formulierung der Randbedingungen

```
REAL    SALT(IX,IY,IZ),S(IX,IY,IZ)
REAL    Z(IX,IY,IZ)
```

Zeit (DT) und Raumschrittweite (DX,DY,DE)  
DE ist die vertikale Raumschrittweite im eta - System

### Achtung:

---

in vertikaler (z - ) Richtung ist das Gitter nicht äquidistant

```
REAL    DT,DX,DY,DE,DZ
```

Phasengeschwindigkeiten für die verschiedenen Randflächen

---

#### a. Phasengeschwindigkeiten normal zum Rand

```
REAL    CXL(IY,IZ),CXR(IY,IZ)
REAL    CYV(IX,IZ),CYH(IX,IZ)
REAL    CEU(IX,IY),CEO(IX,IY)
```

L steht für L-inks  
R steht für r-echts  
V steht für V-orne  
H steht für H-inten  
U steht für Unten  
O steht für O-ben

der mittlere Buchstabe im Namen der Phasengeschwindigkeiten  
gibt die Richtung an.

#### b. Phasengeschwindigkeiten tangential am Rand

```
REAL    CXL(IY,IZ),CXR(IY,IZ)
REAL    CYL(IY,IZ),CYR(IY,IZ)
REAL    CEL(IY,IZ),CER(IY,IZ)
REAL    CXV(IX,IZ),CXH(IX,IZ)
REAL    CYV(IX,IZ),CYH(IX,IZ)
REAL    CEV(IX,IZ),CEH(IX,IZ)
REAL    CXU(IX,IY),CXO(IX,IY)
REAL    CYU(IX,IY),CYO(IX,IY)
REAL    CEU(IX,IY),CEO(IX,IY)
```

Hilfsvektoren zur Berechnung der Randlinien

```
REAL    RLUL(IY),RRUR(IY)
REAL    RVUV(IX),RHUH(IX)
REAL    RLOL(IY),RROR(IY)
REAL    RVOV(IX),RHOH(IX)
REAL    RLVL(IZ),RLHL(IZ)
REAL    RRVR(IZ),RRHR(IZ)
REAL    RLUV(IY),RRUV(IY)
REAL    RVUU(IX),RHUU(IX)
REAL    RLOO(IY),RROO(IY)
REAL    RVOO(IX),RHOO(IX)
REAL    RLVV(IZ),RLHH(IZ)
```

REAL RRVV(IZ),RRHH(IZ)

lokale Variablen

REAL DXRDT,DYRDT,DERDT  
 REAL RDX,RDY,RDE,R2DX,R2DY,R2DE,RDT

Die Vorwärts-Euler-Diskretisierung einer  
 -----  
 dreidimensionalen Transportgleichung :  
 -----

$$\frac{dS}{dt} + CX \frac{dS}{dX} + CY \frac{dS}{dY} + CE \frac{dS}{dE} = 0 \quad \text{zum Zeitpunkt } t^{n-1} \quad \text{am Ort } x_{i,j}$$

lautet:  
 -----

$$\begin{aligned} & \frac{S_{i,j,k}^n - S_{i,j,k}^{n-1}}{t^n - t^{n-1}} + CX_{i,j,k}^{n-1} \frac{S_{i+1,j,k}^{n-1} - S_{i-1,j,k}^{n-1}}{2 DX} + \\ & + CY_{i,j,k}^{n-1} \frac{S_{i,j+1,k}^{n-1} - S_{i,j-1,k}^{n-1}}{2 DY} + \\ & + CE_{i,j,k}^{n-1} \frac{S_{i,j,k+1}^{n-1} - S_{i,j,k-1}^{n-1}}{2 DE} \end{aligned}$$

wobei auf den Randflächen die räumliche Diskretisierung in einer Raumrichtung, auf den Kanten in zwei Raumrichtungen und in den Eckpunkten in allen Raumrichtungen durch einseitige Differenzen ersetzt werden muß.

Wir benutzen diese Gleichung wieder zweifach, zunächst einmal, um die Phasengeschwindigkeiten in der Nähe der Ränder zu schätzen:

die Vorgehensweise wird nicht für alle Ränder aufgezeigt, sondern soll exemplarisch für den linken und den rechten Rand dargestellt werden.

linke Randfläche: Normalrichtung  
 -----

$$\begin{aligned} & \left. \begin{array}{l} \text{CXL}_{2,j,k}^{n-1} \\ \text{CXL}_{2,j,k}^{n-1} \end{array} \right\} = \left\{ \begin{array}{l} - \frac{DX}{DT} \text{ für } \text{CXL}_{2,j,k}^{*n-1} < - \frac{DX}{0 DT} \\ - \frac{DX}{n-1} \text{ für } \text{CXL}_{2,j,k}^{n-1} < 0 \end{array} \right. \end{aligned}$$

$$\left. \begin{array}{l} \\ \\ \\ \end{array} \right\} 0 \quad \text{für} \quad 0 < \text{CXL}_{2,j,k}^{n-1}$$

mit:  
-----

$$\text{CXL}_{2,j,k}^{*n-1} = \frac{\text{DX}}{\text{DT}^{n-1}} * \frac{\text{S}_{2,j,k}^{n-1} - \text{S}_{2,j,k}^{n-2}}{\text{S}_{3,j,k}^{n-2} - \text{S}_{2,j,k}^{n-2}}$$

Um die tatsächliche Phasengeschwindigkeit normal zu linken Rand zu erhalten, muß nun noch die Multiplikation mit dem Normierungsfaktor 1./G durchgeführt werden.

G wird folgendermaßen berechnet:  
-----

zunächst Aufspaltung in die geometrischen Anteile GX, GY und GE  
-----

$$\text{GX} = \text{S}_{3,j,k}^{n-2} - \text{S}_{2,j,k}^{n-2}$$

$$\text{GY} = \text{S}_{2,j+1,k}^{n-2} - \text{S}_{2,j-1,k}^{n-2}$$

$$\text{GE} = \text{S}_{2,j,k+1}^{n-2} - \text{S}_{2,j,k-1}^{n-2}$$

Dann wird gebildet:  $G = \text{GX} * \text{GX} + \text{GY} * \text{GY} + \text{GE} * \text{GE} + \text{EPS}$   
-----

EPS entspricht einem infinitesimal kleinen Wert und wird verwendet, um G nicht Null werden zu lassen.

Achtung: Auch hier kann am Rand natürlich GY und GE nicht zentriert gebildet werden, muß also jeweils einseitig gelöst werden

$$\text{CXL}_{2,j,k}^{n-1} = \text{CXL}_{2,j,k}^{*n-1} * 1./G$$

Berechnung der tangentialen Phasengeschwindigkeit CYL:  
-----

$$\left\{ \begin{array}{l} \\ \\ \end{array} \right. \text{DY} \quad \quad \quad \text{*n-1} \quad \quad \quad \text{DY}$$

$$\text{CYL}_{2,j,k}^{n-1} = \left\{ \begin{array}{l}
 \text{für } - \frac{\text{DT}^{n-1}}{\text{CYL}_{2,j,k}^{*n-1}} < - \frac{\text{DT}^{n-1}}{\text{CYL}_{2,j,k}^{n-1}} \\
 \text{für } - \frac{\text{DT}^{n-1}}{\text{CYL}_{2,j,k}^{*n-1}} < \frac{\text{DY}}{\text{DT}^{n-1}} < \frac{\text{DY}}{\text{DT}^{n-1}} < \frac{\text{DT}^{n-1}}{\text{CYL}_{2,j,k}^{n-1}} \\
 \text{für } \frac{\text{DY}}{\text{DT}^{n-1}} < \frac{\text{DY}}{\text{DT}^{n-1}} < \text{CYL}_{2,j,k}^{*n-1}
 \end{array} \right.$$

Achtung: in y - Richtung ist diese Begrenzung solange zulässig, solange j nicht 1 oder IY ist. In dem Fall muß dann die Beschränkung dergestalt geändert werden, daß nur ein Fluß eines Signals aus dem Modellgebiet hinaus möglich ist.

mit:

$$\text{CYL}_{2,j,k}^{*n-1} = \frac{\text{DY}}{\text{DT}^{n-1}} * \frac{S_{2,j,k}^{n-1} - S_{2,j,k}^{n-2}}{S_{2,j+1,k}^{n-2} - S_{2,j-1,k}^{n-2}}$$

Auch hier muß wieder der Normierungsfaktor G berücksichtigt werden.

$$\text{CYL}_{2,j,k}^{n-1} = \text{CYL}_{2,j,k}^{*n-1} * 1./G$$

Berechnung der tangentialen Phasengeschwindigkeit CEL :

$$\text{CEL}_{2,j,k}^{n-1} = \left\{ \begin{array}{l}
 \text{für } - \frac{\text{DE}}{\text{DT}^{n-1}} < \text{CEL}_{2,j,k}^{*n-1} < - \frac{\text{DE}}{\text{DT}^{n-1}} \\
 \text{für } - \frac{\text{DE}}{\text{DT}^{n-1}} < \text{CEL}_{2,j,k}^{*n-1} < \frac{\text{DE}}{\text{DT}^{n-1}} < \frac{\text{DE}}{\text{DT}^{n-1}} \\
 \text{für } \frac{\text{DE}}{\text{DT}^{n-1}} < \frac{\text{DE}}{\text{DT}^{n-1}} < \text{CEL}_{2,j,k}^{*n-1}
 \end{array} \right.$$

Achtung: in eta - Richtung ist diese Begrenzung solange zulässig, solange k nicht 1 oder IZ ist.

In dem Fall muß dann die Beschränkung dergestalt geändert werden, daß nur ein Fluß eines Signals aus dem Modellgebiet hinaus möglich ist.

mit:  
-----

$$CEL_{2,j,k}^{*n-1} = \frac{DE}{DT^{n-1}} * \frac{S_{2,j,k}^{n-1} - S_{2,j,k}^{n-2}}{S_{2,j,k+1}^{n-2} - S_{2,j,k-1}^{n-2}}$$

Auch hier muß wieder der Normierungsfaktor G berücksichtigt werden.

$$CEL_{2,j,k}^{n-1} = CEL_{2,j,k}^{*n-1} * 1./G$$

Die Behandlung der rechten Seitenfläche erfolgt analog!  
Das Bildungsprinzip für die Berechnung der Phasengeschwindigkeiten ist immer dasselbe:

Die Referenzfläche, von der ausgehend die Differenzen gebildet werden, ist die erste Fläche von Rand weg zur Mitte hin!  
Für die Normalkomponente werden die Differenzen dann einseitig, für die Tangentialkomponenten zentriert gebildet.  
(für Ecken und Kanten selbstverständlich mit den dort notwendigen Einschränkungen)

Die Schranken werden immer so gesetzt, daß Störungen zwar von innen nach außen, aber nicht von umgekehrt gelangen dürfen, d.h. errechnet sich an irgendeinem Punkt auf den Randflächen eine Phasengeschwindigkeit, die von außen nach innen zeigt, so wird sie künstlich auf 0 gesetzt.

So wie hier für den linken Rand, werden dann auch für die anderen Seitenflächen die Phasengeschwindigkeiten gebildet, mit dann entsprechend anderer Verteilung der Normal- und Tangentialrichtungen.

Wichtig: die Behandlung der Ecken und Kanten ist auch in den Originalarbeiten von Miller/Thorpe und Raymond/Kuo nicht näher beschrieben, insofern wird hier der einfachste Weg genommen, d.h. auf die Ecken/Kanten wird die Advektion von Störungen aus allen möglichen Raumrichtungen (x,y,eta) gebildet, sowohl tangential als auch normal, wobei die Differenzen jedoch nur einseitig gebildet werden können (Kanten zweiseitig), und aus den sich dann ergebenden neuen Werten der Kanten und Ecken wird der tatsächliche Wert durch Mittelung ermittelt.

die Phasengeschwindigkeiten werden nun zum neuen Zeitschritt (n-1->n) für die Berechnung der Randpunkte zu benutzen:

Belegung der Ränder mit neuen Werten - dargestellt wird die Berechnung für die Kanten

links:  
-----

$$S_{1,j,k}^n = S_{1,j,k}^{n-1} - \frac{DT}{DX} * CXL_{1,j,k}^n * ( S_{2,j,k}^{n-1} - S_{1,j,k}^{n-1} ) -$$

$$\frac{DT}{DY} * CYL_{1,j,k}^n * ( S_{1,j+1,k}^{n-1} - S_{1,j-1,k}^{n-1} ) -$$

$$\frac{DT}{DE} * CEL_{1,j,k}^n * ( S_{1,j,k+1}^{n-1} - S_{1,j,k-1}^{n-1} )$$

mit:  $\frac{CXL_{1,j,k}^n}{-----} = \frac{CXL_{2,j,k}^{n-1}}{-----},$

$$CYL_{1,j,k}^n = CYL_{2,j,k}^{n-1} \text{ und}$$

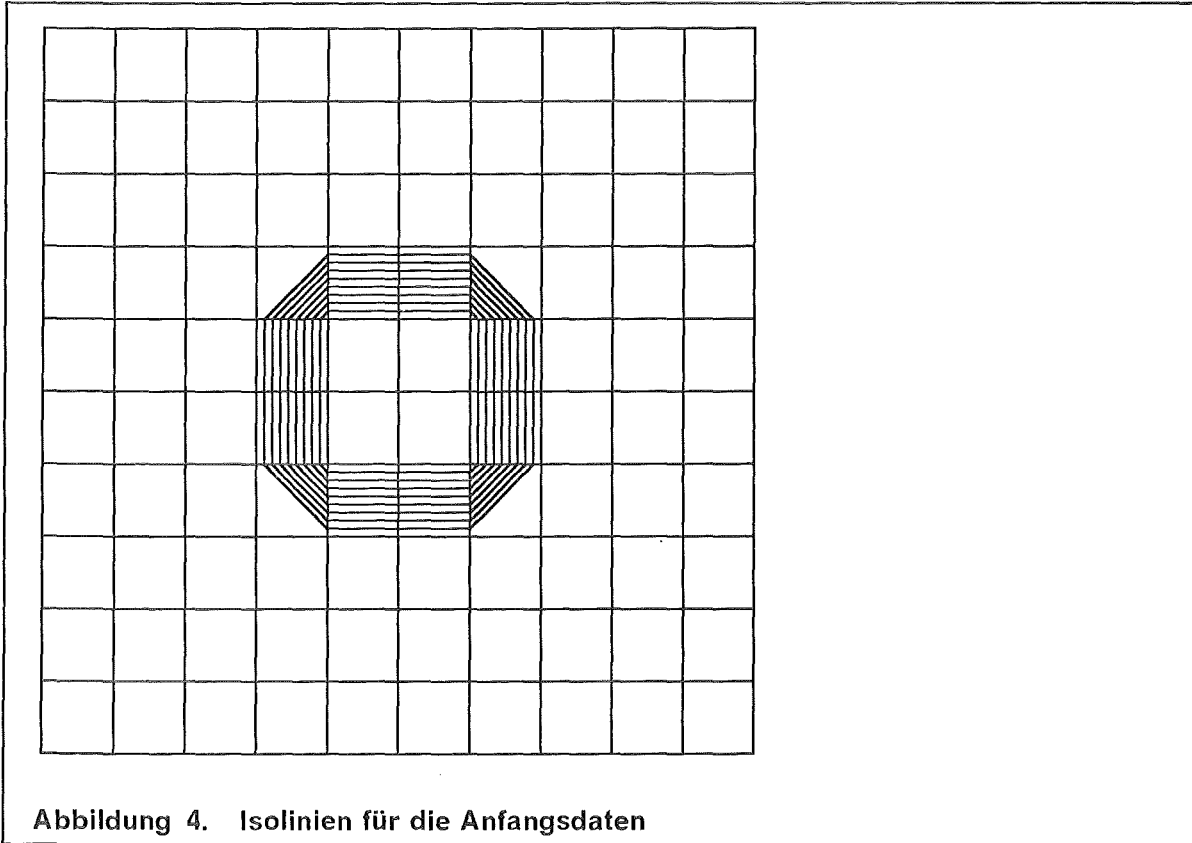
$$CEL_{1,j,k}^n = CEL_{2,j,k}^{n-1}$$

Das vollständige Programm findet sich unter IMK965.FCTTOM.FORT(ORL3D3D)

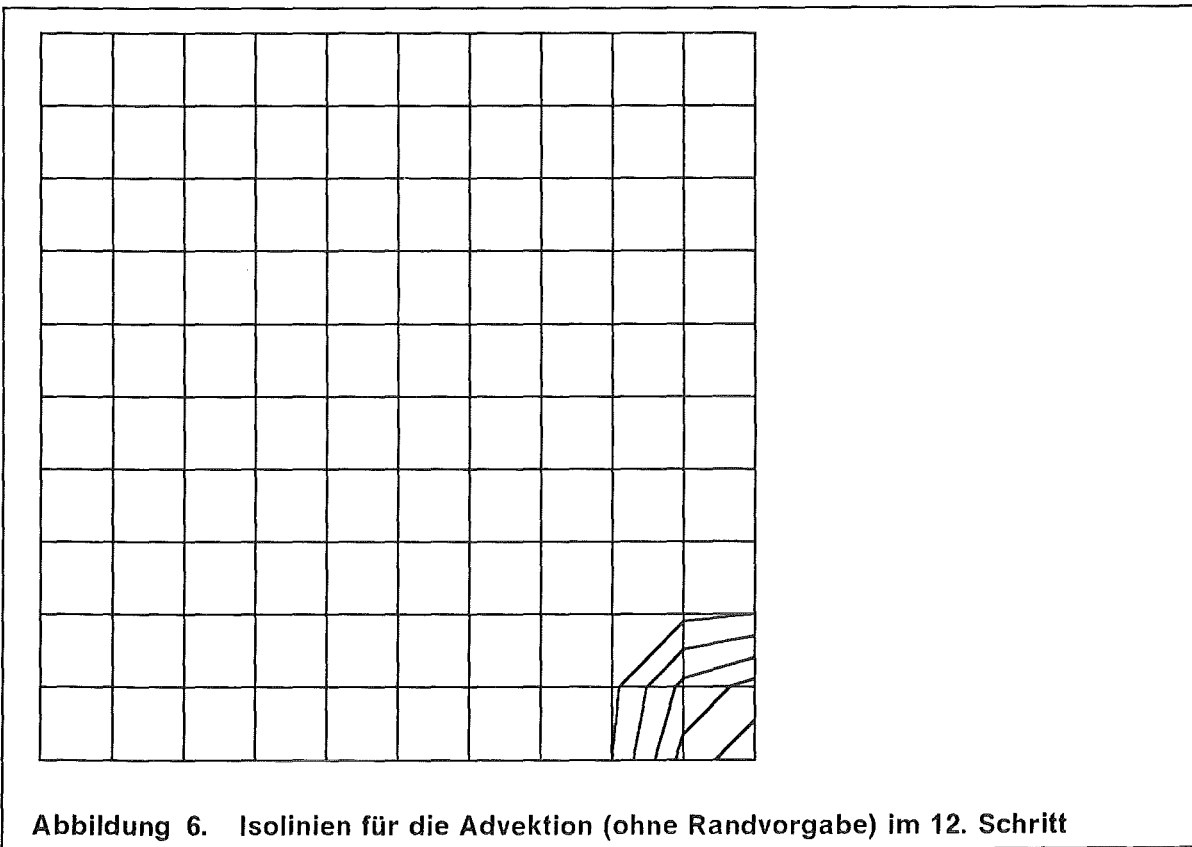
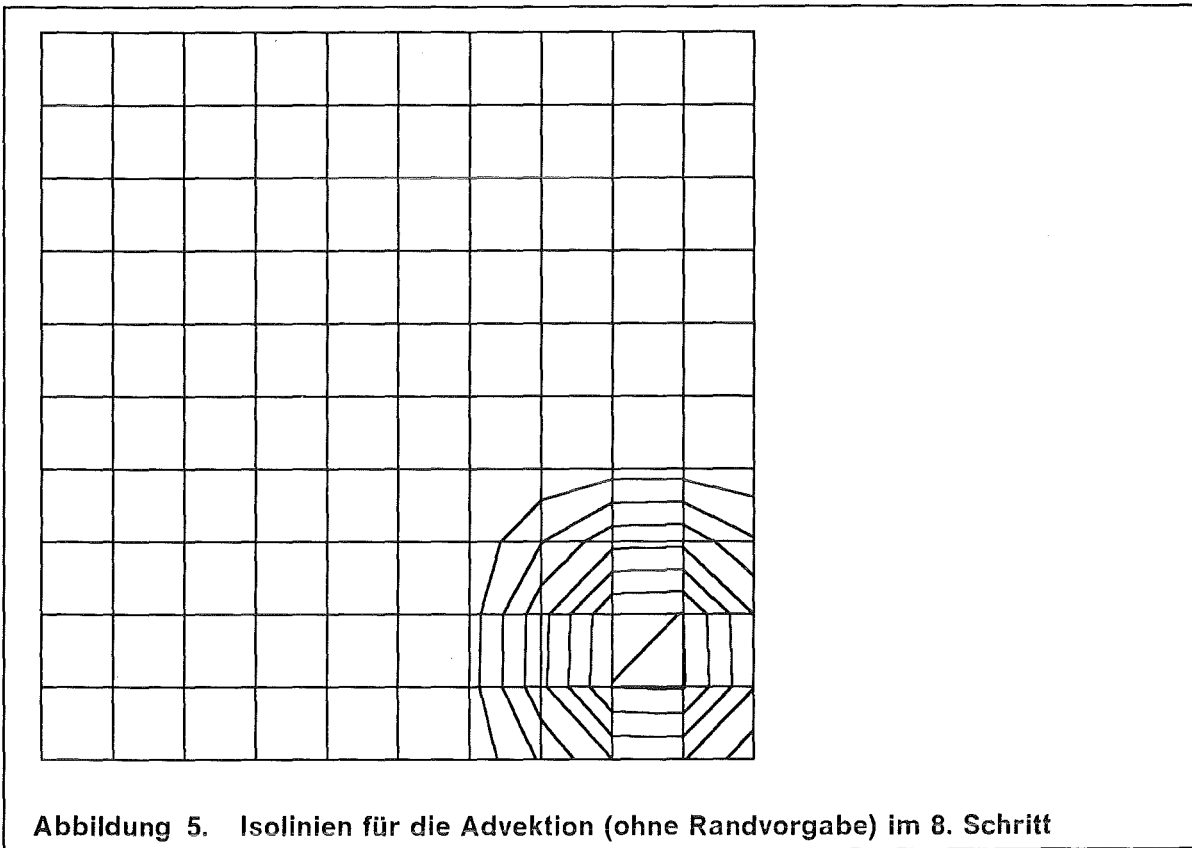
Auch hier sei wieder darauf verwiesen, daß die oben beschriebene Formulierung nicht der Idee von Miller und Thorpe entspricht, da dort wiederum die berechnete Phasengeschwindigkeit zum selben Zeitpunkt, wie sie gebildet wird, auch zur Neubelegung der Ränder verwendet wird.

Ein Unterprogramm, das dieser Idee entspricht, ist aber auch vorhanden, außerdem läßt sich aus dem oben beschriebenen Unterprogramm durch Umstellung im Ablauf (d.h. zuerst Berechnung der Phasengeschwindigkeiten und dann Neubelegung der Ränder) diese Formulierung leicht erreichen.

Diese " Sofortversion " findet man im Member ORL3SOFT .







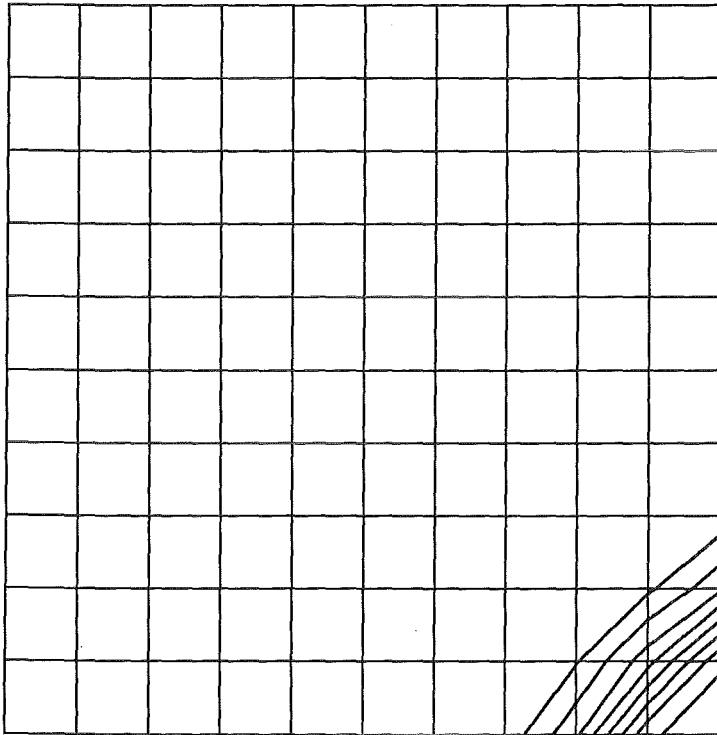


Abbildung 7. Isolinien für die Advektion mit ORL1D2D im 12.Schritt

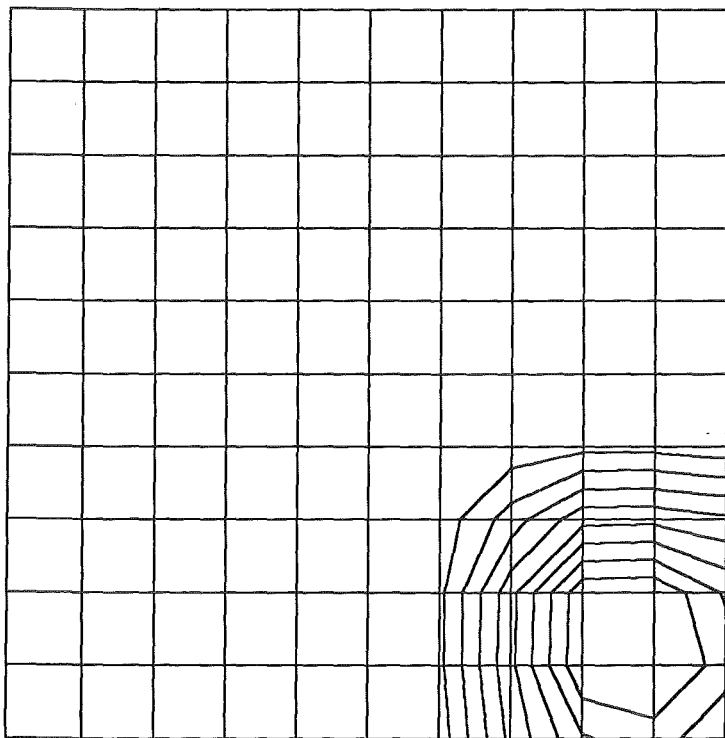


Abbildung 8. Isolinien für die Advektion mit ORL2D2D im 8. Schritt

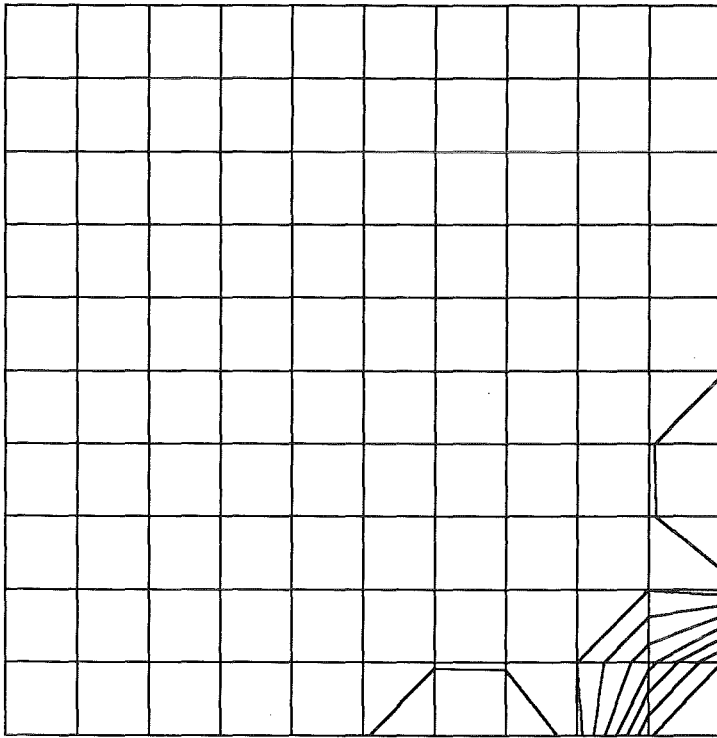


Abbildung 9. Isolinien für die Advektion mit ORL2D2D im 12. Schritt

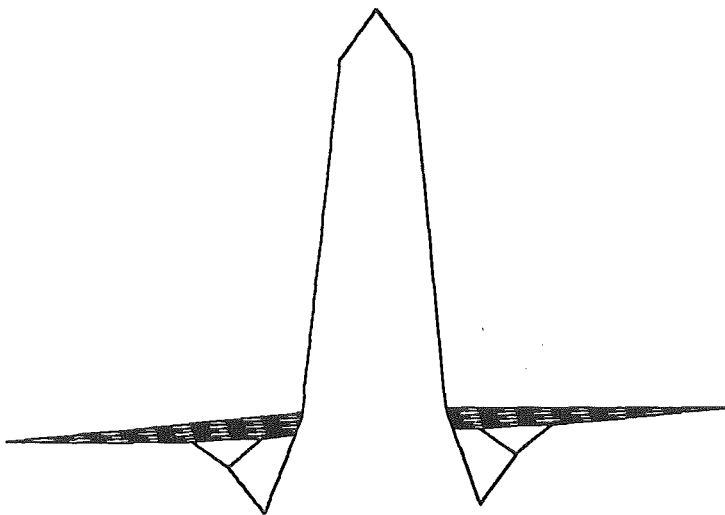
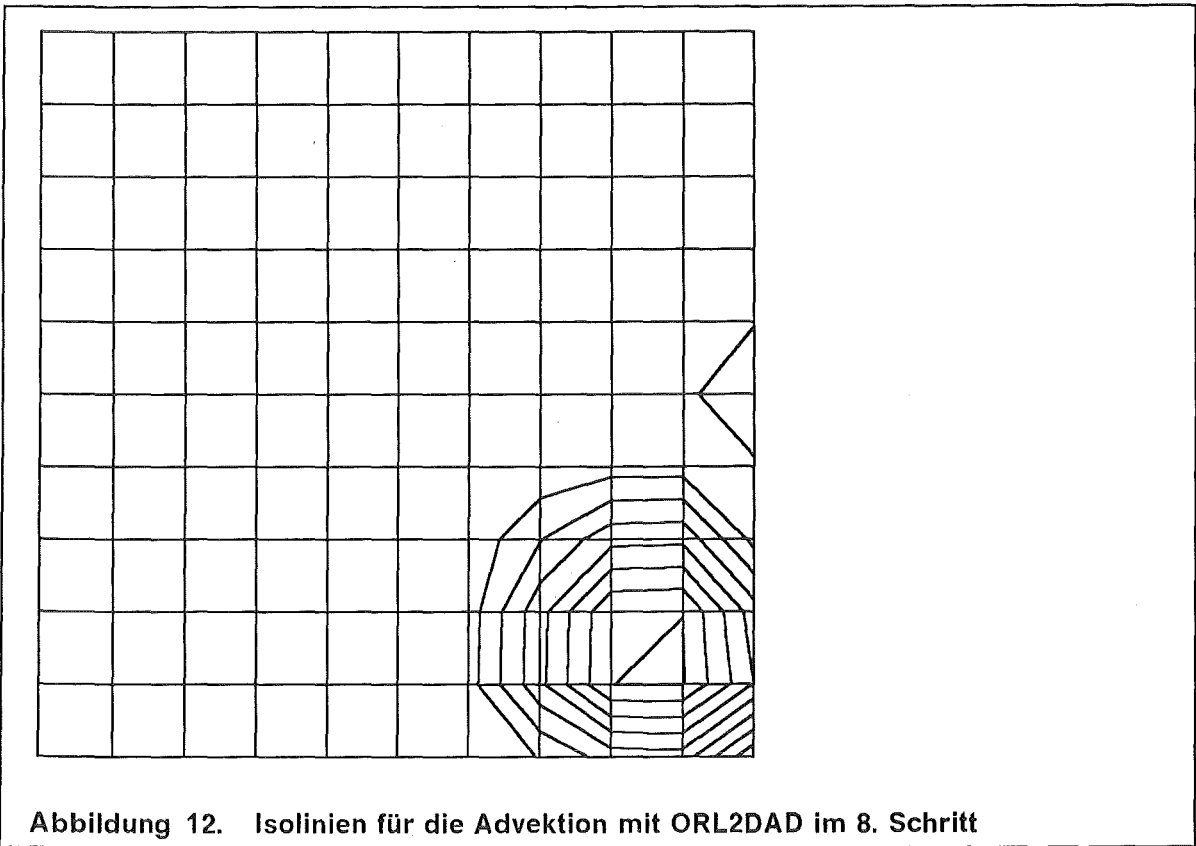
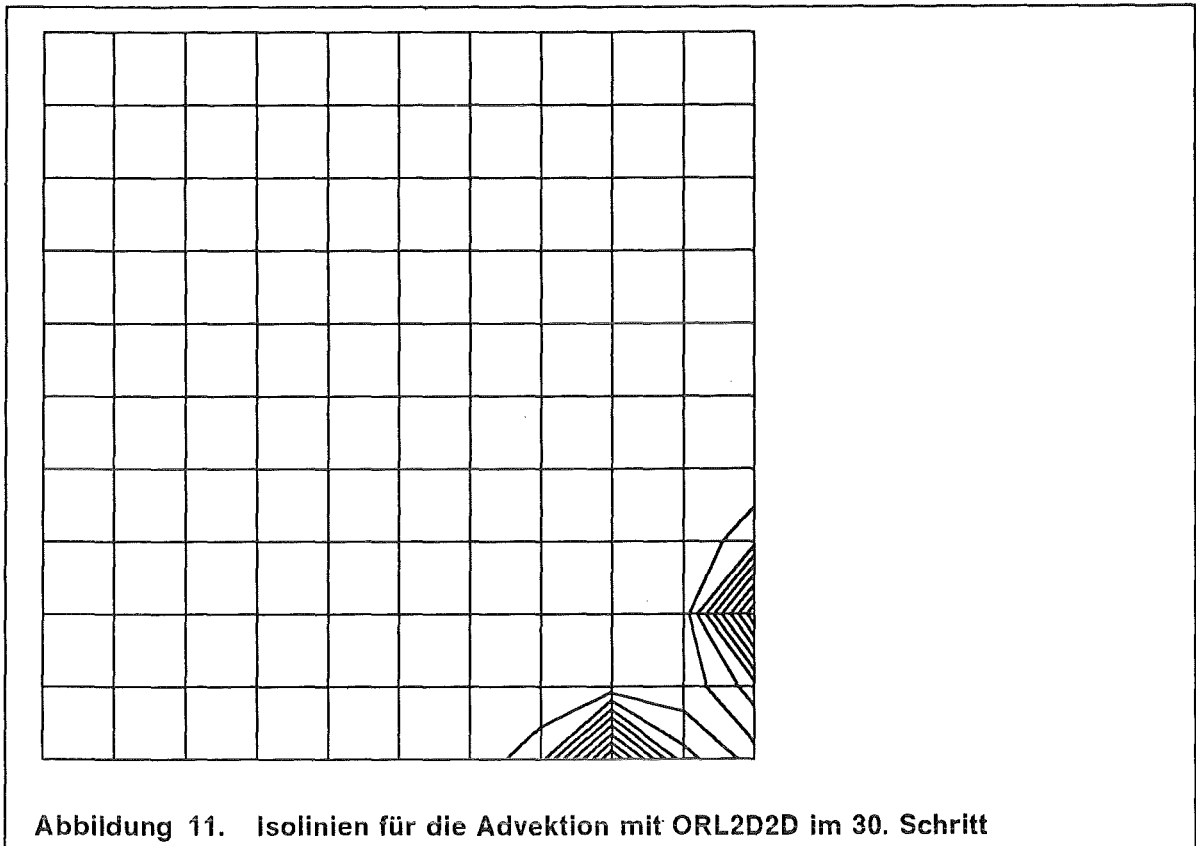
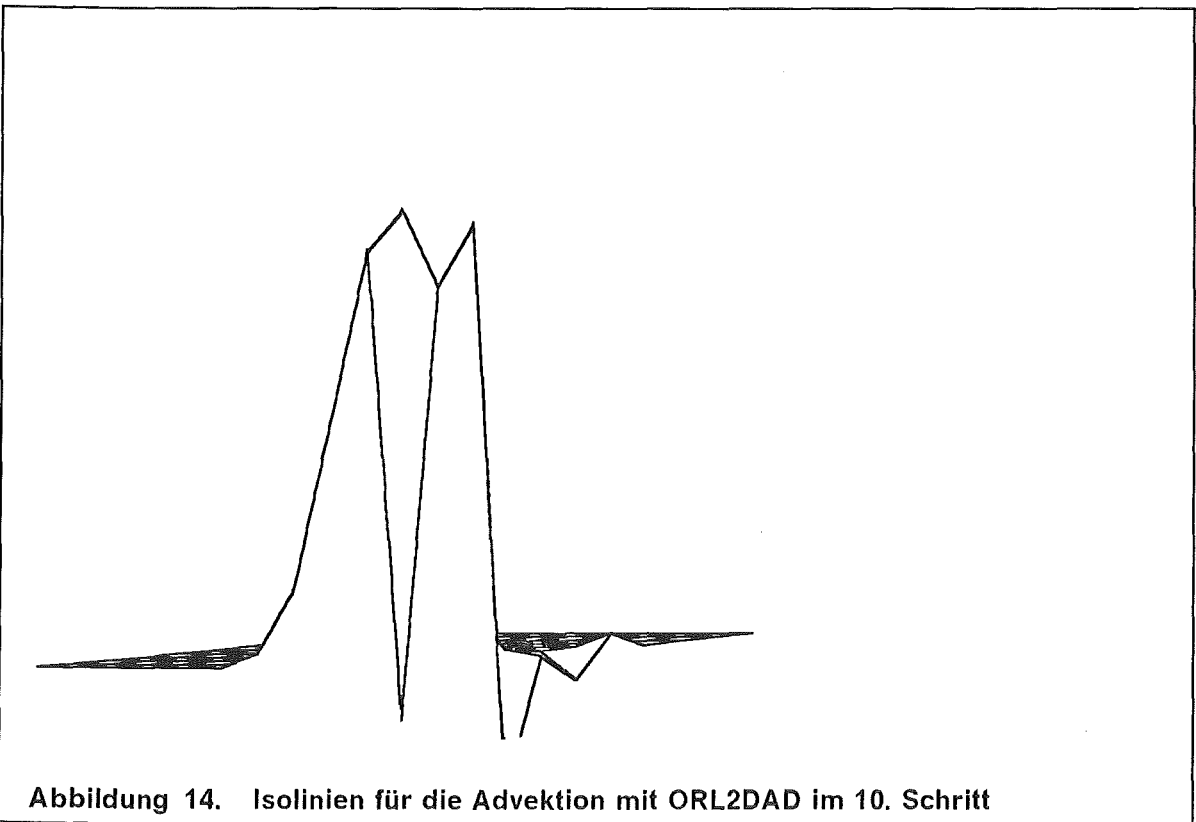
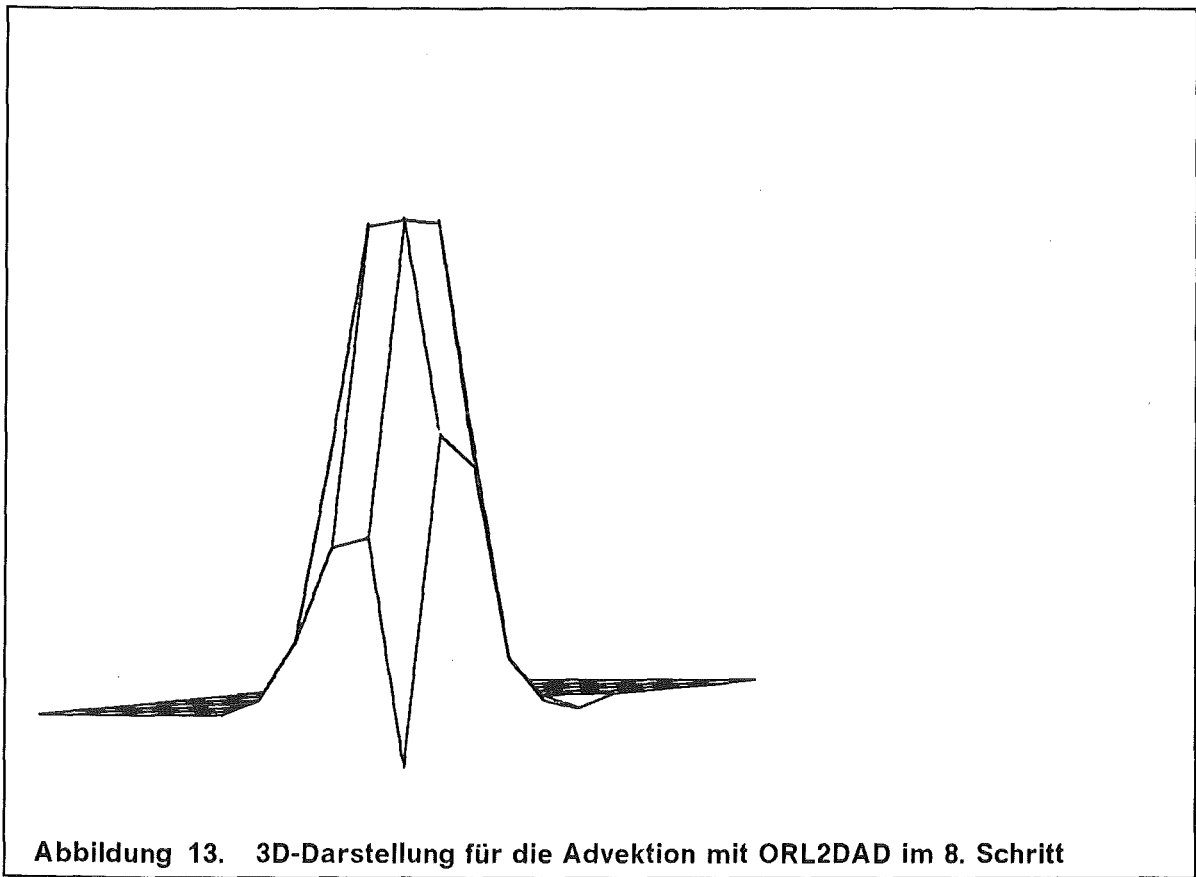


Abbildung 10. 3D-Darstellung für die Advektion mit ORL2D2D im 12. Schritt





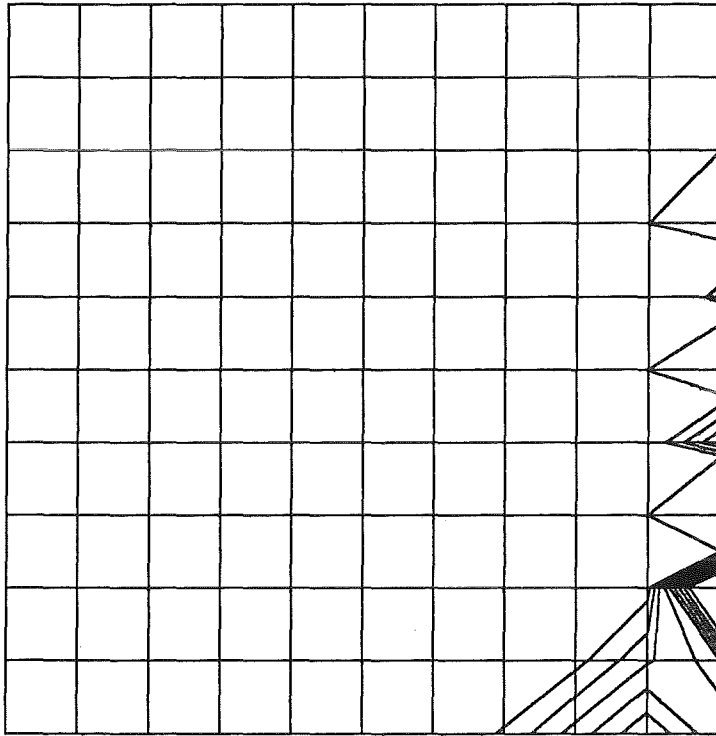


Abbildung 15. Isolinien für die Advektion mit ORL2DAD im 12. Schritt

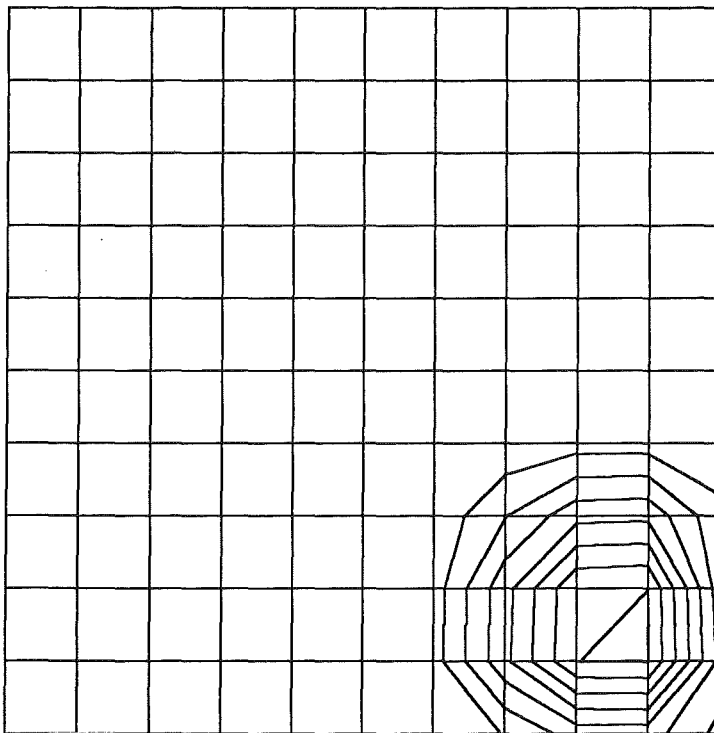
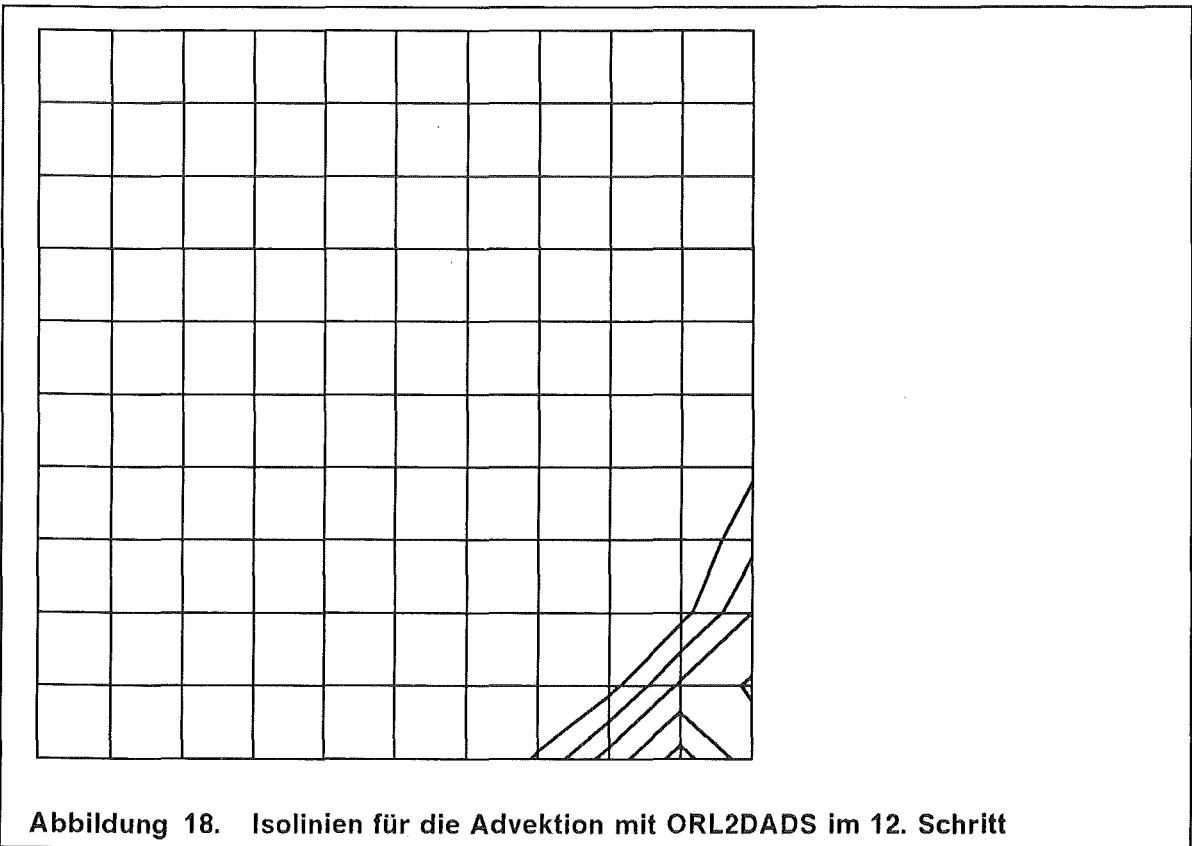
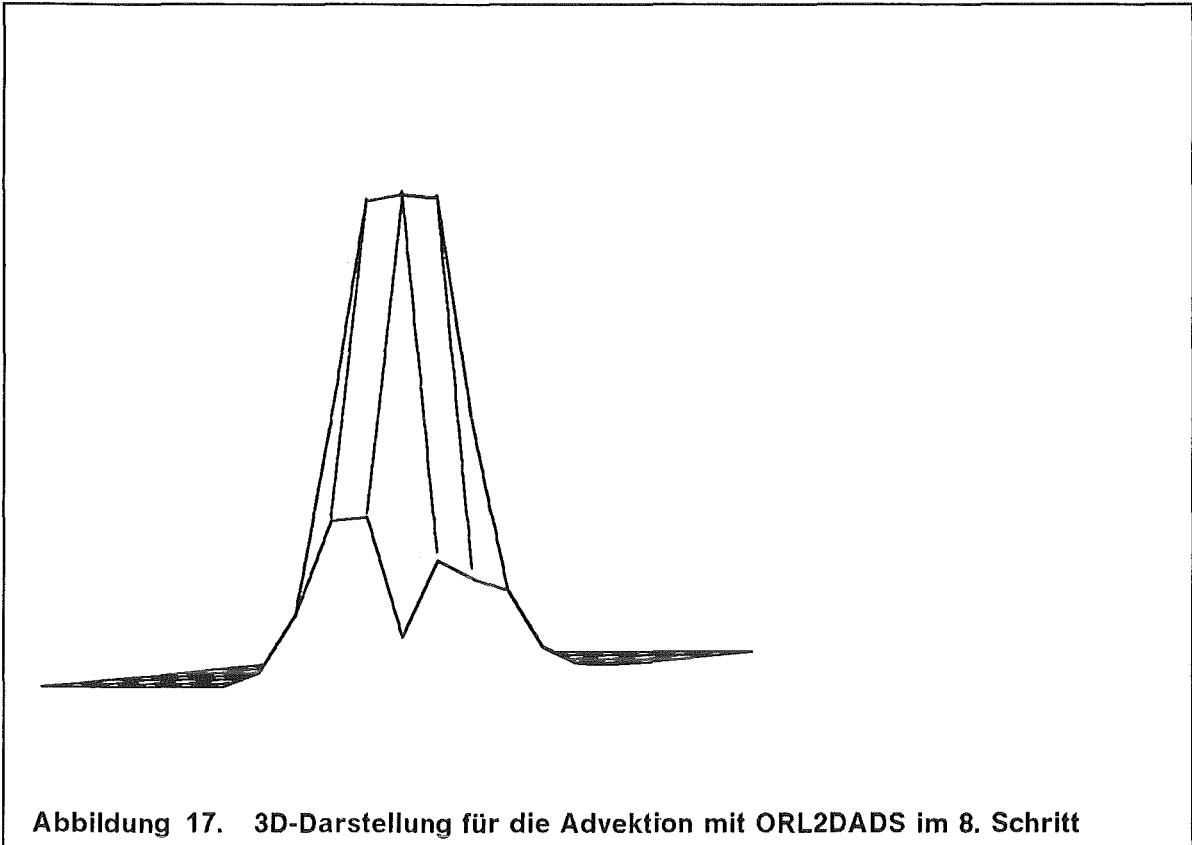
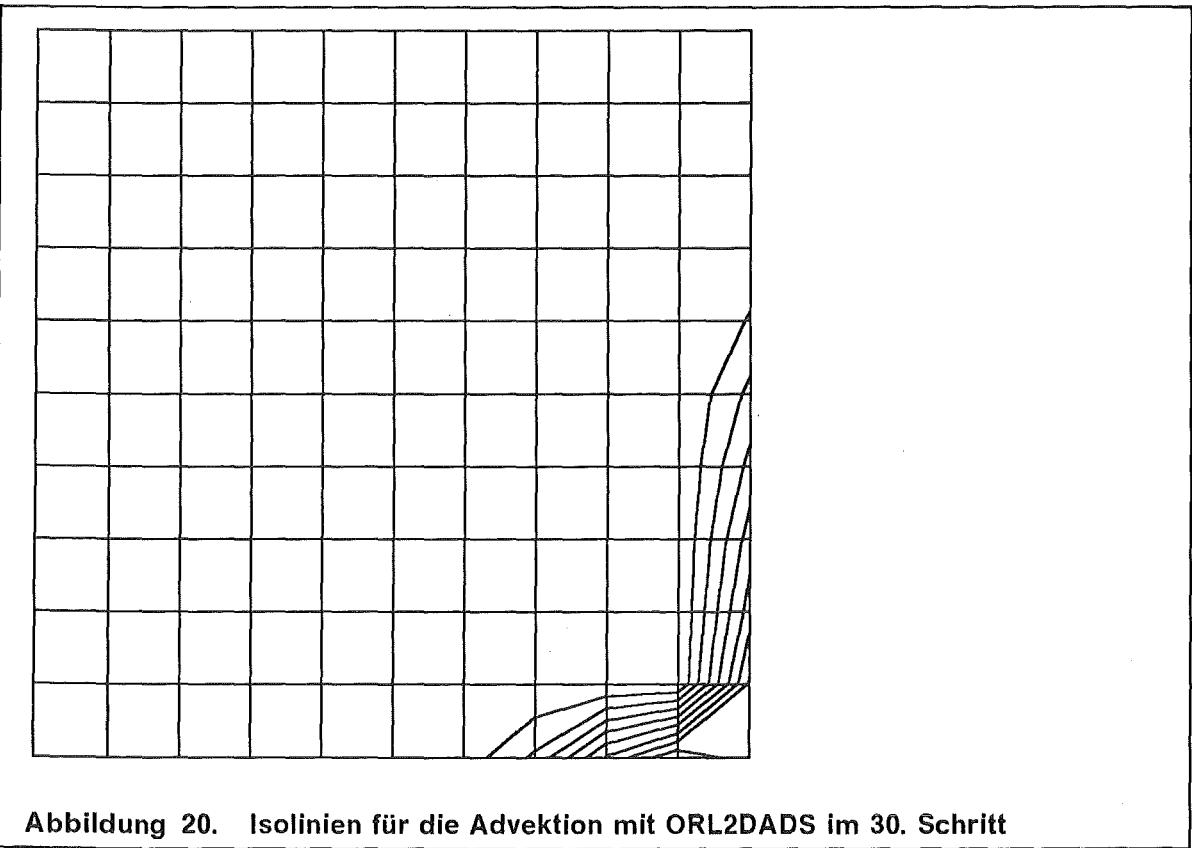
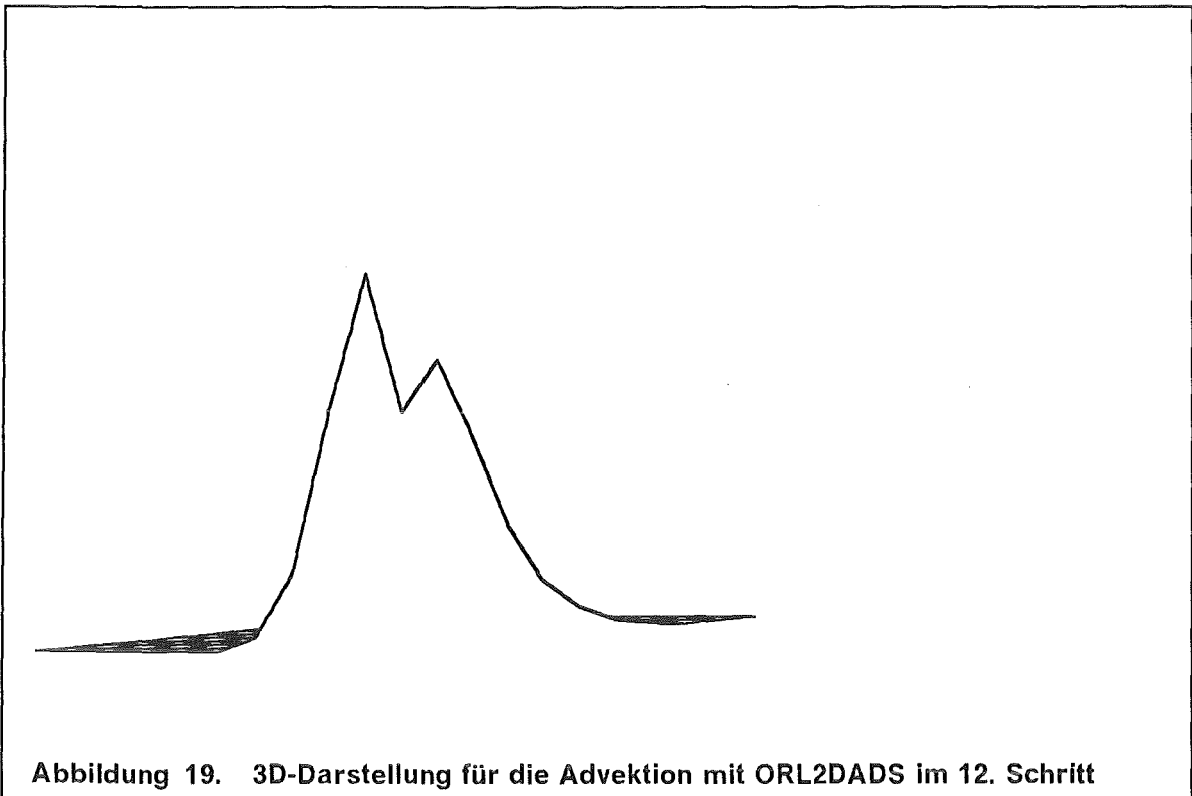
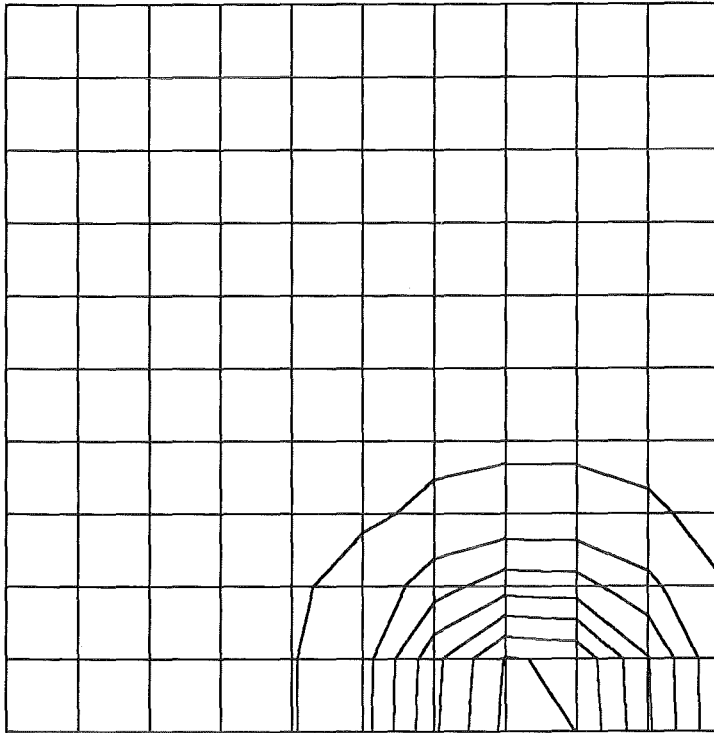


Abbildung 16. Isolinien für die Advektion mit ORL2DADS im 8. Schritt

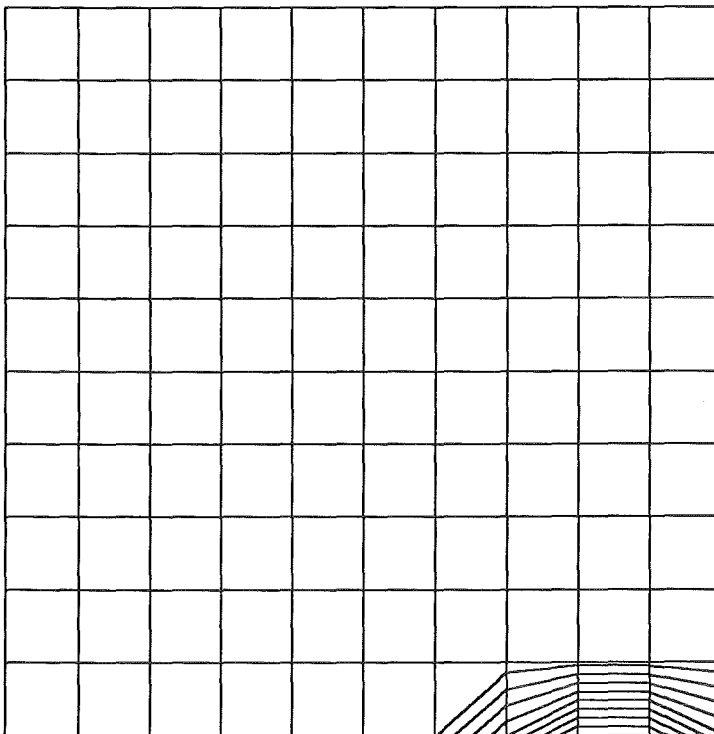








**Abbildung 21. Isolinien für die Advektion (ohne Randvorgaben) im 8. Schritt:**  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )



**Abbildung 22. Isolinien für die Advektion (ohne Randvorgaben) im 12. Schritt:**  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

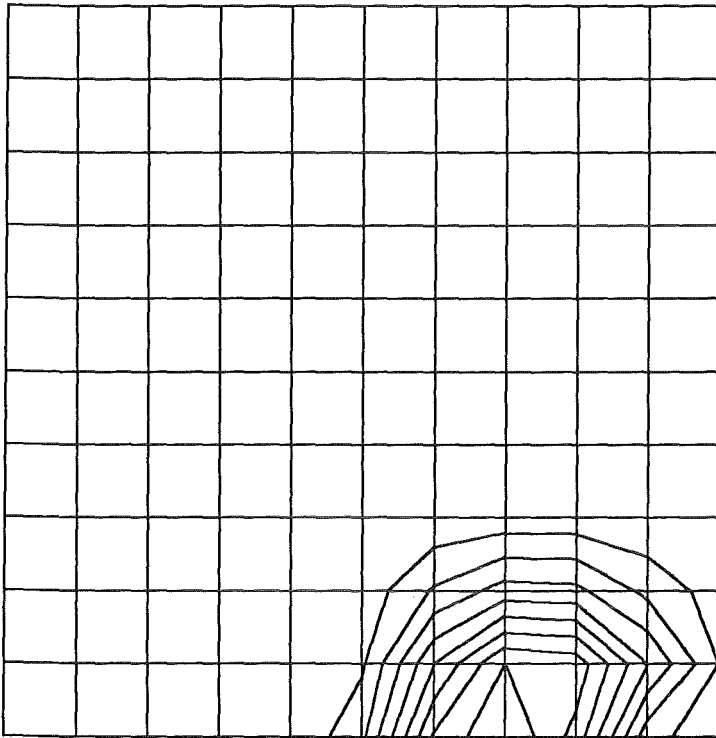


Abbildung 23. Isolinien für die Advektion mit ORL1D2D im 8.Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

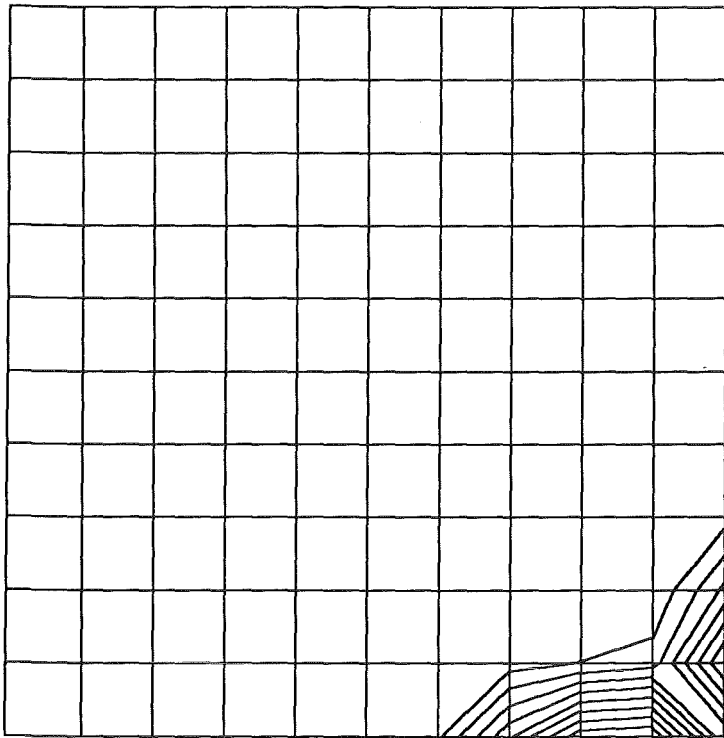


Abbildung 24. Isolinien für die Advektion mit ORL1D2D im 12. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

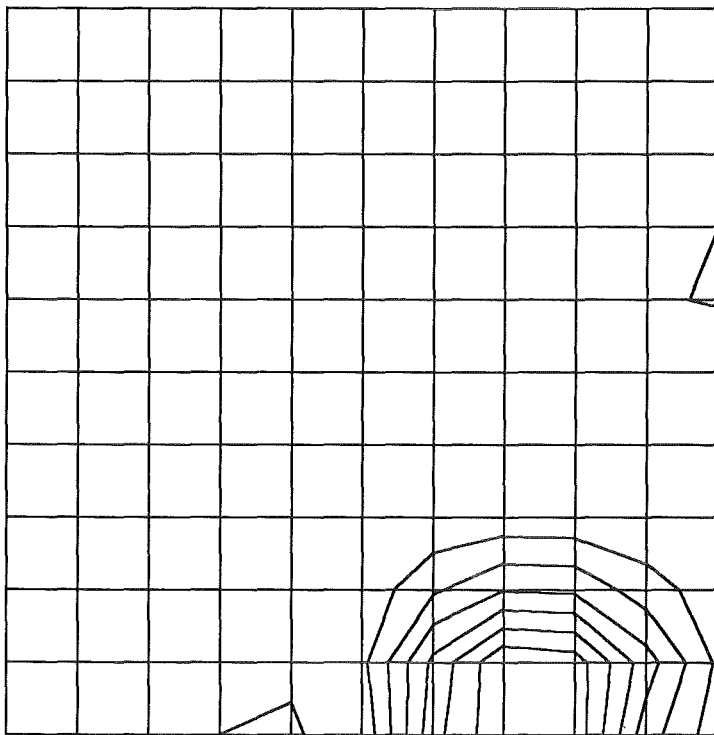


Abbildung 25. Isolinien für die Advektion mit ORL2D2D im 8. Schritt:  
( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

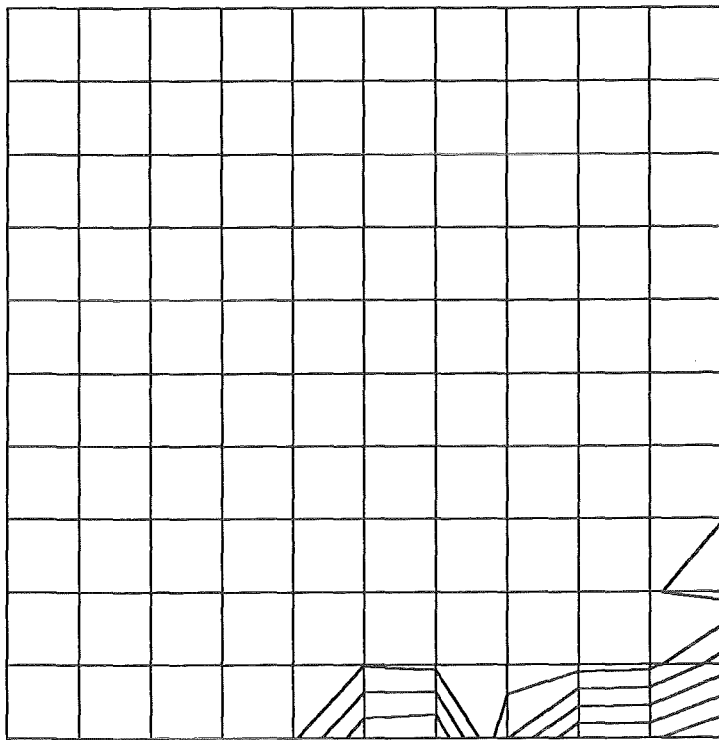


Abbildung 26. Isolinien für die Advektion mit ORL2D2D im 12. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

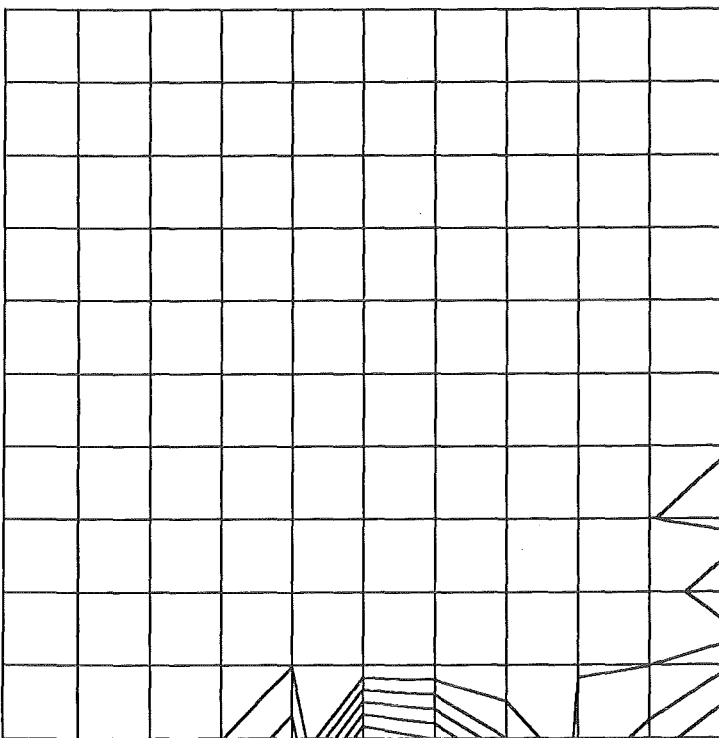


Abbildung 27. Isolinien für die Advektion mit ORL2D2D im 30. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

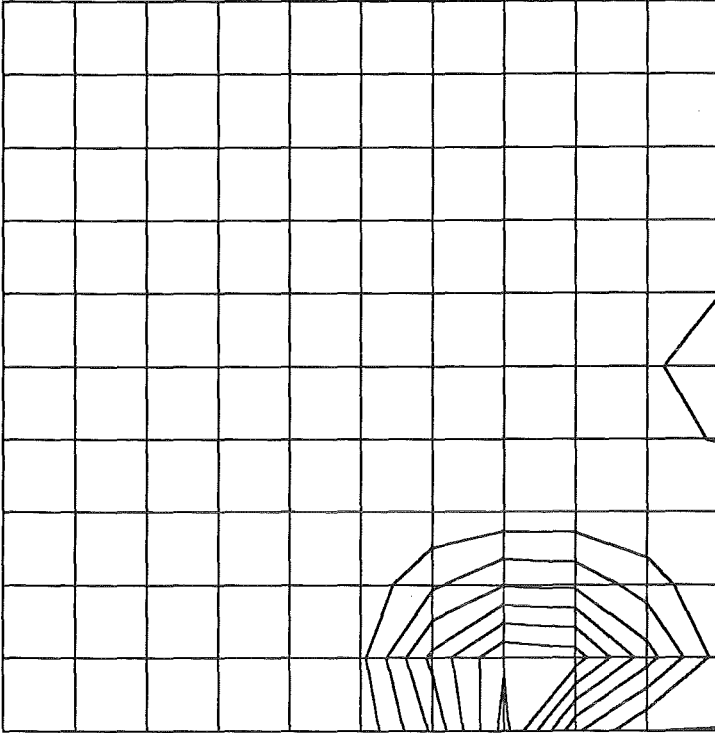


Abbildung 28. Isolinien für die Advektion mit ORL2DAD im 8. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

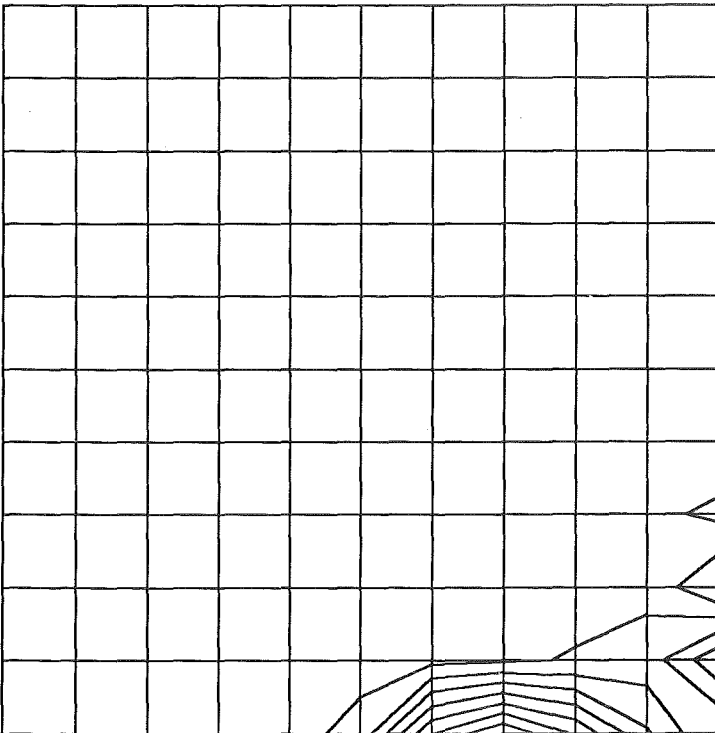
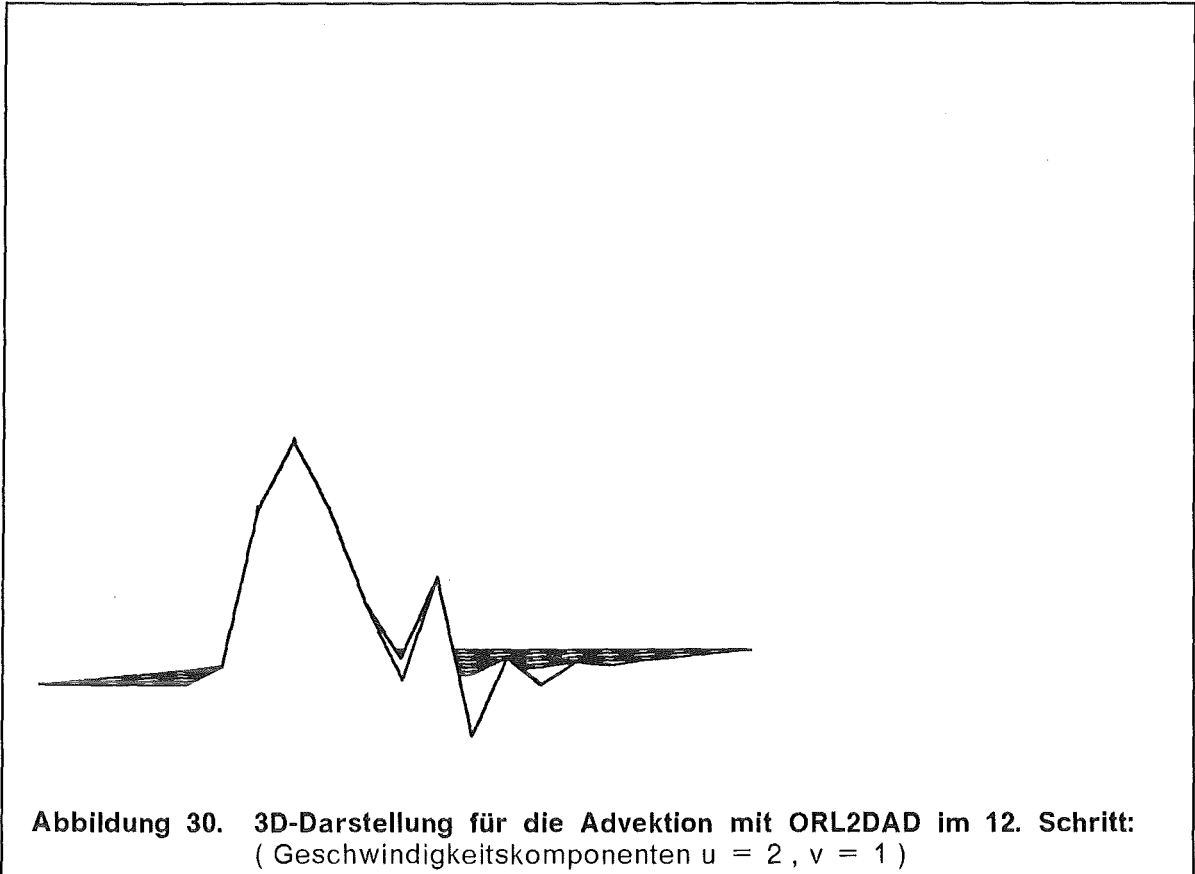


Abbildung 29. Isolinien für die Advektion mit ORL2DAD im 12. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )



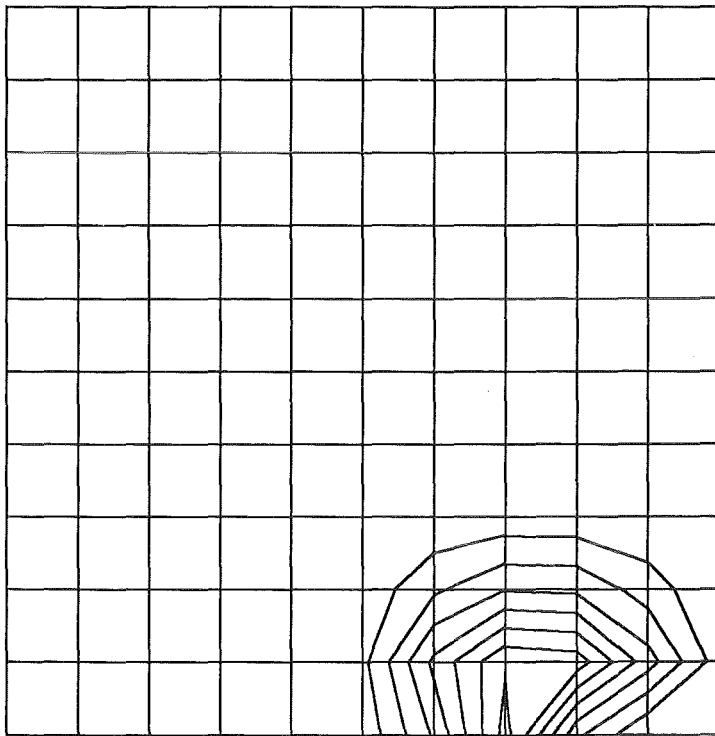


Abbildung 31. Isolinien für die Advektion mit ORL2DADS im 8. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )

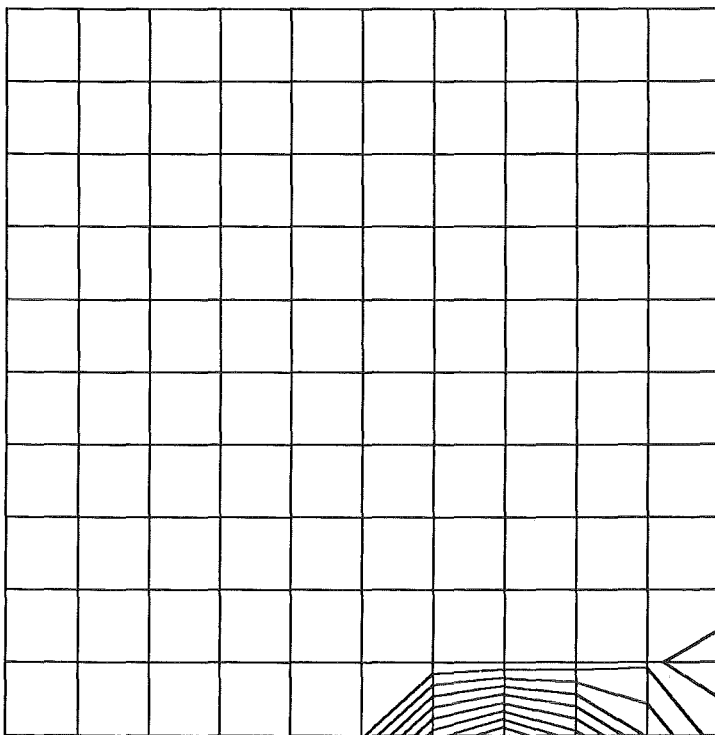
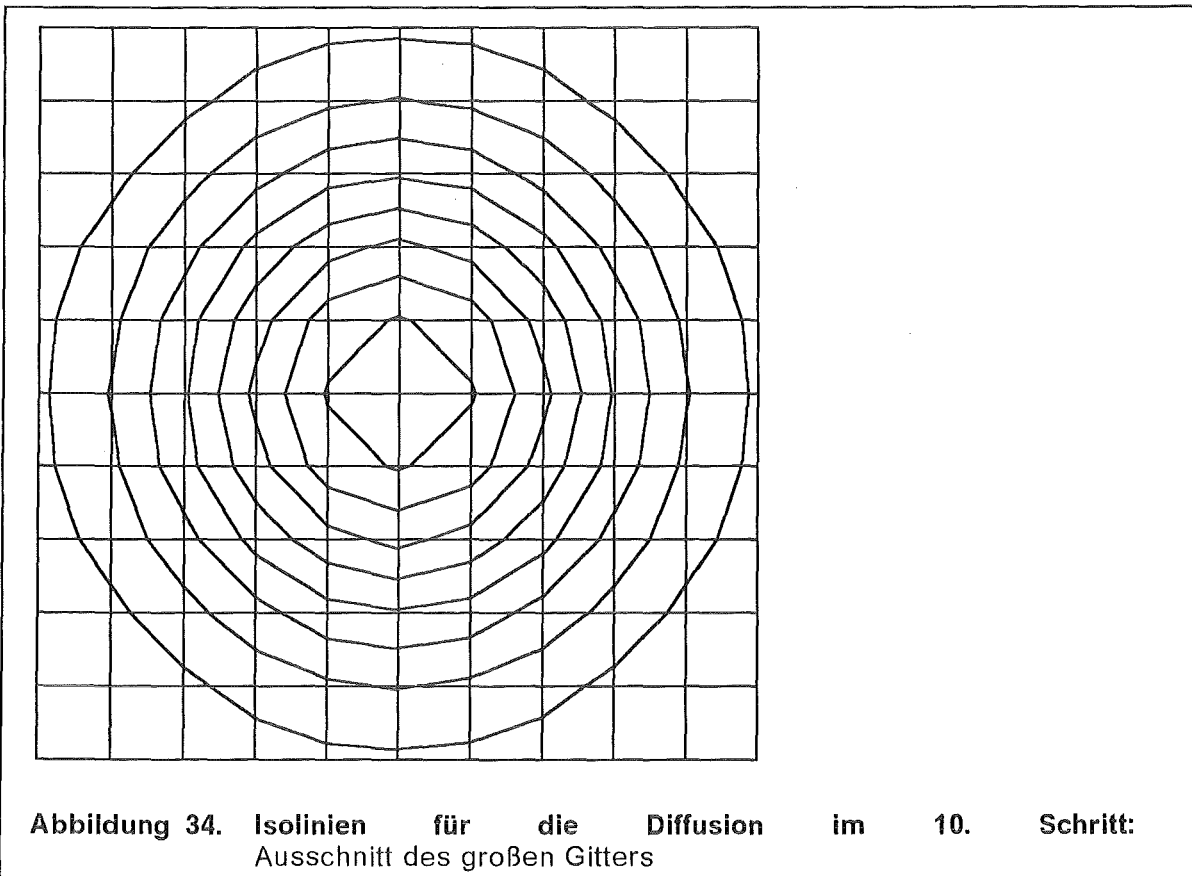
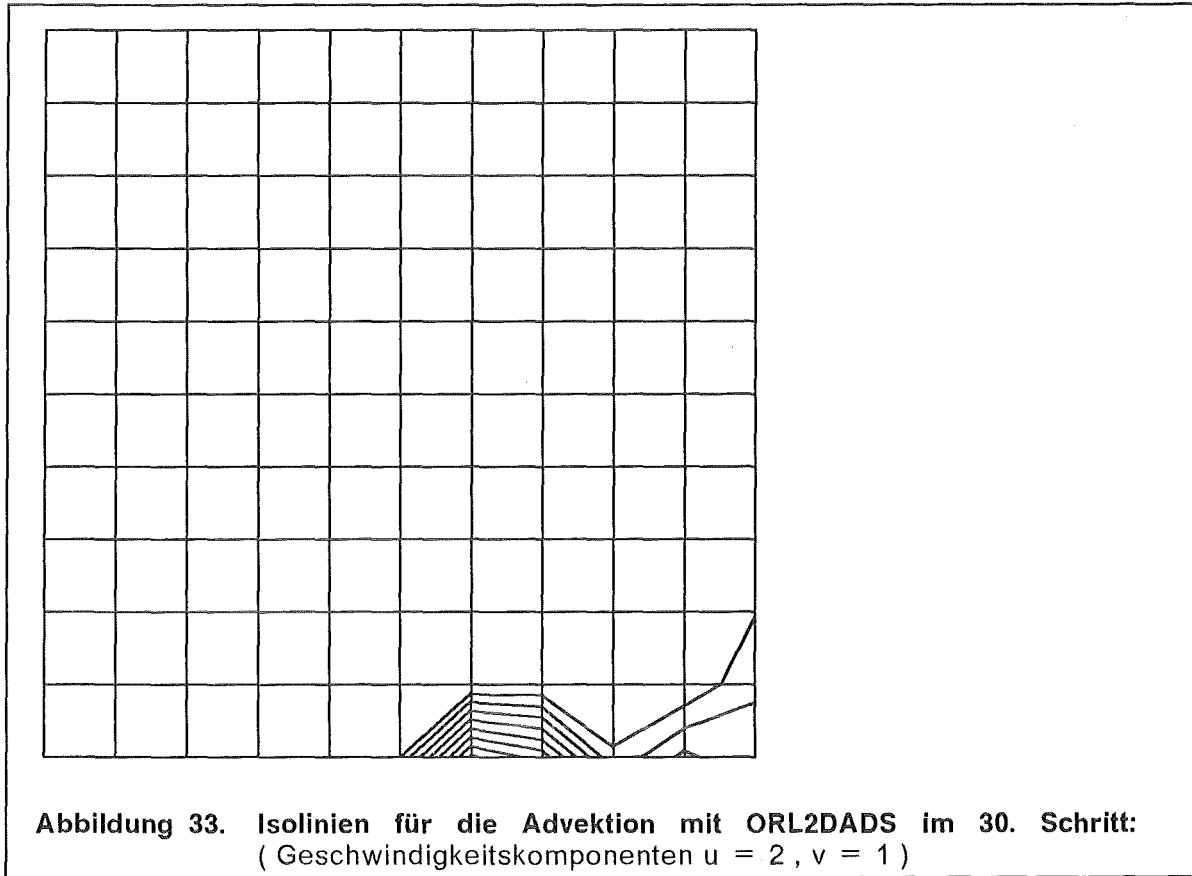


Abbildung 32. Isolinien für die Advektion mit ORL2DADS im 12. Schritt:  
 ( Geschwindigkeitskomponenten  $u = 2, v = 1$  )





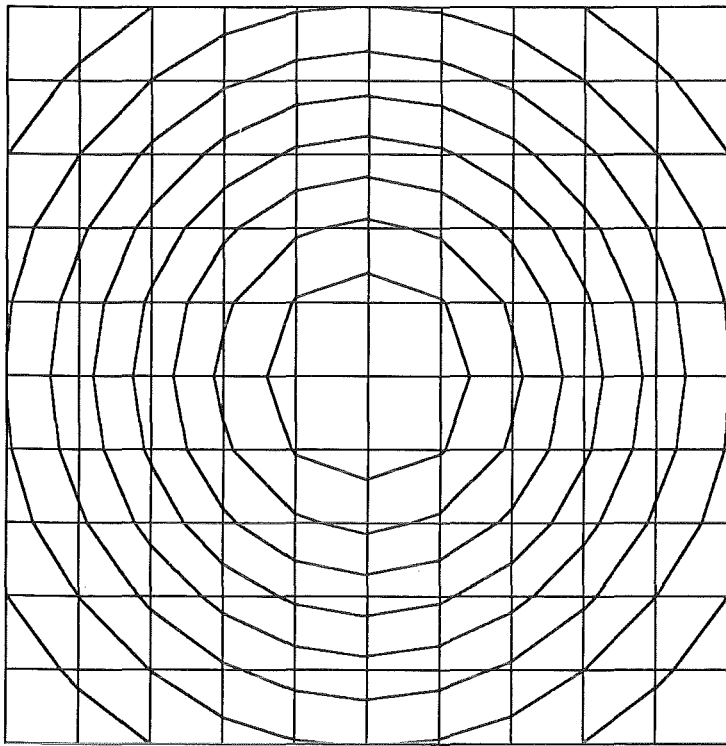


Abbildung 35. Isolinien für die Diffusion im 20. Schritt:  
Ausschnitt des großen Gitters

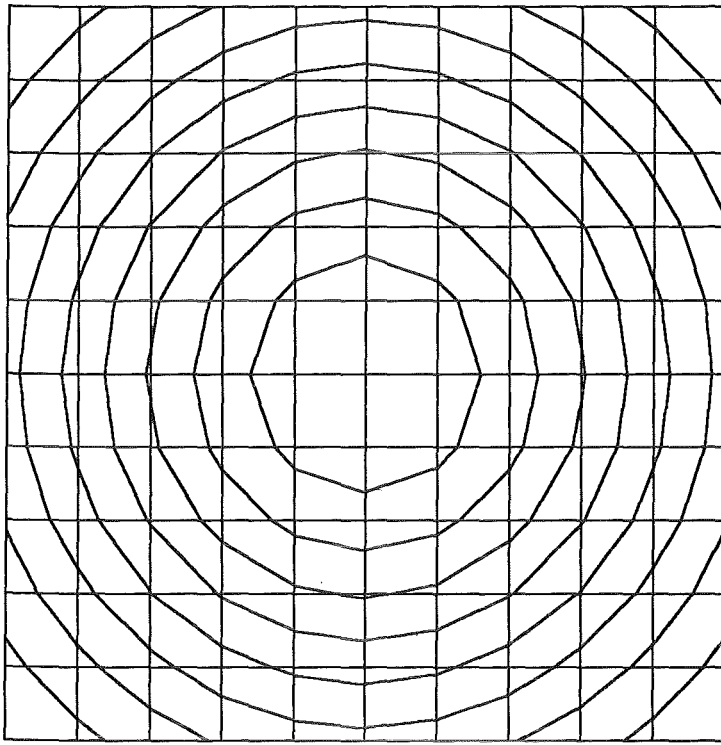


Abbildung 36. Isolinien für die Diffusion im 30. Schritt:  
Ausschnitt des großen Gitters

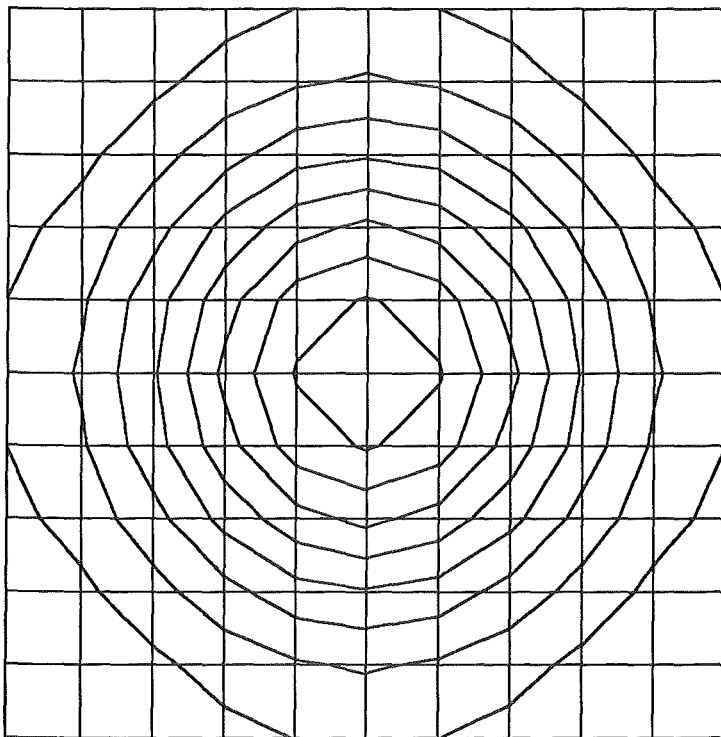


Abbildung 37. Isolinien für die Diffusion mit  $S(0) = S(1)$  im 10. Schritt

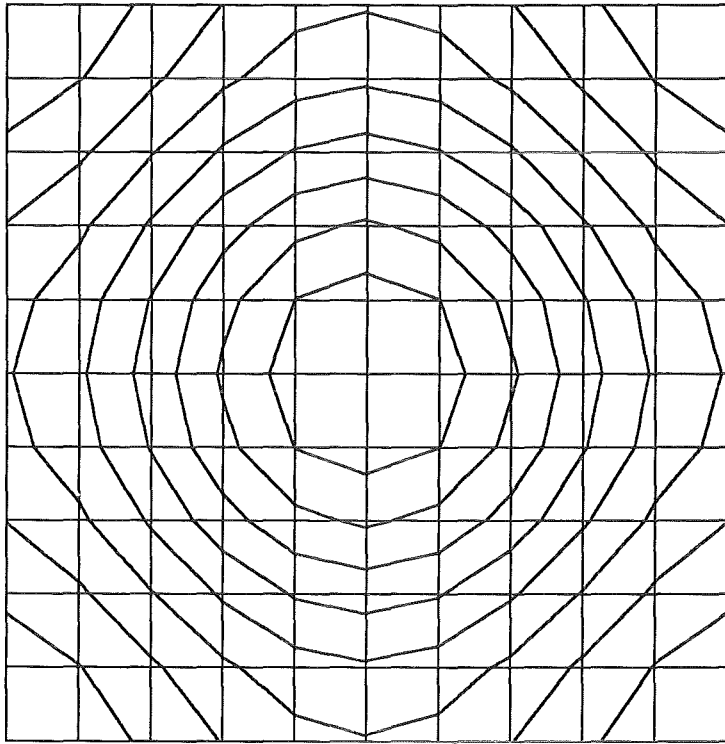


Abbildung 38. Isolinen für die Diffusion mit  $S(0) = S(1)$  im 20. Schritt

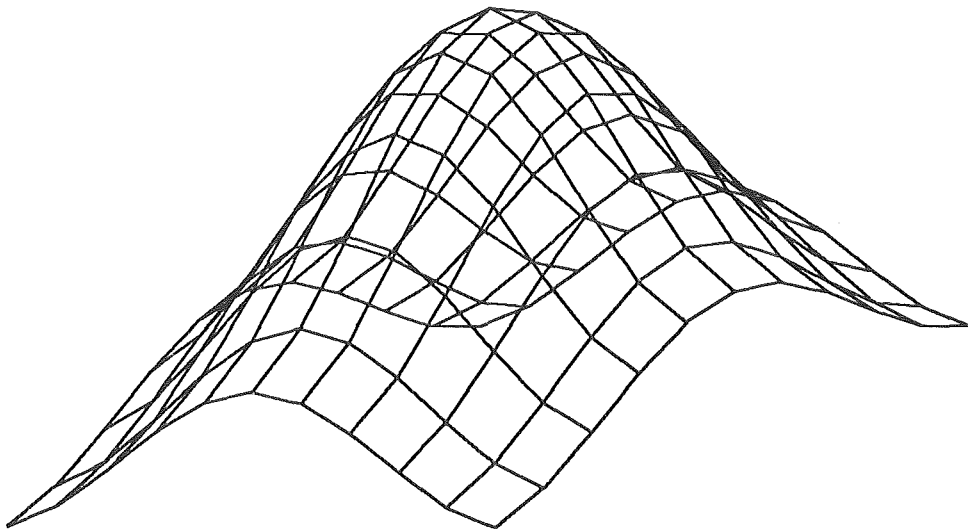


Abbildung 39. 3D-Netz für die Diffusion mit  $S(0) = S(1)$  im 20. Schritt

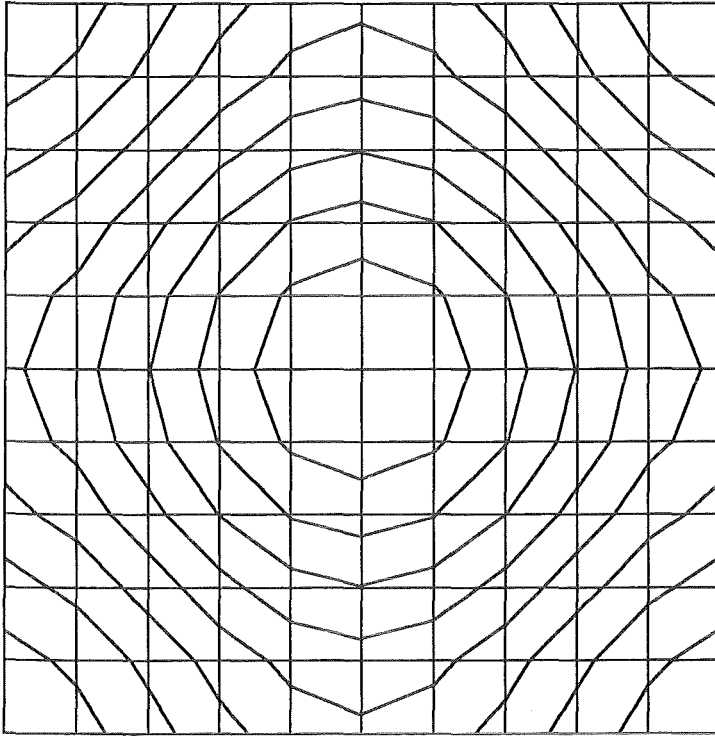


Abbildung 40. Isolinien für die Diffusion mit  $S(0) = S(1)$  im 30. Schritt

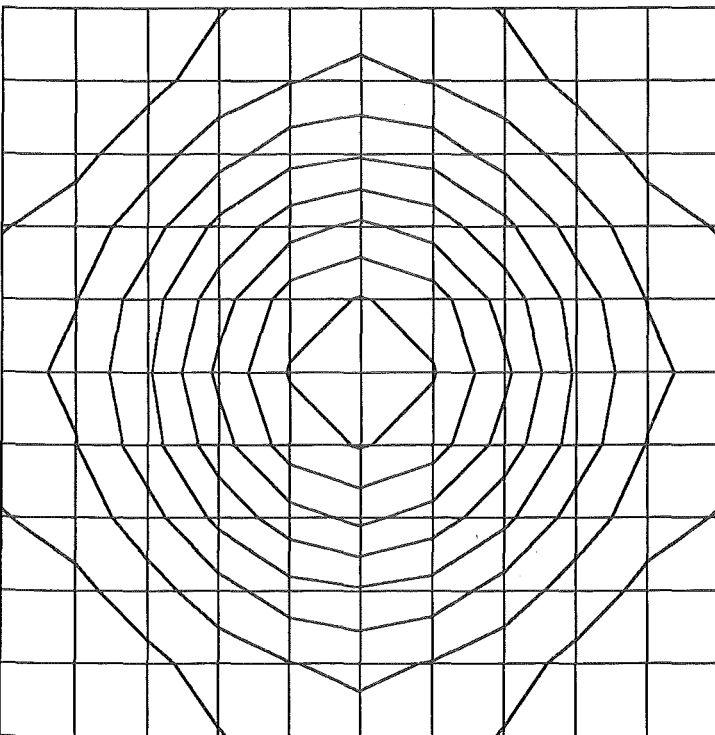
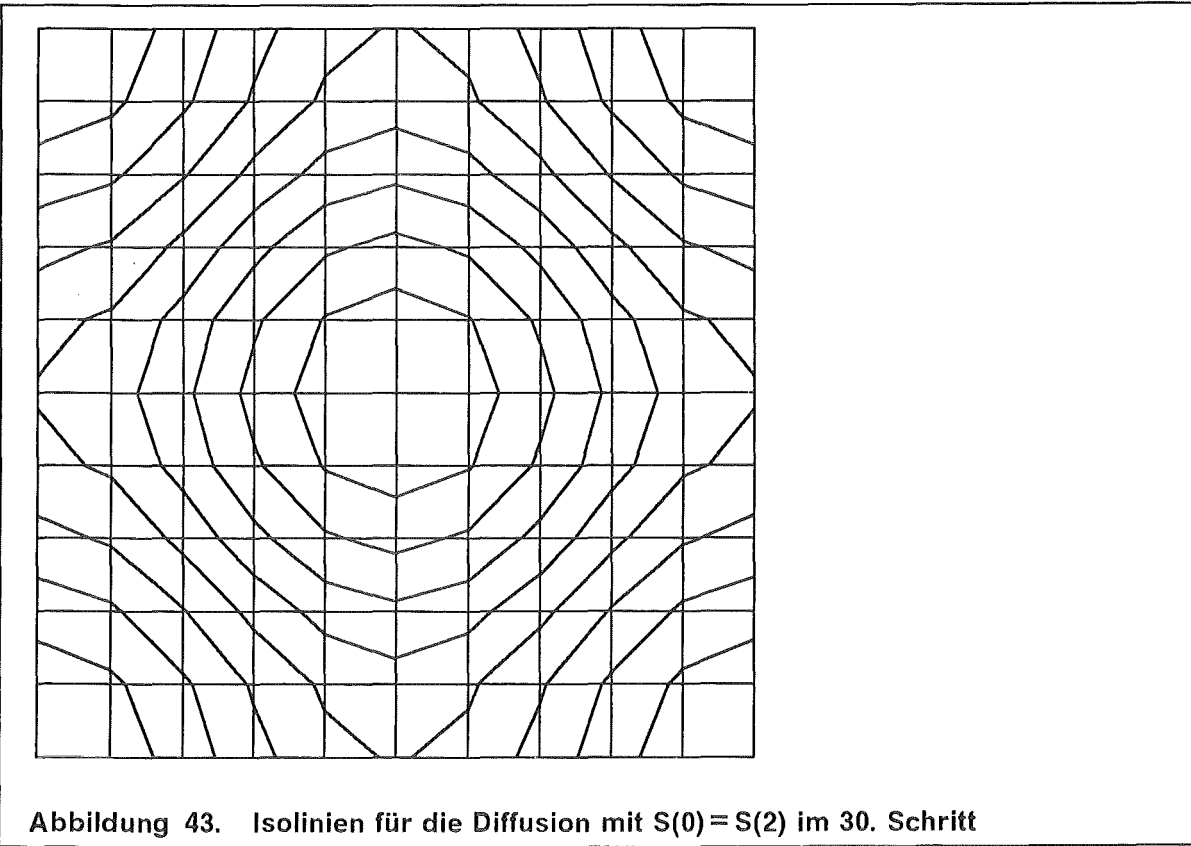
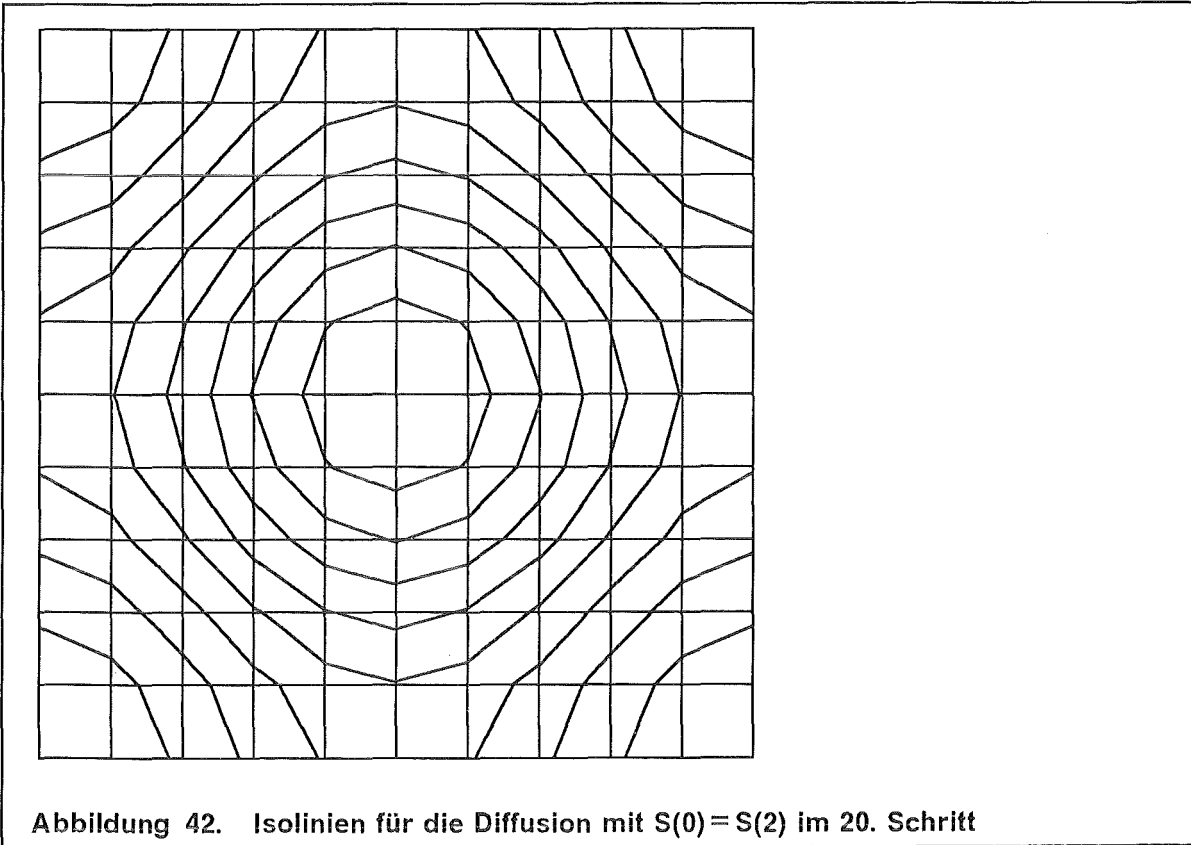


Abbildung 41. Isolinien für die Diffusion mit  $S(0) = S(2)$  im 10. Schritt



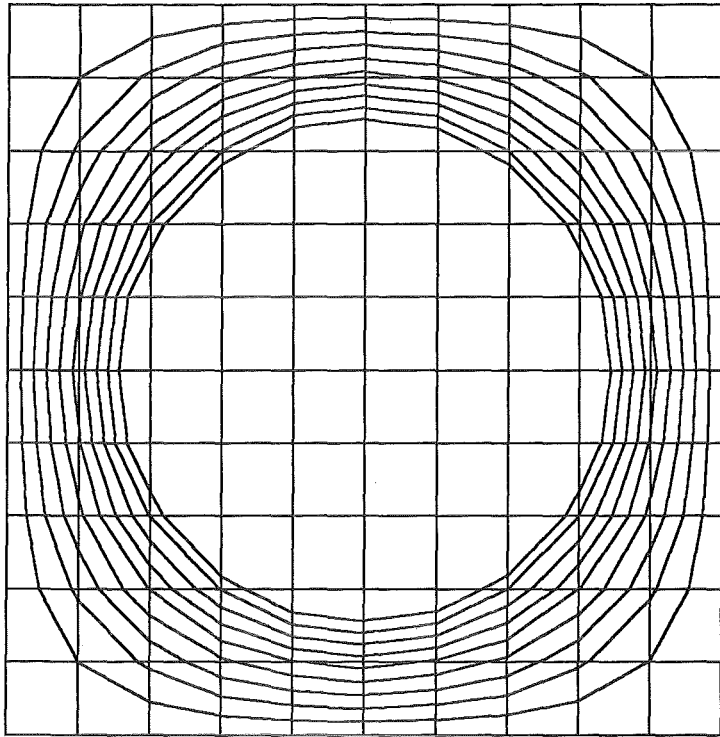


Abbildung 44. Isolinien für die Diffusion mit  $S(0) = 2S(1) - S(2)$  im 10. Schritt

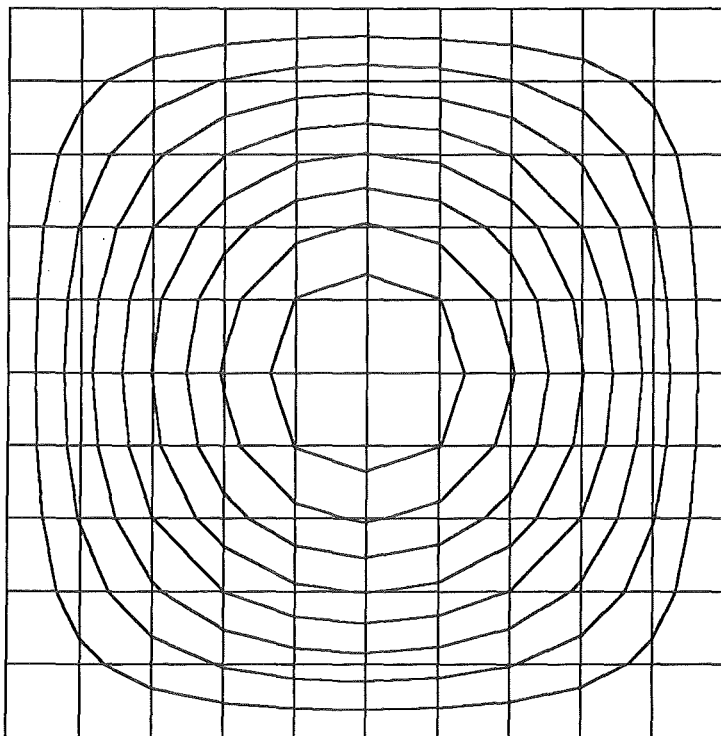


Abbildung 45. Isolinien für die Diffusion mit  $S(0) = 2S(1) - S(2)$  im 20. Schritt

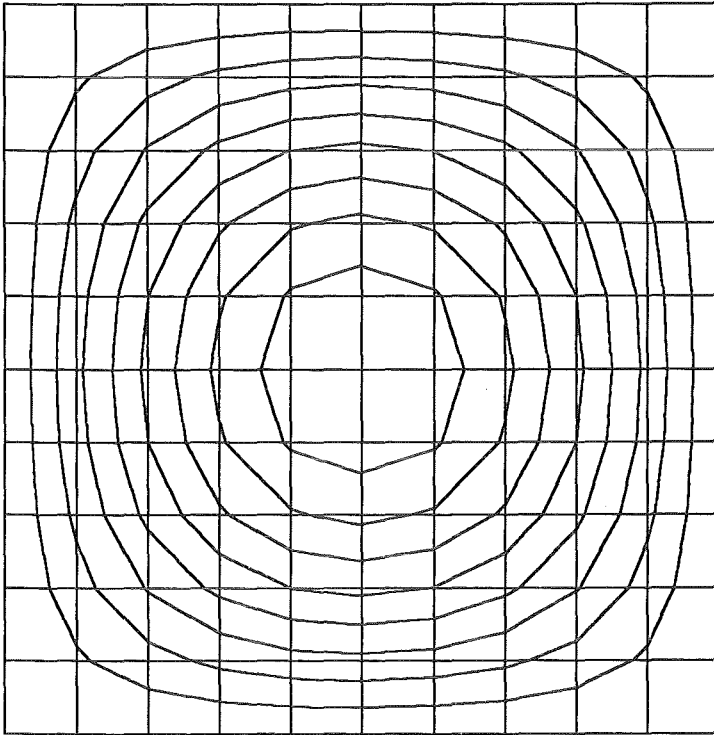


Abbildung 46. Isolinien für die Diffusion mit  $S(0) = 2S(1) - S(2)$  im 30. Schritt

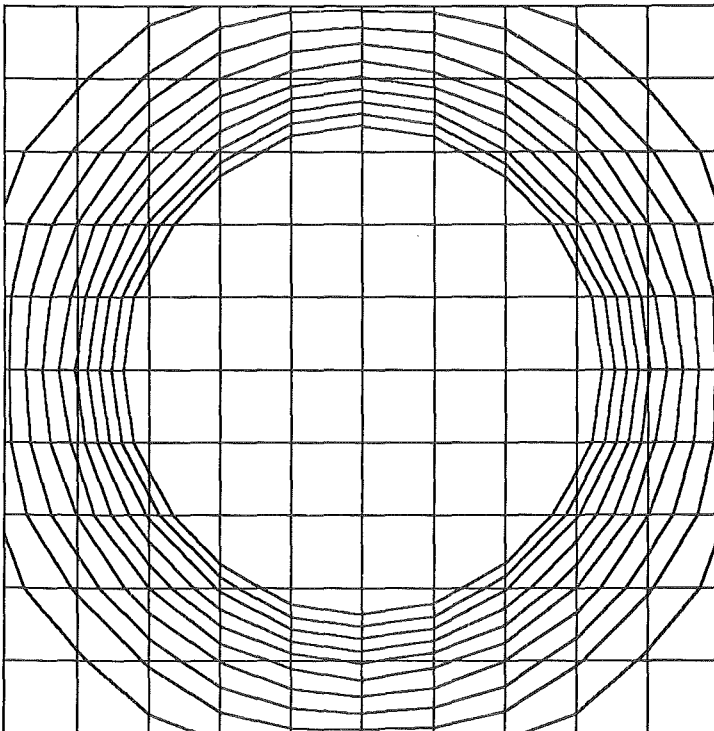


Abbildung 47. Isolinien für die Diffusion mit ORL1D2D im 10. Schritt

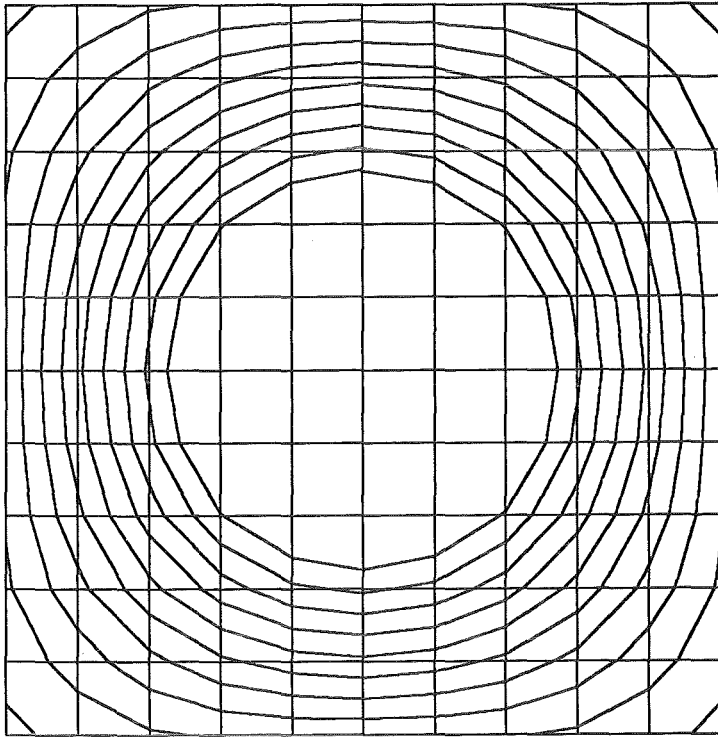


Abbildung 48. Isolinien für die Diffusion mit ORL1D2D im 20. Schritt

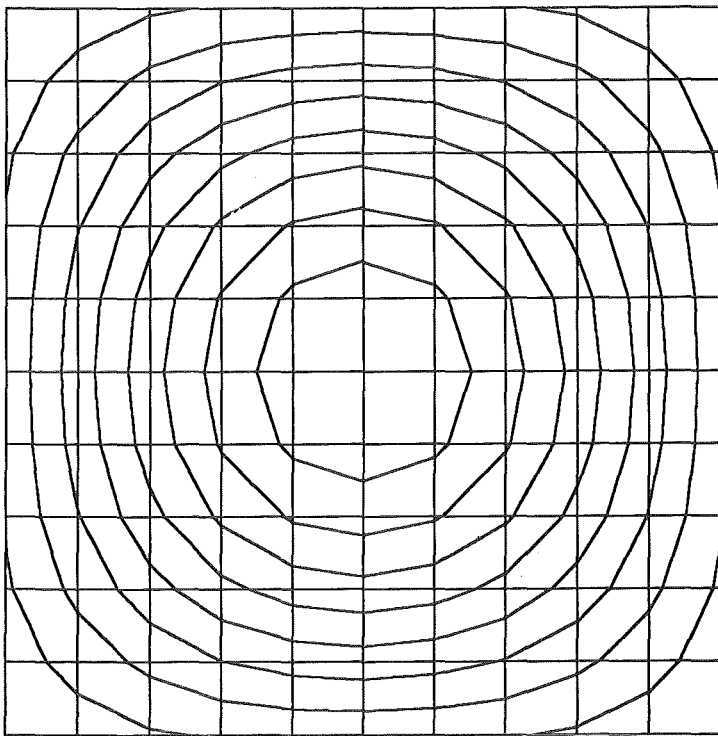
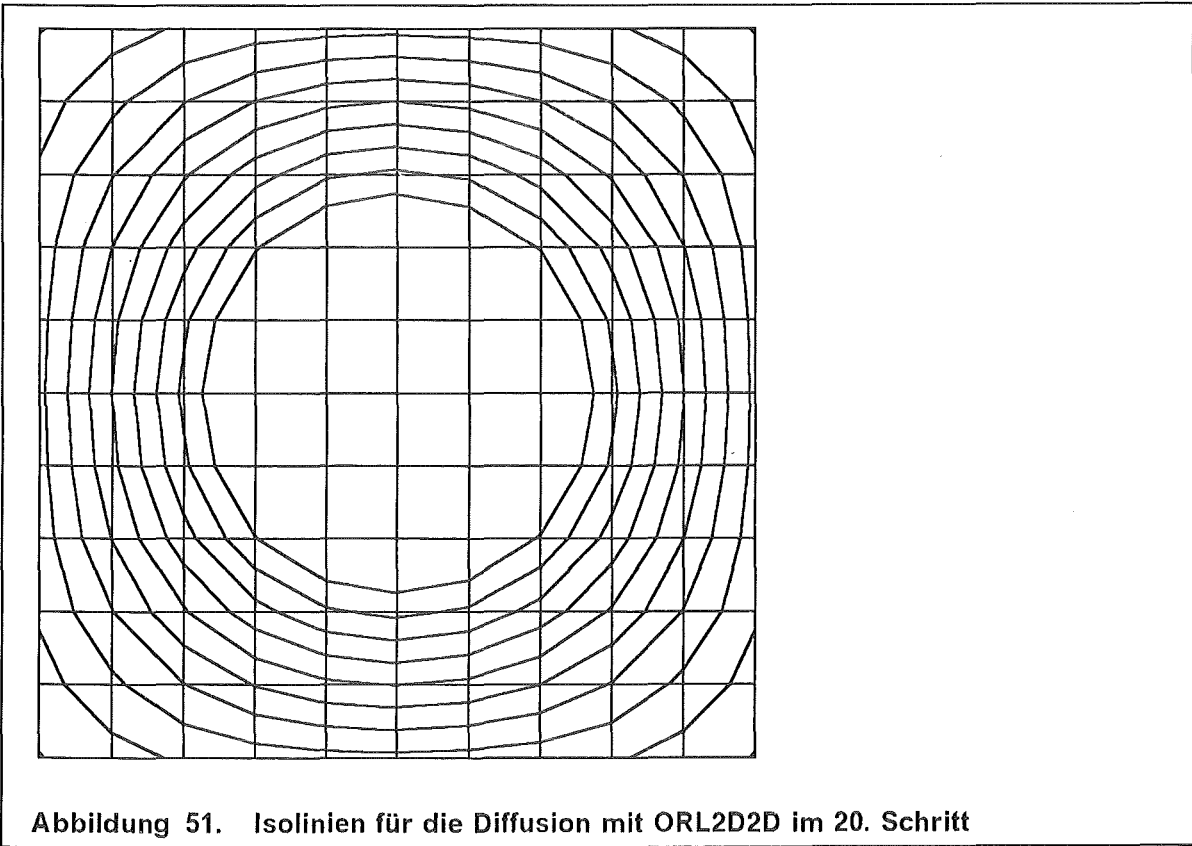
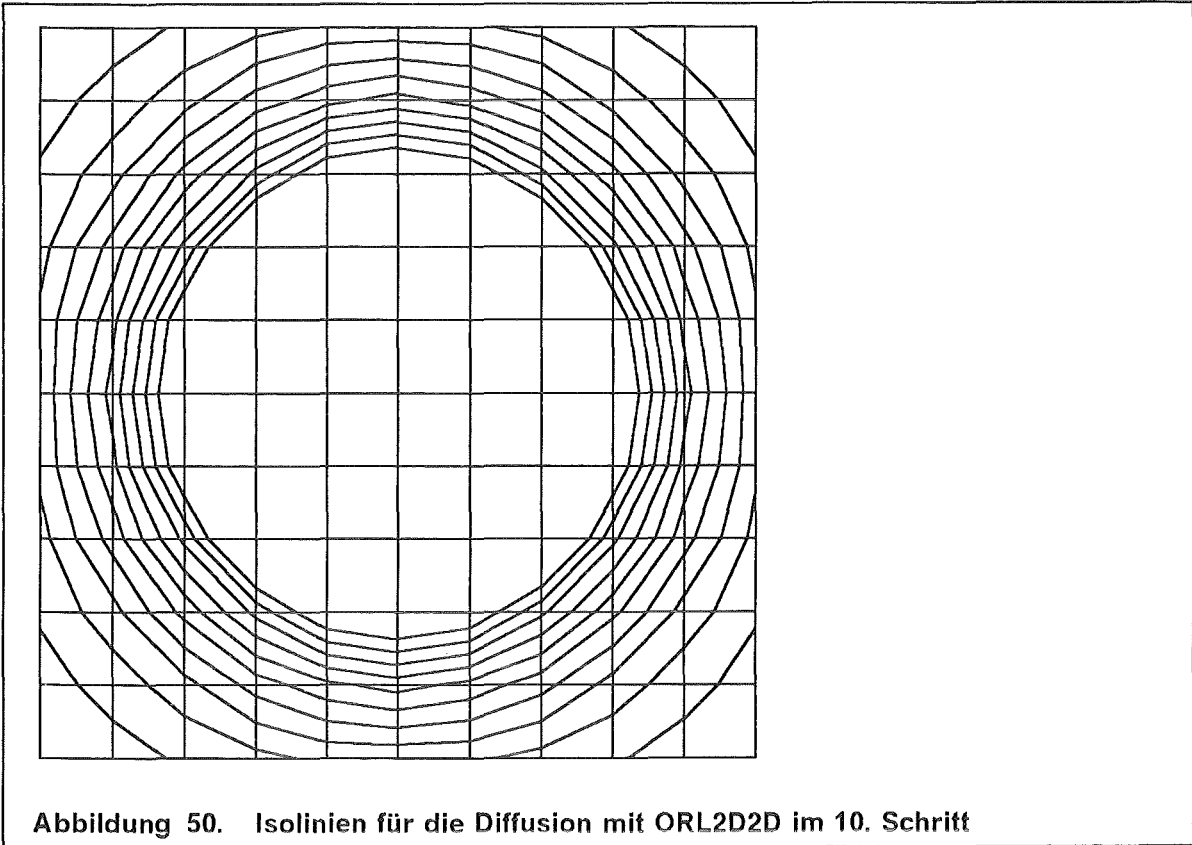


Abbildung 49. Isolinien für die Diffusion mit ORL1D2D im 30. Schritt





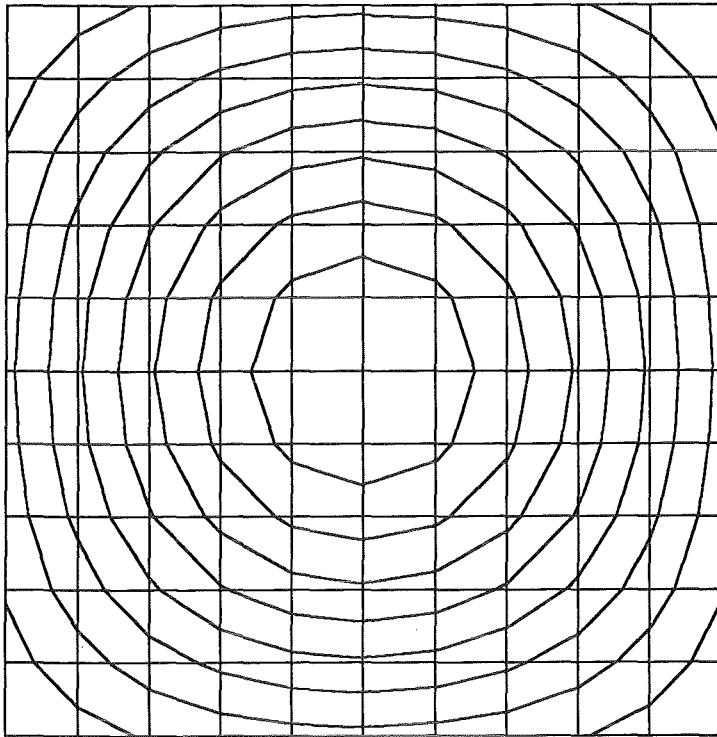


Abbildung 52. Isolinien für die Diffusion mit ORL2D2D im 30. Schritt

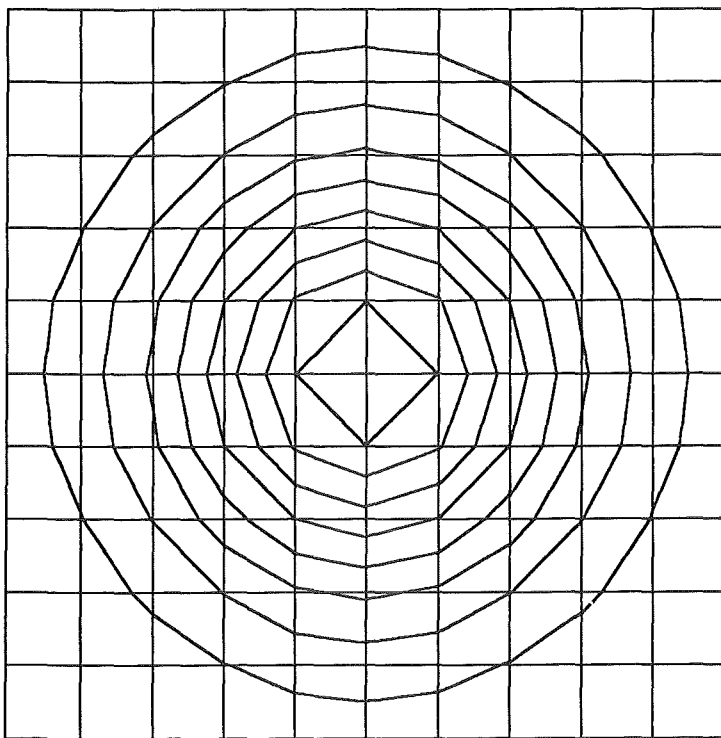


Abbildung 53. Isolinien für die Diffusion (implizit) im 10. Schritt

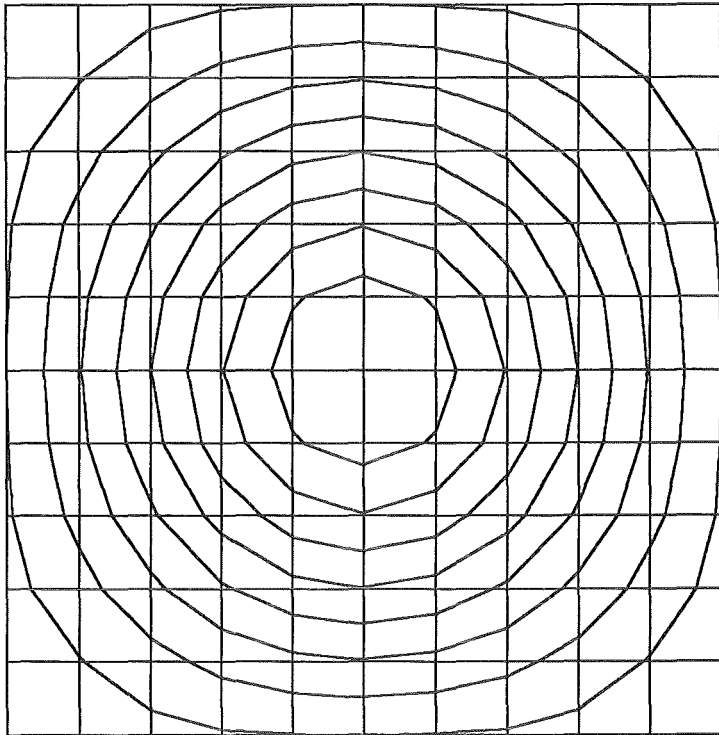


Abbildung 54. Isolinien für die Diffusion (implizit) im 20. Schritt

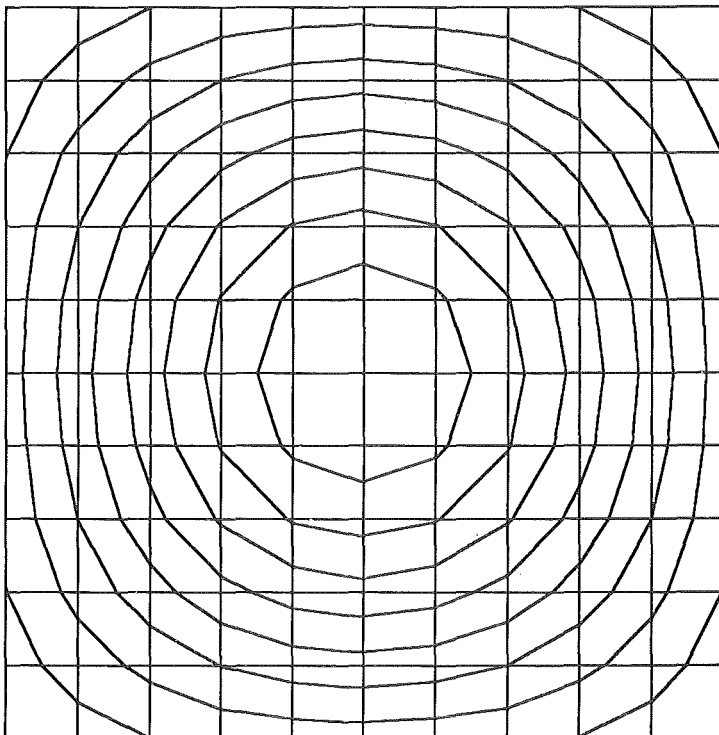


Abbildung 55. Isolinien für die Diffusion (implizit) im 30. Schritt

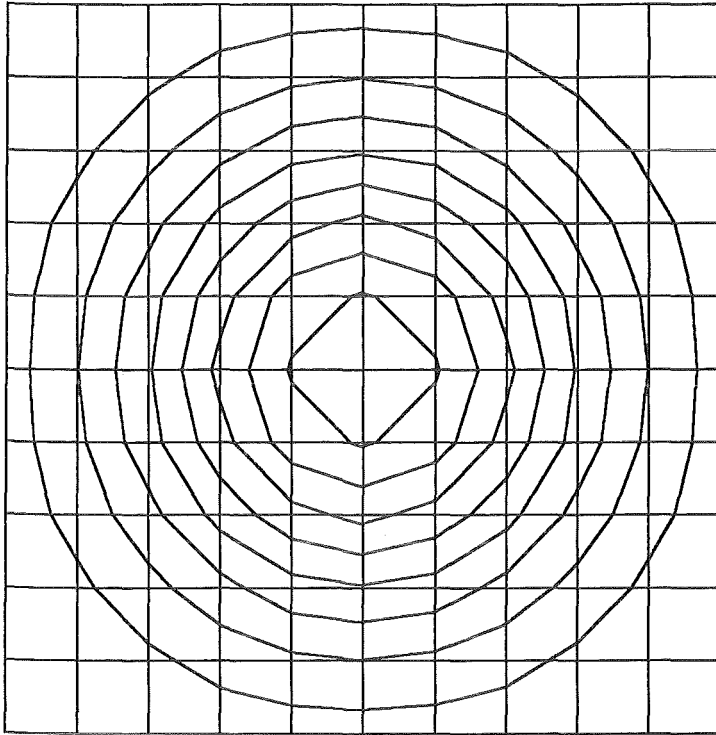


Abbildung 56. Isolinien für die Diffusion mit SORL2D2D im 10. Schritt

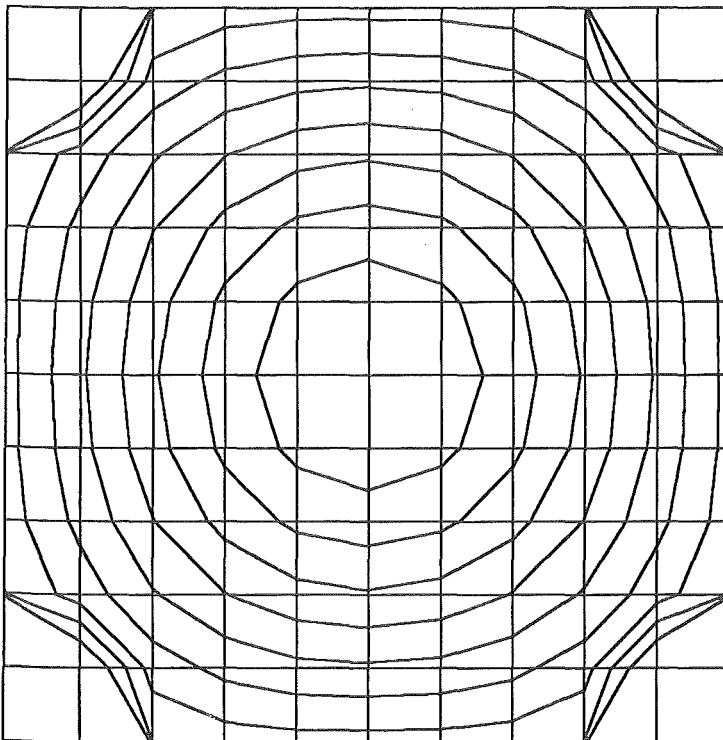
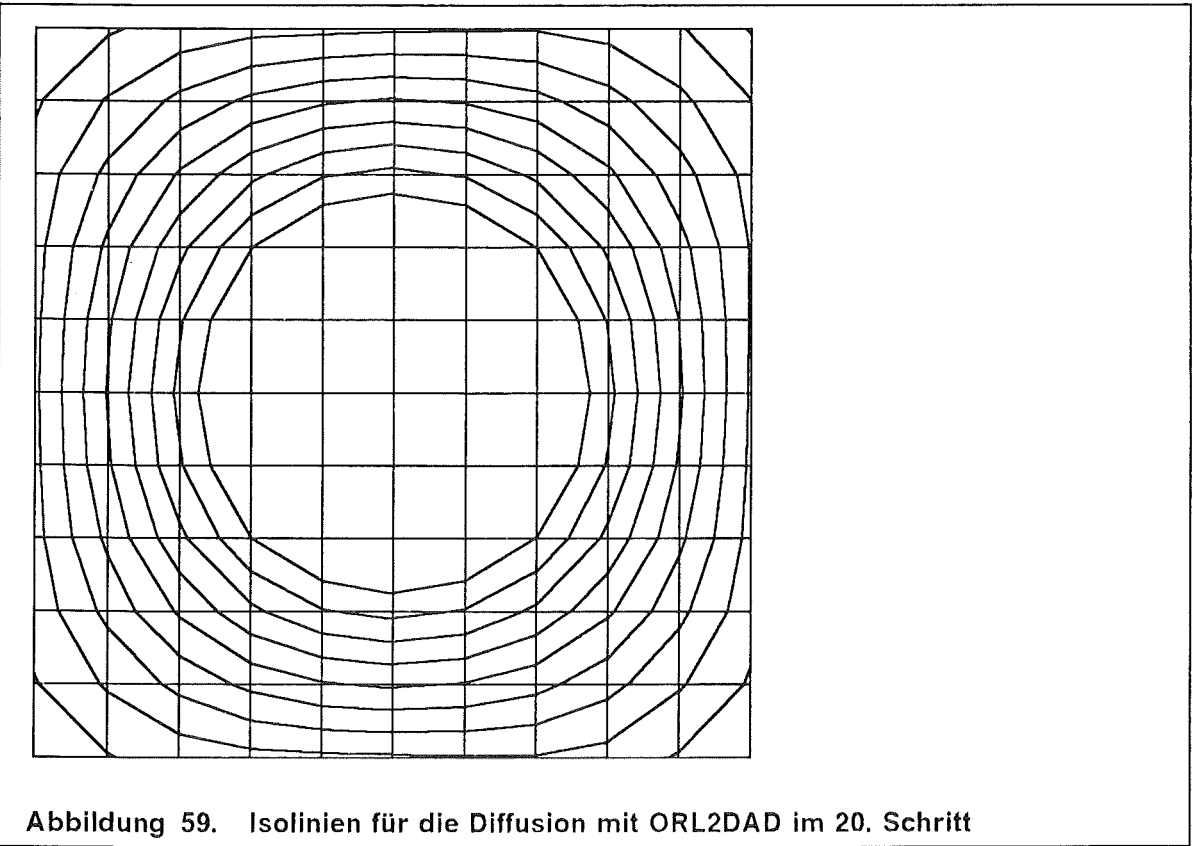
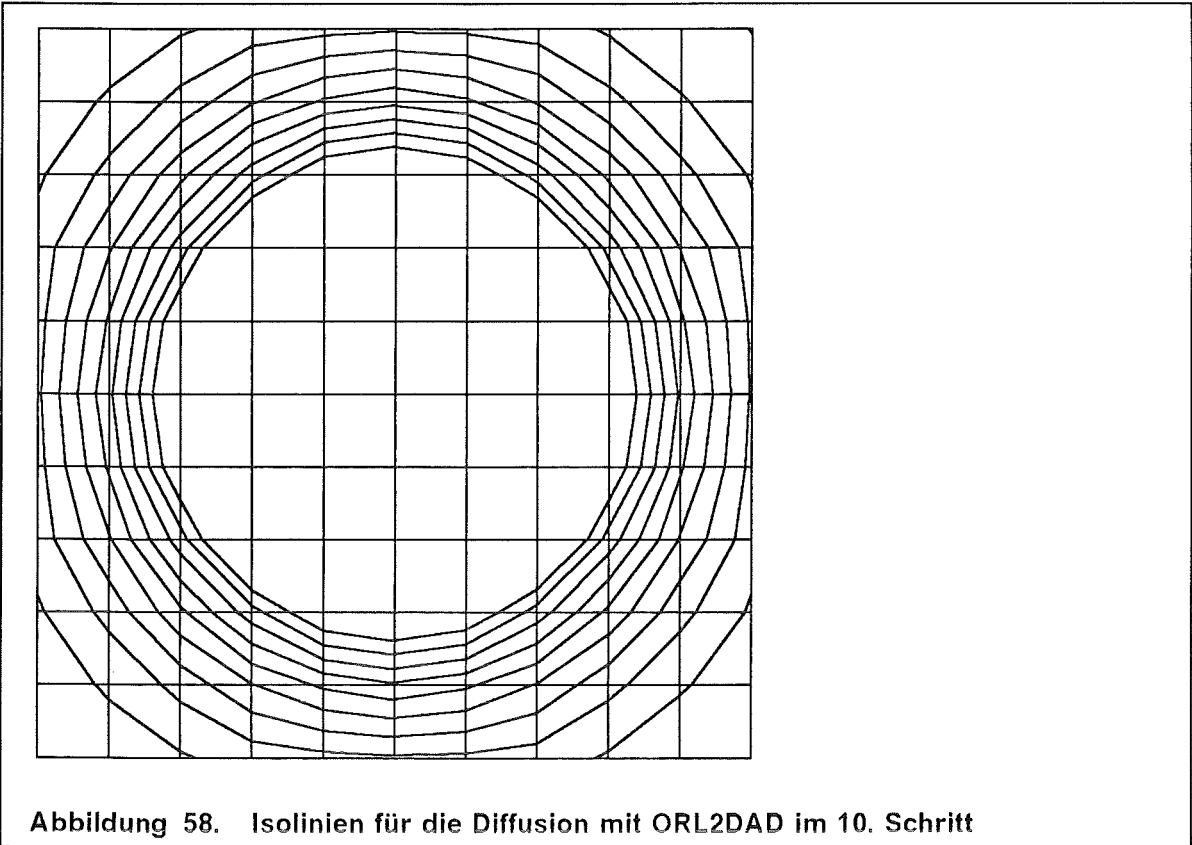


Abbildung 57. Isolinien für die Diffusion mit SORL2D2D im 30. Schritt



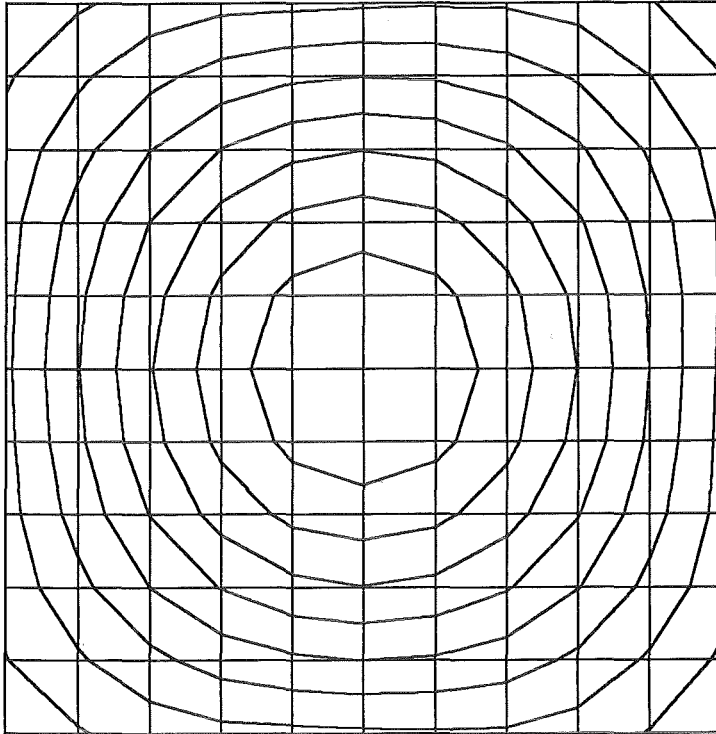


Abbildung 60. Isolinien für die Diffusion mit ORL2DAD im 30. Schritt

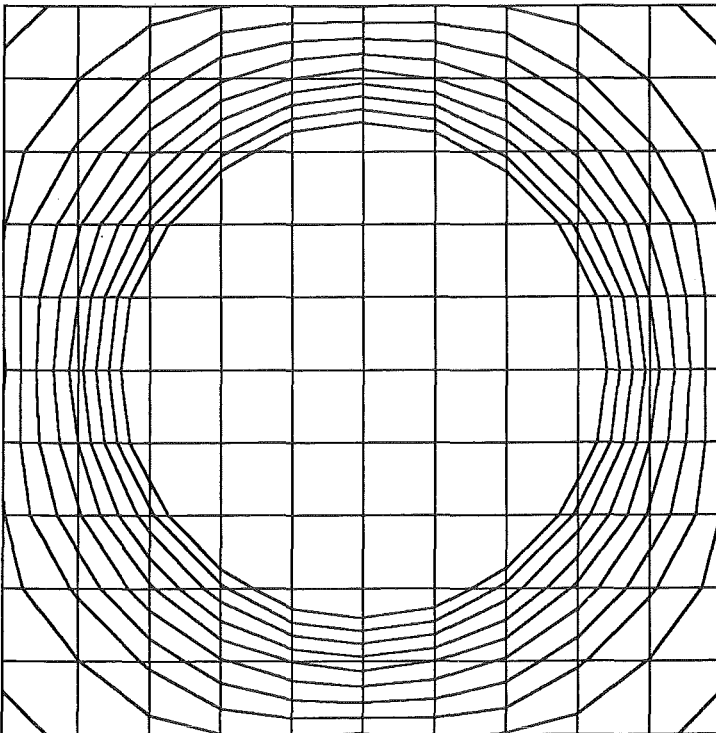


Abbildung 61. Isolinien für die Diffusion mit ORL2DADS im 10. Schritt

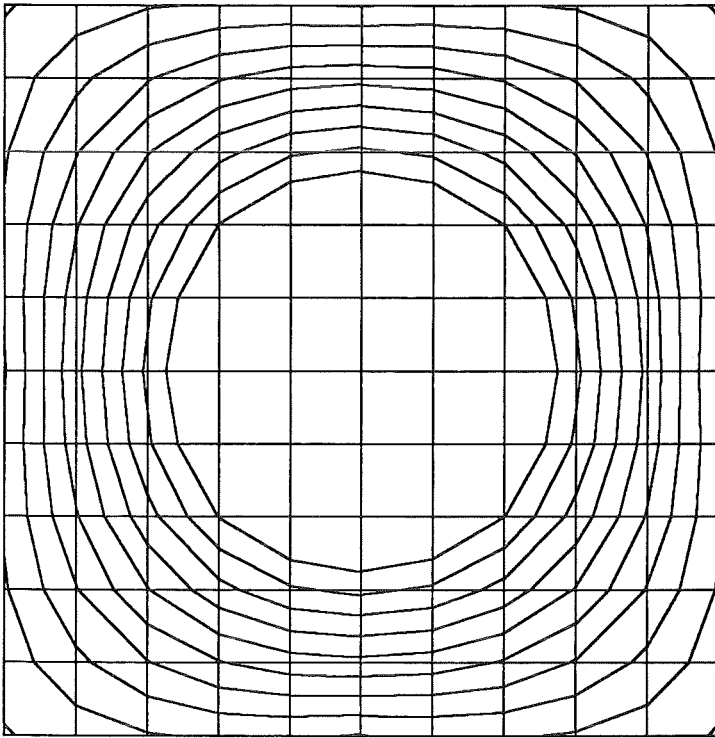


Abbildung 62. Isolinien für die Diffusion mit ORL2DADS im 20. Schritt

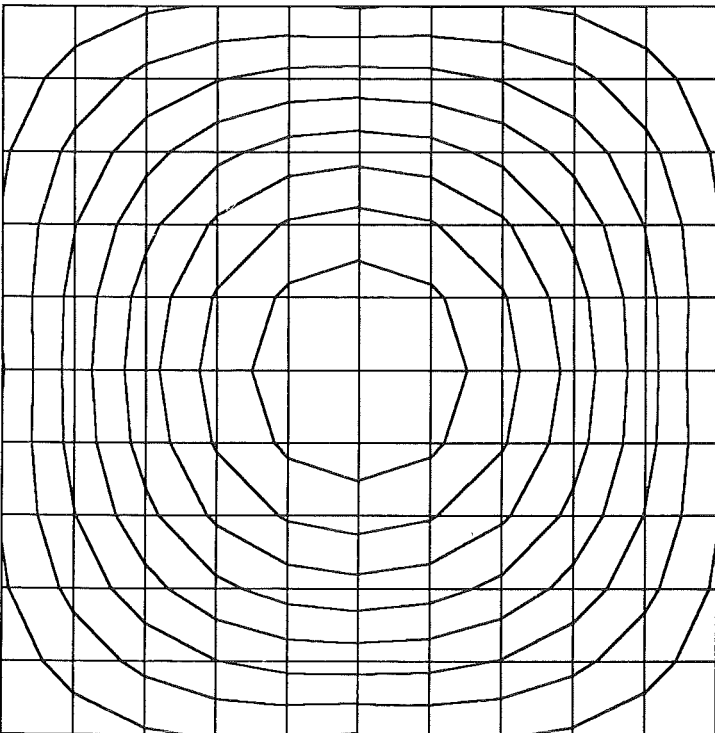


Abbildung 63. Isolinien für die Diffusion mit ORL2DADS im 30. Schritt

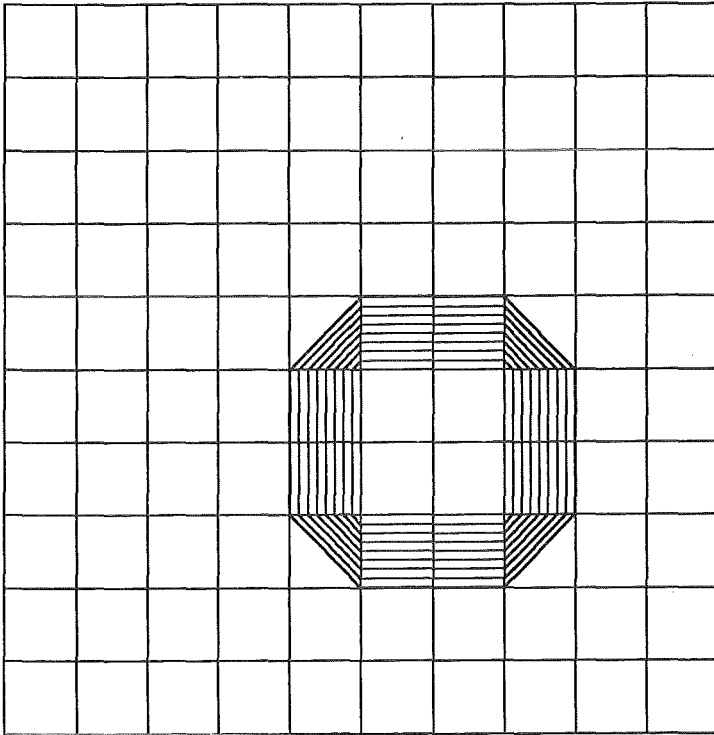


Abbildung 64. Isolinien für die Anfangsbedingungen des versetzten Kastens

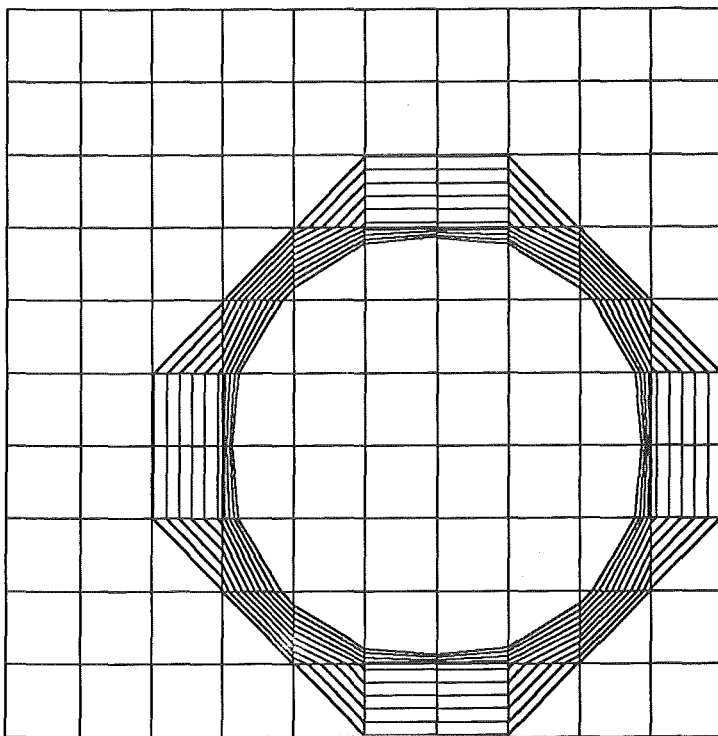
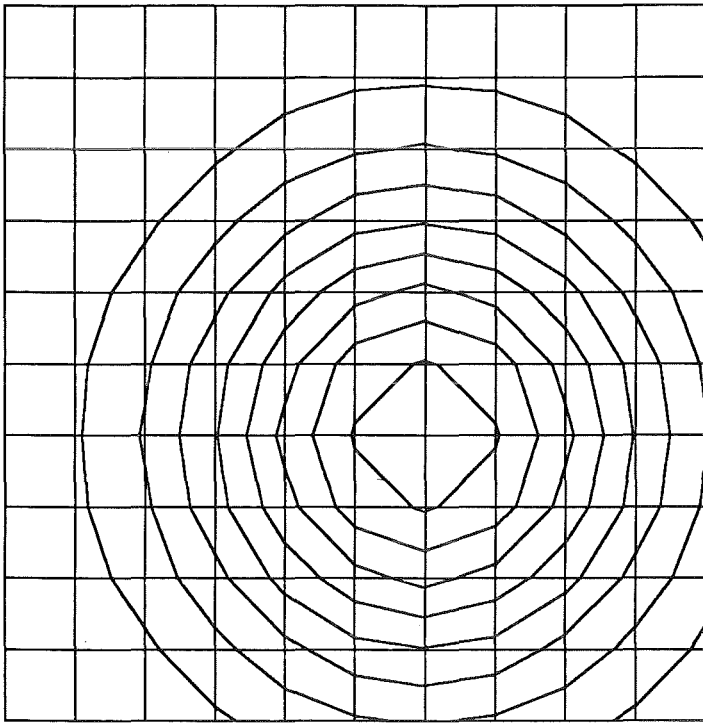
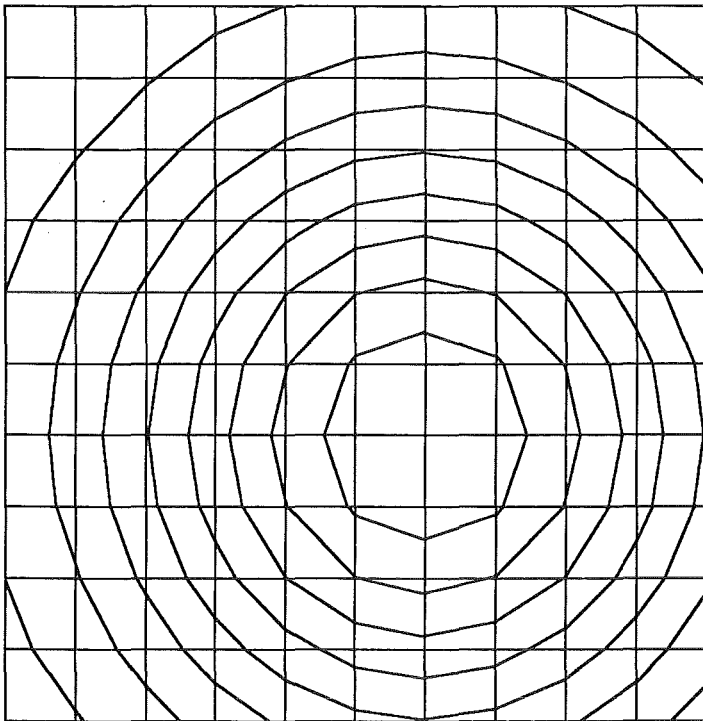


Abbildung 65. Isolinien für die Diffusion nach 2 Schritten:  
Versetzer Kasten

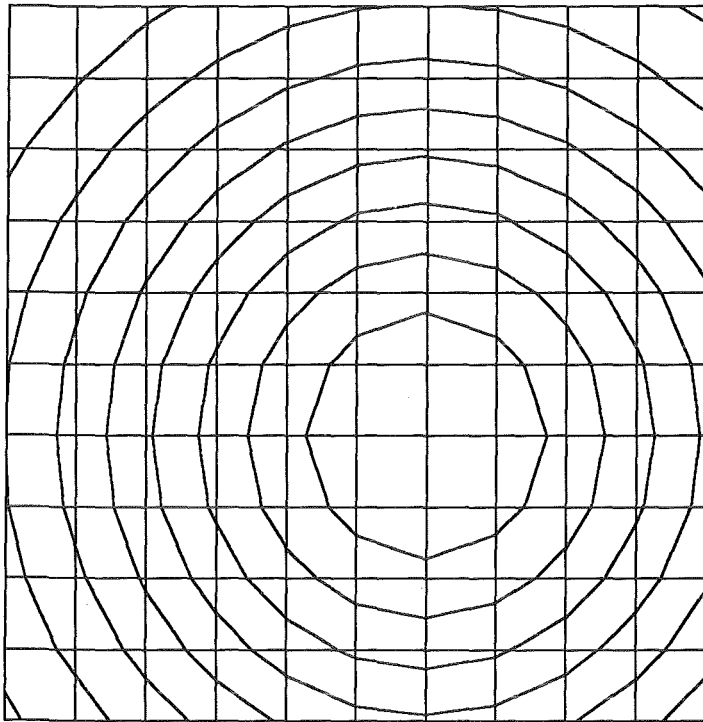




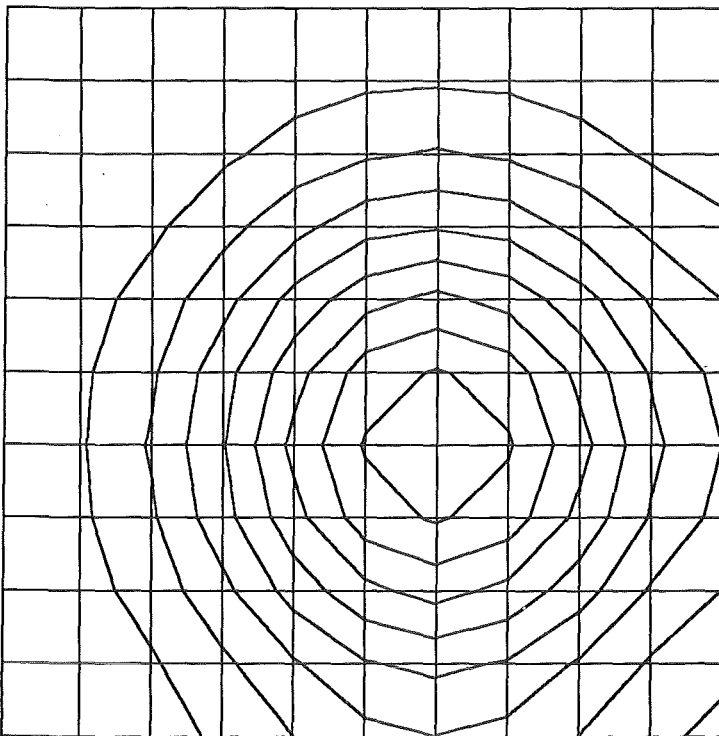
**Abbildung 66. Isolinien für die Diffusion im 10. Schritt:**  
Ausschnitt des großen Gitters, versetzt



**Abbildung 67. Isolinien für die Diffusion im 20. Schritt:**  
Ausschnitt des großen Gitters, versetzt



**Abbildung 68. Isolinien für die Diffusion im 30. Schritt:**  
 Ausschnitt des großen Gitters, versetzt



**Abbildung 69. Isolinien für die Diffusion mit  $S(0)=S(1)$  im 10. Schritt:**  
 Versetzter Kasten

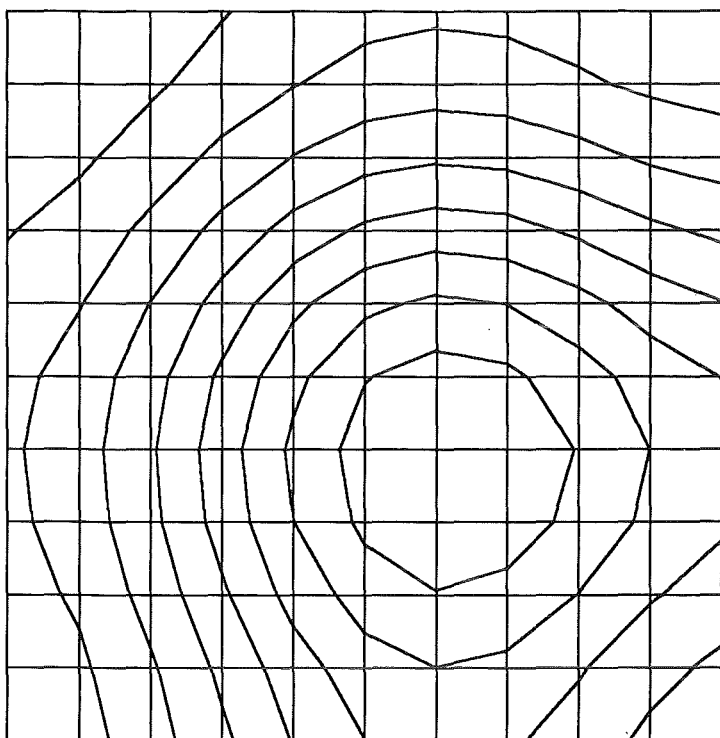


Abbildung 70. Isolinien für die Diffusion mit  $S(0)=S(1)$  im 20. Schritt:  
Versetzter Kasten

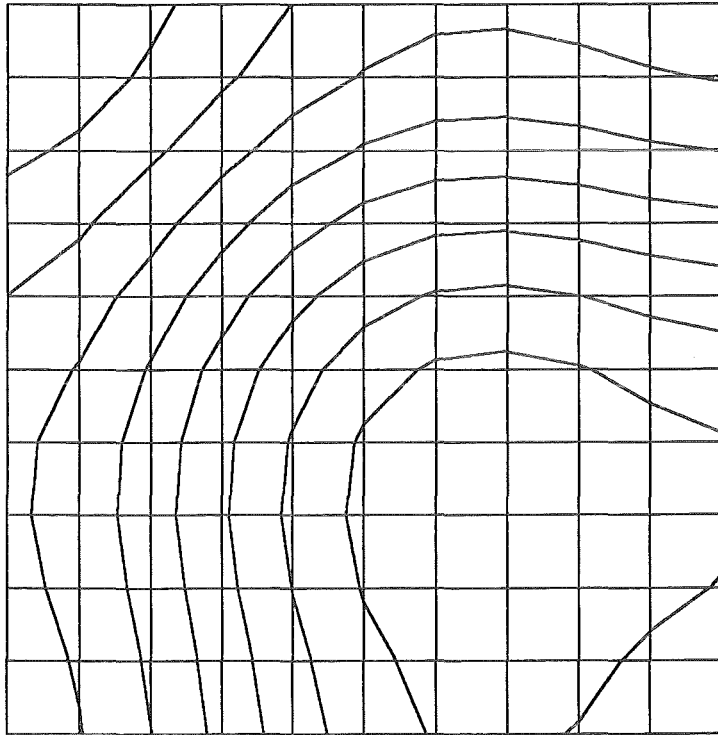


Abbildung 71. Isolinien für die Diffusion mit  $S(0)=S(1)$  im 30. Schritt:  
Versetzter Kasten

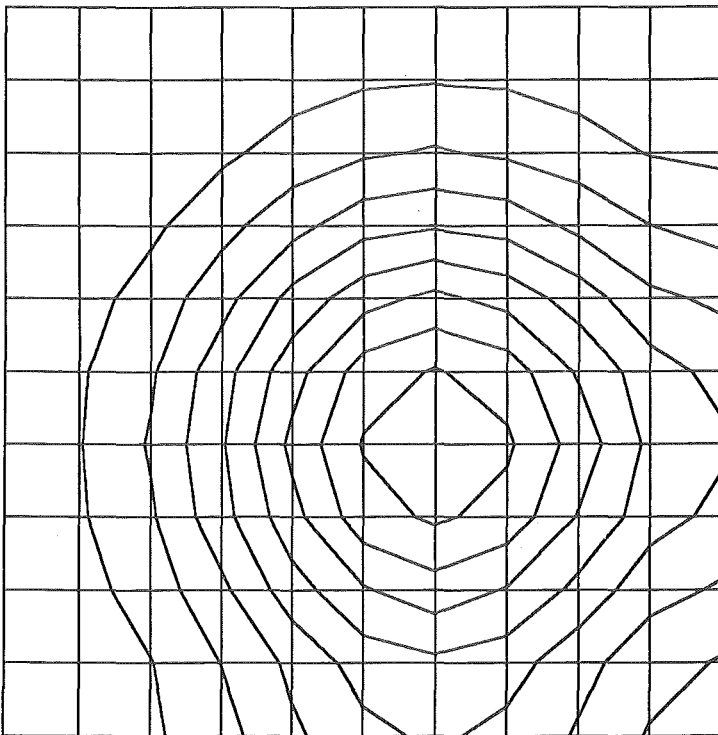


Abbildung 72. Isolinien für die Diffusion mit  $S(0)=S(2)$  im 10. Schritt:  
Versetzter Kasten

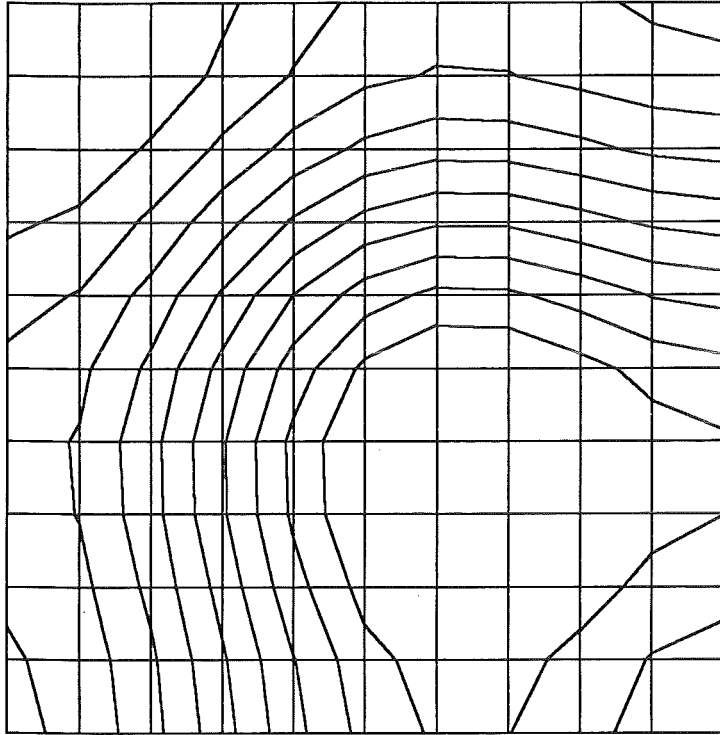


Abbildung 73. Isolinien für die Diffusion mit  $S(0) = S(2)$  im 20. Schritt:  
Versetzter Kasten

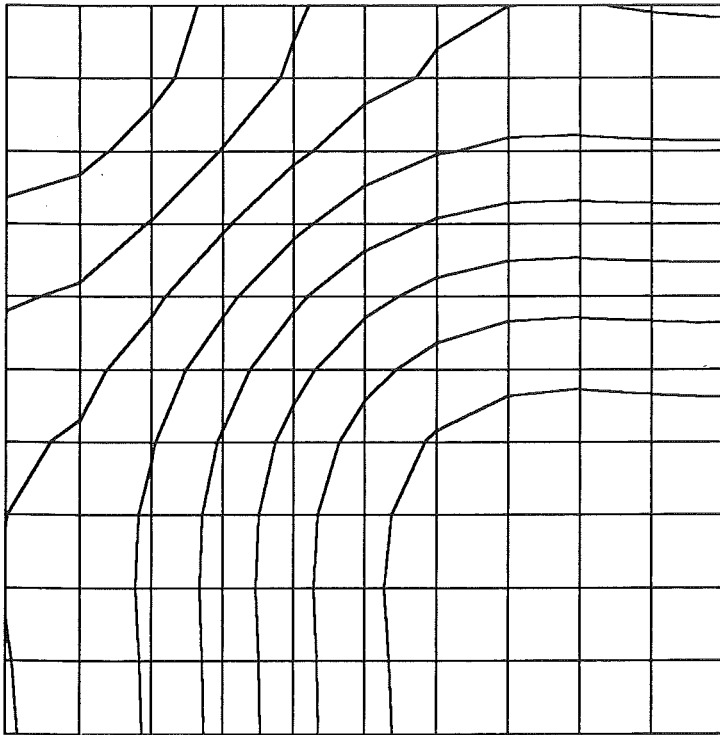
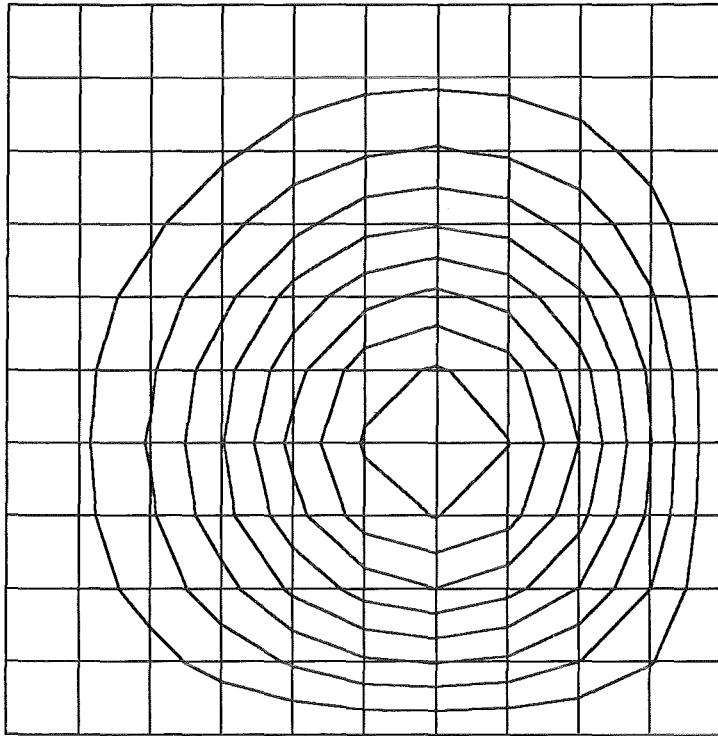


Abbildung 74. Isolinien für die Diffusion mit  $S(0) = S(2)$  im 30. Schritt:  
Versetzter Kasten



**Abbildung 75. Isolinien für die Diffusion mit  $S(0)=2S(1)-S(2)$  im 10. Schritt:  
Versetzter Kasten**

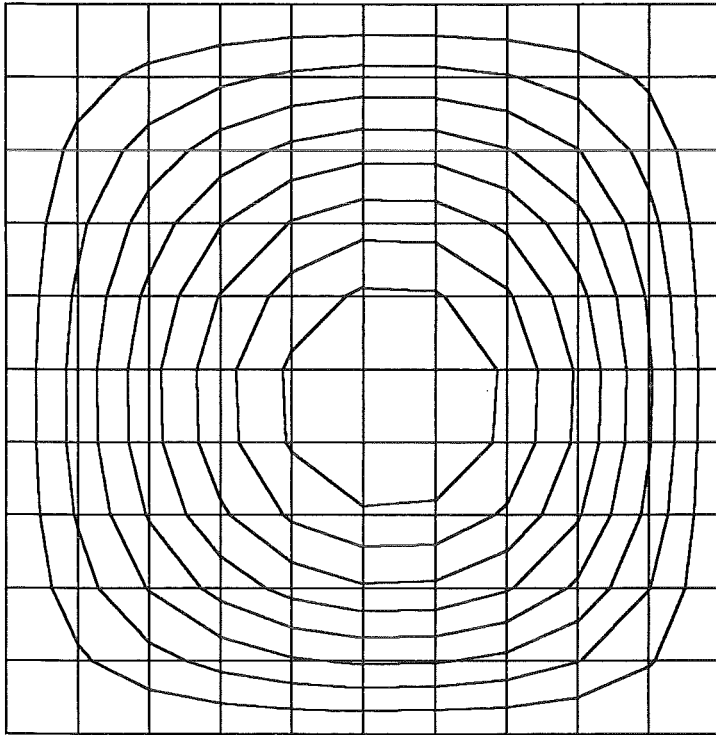


Abbildung 76. Isolinien für die Diffusion mit  $S(0) = 2S(1) - S(2)$  im 30. Schritt:  
Versetzter Kasten

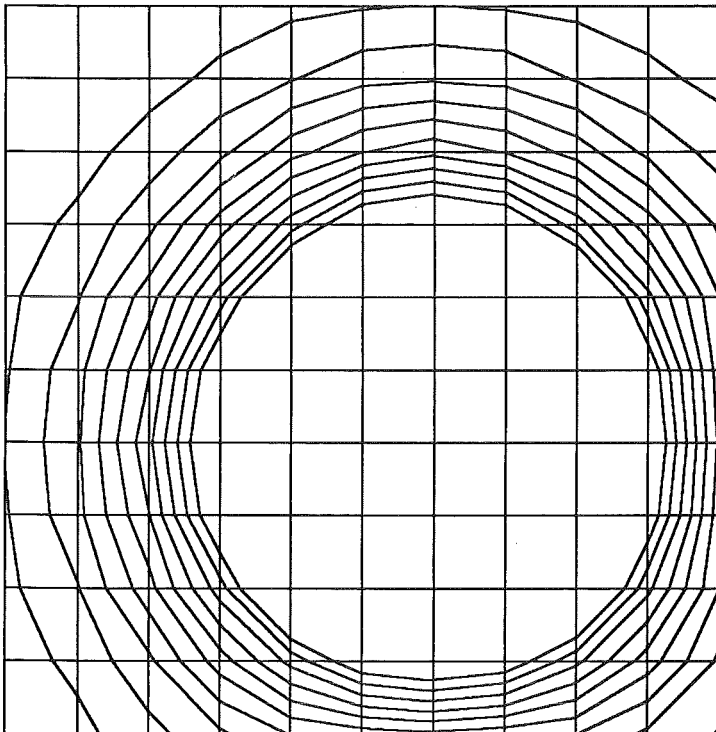
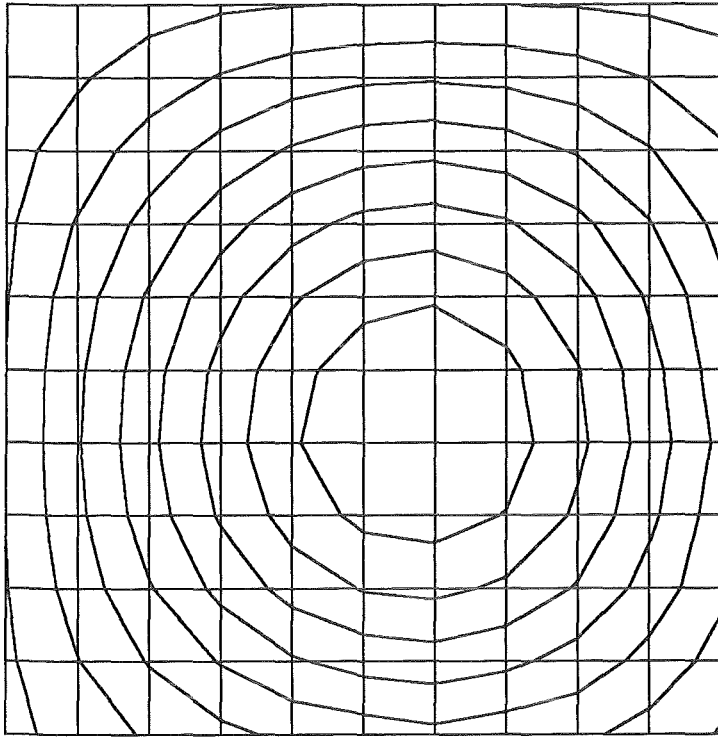
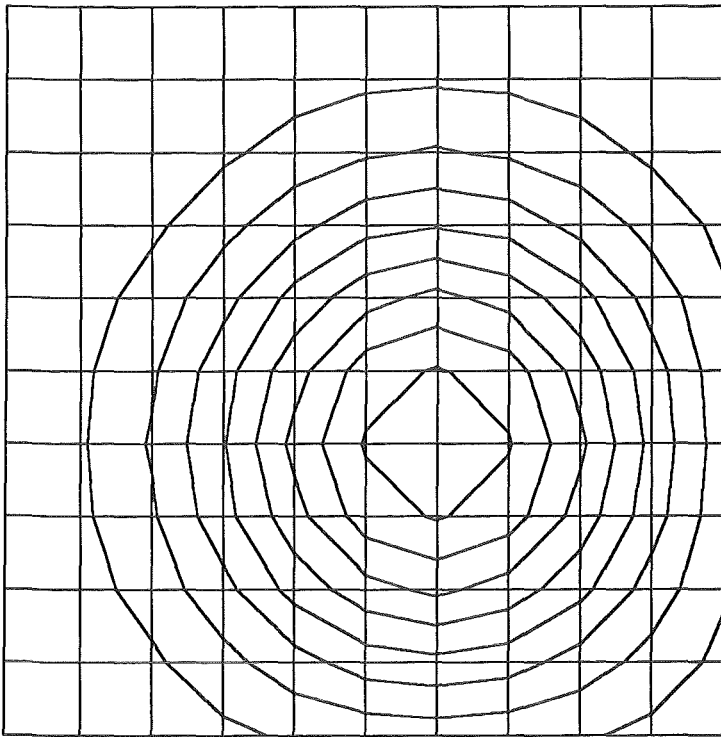


Abbildung 77. Isolinien für die Diffusion mit ORL1D2D im 10. Schritt:  
Versetzter Kasten

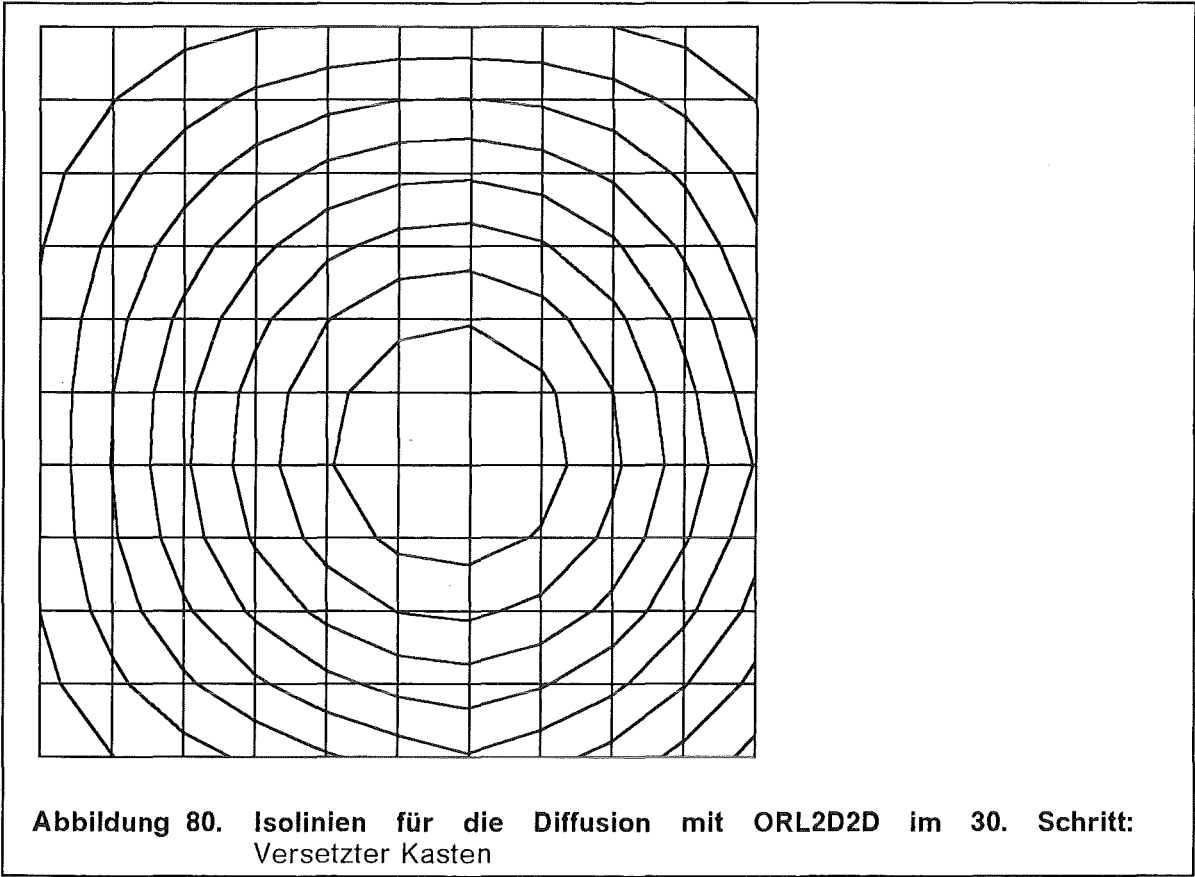


**Abbildung 78. Isolinien für die Diffusion mit ORL1D2D im 30. Schritt:  
Versetzter Kasten**



**Abbildung 79. Isolinien für die Diffusion mit ORL2D2D im 10. Schritt:  
Versetzter Kasten**





**Abbildung 80. Isolinien für die Diffusion mit ORL2D2D im 30. Schritt:  
Versetzter Kasten**

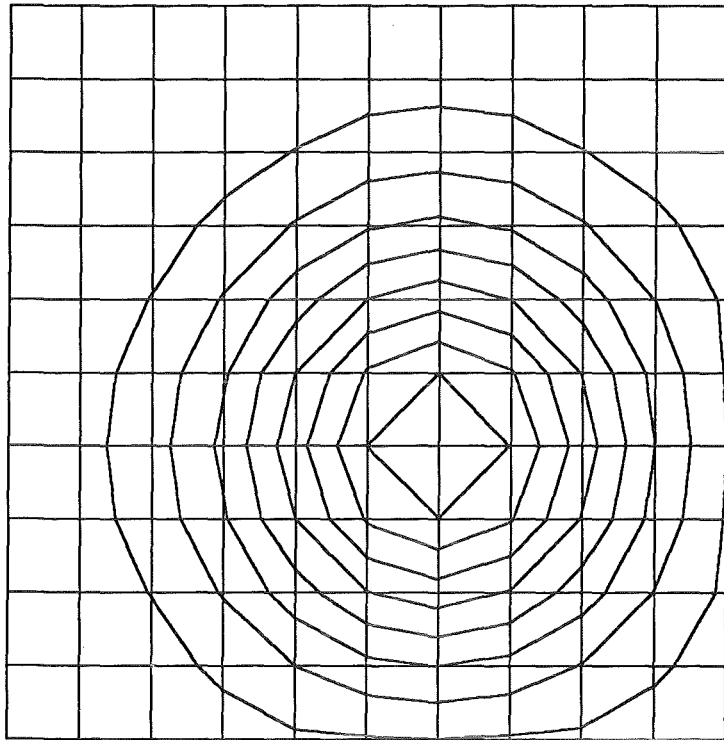


Abbildung 81. Isolinien für die Diffusion (implizit) im 10. Schritt:  
Versetzter Kasten

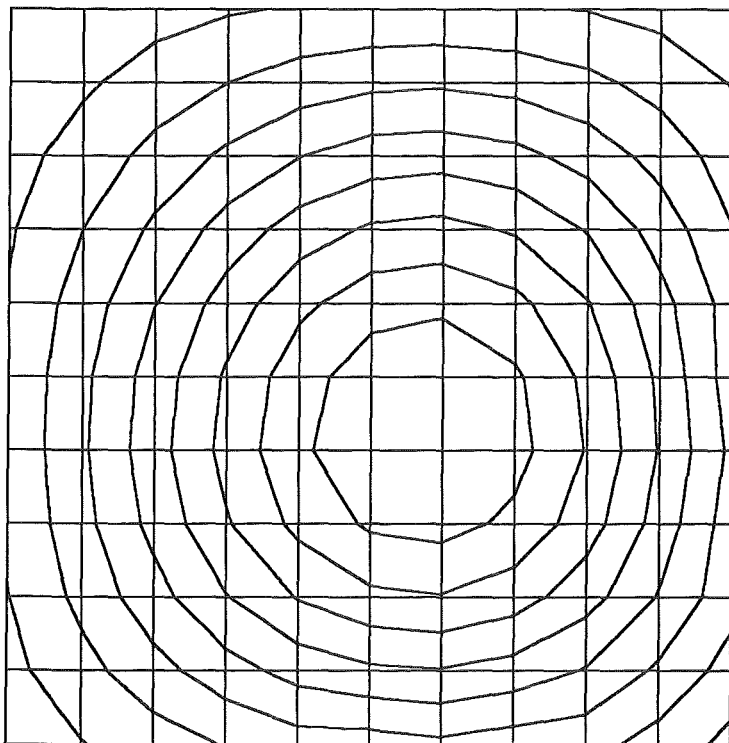


Abbildung 82. Isolinien für die Diffusion (implizit) im 30. Schritt:  
Versetzter Kasten

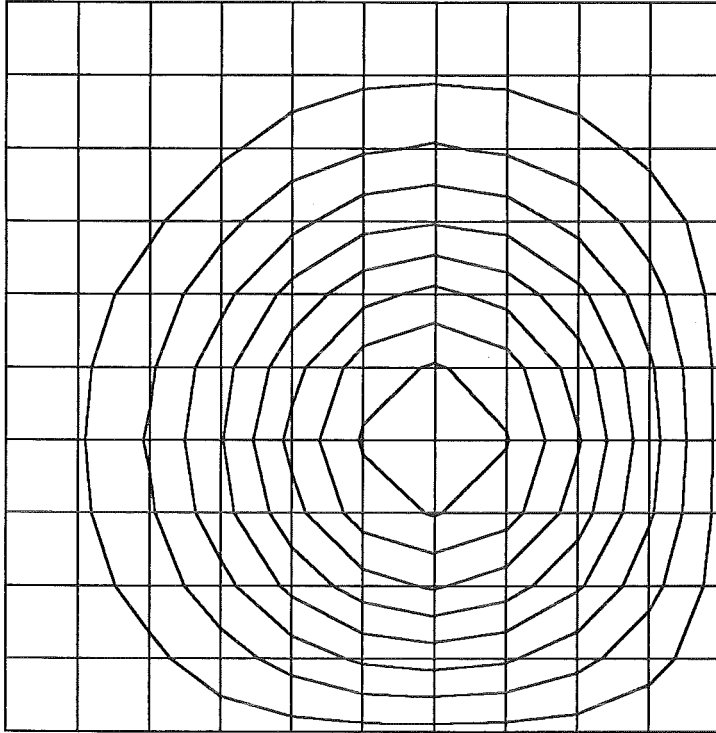


Abbildung 83. Isolinien für die Diffusion mit SORL2D2D im 10. Schritt:  
Versetzter Kasten

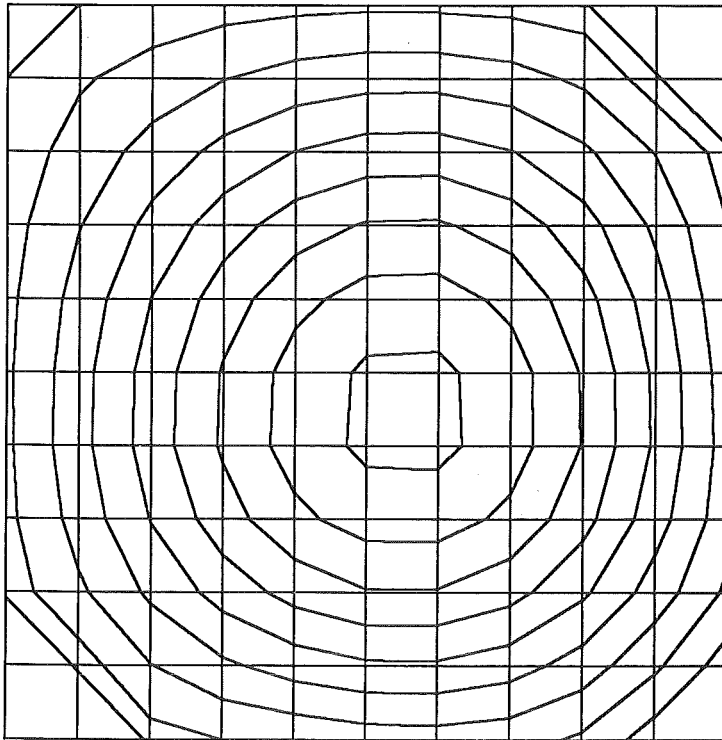
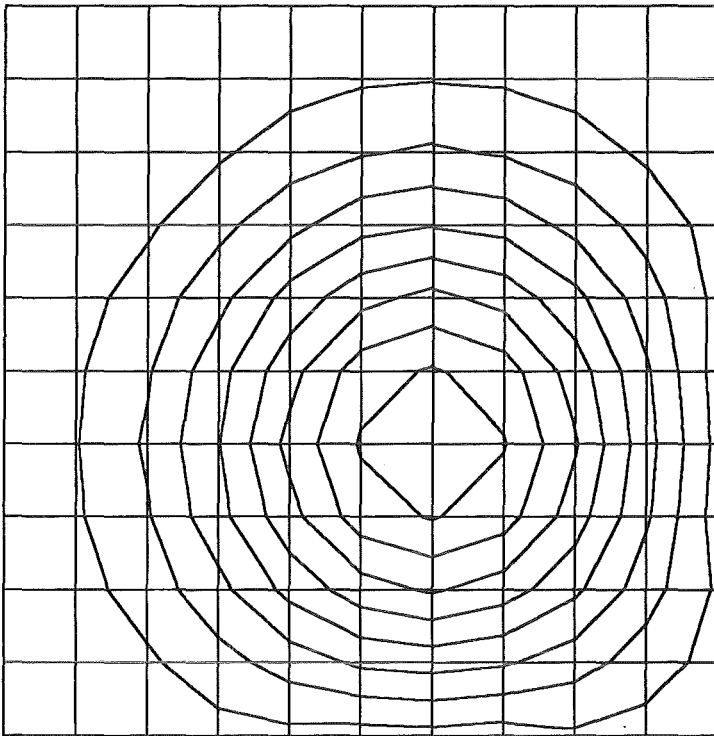


Abbildung 84. Isolinien für die Diffusion mit SORL2D2D im 30. Schritt:  
Versetzter Kasten



**Abbildung 85. Isolinien für die Diffusion mit ORL2DAD im 10. Schritt:  
Versetzter Kasten**

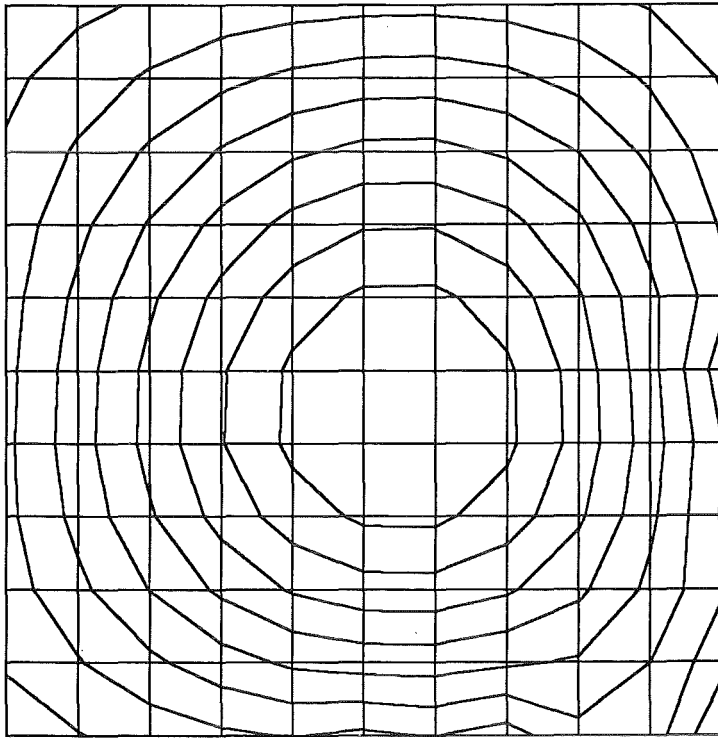


Abbildung 86. Isolinien für die Diffusion mit ORL2DAD im 30. Schritt:  
Versetzter Kasten

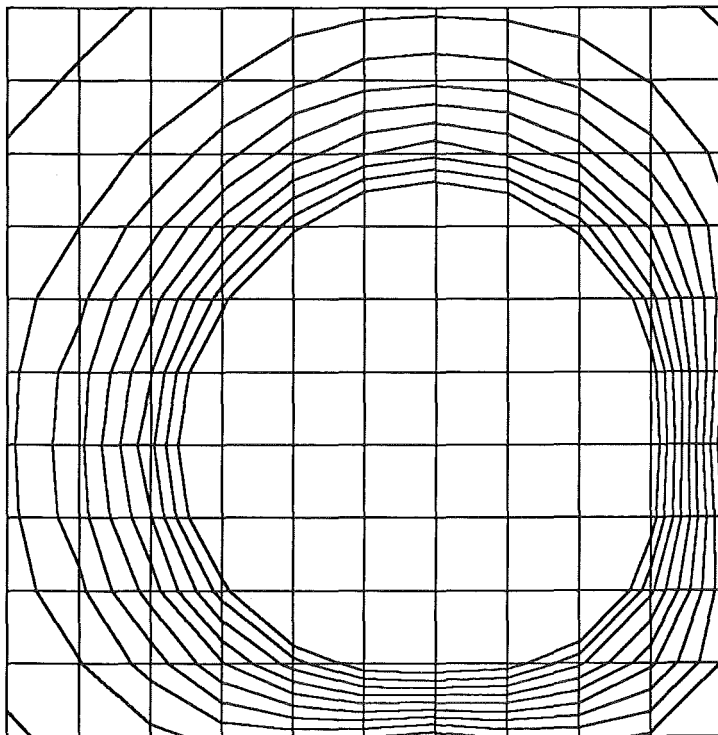
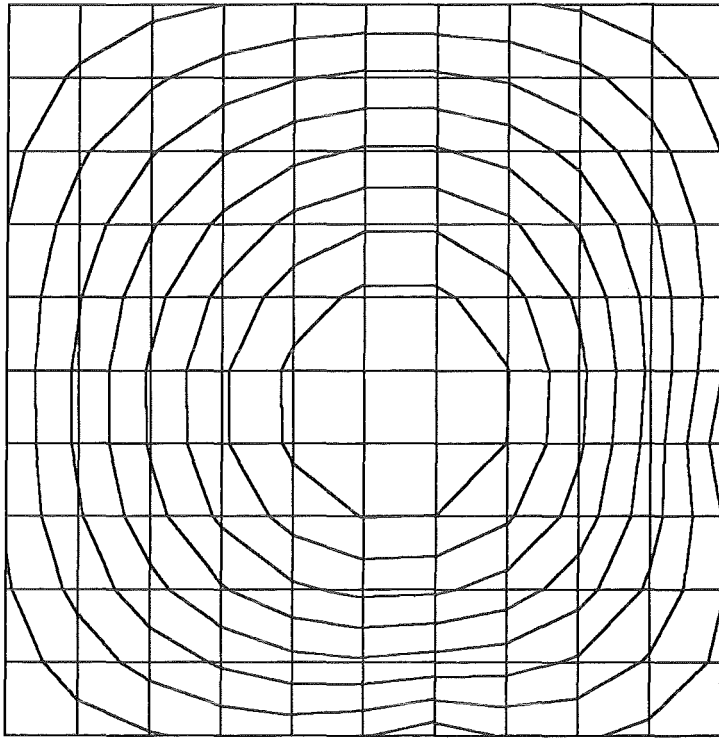


Abbildung 87. Isolinien für die Diffusion mit ORL2DADS im 10. Schritt:  
Versetzter Kasten



**Abbildung 88. Isolinien für die Diffusion mit ORL2DADS im 30. Schritt:  
Versetzter Kasten**



**R.Courant, E.Isaacson, M.Rees**

*On the solution of nonlinear hyperbolic differential equations*

Comm.Pure Appl.Math. (1952), Vol.5, pp243-255

**H.Davies**

*Limitations of some common lateral boundary schemes used in regional NWP models*

Mon.Wea.Rev. (1983), Vol.111, pp1002-1012

**B.Engquist, A.Majda**

*Absorbing boundary conditions for the numerical simulation of waves*

Math.Comp. (1977), Vol.31, pp629-651

**B.Engquist, S.Osher**

*One sided difference approximations for nonlinear conservation laws*

Math.Comp. (1981), Vol.36, pp321-351

**F.H.Harlow, J.E.Welch**

*Numerical calculation of time dependent viscous incompressible flow of fluid with free surface.*

Phys. Fluids (1965), Vol.8, pp2182-2189

**C.-P.Hugelmann**

*Differenzenverfahren zur Behandlung der Advektion*

Wissenschaftliche Berichte des Instituts für Meteorologie und Klimafor- schung der Universität Karlsruhe Nr.8 (1988)

**J.B.Klemp, D.R.Durran**

*An upper boundary condition permit- ting internal gravity wave radiation in numerical mesoscale models*

Mon.Wea.Rev. (1983), Vol.111, pp430-444

**Y.Kurihara, M.A.Bender**

*A numerical scheme to treat the open lateral boundary of a limited area mo- del*

Mon.Wea.Rev. (1983), Vol.111, pp445-454

**P.D.Lax, B.Wendroff**

*Systems of conservation laws*

Comm.Pure Appl.Math. (1960), Vol.13 pp217-237

**E.L.Lindman**

*"Free-Space" boundary conditions for the time dependent wave equation*

J.Comp.Phys. (1975), Vol.18, pp66-78

**M.J.Miller, A.J.Thorpe**

*Radiation conditions for the lateral boundaries of limited-area numerical models.*

Quart.J.R.Met.Soc. (1981), Vol.107, pp.615-628

**C.-D.Munz**

*Godunov-Typ Verfahren für Gleichun- gen der kompressiblen Strömungsme- chanik.*

to appear (1991), Vol.107, pp.47-51

**I.Orlanski**

*A simple boundary condition for un- bounded hyperbolic flows.*

J.Comp.Phys. (1976), Vol.21, pp.251-269

**R.A.Pearson**

*Consistent boundary conditions for numerical models of systems that ad- mit dispersive waves*

J.Atmos.Sci. (1974), Vol.31 , pp.1481-1489

**W.H.Raymond, H.L.Kuo**

*A radiation condition for multi-dimen- sional flows.*

Quart.J.R.Met.Soc. (1984), Vol.110 , pp.535-551

**P.J.Roache**

*Computational fluid dynamics*

Hermosa Publishers Albuquerque (1982), revised Edition

**J.Smoller**

*Shock waves and reaction-diffusion equations*

Springer Verlag, New York-Heidel- berg-Berlin (1983), pp291-301



**A.Sommerfeld**

*Partial differential equations : Lectures in theoretical physics*  
Academic Press, Vol.6 (1949)

**P.K.Sweby**

*High resolution schemes using flux limiter for hyperbolic conservation laws*  
SIAM J.Numer. Anal. (1984), Vol.21, pp217-235

**F.Treves**

*Introduction to pseudodifferential and Fourier integral operators Vol.1,2*  
Plenum Press New York-London (1980)

**L.Wagatha**

*Approximation of pseudodifferential operators in absorbing boundary conditions for hyperbolic equations*  
Numerische Mathematik (1983), Vol.42, pp51-64