
Semidiskretisierung der stationären Liouville Gleichung

Bachelorarbeit

Georg Retschitzegger

0825386

Mathematik in Technik und Naturwissenschaften

Institut für Analysis und Scientific Computing
Technische Universität Wien



TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology

12. August 2012

Inhaltsverzeichnis

1	Einleitung	3
2	Analytische Diskussion	4
2.1	Analytische Lösung	4
2.2	Unkontrollierter Bereich	5
3	Semidiskretisierung in Geschwindigkeitsrichtung	7
3.1	Zentrale Differenzenquotienten	8
3.2	Upwind Differenzenquotienten	10
3.3	Null als diskrete Geschwindigkeit	11
3.3.1	Zentrale Diskretisierung	11
3.3.2	Upwind Diskretisierung	12
3.4	Eindeutige Lösbarkeit	13
3.5	Numerik der Verfahren	16
4	Erhaltungsgrößen	18
4.1	Analytischer Fall	18
4.2	Semidiskreter Fall	19
5	Implementierung in MATLAB	21
6	Beispiele	23
6.1	Beispiel A	23
6.1.1	Lösungen	24
6.1.2	Fehlergrößen	25
6.2	Beispiel B	26
6.2.1	Lösungen	28
6.2.2	Fehlergrößen	29
6.3	Beispiel C	30
6.4	Beispiel D	30
A	3-D Plots	34
B	MATLAB Codes	38

Zusammenfassung

Diese Arbeit beschäftigt sich mit der stationären eindimensionalen Liouville Gleichung und deren numerische Lösung, das heißt mit einer linearen partiellen Differentialgleichung in zwei Variablen mit Einströmrandbedingungen. Wir werden zuerst das Randwertproblem analytisch lösen, erklären warum das Ergebnis nicht eindeutig ist, aber welche Lösung zu favorisieren ist. Anschließend werden mit finiten Differenzen Semidiskretisierungen in der Geschwindigkeitsrichtung definiert und wir rechtfertigen, warum diese auf sinnvolle numerische Probleme mit eindeutigen Lösungen führen. Hier wird auch behandelt, warum man mit Null als diskreter Geschwindigkeit vorsichtig sein muss und wie man das diskrete Geschwindigkeitsgitter wählen muss, um Fehler in der Implementierung zu vermeiden und das physikalische Problem richtig abzubilden. Wir werden auf die erwarteten numerischen Fehlerordnungen und mögliche Schwachstellen der Verfahren eingehen und dann in einigen Beispielen die erhaltenen Resultate überprüfen.

1 Einleitung

Die Liouville Gleichung ist eine fundamentale Gleichung aus der statistischen Mechanik. Wir motivieren mit Resultaten aus [9] und [11] zuerst kurz ihre Herkunft aus der Physik, bevor wir sie mathematisch analysieren. Sie beschreibt die Evolution von konservativen Hamiltonischen Systemen von Partikelbewegungen im Phasenraum. Ihre Besonderheit ist, dass sie sowohl für Gleichgewichtslagen, als auch weg davon gültig ist.

Genauer, die Gleichung beschreibt für ein M -Partikel System mit der Raumdichte $f(q, p, t)$, abhängig von Position, Impuls und Zeit, die Bewegung. Die Herleitung beruht auf der Erhaltung der Energie und den Hamiltonischen Gleichungen und liefert

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \sum_{i=1}^M \left(\frac{\partial f}{\partial q_i} \frac{\partial q}{\partial t} + \frac{\partial f}{\partial p_i} \frac{\partial p}{\partial t} \right) = 0 \quad (1.1)$$

mit der Ortsvariable q und der Impulsvariable p .

Wir werden uns hier auf einen Spezialfall beschränken, nämlich den, dass die Bewegung der Partikel mit Koordinaten (q, p) nicht relativistisch durch ein Kraftfeld F bestimmt wird. Dann vereinfacht sich Gleichung (1.1) zu

$$\frac{\partial f}{\partial t} + \frac{p}{m} \nabla_q f + F \nabla_p f = 0. \quad (1.2)$$

Wir vereinfachen weiter, indem wir für die Masse $m = 1$ annehmen und definieren die eindimensionalen Koordinaten $(q, p) =: (x, v) \in [0, L] \times \mathbb{R}, L > 0$. Außerdem sei das Kraftfeld F durch ein Potential $V(x)$ bestimmt, also $F = -\nabla V = -V'$. Wir können also die Gleichung für $f = f(t, x, v)$ noch mehr vereinfachen und erhalten mit der Notation $\frac{\partial f}{\partial t} = f_t$

$$f_t(t, x, v) + v f_x(t, x, v) - V'(x) f_v(t, x, v) = 0. \quad (1.3)$$

Diese Gleichung werden wir in Folge als *Liouville Gleichung* bezeichnen. Da wir hier an einer stationären Lösung $f = f(x, v)$ interessiert sind, betrachten wir hier hauptsächlich die *stationäre Liouville Gleichung*

$$v f_x(x, v) - V'(x) f_v(x, v) = 0. \quad (1.4)$$

Die Randbedingungen sind gegeben als sogenannte *Einströmrandbedingungen*

$$\begin{aligned} f(0, v) &= f^+(v), & v > 0, \\ f(L, v) &= f^-(v), & v < 0. \end{aligned} \quad (1.5)$$

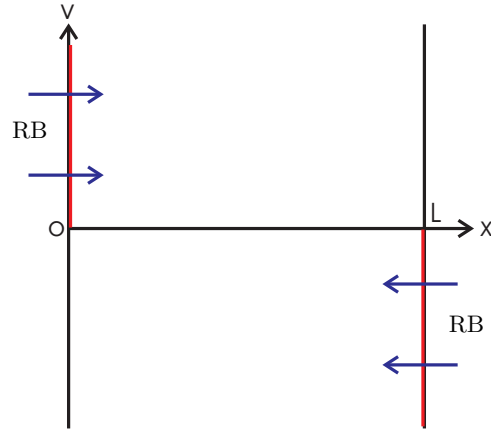


Abbildung 1: Einströmrandbedingungen. Quelle: [1].

Das bedeutet, für positive Geschwindigkeiten v strömen die Randbedingungen am linken Rand des Gebiets hinein, für negative am rechten, vergleiche Abb. 1 bzw. Abb. 2.

Als *Generalvoraussetzung* für die folgenden Diskussionen, falls nicht explizit erwähnt, seien die Träger der Randbedingungen f^+ , f^- kompakte Intervalle, die die Null nicht enthalten, das heißt zum Beispiel $\text{supp } f^+ = [a, b]$, $a, b \in \mathbb{R}^+$, und die Funktionen selber hinreichend glatt, um eine glatte Lösung f zu erhalten. Das Potential $V(x)$ sei auch hinreichend glatt.

2 Analytische Diskussion

In diesem Kapitel befassen wir uns mit der analytischen Lösung der Gleichung (1.4) und zeigen die Probleme auf, die dabei entstehen. Die hier ermittelte Lösung wird später mit der aus dem numerischen Verfahren gewonnenen verglichen, um auf eine numerische Fehlerabschätzung zu kommen.

2.1 Analytische Lösung

Die stationäre Liouville Gleichung

$$vf_x(x, v) - V'(x)f_v(x, v) = 0 \quad (1.4)$$

ist eine homogene partielle Differentialgleichung von erster Ordnung, da nur erste Ableitungen vorkommen, und linear in der Funktion $f(x, v)$. Darum verwenden wir zum Lösen die Methode der Charakteristiken ([8]). Die Randbedingungen sind analog zu vorher mit (1.5) gegeben.

Für die Charakteristikenmethode definieren wir zuerst mit dem Durchlaufungsparameter s :

$$(x, v) := (x(s), v(s)) \quad (2.1)$$

$$z(s) := f(x(s), v(s)) \quad (2.2)$$

$$p(s) := (f_x(s), f_v(s)) \quad (2.3)$$

Die charakteristischen Gleichungen ergeben sich dann zu

$$\dot{x}(s) = v(s) \quad (2.4)$$

$$\dot{v}(s) = -V'(x(s))$$

$$\dot{z}(s) = 0. \quad (2.5)$$

Aus (2.2) und (2.5) folgt

$$f(x(s), v(s)) = z(s) = \text{const} =: z_0, \quad (2.6)$$

also ist die Lösung entlang der charakteristischen Kurven konstant. Damit wir dieses Resultat besser verstehen können, betrachten wir das Phasenportrait der Trajektorien. Aus (2.4) sehen wir, dass die Gestalt der Kurven alleine vom gegebenen Potential $V(x)$ abhängt.

Lemma 2.1. *Die Charakteristiken (2.4) der stationären Liouville Gleichung (1.4) erfüllen die implizite Durchlaufungsform der Energieerhaltung*

$$V(x(s)) + \frac{v(s)^2}{2} = \text{const}. \quad (2.7)$$

Beweis. Es gilt

$$0 = -\dot{v}(s)v(s) + v(s)\dot{v}(s) \stackrel{(2.4)}{=} V'(x(s))\dot{x}(s) + v(s)\dot{v}(s) = \frac{d}{ds} \left(V(x(s)) + \frac{v(s)^2}{2} \right),$$

woraus die Aussage folgt. \square

Abbildung 2 illustriert das Verhalten. Ein lokales Minimum des Potential führt auf geschlossene Trajektorien, während eine Maximum einen Sattelpunkt erzeugt. Das Potential hängt nur vom Ort x ab und darum ist das Bild im Phasenraum (x, v) symmetrisch um die x -Achse. Das Vorzeichen der Geschwindigkeit v bestimmt die Strömungsrichtung. Aus den obigen Resultaten wissen wir, dass die Lösung entlang der Kurven konstant ist, also werden die Einströmrandbedingungen einfach entlang der Charakteristiken transportiert. Wie man jedoch in Abb. 2 erkennen kann, haben die Charakteristiken um das lokale Minimum keine Verbindung zum Rand, sie werden also nicht durch die Randbedingungen kontrolliert. In diesem *unkontrollierten Bereich* haben wir keinerlei Hinweise auf ein mögliches Verhalten der Lösung. Damit ist die Lösung von (1.4), (1.5) im Allgemeinen nicht eindeutig. Wir fassen unsere Ergebnisse zusammen in

Theorem 2.2. *Die stationäre Liouville Gleichung (1.4) mit den Randbedingungen (1.5) hat die, im Allgemeinen, nicht eindeutige Lösung entlang der Charakteristiken*

$$f((x(s), v(s)) = z_0 = \begin{cases} f^+(v(s_0)) & , \text{ Randbedingungen von linken Rand,} \\ f^-(v(s_0)) & , \text{ Randbedingungen von rechten Rand,} \\ \bar{f} & , \text{ im unkontrollierten Bereich,} \end{cases} \quad (2.8)$$

mit $\bar{f} \in \mathbb{R}$ beliebig und s_0 so, dass $(x(s_0), v(s_0))$ der passende Startwert der Charakteristik in Durchlaufungsform am Rand ist.

2.2 Unkontrollierter Bereich

Die Lösung aus dem vorherigen Kapitel kann schlecht benutzt werden, um ein numerisches Verfahren zu vergleichen, solange Bereiche existieren, in denen man gar nichts über das Verhalten weiß. Wir modifizieren die Gleichung nun, um dadurch eine der möglichen Lösungen aus Theorem 2.2 zu favorisieren. Die Störung durch die Modifikation lassen wir dann gegen Null gehen und vergleichen die Lösungen mit einander.

Wir definieren nun die *gedämpfte stationäre Liouville Gleichung*

$$vf_{\varepsilon,x}(x, v) - V'(x)f_{\varepsilon,v}(x, v) = -\varepsilon f_{\varepsilon}(x, v) \quad (2.9)$$

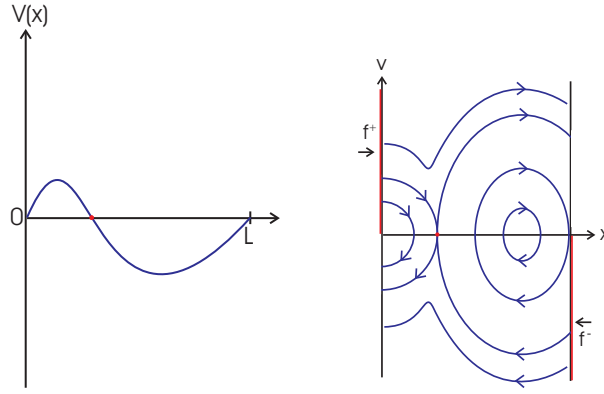


Abbildung 2: Das gegebene Potential $V(x)$ auf dem linken Bild erzeugt die Charakteristiken im rechten Bild. Die Einströmrandbedingungen werden entlang dieser transportiert und wie man sieht, können die geschlossenen Kurven um das Minimum nicht durch die Randbedingungen kontrolliert werden. Quelle: [1].

mit den gleichen Einströmrandbedingungen

$$\begin{aligned} f_\varepsilon(0, v) &= f^+(v), & v > 0 \\ f_\varepsilon(L, v) &= f^-(v), & v < 0. \end{aligned} \quad (2.10)$$

Lässt man ε in der gedämpften Gleichung (2.9) gegen Null gehen, erhält man genau die ursprüngliche Gleichung (1.4). Wir werden die Lösung von (2.9) berechnen um zu sehen, ob dort eine Aussage über den unkontrollierten Bereich gemacht werden kann. (2.9) ist wie vorher wieder eine partielle Differentialgleichung erster Ordnung und linear in f_ε mit einer Inhomogenität εf , also werden wir wieder die Charakteristikenmethode anwenden, um eine analytische Lösung zu erhalten. Wir definieren wieder zuerst

$$(x, v) := (x(s), v(s)) \quad (2.11)$$

$$z_\varepsilon(s) := f_\varepsilon(x(s), v(s)) \quad (2.12)$$

$$p_\varepsilon(s) := (f_{\varepsilon,x}(s), f_{\varepsilon,v}(s)). \quad (2.13)$$

Die charakteristischen Gleichungen ergeben sich dann zu

$$\begin{aligned} \dot{x}(s) &= v(s) \\ \dot{v}(s) &= -V'(x(s)) \end{aligned} \quad (2.14)$$

$$\dot{z}(s) = -\varepsilon z_\varepsilon(s). \quad (2.15)$$

Wir erkennen sofort, dass das Bild der Charakteristiken dasselbe wie im ungedämpften Fall ist, da (2.14) und (2.4) ident sind. Mit (2.12) und (2.15) erhalten wir

$$f_\varepsilon(x(s), v(s)) = z_\varepsilon(s) = \text{const} e^{-s\varepsilon} =: z_{\varepsilon 0} e^{-s\varepsilon}. \quad (2.16)$$

Das heißt, die Werte entlang der Trajektorien sind nicht mehr konstant, sondern werden gedämpft und nehmen mit dem Faktor $e^{-s\varepsilon}$ ab. Wir können für jede Kurve ohne Beschränkung der Allgemeinheit eine Durchlaufung wählen, so dass $s_0 = 0$ ist, wobei s_0 wie vorher der Startwert der Kurve am passenden Rand ist. Also erhalten wir für den von den Randwerten kontrollierten Bereich $z_{\varepsilon 0} = z_0$.

Die besondere Bauart der Lösung erlaubt uns nun aber noch eine Aussage über den unkontrollierten Bereich. Wie in Kapitel 2.1 bereits erwähnt sind die Charakteristiken um das Minimum des Potentials geschlossene Ellipsen. Durch die Parameterdarstellung in s wird die Ellipse für $s \rightarrow \infty$ beliebig oft durchlaufen. Sei also (\bar{x}, \bar{v}) ein beliebiger, fester Punkt auf einer festen geschlossenen Trajektorie. Dann folgt aus (2.16) die Existenz eines $\bar{s} \in \mathbb{R}$, so dass

$$f_\varepsilon(\bar{x}, \bar{v}) = z_\varepsilon(\bar{s}) = z_0 e^{-\bar{s}\varepsilon}. \quad (2.17)$$

Da die Ellipse beliebig oft durchlaufen werden kann, existiert aber auch ein $\bar{k} \in \mathbb{R}$, $\bar{k} \neq \bar{s}$, mit

$$f_\varepsilon(\bar{x}, \bar{v}) = z_\varepsilon(\bar{k}) = z_0 e^{-\bar{k}\varepsilon}. \quad (2.18)$$

Setzt man nun $\bar{k} = \bar{s} + \alpha$ mit $\alpha \in \mathbb{R} \setminus \{0\}$ passend, dann folgt aus (2.17) und (2.18)

$$\begin{aligned} f_\varepsilon(\bar{x}, \bar{v}) &= z_\varepsilon(\bar{s}) = z_\varepsilon(\bar{s} + \alpha) \\ &= z_{\varepsilon 0} e^{-\bar{s}\varepsilon} = z_0 e^{-(\bar{s} + \alpha)\varepsilon}. \end{aligned} \quad (2.19)$$

Die Exponentialfunktion ist injektiv, also folgt aus (2.19) $z_{\varepsilon 0} = 0$ für den unkontrollierten Bereich der gedämpften Gleichung. Wir fassen die Ergebnisse wieder zusammen in

Theorem 2.3. *Die gedämpfte stationäre Liouville Gleichung (2.9) mit den Randbedingungen (2.10) hat die eindeutige Lösung entlang der Charakteristiken in Durchlaufungsform*

$$f_\varepsilon(x(s), v(s)) = z_{\varepsilon 0} e^{-s\varepsilon} = \begin{cases} f^+ e^{-s\varepsilon} & , \text{Information vom rechten Rand} \\ f^- e^{-s\varepsilon} & , \text{Information vom linken Rand} \\ 0 & , \text{Information vom rechten Rand} \end{cases} \quad (2.20)$$

Für den kontrollierten Bereich erkennen wir aus dieser Form

$$\lim_{\varepsilon \rightarrow 0} f_\varepsilon(x, v) = f(x, v). \quad (2.21)$$

Darum erwarten wir für die Lösung der ungedämpften Gleichung (2.8) auch $f \equiv 0$ im unkontrollierten Bereich als sinnvolle Lösung eines passenden numerischen Verfahrens.

3 Semidiskretisierung in Geschwindigkeitsrichtung

Die Gleichung (1.4) soll in weiterer Folge numerisch gelöst werden. Um diesen Vorgang zu realisieren, werden wir die Gleichung zuerst in Richtung der Geschwindigkeit v diskretisieren, während die Ortsvariable x kontinuierlich bleibt. Diese sogenannte Semidiskretisierung ist in der Literatur auch unter dem Begriff Linienmethode zu finden ([6]). Die horizontale Linienmethode bezeichnet das Diskretisieren der Zeitvariable und bei der vertikalen Linienmethode ersetzt man die Ortsvariable durch ein Gitter. Der Zweck dieser Methode ist es, aus der partiellen Differentialgleichung ein System von gewöhnlichen Differentialgleichungen zu machen. Man könnte die Gleichung auch in beiden Variablen diskretisieren, aber durch Freilassen einer Variable erwarten wir uns leichtere Aussagen über mögliche Lösungen und Konvergenz aus der Theorie der gewöhnlichen Differentialgleichungen. Die Diskretisierung wird mittels finiter Differenzen realisiert.

Wir müssen nun ein Gitter festlegen, welches die Diskretisierung von v repräsentiert. Der Einfachheit halber nehmen wir eine konstante Schrittweite $\Delta v > 0$ und die diskreten Geschwindigkeiten $v_j \in \mathbb{R}$ als streng monoton steigend, das heißt $v_j < v_{j+1} = v_j + \Delta v$, $j \in J$ mit $J \subset \mathbb{Z}$.

Dies soll nur die allgemeine Notation für das Gitter illustrieren. Die genaue Gestalt wird dann in Kapitel 5 behandelt. Weiters gelte

$$\begin{aligned} v_j < 0, j \in J^- := J \cap (-\mathbb{N}) \\ v_j > 0, j \in J^+ := J \cap \mathbb{N} \end{aligned}$$

und $v_0 = 0$. Der Index Null soll aber nicht in unserer Indexmenge enthalten sein, da wir vorerst Null als diskrete Geschwindigkeit vermeiden, also $0 \notin J$. Die genauere Diskussion über diese Annahme findet in Kapitel 3.3 statt. Weiters definieren wir die semidiskrete Funktion als Vektor

$$\begin{aligned} w_j(x) &:= f(x, v_j) \\ w(x) &:= (w_j)_{j \in J} \in \mathbb{R}^{|J|}. \end{aligned} \tag{3.1}$$

Der Einfachheit halber wählen wir das v -Gitter symmetrisch um Null, es sollen also sowohl in die negative als auch in die positive Richtung gleich viele Punkte aufgetragen werden. Damit gilt

$$|J^+| = |J^-| = N.$$

Welche Art von finiten Differenzen verwendet werden sollen, um die Ableitung f_v in (1.4) zu realisieren, ist ad hoc nicht klar und wir werden in den folgen Sektionen zwei verschiedene Varianten vorstellen und analysieren.

3.1 Zentrale Differenzenquotienten

Der erste Ansatz, um die Differentiation zu realisieren, ist eine Variante, die aus dem zentralen Differenzenquotient

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h} \tag{3.2}$$

gewonnen wird. In diskreter Version nimmt dieser die Gestalt

$$f'(x_j) \approx \frac{f_{j+1} - f_{j-1}}{2h} \tag{3.3}$$

an, mit $f_j = f(x_j)$, $h = x_j - x_{j-1}$. Die zentralen Differenzen funktionieren bei uns im zweidimensionalen Fall analog, also

$$f_v(x, v_j) \approx \frac{w_{j+1}(x) - w_{j-1}(x)}{2\Delta v}. \tag{3.4}$$

(1.4) wird also mittels diesem Ansatz auf die $2N$ Gleichungen

$$v_j w'_j(x) - V'(x) \frac{1}{2\Delta v} (w_{j+1}(x) - w_{j-1}(x)) = 0 \quad , j \in J \tag{3.5}$$

semidiskretisiert.

Wir haben damit die partielle Differentialgleichung in ein System von gewöhnlichen Differentialgleichungen verwandelt. In dem in (3.5) beschriebenen Verfahren kommen aber auch die Werte von w auf den virtuellen Gitterlinien v_{N+1} und v_{N-1} vor, die eigentlich nicht in der Diskretisierung enthalten sind. Um diese passend in unser Verfahren einzubauen, benötigen wir zuerst noch eine Vorgabe an das Gebiet. Die Randbedingungen sind bei uns mit kompaktem Träger $[a, b] \subset \mathbb{R}$ gegeben. Aus der analytischen Diskussion in 2.1 wissen wir, dass diese entlang der

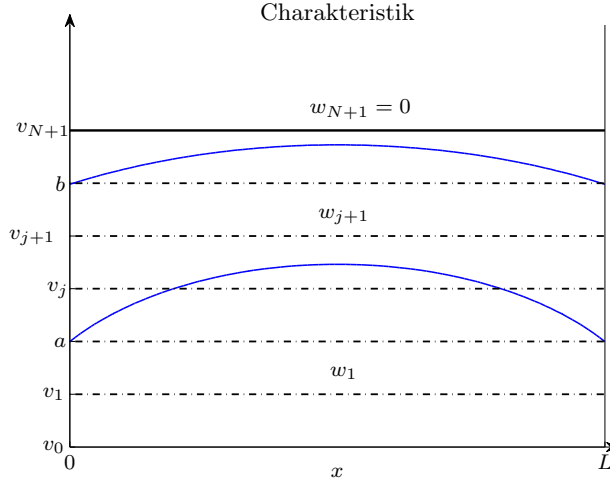


Abbildung 3: Die Grafik zeigt, wie man das Gitter wählen kann, um $w_{N+1} = 0$ garantieren zu können.

Charakteristiken transportiert wird, vergleiche Abb. 2. Bei gegebene Potential $V(x)$ können wir nun sofort feststellen, wie weit die Lösung $f(x, v)$ sich im (x, v) Phasenraum ausbreitet und das v -Gitter so wählen, dass stets $w_{N+1}(x) = 0$ ist. Abb. 3 illustriert dies für ein parabolisches Potential, welches auch im Beispiel A in Kapitel 6.1 verwendet wird. Für w_{-N-1} funktioniert das natürlich analog. Für weitere Diskussionen können wir das Ganze nun in Matrixschreibweise setzen und wir definieren die zentrale Semidiskretisierung der stationären Liouville Gleichung (1.4) mit Randbedingungen (1.5).

Definition 3.1. Die *zentrale Semidiskretisierung* der stationären Liouville Gleichung (1.4) ist komponentenweise definiert durch

$$\begin{aligned}
v_j w'_j(x) - V'(x) \frac{1}{2\Delta v} (w_{j+1}(x) - w_{j-1}(x)) &= 0 \quad , j \in J \setminus \{N, -N\}, \\
v_{-N} w'_{-N}(x) - V'(x) \frac{1}{2\Delta v} (w_{-N+1}(x)) &= 0, \\
v_N w_N(x)' - V'(x) \frac{1}{2\Delta v} (-w_{N-1}(x)) &= 0,
\end{aligned} \tag{3.6}$$

und in Matrixschreibweise durch

$$T w'(x) - V'(x) \frac{1}{2\Delta v} B w(x) = 0, \tag{3.7}$$

mit

$$\begin{aligned}
T &:= \text{diag}(v_N, v_{N-1}, \dots, v_1, v_{-1}, \dots, v_{-N+1}, v_{-N}) \in \mathbb{R}^{|J| \times |J|}, \\
B &:= \begin{pmatrix} 0 & -1 & & & \\ 1 & 0 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & -1 \\ & & & 1 & 0 \end{pmatrix} \in \mathbb{R}^{|J| \times |J|}.
\end{aligned}$$

Die *semidiskreten Randbedingungen* sind durch

$$\begin{aligned}
w_j(0) &= w_j^+ := f^+(v_j), \quad j \in J^+, \\
w_j(L) &= w_j^- := f^-(v_j), \quad j \in J^-
\end{aligned} \tag{3.8}$$

gegeben.

3.2 Upwind Differenzenquotienten

Der zweite Ansatz um die Differentiation zu diskretisieren wird mit dem Upwind Schema realisiert. Das Upwind Schema ist eigentlich ein volldiskretes Verfahren erster Ordnung, welches sich aus der Flussrichtung der Charakteristiken der linearen Advektionsgleichung

$$u_t + au_x = 0 \quad (3.9)$$

herleitet. Der Faktor a bestimmt hier die Richtung der Charakteristiken und die Gitterpunkte werden so gewählt, dass immer Punkte aus der Richtung des Ursprungs der Information genommen werden. Das heißt mit $u(t_n \pm \Delta t, x_i) =: u_i^{n \pm 1}$ und $u(t_n, x_i \pm \Delta x) =: u_{i \pm 1}^n$ diskretisiert sich (3.9) zu

$$\begin{aligned} \frac{u_i^{n+1} - u_i^n}{\Delta t} + a \frac{u_i^n - u_{i-1}^n}{\Delta x} &= 0, \quad a > 0, \\ \frac{u_i^{n+1} - u_i^n}{\Delta t} + a \frac{u_{i+1}^n - u_i^n}{\Delta x} &= 0, \quad a < 0. \end{aligned} \quad (3.10)$$

Eine genaue Herleitung und Diskussion findet man zum Beispiel in [12]. Um das Upwind Schema auf unsere Gleichung anwenden zu können, betrachten wir die zeitabhängige Gleichung

$$f_t(t, x, v) + v f_x(t, x, v) - V'(x) f_v(t, x, v) = 0. \quad (1.3)$$

Mit der Notation $f(t_n \pm \Delta t, x_i, v_j) =: f_{i,j}^{n \pm 1}$, $f(t_n, x_i \pm \Delta x, v_j) =: f_{i \pm 1,j}^n$ und $f(t_n, x_i, v_j \pm \Delta v) =: f_{i,j \pm 1}^n$ liefert analoges Vorgehen wie bei der Advektionsgleichung

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + v_j \frac{f_{i,j}^n - f_{i-1,j}^n}{\Delta x} - V'(x) \frac{f_{i,j}^n - f_{i,j-1}^n}{\Delta v} = 0, \quad v_j > 0, V'(x) < 0, \quad (3.11)$$

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + v_j \frac{f_{i,j}^n - f_{i-1,j}^n}{\Delta x} - V'(x) \frac{f_{i,j+1}^n - f_{i,j}^n}{\Delta v} = 0, \quad v_j > 0, V'(x) > 0, \quad (3.12)$$

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + v_j \frac{f_{i+1,j}^n - f_{i,j}^n}{\Delta x} - V'(x) \frac{f_{i,j}^n - f_{i,j-1}^n}{\Delta v} = 0, \quad v_j < 0, V'(x) < 0, \quad (3.13)$$

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + v_j \frac{f_{i+1,j}^n - f_{i,j}^n}{\Delta x} - V'(x) \frac{f_{i,j+1}^n - f_{i,j}^n}{\Delta v} = 0, \quad v_j < 0, V'(x) > 0. \quad (3.14)$$

Diese Volldiskretisierung garantiert uns, dass die Information aus der passenden Richtung genommen wird, um die neuen Werte numerisch zu berechnen. Hier ist aber die stationäre Gleichung von Interesse und es soll nur eine Semidiskretisierung durchgeführt werden. Das heißt, wir lassen den ersten Term in den Gleichungen (3.12) bis (3.14) wegfallen und ersetzen die finiten Differenzen für x wieder durch die Ableitung, wodurch die Bedingung an v_j wegfällt. Für unsere semidiskrete Gleichung gehen wir jetzt wieder zur in (3.1) eingeführten Notation über und erhalten die komponentenweise Darstellung

$$\begin{aligned} v_j w'_j(x) - V'(x) \frac{1}{\Delta v} (w_j(x) - w_{j-1}(x)) &= 0, \quad V'(x) < 0, \\ v_j w'_j(x) - V'(x) \frac{1}{\Delta v} (w_{j+1}(x) - w_j(x)) &= 0, \quad V'(x) > 0. \end{aligned}$$

Mit dem selben Argument, wie vorher bei dem zentralen Verfahren aus Definition 3.1 können wir das Gitter so wählen, dass die Werte w_{N+1} und w_{-N-1} stets verschwinden und wir können das Upwind Verfahren auch in Matrixschreibweise definieren.

Definition 3.2. Wir definieren die *Upwind Semidiskretisierung* der stationären Liouville Gleichung durch

$$\begin{cases} v_j w'_j(x) - V'(x) \frac{1}{\Delta v} (w_j(x) - w_{j-1}(x)) = 0, & V'(x) < 0, \\ v_j w'_j(x) - V'(x) \frac{1}{\Delta v} (w_{j+1}(x) - w_j(x)) = 0, & V'(x) > 0, \end{cases} \quad j \in J, \quad (3.15)$$

mit $w_{N+1} = w_{-N-1} = 0$.

In Matrizenform lässt sich (3.15) auch schreiben als

$$T w'(x) - V'(x) \frac{1}{\Delta v} B_u(x) w(x) = 0, \quad (3.16)$$

mit

$$\begin{aligned} T &:= \text{diag}(v_N, v_{N-1}, \dots, v_1, v_{-1}, \dots, v_{-N+1}, v_{-N}) \in \mathbb{R}^{|J| \times |J|}, \\ B_u &:= \begin{cases} B^+ & V'(x) < 0 \\ B^- & V'(x) > 0, \end{cases} \\ B^+ &:= \begin{pmatrix} 1 & -1 & 0 & & \\ 0 & 1 & -1 & 0 & \\ & \ddots & \ddots & \ddots & \\ & & 0 & 1 & -1 \\ & & & 0 & 1 \end{pmatrix} \in \mathbb{R}^{|J| \times |J|}, \\ B^- &:= \begin{pmatrix} -1 & 0 & & & \\ 1 & -1 & 0 & & \\ 0 & \ddots & \ddots & \ddots & \\ & & 1 & -1 & 0 \\ & & & 1 & -1 \end{pmatrix} \in \mathbb{R}^{|J| \times |J|}. \end{aligned}$$

Die beiden in diesem Kapitel definierten Verfahren werden wir nun in Folge genauer betrachten und testen, ob sie sich dafür eignen, das gestellte Problem numerisch zu lösen.

3.3 Null als diskrete Geschwindigkeit

Wir werden jetzt diskutieren, warum wir $v_0 = 0$ als diskrete Geschwindigkeit ausgenommen haben und welche Probleme dadurch entstehen können. Das heißt, wir behalten die Notation in (3.1) bei und inkludieren einfach 0 in die Menge J .

3.3.1 Zentrale Diskretisierung

Zuerst betrachten wir das zentrale Verfahren

$$T w'(x) - V'(x) \frac{1}{2\Delta v} B w(x) = 0. \quad (3.7)$$

Aus dieser Gleichung erkennen wir, dass sich das lineare Differentialgleichungssystem mit 0 als diskreter Geschwindigkeit in ein Differential-Algebraisches-System (DAE) wandelt, da die Matrix der Geschwindigkeiten T nicht mehr regulär ist. Wir werden an einem einfachen Beispiel illustrieren, warum dadurch Probleme auftreten können.

Beispiel 3.3. Wir betrachten drei Stützstellen $v_1 = 1, v_0 = 0$ und $v_{-1} = 1$. Das Verfahren nimmt dann in Matrixform die Gestalt

$$\begin{pmatrix} 1 & & \\ & 0 & \\ & & -1 \end{pmatrix} \begin{pmatrix} w_1 \\ w_0 \\ w_{-1} \end{pmatrix}'(x) - V'(x) \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ w_0 \\ w_{-1} \end{pmatrix}(x) = 0 \quad (3.17)$$

mit den Randbedingungen

$$w_1(0) = w^+, \quad w_{-1}(L) = w^- \quad (3.18)$$

an.

Das DAE System mit konstanten Koeffizienten ist eindeutig lösbar, falls die Determinante

$$\det(T - \lambda V'(x) \frac{1}{2\Delta v} B) \quad (3.19)$$

nicht für alle $\lambda \in \mathbb{C}$ verschwindet ([10], [5]). Für unser System gilt aber $\det(T - \lambda V'(x) \frac{1}{2\Delta v} B) \equiv 0$ und damit können wir für gegebene Randbedingungen unendlich viele oder keine Lösungen erwarten, etwas das für ein numerisches Verfahren nicht sinnvoll ist.

Theorem 3.4. Für allgemeines Gitter $v_j, j \in J \cup \{0\}$ ist die DAE (3.7) mit Randbedingungen (3.8) nicht eindeutig lösbar.

Beweis. Für den Beweis siehe [3, Th. 4.4]. Dort wird die Wigner Gleichung semidiskretisiert und unser DAE System ist ein Spezialfall, des dort behandelten. \square

Dieses Theorem sagt uns, dass es für dieses Verfahren nicht möglich ist, das Gitter so zu wählen, dass $v_0 = 0$ enthalten ist.

3.3.2 Upwind Diskretisierung

Bei der Betrachtung des Upwind Verfahrens für diskrete Geschwindigkeit 0 betrachten wir die komponentenweise Darstellung (3.15) für $j = -1, 0, 1$. Sei zunächst $V'(x) < 0$, dann gilt

$$v_1 w_1'(x) - V'(x) \frac{1}{\Delta v} (w_1(x) - w_0(x)) = 0 \quad (3.20)$$

$$\underbrace{v_0}_{=0} w_0'(x) - V'(x) \frac{1}{\Delta v} (w_0(x) - w_{-1}(x)) = 0 \quad (3.21)$$

$$v_{-1} w_{-1}'(x) - V'(x) \frac{1}{\Delta v} (w_{-1}(x) - w_{-2}(x)) = 0. \quad (3.22)$$

Aus (3.21) folgt wegen $V'(x), \Delta v \neq 0$, dass $w_0(x) = w_{-1}(x)$. Substituiert man das in die Gleichung für $j = 1$ erhält man

$$v_1 w_1'(x) - V'(x) \frac{1}{\Delta v} (w_1(x) - w_{-1}(x)) = 0. \quad (3.23)$$

Das bedeutet, dass das Verfahren für $j = 0$ eigentlich keinen Beitrag liefert und man die Stützstelle auslassen kann, wenn man passend substituiert. Der Unterschied zum Upwind Verfahren aus Definition 3.2 ist, dass zwischen $j = 1$ und $j = -1$ ein Schritt der Länge $2\Delta v$ gemacht

wird. Damit ist aber auch $\frac{w_1(x) - w_{-1}(x)}{\Delta v}$ keine passende Approximation an $f_v(x, v_1)$, sondern approximiert in Wirklichkeit $2f_v(x, v_1)$ ¹. Für $V'(x) > 0$ können wir die Schritte mit vertauschten Rollen von $j - 1$ und $j + 1$ wiederholen und erhalten

$$v_{-1}w'_{-1}(x) - V'(x)\frac{1}{\Delta v}(w_1(x) - w_{-1}(x)) = 0. \quad (3.24)$$

Nehmen wir also Null implizit als diskrete Geschwindigkeit auf, können wir ein alternatives Upwindverfahren definieren.

Definition 3.5. Wir definieren die *Upwind Diskretisierung mit Null* der stationären Liouville Gleichung, wie in Definition 3.2 mit dem Unterschied, dass das Gitter $(v_j)_{j \in J}$ so gewählt wird, dass gilt

$$\begin{aligned} v_j - v_{j-1} &= 2\Delta v, & j = 1, \\ v_j - v_{j-1} &= \Delta v, & \text{sonst.} \end{aligned} \quad (3.25)$$

In der Implementierung unterscheiden sich die beiden Upwindverfahren aber kaum und liefern auch annähernd idente Ergebnisse, wie wir in Kapitel 6 noch sehen werden.

3.4 Eindeutige Lösbarkeit

Die definierten numerischen Verfahren machen nur Sinn, wenn sie für gegebene Randbedingungen eine eindeutige Lösung haben, da sonst keine Konvergenz erhofft werden kann. Wir können zeigen, dass die Differentialgleichungssysteme mit Randbedingungen, die bei den beiden Verfahren entstehen, eine eindeutige Lösung haben. Hauptsächlich ist diese Lösbarkeit abhängig von der Struktur der Differenzenmatrix B bzw. B_u .

Zuerst betrachten wir wieder die zentrale Diskretisierung. Wir werden uns jetzt an das Vorgehen in dem Artikel [3, Kapitel III] halten, in dem die Wignergleichung semidiskretisiert wird und ein ähnliches Differentialgleichungssystem entsteht, genauer eine Verallgemeinerung unseres Systems, in der die schiefe Symmetrie der Matrix B eine entscheidende Rolle spielt.

Theorem 3.6. *Das Randwertproblem der zentralen Semidiskretisierung (3.7), (3.8) ist eindeutig lösbar.*

Beweis. Wir werden hier den Beweis aus [3] skizzieren, um die Idee zu vermitteln. Wir betrachten also das Randwertproblem (3.7), (3.8).

- 1. Schritt: Transformation des DGL Systems

Es gelte die Notation für Matrizen in Blockstruktur

$$A := \frac{V'(x)}{2\Delta v}B = \begin{pmatrix} A^{++} & A^{+-} \\ A^{-+} & A^{--} \end{pmatrix}, \quad (3.26)$$

$$D := T^{-1} = \text{diag} \left(\left(\frac{1}{v_j} \right)_{j \in J} \right) = \begin{pmatrix} D^+ & 0 \\ 0 & D^- \end{pmatrix}, \quad (3.27)$$

$$|D| = \begin{pmatrix} D^+ & 0 \\ 0 & -D^- \end{pmatrix}, \quad (3.28)$$

wobei D^+ , bzw. $v = (v^+, v^-)^T$ hier immer die positiven bzw. negativen Indizes anspricht. A ist schiefssymmetrisch und es gilt $A^{++} = A^{--}$, $A^{+-} = (-A^{-+})^T$.

¹Taylorentwicklung einer glatten Funktion f liefert die Aussage.

Wir definieren den neuen Vektor $\sqrt{|D|}^{-1}z(x) := w(x)$ und transformieren (3.7) auf

$$z'(x) - \underbrace{\sqrt{|D|}^{-1}DA\sqrt{|D|}}_{=:C(x)}z(x) = 0 \quad (3.29)$$

mit den Randbedingungen

$$\begin{aligned} z^+(0) &= \sqrt{|D|}^{-1}w^+(0) =: g^+, \\ z^-(L) &= \sqrt{|D|}^{-1}w^-(L) =: g^-. \end{aligned} \quad (3.30)$$

• 2. Schritt: Spezielle Gestalt von $C(x)$

Die Transformation hat den Nutzen, dass die Matrix $C(x)$ eine besondere Gestalt hat. Wie man mit der Blockstrukturdarstellung nachrechnen kann, gilt

$$\begin{aligned} C^{++} &= (-C^{++})^T, & C^{--} &= (-C^{--})^T, \\ C^{-+} &= (C^{-+})^T, & C^{+-} &= (C^{+-})^T. \end{aligned} \quad (3.31)$$

• 3. Schritt: Anfangswertproblem ist eindeutig lösbar

Lemma 3.1 in [3] liefert uns, dass das Anfangswertproblem

$$z'(x) - C(x)z(x) = 0, \quad 0 < x < L \quad (3.32)$$

$$z(0) = z_0. \quad (3.33)$$

eindeutig lösbar ist und ein beschränkter, stetiger Propagator $U(x, y)$, $x, y \in (0, L)$ existiert, der die glatten Funktionen $F(x) = U(x, 0)$, $G(x) = U(0, x)$, $F^T(x)$ mit

$$F'(x) - C(x)F(x) = 0, \quad G'(x) + G(x)C(x) = 0, \quad F^{T'}(x)F^T(x) - F^T(x)C^T(x), \quad (3.34)$$

$$F(0) = G(0) = F^T(0) = Id, \quad (3.35)$$

liefert.

• 4. Schritt: Randwertproblem auf Anfangswertproblem umschreiben

In Analogie zu den Einströmdaten g^+, g^- bezeichnen wir die unbekanntenen Ausströmdaten mit h^+, h^- . Dann gilt für die Lösung z von (3.32) $z(0) = (g^+, h^-)^T$, $z(L) = (h^+, g^-)^T$ und falls die Ausströmdaten bekannt wären,

$$z(x) = U(x, 0) \begin{pmatrix} g^+ \\ h^- \end{pmatrix} = U(x, L) \begin{pmatrix} h^+ \\ g^- \end{pmatrix}, \quad 0 \leq x \leq L. \quad (3.36)$$

Die Idee ist jetzt h^- aus der letzten Gleichung zu eliminieren und h^+ daraus zu bestimmen, da der Rest aus bekannten Funktionen U, g^\pm besteht. Wir bezeichnen die Projektionen auf die Halbebenen der positiven bzw. negativen Geschwindigkeiten mit P^+ bzw. P^- und erhalten für h^+

$$[Id - \underbrace{P^+F(L)P^-G(L)P^+}_{=:K}] \begin{pmatrix} h^+ \\ 0 \end{pmatrix} = P^+F(L) \left[\begin{pmatrix} g^+ \\ 0 \end{pmatrix} + P^-G(L) \begin{pmatrix} 0 \\ g^- \end{pmatrix} \right] =: l(g), \quad (3.37)$$

$$[Id - K] \begin{pmatrix} h^+ \\ 0 \end{pmatrix} = l(g), \quad (3.38)$$

wobei die rechte Seite $l(g)$ bekannt ist. Falls das lineare Gleichungssystem (3.38) eindeutig lösbar ist, können wir h^+ und in Folge auch h^- aus (3.36) bestimmen.

• 5. Schritt: $[Id - K]$ ist regulär

Eine hinreichende Bedingung ist $K \leq 0$, denn daraus folgt $Id - K \geq Id$. Wir betrachten die Struktur von K .

$$K = \begin{pmatrix} F(L)^{+-}G(L)^{-+} & 0 \\ 0 & 0 \end{pmatrix} \quad (3.39)$$

Das heißt es genügt uns $K^{++} = F(L)^{+-}G(L)^{-+} \leq 0$ zu zeigen. Mit Hilfe der speziellen Struktur der Matrix $C(x)$ und den Funktionen G, F^T erhalten wir $G(L)^{-+} = (-F(L)^{+-})^T$ und damit

$$K^{++} = -F(L)^{+-}(F(L)^{+-})^T \leq 0. \quad (3.40)$$

Also haben wir erreicht, dass wir unser Randwertproblem (3.7), (3.8) als Anfangswertproblem (3.32), (3.33) schreiben können. Letzteres hat eine eindeutige Lösung und die eindeutige Rücktransformation liefert uns dann die Eindeutigkeit der Lösung der ursprünglichen Gleichung (3.7). \square

Um die eindeutige Lösbarkeit der Upwind Diskretisierung zu diskutieren, werden wir uns an [2] orientieren. Wir werden die nötigen Voraussetzungen an unsere Matrix nachrechnen um die dort angegebene Lösungstheorie verwenden zu können.

Theorem 3.7. *Die Semidiskretisierung mittels Upwind Verfahren und Upwind Verfahren mit Null führt auf ein eindeutig lösbares Randwertproblem (3.16), (3.8).*

Beweis. Wir betrachten das Upwind Verfahren und definieren zuerst die Hilfsmatrix A aus der Matrixdarstellung.

$$Tw'(x) + \underbrace{(-V'(x))\frac{1}{\Delta v}B_u(x)}_{:=A(x)}w(x) = 0$$

$A(x)$ ist stückweise stetig in x und beschränkt für $V(x)$ hinreichend glatt. Für $A^s := \frac{1}{2}(A + A^T)$ gilt

$$A^s(x) = \left\{ \begin{array}{l} -\frac{V'(x)}{2\Delta v} \begin{pmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & \ddots & \ddots & \ddots \end{pmatrix} & V'(x) < 0, \\ \frac{V'(x)}{2\Delta v} \begin{pmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & \ddots & \ddots & \ddots \end{pmatrix} & V'(x) > 0. \end{array} \right\} =: \left| \frac{V'(x)}{2\Delta v} \right| M \quad (3.41)$$

Wir erkennen M als Matrix des zentralen Differenzenquotienten für die zweite Ableitung. Es gilt $M > 0$ im Sinne der Matrix Definitheit, da aus

- M ist symmetrisch
- M ist diagonal dominant im Sinne von $M_{ii} \geq \sum_{i \neq j} |M_{ij}|$
- $M_{ii} > 0$

- M ist tridiagonal und für die nichttrivialen Einträge gilt $M_{ij} < 0, i \neq j$

die positiv Definitheit folgt ([7]). Also gilt $A^s(x) \geq 0$ für alle $x \in (0, L)$. Damit sind die Voraussetzungen (H1) und (H2) aus [2, Th. 1] erfüllt und letzteres Theorem garantiert die eindeutige Lösbarkeit. Für das Upwind Verfahren mit Null gelten die Schritte analog. \square

3.5 Numerik der Verfahren

Wir wollen hier diskutieren, welches Verhalten wir von den in Kapitel 3.1 und 3.2 vorgestellten Verfahren von Semidiskretisierung erwarten können. Wir werden uns dabei hauptsächlich auf Resultate aus [12] berufen. In dieser Referenz von LeVeque werden Volldiskretisierungen von hyperbolischen Erhaltungsgesetzen, also partielle Differentialgleichungen der Form

$$u_t + f(u)_x = 0, \quad (3.42)$$

behandelt. Wir wollen die Erwartungen an unsere Verfahren mit den dortigen Ergebnissen motivieren, obwohl der Zusammenhang mit unseren Problemen auf den ersten Blick nicht offensichtlich ist. Eine Semidiskretisierung ist aber nur eine schwächere Form der Diskretisierung und mittels der Transformation $u := vf$ gilt für die stationäre Liouville Gleichung (1.4)

$$u_x(x, v) + \left(-V'(x) \frac{1}{v} u(x, v) \right)_v = 0, \quad (3.43)$$

also genau die Erhaltungsform (3.42), mit x interpretiert als Zeitvariable und v als Ort. Wir definieren zuerst, wie bei Volldiskretisierungen, den lokalen Abschneidefehler und den Begriff der Konsistenz ([12, Kapitel 10.3]).

Definition 3.8. Der *lokale Abschneidefehler*² $L_{\Delta v}(x, v)$ für glatte Lösungen eines numerische Verfahrens ist definiert als der Fehlerterm dominanter Ordnung, für $\Delta v \rightarrow 0$, der beim Einsetzen der exakten Lösung in das numerische Verfahren entsteht.

Definition 3.9. Eine numerische Methode heißt *konsistent*, falls

$$\|L_{\Delta v}(\cdot, \cdot)\| \rightarrow 0 \quad \text{für} \quad \Delta v \rightarrow 0. \quad (3.44)$$

Genauer, ein Verfahren ist *konsistent von der Ordnung p* , falls

$$\|L_{\Delta v}(\cdot, \cdot)\| = O(\Delta v^p) \quad \text{für} \quad \Delta v \rightarrow 0. \quad (3.45)$$

Wir werden in Folge diese Größen für unsere beiden Verfahren betrachten und mit Resultaten aus [12] verknüpfen. Wir beginnen mit der zentralen Diskretisierung und folgendem Lemma.

Lemma 3.10. *Die zentrale Semidiskretisierung (3.6) ist von Konsistenzordnung 2 mit lokalem Abschneidefehler*

$$L_{\Delta v}^z(x, v) = -\frac{1}{6} V'(x) \Delta v^2 f_{vvv}(x, v) \quad (3.46)$$

und approximiert die modifizierte Gleichung

$$vf_x(x, v) - V'(x) f_v(x, v) = \frac{1}{6} V'(x) \Delta v^2 f_{vvv}(x, v) \quad (3.47)$$

mit 3. Ordnung.

²Wir definieren hier nur verbal um das Einführen neuer Notation zu sparen.

Beweis. Laut Definition des Abschneidefehlers liefert mit Einsetzen der glatten Lösung von (1.4) in (3.6) und dem Satz von Taylor ([13])

$$\begin{aligned}
L_{\Delta v}^z(x, v) &= v f_x(x, v) - V'(x) \frac{1}{2\Delta v} (f(x, v + \Delta v) - f(x, v - \Delta v)) \\
&\stackrel{\text{Taylor}}{=} v f_x(x, v) - V'(x) \frac{1}{2\Delta v} (f(x, v) + \Delta v f_v(x, v) + \frac{1}{2} \Delta v^2 f_{vv}(x, v) + \frac{1}{6} \Delta v^3 f_{vvv}(x, v) \\
&\quad - f(x, v) + \Delta v f_v(x, v) - \frac{1}{2} \Delta v^2 f_{vv}(x, v) + \frac{1}{6} \Delta v^3 f_{vvv}(x, v) + O(\Delta v^4)) \\
&= \underbrace{v f_x(x, v) - V'(x) f_v(x, v)}_{=0, \quad (1.4)} - \frac{1}{6} V'(x) \Delta v^2 f_{vvv}(x, v) + O(\Delta v^3) \\
&= -\frac{1}{6} V'(x) \Delta v^2 f_{vvv}(x, v).
\end{aligned}$$

Der zweite Teil des Lemmas folgt daraus sofort. \square

Die modifizierte Gleichung des zentralen Schemas (3.47) ist wegen f_{vvv} eine dispersive Gleichung. Für Verfahren zweiter Ordnung führt dieser Dispersionfehler bei der numerischen Lösung hyperbolischer Erhaltungsgesetze auf Oszillationen in der Nähe von Unstetigkeitsstellen ([12, Kapitel 11.1.2]). Ein Phänomen, welches wir auch hier erwarten. Das heißt wir rechnen trotz guter Konsistenzordnung mit oszillatorischen Fehlern in Form von Schwingungen in der Nähe von Knicken und Sprungstellen der Lösung. Für das Upwind Verfahren diskutieren wir analog in

Lemma 3.11. *Die Upwind Semidiskretisierung ist von erster Konsistenzordnung mit lokalem Abschneidefehler*

$$L_{\Delta v}^u(x, v) = -\frac{1}{2} |V'(x)| \Delta v f_{vv}(x, v) \quad (3.48)$$

und approximiert die modifizierte Gleichung

$$v f(x, v)_x - V'(x) f(x, v)_v = \frac{1}{2} |V'(x)| \Delta v f_{vv}(x, v) \quad (3.49)$$

mit 2. Ordnung.

Beweis. Für den Abschneidefehler betrachten wir wieder eine glatte Lösung von (1.4) und setzen in das numerische Verfahren (3.15) ein. Sei zunächst $V'(x) < 0$.

$$\begin{aligned}
L_{\Delta v}^u(x, v) &= v f_x(x, v) - V'(x) \frac{1}{\Delta v} (f(x, v) - f(x, v - \Delta v)) \\
&\stackrel{\text{Taylor}}{=} v f_x(x, v) - V'(x) \frac{1}{\Delta v} \left(f(x, v) - f(x, v) + \Delta v f_v(x, v) - \frac{1}{2} \Delta v^2 f_{vv}(x, v) + O(\Delta v^3) \right) \\
&= \underbrace{v f_x(x, v) - V'(x) f_v(x, v)}_{=0, \quad (1.4)} + V'(x) \frac{1}{2} \Delta v f_{vv}(x, v) + O(\Delta v^2) \\
&= \frac{1}{2} V'(x) \Delta v f_{vv}(x, v) + O(\Delta v^2).
\end{aligned} \quad (3.50)$$

Für $V'(x) < 0$ gilt analog

$$\begin{aligned}
L_{\Delta v}^u(x, v) &= v f_x(x, v) - V'(x) \frac{1}{\Delta v} (f(x, v + \Delta v) - f(x, v)) \\
&= -\frac{1}{2} V'(x) \Delta v f_{vv}(x, v) + O(\Delta v^2).
\end{aligned} \quad (3.51)$$

(3.50) und (3.51) implizieren (3.48) und die zweite Aussage des Lemmas folgt sofort aus der ersten. \square

Die modifizierte Gleichung des Upwind Schemas (3.49) ist eine Diffusionsgleichung wegen des Diffusionsterms f_{vv} . In der hyperbolischen Erhaltungstheorie ist die Stabilität eines voll-diskreten Verfahrens erster Ordnung äquivalent dazu, dass der Koeffizient des Diffusionsterms positiv semidefinit ist. Bei uns ist das offensichtlich der Fall, da $\Delta v > 0$ und $|V'| \geq 0$ ist. Hyperbolische Erhaltungsgesetze (3.42) werden mit zunehmender Zeitevaluation in der Nähe von Unstetigkeiten durch den Diffusionsfehler ausgeschmiert. Das heißt die Lösung wird an diesen Sprungstellen oder auch an Knicken geglättet ([12, Kapitel 11.1.1]). Diese Aussage motiviert auch eine Glättung für das Upwind Schema bei unserem Problem, da x wieder als Zeitvariable interpretiert werden kann.

Bei den Referenzen auf die hyperbolischen Erhaltungsgesetze wurde die Ortsvariable x stets als Zeit t interpretiert. Führen wir diesen Gedanken weiter, dann lässt sich noch eine weitere Art von Upwind Verfahren als sinnvoll vermuten. Für $v \neq 0$ können wir (1.4) durch v dividieren und erhalten

$$0 = f_x(x, v) + \left(\frac{-V'(x)}{v} f_v(x, v) \right) := f_x(x, v) + a(x, v) f_v(x, v). \quad (3.52)$$

Definieren wir jetzt das Upwind Verfahren in Abhängigkeit des Vorzeichens von $a(x, v)$, dann erhalten wir vier Bedingungen an die Richtung der Diskretisierung in v . Warum dieses Verfahren aber nicht weiter betrachtet wird, werden wir im nächsten Kapitel in Lemma 4.5 erläutern.

4 Erhaltungsgrößen

In diesem Kapitel werden wir uns den Erhaltungsgrößen der stationären Liouville Gleichung (1.4) widmen. Eine Erhaltungsgröße ist ein Wert der sich während einem physikalischen Prozess nicht ändert. Das prominenteste Beispiel dafür ist wohl die Energie in einem konservativen System, die bekanntlich konstant bleibt.

Betrachtet man die Energie eines Teilchens mit Masse m , welches sich unter Einwirkung des Potentials $V(x)$ mit der Geschwindigkeit v bewegt, erhält man mit Hilfe des Noether Theorems ([4])

$$E(t, x, v) = E_{kin} + E_{pot} = \frac{1}{2}mv^2 + V(x) = \text{const.} \quad (4.1)$$

Für verschiedene Energiewerte führt (4.1) genau auf die Charakteristiken der stationären Liouville Gleichung (1.4) mit Masse $m = 1$, welche genau diese Art von Partikelbewegung beschreibt. Wir sind hier nun an Erhaltungsgrößen interessiert, die von der Lösung von (1.4) abhängen. Für Lösungen eines numerischen Verfahrens wollen wir dann das gleiche Verhalten für die diskrete Version dieser Größe haben. Dadurch können wir eventuell Verfahren ausschließen, die den physikalischen Hintergrund des Problems nicht korrekt abbilden. Deshalb betrachten wir den Strom, also den Partikelfluss pro Zeiteinheit durch einen Punkt, da wir uns auf ein eindimensionales Problem beschränkt haben. Der physikalische Strom \mathfrak{J} ist definiert durch

$$\mathfrak{J}(x) := \int_{\mathbb{R}} v f(x, v) dv. \quad (4.2)$$

4.1 Analytischer Fall

Wir zeigen jetzt, dass der Strom \mathfrak{J} von analytischen Lösungen f der stationären Liouville Gleichung (1.4) erhalten wird, das heißt $\mathfrak{J}(x) = \text{const}$, beziehungsweise $\frac{d}{dx}\mathfrak{J} = 0$.

Lemma 4.1. Sei $f(x, v)$ eine glatte Lösung der stationären Liouville Gleichung

$$vf_x(x, v) = V'(x)f_v(x, v) \quad (1.4)$$

mit den Randbedingungen mit kompaktem Träger

$$\begin{aligned} f(0, v) &= f^+, v > 0 \\ f(L, v) &= f^-, v < 0 \end{aligned} \quad (1.5)$$

Dann erhält f den Strom $\mathfrak{J}(x)$.

Beweis. Wir integrieren die komplette Gleichung (1.4) nach $\int_{\mathbb{R}} dv$ und erhalten

$$\begin{aligned} \int_{\mathbb{R}} vf_x(x, v)dv &= \int_{\mathbb{R}} V'(x)f_v(x, v)dv \\ \int_{\mathbb{R}} (vf(x, v))_x dv &= V'(x) \int_{\mathbb{R}} f_v(x, v)dv. \end{aligned}$$

$f(x, \cdot)$ ist auf Grund der Lösungsdarstellung in Kapitel 2.1 für jedes feste x in v integrierbar und hat kompakten Träger, falls die Randbedingungen kompakten Träger haben. Damit hat auch $vf(x, v)$ kompakten Träger und ist integrierbar in v für festes x . Weiters haben wir f als hinreichend glatt angenommen, also können wir die Differentiation und das Integral auf der linken Seite im letzten Schritt vertauschen. Das Integral auf der rechten Seite können wir direkt ausrechnen und durch den kompakten Träger von $f(x, \cdot)$ verschwindet es.

$$\begin{aligned} \partial_x \underbrace{\int_{\mathbb{R}} vf(x, v)dv}_{=\mathfrak{J}(x)} &= \underbrace{V'(x)}_{< \infty} \lim_{\alpha \rightarrow \infty} \underbrace{(f(x, \alpha) - f(x, -\alpha))}_{\rightarrow 0} = 0 \\ \frac{d}{dx} \mathfrak{J}(x) &= 0 \end{aligned}$$

□

4.2 Semidiskreter Fall

Wir wollen nun zeigen, dass unsere in Kapitel 3 definierten numerischen Verfahren auch den Strom erhalten. Im semidiskreten Fall wird dv zu einem diskreten Maß auf $\mathbb{R}^{|J|}$. Wir definieren den Strom daher analog zu (4.2) für $w(x) \in \mathbb{R}^{|J|}$ durch

$$\mathfrak{J}(x) := \sum_{j \in J} v_j w_j(x) = \sum_{j=-N}^N v_j w_j(x). \quad (4.3)$$

In den nächsten beiden Lemmata werden wir Voraussetzungen an das v -Gitters herleiten, dass sowohl das zentrale Schema aus Definition 3.1, als auch das Upwindschema aus 3.2, auf eine Invarianz von $\mathfrak{J}(x)$ führen.

Lemma 4.2. Für die Lösung $w(x)$ der zentral semidiskretisierten Liouville Gleichung

$$Tw(x) = V'(x) \frac{1}{2\Delta v} Bw(x) \quad (3.16)$$

mit den Randbedingungen mit kompaktem Träger

$$\begin{aligned} w_j(0) &= w_j^+, j \in J^+ \\ w_j(L) &= w_j^-, j \in J^- \end{aligned} \quad (3.8)$$

gilt $\frac{d}{dx} \mathfrak{J}(x) = 0$ für passendes v -Gitter.

Beweis. Betrachten wir die Gleichung (3.16) komponentenweise. Unter Berücksichtigung der Randbedingung $w_{N+1}(x) = w_{-(N+1)}(x) = 0$ für alle $x \in [0, L]$ erhalten wir

$$v_j w'_j(x) = V'(x) \frac{1}{2\Delta v} (w_{j+1}(x) - w_{j-1}(x)), \quad j \in J.$$

Wir gehen jetzt analog zu Lemma 4.1 vor und summieren die Gleichung über J auf und nutzen von erster auf zweite Zeile $|J| < \infty$.

$$\begin{aligned} \sum_{j \in J} v_j w'_j(x) &= \sum_{j \in J} V'(x) \frac{1}{2\Delta v} (w_{j+1}(x) - w_{j-1}(x)) \\ \frac{d}{dx} \sum_{j \in J} v_j w_j(x) &= V'(x) \frac{1}{2\Delta v} \sum_{j \in J} (w_{j+1}(x) - w_{j-1}(x)) \end{aligned}$$

Der Großteil der Terme in der Summe hebt sich aufgrund der Periodizität der Indizes auf und es verbleibt nur

$$\begin{aligned} \frac{d}{dx} \mathfrak{J}(x) &= V'(x) \frac{1}{2\Delta v} \sum_{j=-N}^N (w_{j+1}(x) - w_{j-1}(x)) \\ \frac{d}{dx} \mathfrak{J}(x) &= V'(x) \frac{1}{2\Delta v} \underbrace{(-w_{-N-1}(x) - w_{-N}(x) + w_N(x) + w_{N+1}(x))}_{=0} \\ \frac{d}{dx} \mathfrak{J}(x) &= V'(x) \frac{1}{2\Delta v} (-w_{-N}(x) + w_N(x)). \end{aligned} \quad (4.4)$$

Mit dem gleichen Argument wie bei der Herleitung des Schemas in Kapitel 3.1, können wir das Gitter so wählen, dass $w_N(x) = w_{-N}(x) \equiv 0$ für jede beliebige Randbedingung mit kompaktem Träger erfüllt ist. (4.4) liefert dann die Aussage $\mathfrak{J}(x) = \text{const.}$ \square

Lemma 4.3. *Die Lösung $w(x)$, der mittels Upwind Schema semidiskretisierten Liouville Gleichung*

$$Tw(x) = V'(x) \frac{1}{\Delta v} B_u(x) w(x) \quad (3.16)$$

mit den Randbedingungen mit kompaktem Träger (3.8), erhält ebenfalls den semidiskreten Strom $\mathfrak{J}(x)$ auf geeignetem v -Gitter, das heißt $\frac{d}{dx} \mathfrak{J}(x) = 0$.

Beweis. Wir betrachten zuerst den Fall $V'(x) < 0$ und die Gleichung (3.16) wieder komponentenweise. Mit den Randbedingung $w_{N+1}(x) = w_{-(N+1)}(x) \equiv 0$ erhalten wir

$$v_j w'_j(x) = V'(x) \frac{1}{\Delta v} (w_j(x) - w_{j-1}(x)), \quad j \in J.$$

Analoges Vorgehen, wie beim Beweis des vorherigen Lemmas 4.2, führt auf

$$\frac{d}{dx} \mathfrak{J}(x) = V'(x) \frac{1}{\Delta v} (w_N(x) - \underbrace{w_{-N-1}(x)}_{=0}) = V'(x) \frac{1}{\Delta v} (w_N(x)). \quad (4.5)$$

Für $V'(x) > 0$ liefern die gleichen Schritte

$$\frac{d}{dx} \mathfrak{J}(x) = V'(x) \frac{1}{\Delta v} (\underbrace{w_{N+1}(x)}_{=0} - w_{-N}(x)) = V'(x) \frac{1}{\Delta v} (w_{-N}(x)) \quad (4.6)$$

und für $V'(x) = 0$ ist die Aussage natürlich auch erfüllt. Also können wir mit den gleichen Bedingungen an das Gitter, wie in Lemma 4.2 wieder den Erhalt des Stromes wegen (4.5) und (4.6) garantieren. \square

Korollar 4.4. Das Upwind Schema mit Null erhält auch den semidiskreten Strom $\mathfrak{J}(x)$.

Lemma 4.5. Das in Kapitel 3.5, (3.52) vorgestellte Upwind Schema, in dem x als Zeitvariable interpretiert wird, erhält den Strom im Allgemeinen nicht.

Beweis. Wir betrachten nur $V'(x) < 0$, da dies ausreichend ist, um die Probleme aufzuzeigen. Dann nimmt das Verfahren die Gestalt

$$v_j w'_j(x) = V'(x) \frac{1}{\Delta v} (w_j(x) - w_{j-1}(x)), \quad j \in J^+ \quad (4.7)$$

$$v_j w'_j(x) = V'(x) \frac{1}{\Delta v} (w_{j+1}(x) - w_j(x)), \quad j \in J^- \quad (4.8)$$

an. Aufsummieren über J liefert dann

$$\begin{aligned} \frac{d}{dx} \mathfrak{J}(x) &= V'(x) \frac{1}{\Delta v} \left(\sum_{j \in J^+} (w_j(x) - w_{j-1}(x)) + \sum_{j \in J^-} (w_{j+1}(x) - w_j(x)) \right) \\ \frac{d}{dx} \mathfrak{J}(x) &= V'(x) \frac{1}{\Delta v} (w_N(x) - w_{-1} + w_1(x) - w_{-N}(x)). \end{aligned}$$

Das Gitter kann wieder so gewählt werden, dass $w_N(x)$ und $w_{-N}(x)$ für alle x verschwinden. $w_{-1}(x)$ und $w_1(x)$ sind aber die Werte auf den ersten Diskretisierungslinien v_{-1} und v_1 in die negative und positive v Richtung. Dort kann man für allgemeine Randbedingungen und Potentiale aber nichts garantieren, da die Charakteristiken immer durch dieses Gebiet laufen. \square

5 Implementierung in MATLAB

In diesem Kapitel werden wir auf die Probleme und Besonderheiten eingehen, die bei der Implementierung der Verfahren aus Definition 3.1, 3.2 und 3.5 in die Programmiersprache MATLAB entstehen und gegeben sind.

Als Erstes müssen wir auf die Indizierung aufpassen. Der Vektor $(v_j)_{j \in J} = (v_N, \dots, v_{-N})^T$ hat in MATLAB die Gestalt $(v(1), v(2), \dots, v(2N))^T$. Wir werden die Diskretisierungen aber mittels ihrer Matrizenformen realisieren, um eine Umm Nummerierung der Indizes zu vermeiden. Die Position der Werte innerhalb des Vektors ändert sich ja nicht.

Weiters möchten wir die Wahl des Gitters $(v_j)_{j \in J}$ diskutieren. Wir haben vorgegeben, dass die Randbedingungen kompakten Träger haben müssen. Dadurch wird die Information nur auf ein beschränktes Gebiet in der (x, v) -Ebene ausgebreitet, vergleiche Kapitel 2.1. Es ist daher wünschenswert unser Gebiet nicht zu groß zu wählen, um keine unnötige Rechenleistung zu verschwenden und die Diskretisierungspunkte dort zu haben, wo auch etwas passiert. Wir schneiden daher den Bereich in der v -Richtung so ab, dass nur wenig Fläche zwischen dem Rand des Gebiets und dem nichttrivialen Bereich der Lösung verbleibt. Das folgende Beispiel illustriert dieses Vorgehen am Beispiel des Potentials aus Kapitel 6.1.

Beispiel 5.1. Sei $L = 2$ und $V(x) = (x - \frac{L}{2})^2$. Für die Randbedingungen gelte $f^- \equiv 0$ und $\text{supp } f^+ = [a, b] \subset \mathbb{R}^+$, $f^+(v) = \sin(\frac{\pi}{b-a}(v - a))$.

Dann erfüllen die Charakteristiken die implizit gegebene Durchlaufungsform

$$(x - \frac{L}{2})^2 + \frac{v}{2} = s^2, s \in \mathbb{R}^+,$$

siehe Lemma 2.1. Der höchste v -Wert v_{max} , für den $f(x, v_{max})$ noch Information liefert, liegt auf der Charakteristik, startend von $(x = 0, v = b)$, vergleiche Abb. 3. Wir berechnen daher den maximalen v -Wert dieser Kurve.

$$(x = 0, v = b) : \quad \left(0 - \frac{L}{2}\right)^2 + \frac{b}{2} = s^2 \Rightarrow s = \sqrt{\frac{b}{2} + \frac{L^2}{4}}$$

$$(x = \frac{L}{2}, v = v_{max}) : \quad \left(\frac{L}{2} - \frac{L}{2}\right)^2 + \frac{v_{max}}{2} = s^2 \Rightarrow v_{max} = \sqrt{2}s = \sqrt{2}\sqrt{\frac{b}{2} + \frac{L^2}{4}}$$

Um das Gebiet nicht zu knapp abzuschneiden und eventuell auftretende Oszillation, Fehler oder numerische Ungenauigkeit zu verursachen, addieren wir noch einen Sicherheitsterm δ , welcher der Lösung etwas Spielraum lässt. In der Anwendung hat sich zum Beispiel $\delta = \frac{v_{max}}{6}$ als ausreichend erwiesen.

In Kapitel 4.2 haben wir aber noch weitere Voraussetzungen an das Gitter festgelegt, um die Verfahren sinnvoll zu definieren. Es sollen nämlich die letzten beiden Gitterpunkte in positive und negative v -Richtung bereits die Nullfunktion für die Lösung $w_j(x)$ implizieren, um den Strom zu erhalten. Wir fassen das in Beispiel 5.1 definierte v_{max} als exakte Abgrenzung an das Gebiet ohne Information auf, der Sicherheitsterm ist beliebig. Ein Gitter mit $v_N = v_{max}$ kann die geforderte Bedingung im Allgemeinen also nicht garantieren. Wir wählen daher das Gitter so, dass $v_{max} = v_{N-2}$ gilt, woraus $w_N - 1 = w_N = w_{N+1} = 0$ folgt. Für die negative Richtung kann man das Ganze analog durchspielen.

Wir wollen dem Programm später die Anzahl der Stützstellen $2N$ vorgeben. Darum definieren wir für das zentrale Verfahren aus Definition 3.1 und das Upwind Verfahren aus Definition 3.2 das Gitter mit

$$\Delta v := \frac{2v_{max}}{2N - 5}, \quad \begin{aligned} v_j &:= \left(j - \frac{1}{2}\right) \Delta v, & j = 1, 2, \dots, N \\ v_j &:= \left(j + \frac{1}{2}\right) \Delta v, & j = -1, -2, \dots, -N. \end{aligned} \quad (5.1)$$

Dann gilt $v_j - v_{j-1} = \Delta v$ und

$$\begin{aligned} v_N &= \left(N - \frac{1}{2}\right) \Delta v = \left(N - \frac{1}{2}\right) \frac{2v_{max}}{2N - 5} = \frac{v_{max}(2N - 1)}{2N - 5} \\ &= \frac{v_{max}(2N - 5 + 4)}{2N - 5} = \frac{v_{max}(2N - 5)}{2N - 5} + \frac{4v_{max}}{2N - 5} \\ &= v_{max} + 2\Delta v. \end{aligned}$$

Für v_{-N} gilt dann analog $v_{-N} = -v_{max} - 2\Delta v$. Für das Upwind Verfahren mit Null aus Definition 3.5 muss das Gitter etwas anders gewählt werden.

$$\Delta v := \frac{v_{max}}{N - 2}, \quad v_j := j\Delta v, \quad j = -N, \dots, -1, 1, \dots, N. \quad (5.2)$$

Damit haben wir erreicht, dass $v_1 - v_{-1} = 2\Delta v$ und $v_j - v_{j-1} = \Delta v$ für die verbleibenden Indizes gilt, wie es in der Definition verlangt ist, und

$$\begin{aligned} v_N &= N\Delta v = N \frac{v_{max}}{N - 2} = \frac{v_{max}(N - 2 + 2)}{N - 2} \\ &= \frac{v_{max}(N - 2)}{N - 2} + \frac{2v_{max}}{N - 2} \\ &= v_{max} + 2\Delta v. \end{aligned}$$

$v_{-N} = -v_{max} - 2\Delta v$ funktioniert wieder analog.

Die Semidiskretisierungen führen im Gegensatz zu einer Volldiskretisierung nicht auf ein algebraisches Gleichungssystem oder einen Algorithmus, sondern auf ein System von Differentialgleichungen (3.7), (3.16) mit Randbedingungen (3.8). Dieses System werden wir mittels programminternen Randwertsolver lösen, da Kapitel 3.4 die Existenz einer Lösung sichert. In MATLAB stehen dafür die Routinen vierter Ordnung `bvp4c` und fünfter Ordnung `bvp5c` zur Verfügung. Wir begnügen uns hier mit dem Solver vierter Ordnung, um mehr Effizienz zu erhalten. Das Lösen der großen Anzahl an Differentialgleichungen benötigt genügend Rechenleistung. `bvp4c` ist ein finites Differenzen Verfahren, welches mit der drei-stufigen Lobatto Formel arbeitet und eine stetig differenzierbare Polynom-Lösung liefert bzw. sucht ([14]). Die Syntax

```
sol = bvp4c( odefun, bcfun, solinit, options)
```

erfordert neben den Differentialgleichung `odefun` und den Randbedingungen `bcfun` auch einen Startwert für die Iteration des Solvers auf einem x -Gitter `solinit`. Für diesen Startwert nehmen wir die Lösung an, welche die Randbedingungen konstant in x -Richtung transportiert. Wir werden sehen für Beispiel A in Kapitel 6.1 ist das eine gute Annahme und für das Beispiel B in Kapitel 6.2 reicht das immer noch aus, um gegen die zu erwartende Lösung zu konvergieren. Weiters muss darauf aufgepasst werden, dass der Solver standardmäßig nur eine beschränkte Anzahl an Punkten für das x -Gitter zulässt, unabhängig von dem Gitter auf dem man den Startwert definiert. Dieses Limit skaliert mit der Anzahl der Differentialgleichung in `odefun`, genauer gilt $N_{max} = \lfloor \frac{1000}{n} \rfloor$, wobei N_{max} die Anzahl der x -Punkte, n die Anzahl der Gleichungen und die Klammer die abrundende Gaußklammer ist. Gehen wir von einer Semidiskretisierung mit $2N = 200$ Punkten für v aus, verbleiben uns nur fünf Punkte für das Lösen des Differentialgleichungssystems in x über. Abhilfe schafft man hier mit `options = bvpset('Nmax', 200)`, was das Limit der Punkte auf beispielsweise 200 erhöht. Vor allem beim Lösen des "Schwierigeren" Beispiel B's ist das nötig, da es sonst zu großen Fehlerresiduen kommen kann.

Ein weiteres Problem, welches bei der Implementierung des zentralen Schemas passiert, ist, dass die Wahl des Gitters, für Δv klein genug, nicht mehr vom Randwertsolver in MATLAB verarbeitet werden kann. Sobald die Werte der Gitterpunkte $(v_j)_{j \in J}$ zu nahe an Null sind stößt die Routine `bvp4c` auf den Fehler `SingularJacobion encountered`. Dieser kann mehrere Ursachen haben, entweder treten Singularitäten in der Gleichung auf, die Lösung ist nicht mehr eindeutig oder die initiale Vermutung an die Lösung konvergiert nicht mehr. Feineres x -Gitter und andere Schätzlösungen schaffen hier keine Abhilfe. Eine Lösung kann erhalten werden, wenn man ein Sicherheitsdelta um $v = 0$ legt, das heißt alle Gitterpunkte mit $|v_j| < \delta$ werden weggelassen. Das Sicherheitsdelta führt aber dazu, dass die finiten Differenzenquotienten nicht mehr die Ableitung um Null approximieren, wie auch beim Upwind Verfahren mit Null. Für qualitatives Verhalten kann man in der Praxis aber mit diesem Fehler arbeiten, da er im Allgemeinen nicht zu groß ist. Der Grund, für den vom Solver produzierten Fehler, ist jedoch nicht offensichtlich. Die Matrix der diskreten Geschwindigkeiten T wird zwar invertiert um das Differentialgleichungssystem aufzustellen, siehe (3.7), und dort entstehen für kleine v_j große Werte, jedoch geschieht dies auch bei den beiden Upwind Implementierungen und dort treten keine Probleme auf und Theorem 3.6 garantiert uns eindeutige Lösbarkeit des Differentialgleichungssystems.

6 Beispiele

6.1 Beispiel A

Wir wollen die angeführte Theorie nun in praktischen Beispielen testen und das numerische Verhalten der Verfahren überprüfen. In unserem ersten Beispiel betrachten wir eine relativ

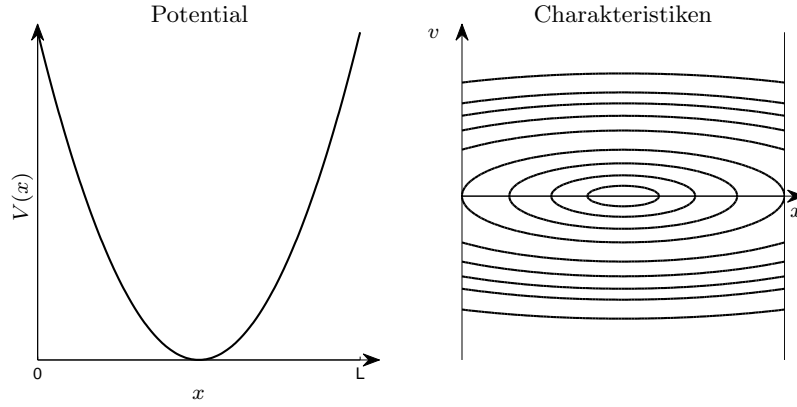


Abbildung 4: Das in Beispiel A gegebene Potential links und die dadurch bestimmten Charakteristiken rechts.

brave Situation und legen das Hauptaugenmerk neben dem Vergleich mit der exakten Lösung auf die Diffusion des Upwind Schemas, der Erhaltung des Stromes und den Oszillationen des zentralen Verfahrens.

Als Randbedingungen für $x \in [0, L]$, $L = 2$, $v \in \mathbb{R}$ seien

$$\text{supp} f^+ = [2, 4] =: [a, b], \quad \text{supp} f^- = \emptyset, \quad (6.1)$$

$$f^+(v) = \sin\left(\frac{\pi}{b-a}(v-a)\right), \quad f^-(v) = 0, \quad (6.2)$$

$$V(x) = \left(x - \frac{L}{2}\right)^2, \quad (6.3)$$

gegeben.

6.1.1 Lösungen

Wir werden zuerst die exakte Lösung bestimmen, um die numerischen Ergebnisse dann sinnvoll vergleichen zu können. Aus dem gegebenen Potential V können wir mithilfe von Lemma 2.1 sofort die Gestalt der Charakteristiken als die Ellipsen der Form

$$\left(x - \frac{L}{2}\right)^2 + \frac{v^2}{2} = \text{const} \in \mathbb{R}^+ \quad (6.4)$$

identifizieren, siehe Abb. 4. Wir wissen nun aus Kapitel 2.1, dass die Randbedingungen entlang dieser Charakteristiken transportiert werden und können die Lösung als Referenz für die numerischen Verfahren nehmen.

Bemerkung 6.1. Für den unkontrollierten Bereich nehmen wir Null als exakte Lösung. Wir motivieren diese Annahme mit den Ergebnissen aus Kapitel 3.5. In Lemma 3.11 wird der dominante Fehler für das Upwind Verfahren als Diffusionsfehler identifiziert. Diese Diffusion wird die bekannten Werte entlang des Randes der Ellipse des unkontrollierten Bereiches in das Innere transportieren. Die Randwerte entstehen aus den Charakteristiken, startend von $(x, v) = (0, 0)$ und $(x, v) = (L, 0)$, und ergeben sich zu $f^+(0) = 0$, $f^-(0) = 0$, da wir die Träger der Randbedingungen so gewählt haben, dass Null nicht enthalten ist. Für das zentrale Verfahren werden wir sehen, dass diese Annahme auch zutrifft.

Sollten die Randbedingungen an Null jedoch nicht trivial sein, ergibt sich ein Plateau in der Höhe des Mittelwerts der beiden Randwerte, mehr dazu in den Kapiteln 6.3 und 6.4.

Die Rechnungen in MATLAB werden mit einer maximalen Anzahl von 100 x - Punkten durchgeführt auf einem v -Gitter von $2N = 200$ Punkten, vergleiche Kapitel 5. In Abb. 12³ sieht man die Lösung mittels zentralem Verfahren. Die Draufsicht auf dem linken Bild zeigt deutlich den Transport der Randbedingungen während das vergrößerte rechte Bild die entstandenen Oszillationen hervorhebt. Man erkennt auch gut, das um die Null gelegte Sicherheitsdelta, welches den Durchlauf des Rechenalgorithmus erst ermöglicht.

Die beiden Upwind Verfahren, mit und ohne Null, unterscheiden sich in ihren numerischen Ergebnissen kaum, siehe Abb. 13. Wir werden deshalb im Weiteren nur mehr das zentrale und das Upwind Verfahren vergleichen, wenn wir spezielle Beispiele herausgreifen. Die Upwind Verfahren zeigen beide eine deutliche Diffusion, also eine Glättung entlang der Charakteristiken, wieder in selbem Ausmaß. Dadurch werden die Randwerte nicht konstant vom linken an den rechten Rand transportiert, sondern fallen leicht und der Träger vergrößert sich etwas, wodurch der Partikelstrom konstant bleibt.

Das Upwind Schema zeigt jedoch weit weniger Fehler in Bezug auf Oszillationen. Abbildung 14 vergleicht die Ergebnisse mit dem zentralen Verfahren mittels eines `contourplot`'s, welcher die Höhenschichtlinien der Figur darstellt. In dem Bereich, der die Null Lösung sein sollte, ist das Upwind Verfahren um einiges „ruhiger“, während das zentrale Verfahren vor allem innerhalb der Charakteristiken einige Oszillationen aufweist.

6.1.2 Fehlergrößen

Wir wollen nun das Verhalten der Fehler, die wir auf den Abbildungen erkennen, betrachten. Das heißt wir betrachten nicht ein fixes Gitter, wie im vorherigen Kapitel, sondern lassen die Gitterbreite Δv gegen Null gehen. Die Einstellungen des x -Gitters behalten wir bei, da es für eine Semidiskretisierung als kontinuierlich angenommen wird. Da für $2N = 200$ schon viel Rechenleistung benötigt wird, starten wir bei $2N = 100$ und verfeinern Gitter sukzessive. Wir testen zuerst den Fehler gegen die exakte Lösung, siehe Abb. 5. Wir definieren dazu die (semi-)diskreten Normen

$$\|w\|_\infty := \max_{ij} |w_j(x_i)| \quad (6.5)$$

$$\|w\|_2 := \Delta x \Delta v \sqrt{\sum_{ij} w_j(x_i)^2}. \quad (6.6)$$

Wie man sieht, fallen die Fehler mit kleiner werdendem Δv in der Maximumsnorm und die absoluten Fehlerwerte sind auch in einem akzeptablen Bereich. Auf dem rechten Bild erkennt man, dass der L^2 -Fehler für alle drei Verfahren proportional zu Δv ist.

Im linken Plot von Abb. 6 testen wir den Fehler, der durch die Diffusion entsteht. Wir erwarten, dass die Werte von $w_j(0)$ und $w_j(L)$ für alle $j \in J$ übereinstimmen, da in der analytischen Lösung die Randbedingung von einem Rand auf den andern transportiert werden und aufgrund des Potentialminimums in der Mitte auf gleicher Höhe wieder auftreffen. Als den Diffusionsfehler bezeichnen wir $\max_j |w_j(0) - w_j(L)|$. Das zentrale Verfahren ist nach den Beobachtungen in Kapitel 3.5 diffusionfrei und hier wird das bestätigt, wie man an der Größenordnung des Diffusionsfehlers erkennt. Für die Upwind Schemata fällt der Diffusionsfehler proportional zu Δv , wie auch die modifizierte Gleichung (3.49) motiviert.

Im rechten Bild der Abb. 6 wird überprüft, ob der Strom konstant bleibt, indem wir ihn mit seinem Mittelwert vergleichen. Durch die Wahl des Gitters in Kapitel 5 ist die Größenordnung des Fehlers gering genug, um diese Aussage zu bestätigen.

³3-D Plots zu diesem Bereich im Anhang um die Form zu behalten.

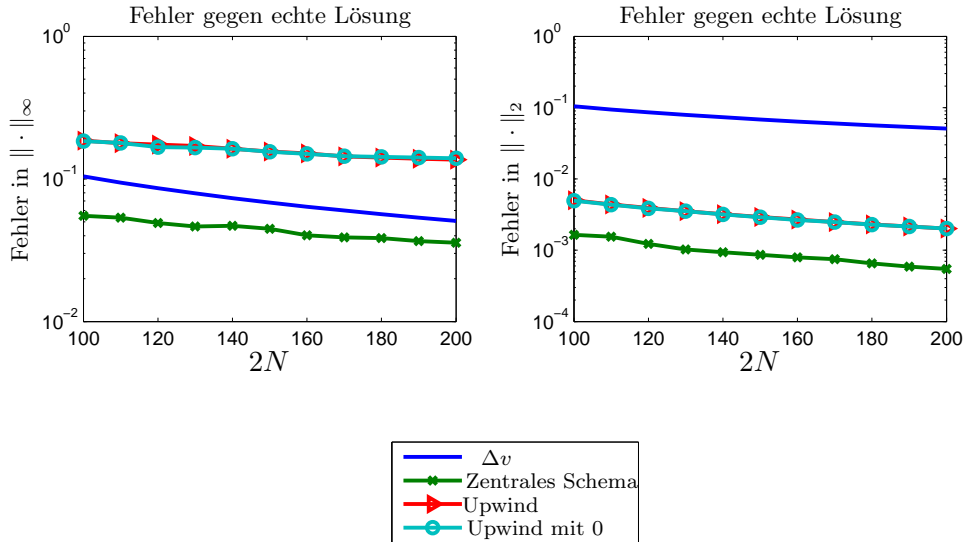


Abbildung 5: Fehlergrößen gegen exakte Lösung, Beispiel A.

Wir fassen zusammen, das Upwind Verfahren und Upwind mit Null liefern die glatteren Lösungen und besitzen adäquates Fehlerverhalten für $\Delta v \rightarrow 0$. Die Lösung des zentralen Verfahrens wirkt auf den ersten Blick weniger gut durch die Oszillationen, die Größenordnung der Fehler ist jedoch etwas besser als bei den anderen Implementierungen und der Fehler fällt genauso für feiner werdendes Gitter. Jedoch muss stets im Hinterkopf behalten werden, dass das Gitter wegen des Sicherheitsdeltas um Null ein großer Unsicherheitsfaktor für dieses Verfahren ist.

6.2 Beispiel B

Wir betrachten nun ein weniger braves Problem, bei dem sich die Problemstellen der Verfahren deutlicher zeigen. Wir werden hier das Hauptaugenmerk, neben dem Vergleich mit der exakten Lösung, auf den unkontrollierten Bereich legen. Wir sind im vorherigen Beispiel auf diesen Bereich kaum eingegangen, da die Information relativ weit davon entfernt war und sich eine gute Null Lösung dort eingestellt hat. Hier betrachten wir die Randbedingungen so, dass der Träger der Lösung, den unkontrollierten Bereich einschließt.

Es sei also gegeben für $x \in [0, L]$, $L = 2$, $v \in \mathbb{R}$,

$$\text{supp} f^+ = [0.7, 1.5] =: [a, b], \quad \text{supp} f^- = \emptyset, \quad (6.7)$$

$$f^+(v) = \sin\left(\frac{\pi}{b-a}(v-a)\right), \quad f^-(v) = 0, \quad (6.8)$$

$$V(x) = \left(x - \frac{L}{4}\right)^2. \quad (6.9)$$

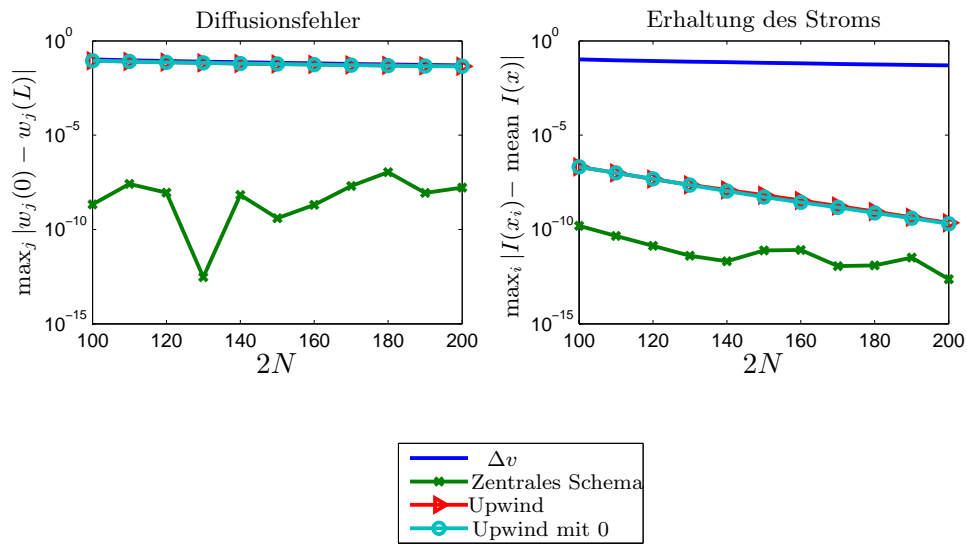


Abbildung 6: Fehlergrößen der Erhaltung, Beispiel A.

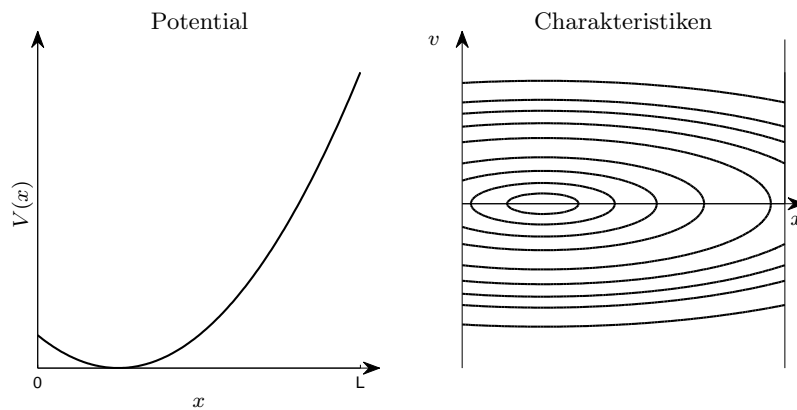


Abbildung 7: Das in Beispiel B gegebene Potential links und die dadurch bestimmten Charakteristiken rechts.

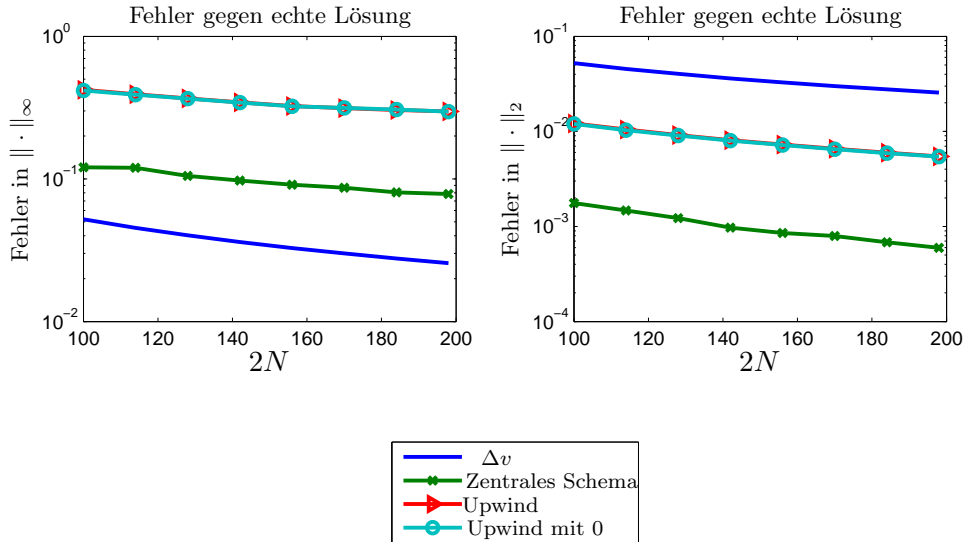


Abbildung 8: Fehler im gesamten Gebiet, Beispiel B.

6.2.1 Lösungen

Die exakte Lösung mit der Nullfunktion im unkontrollierten Bereich soll, wegen Bemerkung 6.1, wieder als Referenz gelten. Wir betrachten daher wieder die Charakteristiken nach Lemma 2.1 und erhalten

$$\left(x - \frac{L}{4}\right)^2 + \frac{v^2}{2} = \text{const} \in \mathbb{R}^+, \quad (6.10)$$

vergleiche Abb. 7. Wir bestimmen die numerischen Lösungen für maximal 100 x -Punkte und $2N = 200$ v -Punkte.

In Abb. 15 sehen wir die Lösungen mittels zentralem und Upwind Verfahren. Man sieht wie die Randbedingung entlang der Charakteristiken transportiert wird und wie diese, auf Grund der Lage des Trägers von f^+ den rechten Rand des Gebiets nicht erreicht, sondern vollständig umkehrt. Hauptaugenmerk liegt für uns jetzt auf dem unkontrollierten Bereich im Inneren. In Abb. 16 erkennen wir auf der linken Grafik, dass das zentrale Schema, durch die, im Vergleich zu Beispiel A in Kapitel 6.1, wenigen v -Punkte innen und zusätzlichen Störungen von der negativen Richtung her, deutlich mehr Oszillationen aufweist. Auch das Sicherheitsdelta, welches die Werte einfach fortsetzt, mindert diese Störungen nicht. In dem rechten Plot erkennen wir die Diffusion des Upwind Schemas. Diese ist auch um einiges stärker als im ersten Beispiel, verhält sich aber wieder proportional zu Δv für feiner werdendes Gitter und wird deshalb nur der Vollständigkeit halber gezeigt. Auch für das Upwind Verfahren bekommt der unkontrollierte Bereich hier mehr Beachtung.

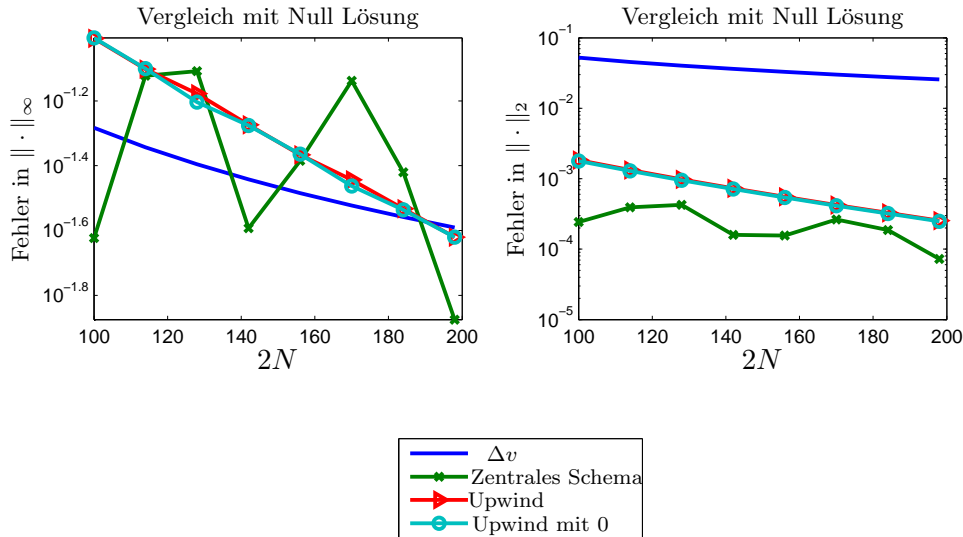


Abbildung 9: Fehler im unkontrollierten Bereich, Beispiel B.

6.2.2 Fehlergrößen

Hier wollen wir wieder das Fehlerverhalten für $\Delta v \rightarrow 0$ beobachten. Wir werden in den vorher definierten Normen messen und mit der exakten Lösung mit Null im unkontrollierten Bereich vergleichen. Abb. 8 zeigt das Fehlerverhalten auf dem gesamten Gebiet. Das Verhalten der Fehler ähnelt dem aus Beispiel A stark. Die maximalen Fehler werden leicht geringer, wie das linke Bild zeigt, und die L^2 -Fehler Verhalten sich wieder proportional zu Δv .

Betrachten wir nun speziell den unkontrollierten Bereich in Abb. 9. Die auftretenden Konvergenzordnungen gegen die Nulllösung sind für beide Fehlernormen besser als auf dem gesamten Gebiet. Besonders auffällig ist das Verhalten des zentralen Verfahrens in der Maximumsnorm, man sieht hier gut, dass verschiedene Gittergrößen die Oszillationen verstärken können. Die allgemeine Fehlergröße bleibt aber in akzeptabler Höhe und auch in L^2 -Norm sieht man das passende Verhalten. Das Upwind Verfahren zeigt in beiden Normen, aufgrund des Diffusionsterms, ein gutes Konvergenzverhalten.

6.3 Beispiel C

In den beiden abschließenden Kapiteln werden wir den Einfluss der Randbedingungen auf den unkontrollierten Bereich, den wir in Bemerkung 6.1 angesprochen haben, genauer untersuchen. In den vorherigen Beispielen war der Träger der Randbedingungen immer so vorgegeben, dass er $v = 0$ nicht einschloss. Es hat sich Innen dann eine Null Lösung eingestellt.

Es seien nun also die Randbedingungen f^+ und f^- mit $\text{supp } f^+ = [0, 4] =: [a, b]$, $\text{supp } f^- = [-3, 0] =: [c, d]$ und

$$f^+(v) = 2 \cos\left(\frac{\pi}{2b}v\right), \quad f^-(v) = \cos\left(\frac{\pi(v+d-2c)}{2(d-c)}\right) \quad (6.11)$$

gegeben. Für $v = 0$ erkennt man $f^+(0) = 2$ und $f^-(0) = -1$ und $f^+(b) = f^-(c) = 0$. Das Potential und das Gebiet sei wie in Beispiel A gegeben, siehe Abb. 4. Wir implementieren das Randwertproblem für das zentrale und das Upwind Verfahren mit den selben Konfigurationen wie in Beispiel A. Das Upwind Verfahren mit Null wird in dieser Diskussion nicht mehr behandelt, da die Unterschiede zum Upwind Verfahren zu gering sind. Wir wollen betrachten, ob sich das in Bemerkung 6.1 erwartete Verhalten im unkontrollierten Bereich einstellt. In Abb. 17 sehen wir die numerischen Ergebnisse der Verfahren. Im kontrollierten Bereich wird die Information wieder entlang der Charakteristiken transportiert. Im Inneren erkennt man jedoch, dass diesmal nicht die Nulllösung approximiert wird, sondern dass sich ein eigenes Plateau in der Höhe von circa 0.5 einstellt. Dieser Wert entsteht genau durch den Einfluss der letzten Information tragenden Charakteristiken, welche von $(x, v) = (0, 0)$ und $(x, v) = (L, 0)$ weglaufen. Genauer gilt

$$\frac{f^+(0) + f^-(0)}{2} = \frac{2 - 1}{2} = 0.5. \quad (6.12)$$

Wir wollen überprüfen, ob dieses Verhalten auch für feiner werdendes Gitter zu beobachten ist und lassen daher Δv , wie in den vorherigen Beispielen, kleiner werden. Abb. 10 zeigt den Vergleich im unkontrollierten Bereich mit der Lösung eines konstanten Plateaus. Der maximale Fehler bleibt etwa konstant aber in einer geringen Größenordnung und der L^2 -Fehler fällt mit $O(\Delta v)$ für beide Verfahren. Der Fehler auf dem gesamten Gebiet wird hier nicht mehr betrachtet.

6.4 Beispiel D

Wir haben verifiziert, dass das Verhalten im unkontrollierten Bereich von den umschließenden Randbedingungen vorgegeben wird. Wir testen diese Erkenntnis noch für den Fall, dass die Charakteristiken den Bereich umschließen, wie in Beispiel B. Es sei also das Potential aus Beispiel B gegeben, vergleiche Abb. 7, und die Randbedingungen

$$f^+(v) = 2 \cos\left(\frac{\pi}{2b}v\right), \quad f^-(v) = 0, \quad (6.13)$$

mit $\text{supp } f^+ = [0, 1.5] =: [a, b]$ und $\text{supp } f^- = \emptyset$.

Die numerischen Lösungen sind in Abb. 18 zu sehen. Man erkennt, dass der unkontrollierte Bereich für beide Verfahren die Werte der letzten umschließenden Charakteristik annimmt. Das heißt $f^+(0)$ ist der ausschlaggebende Faktor für die Höhe des Plateaus. Wir testen diese Annahme für feiner werdendes v -Gitter. Die Ergebnisse in Abb. 11 zeigen wieder das gleiche Verhalten wie in Beispiel C. Der Maximumfehler auf dem rechten Bild bleibt etwa konstant, bzw. fällt leicht und der L^2 -Fehler ist proportional zu Δv .

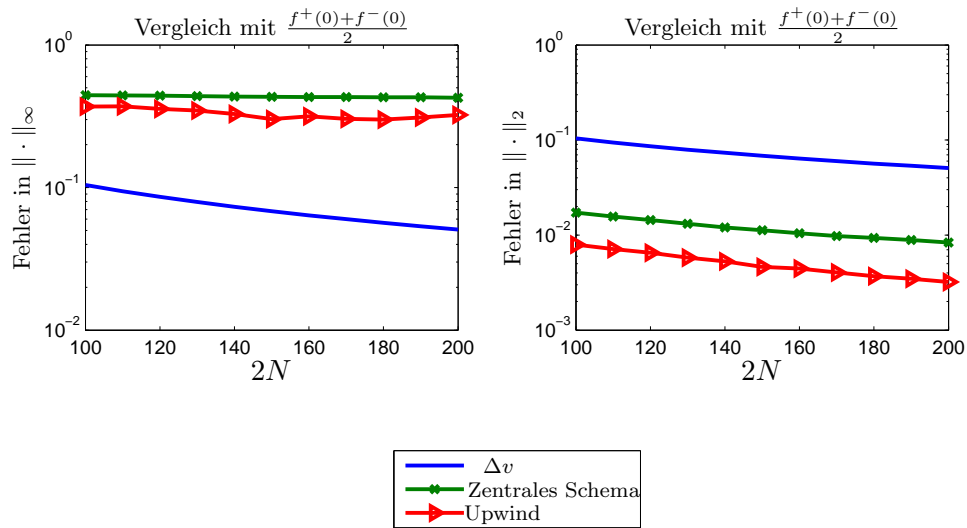


Abbildung 10: Fehler im unkontrollierten Bereich, Beispiel C.

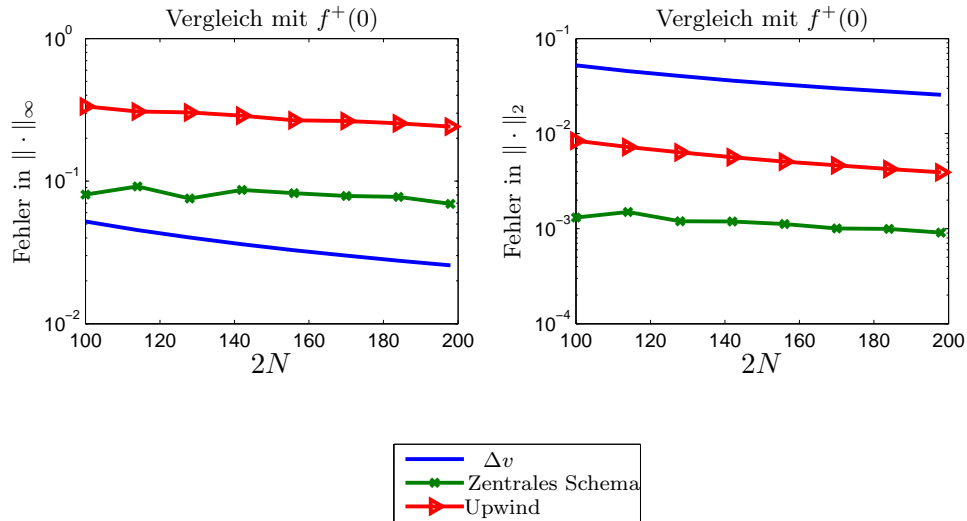


Abbildung 11: Fehler im unkontrollierten Bereich, Beispiel D.

Literatur

- [1] A. Arnold: Mathematical Properties of Quantum Evolution Equations
- [2] A. Arnold, J.A. Carrillo, M.D. Tidriri: Large-time Behavior of Discrete Kinetic Equations with Non-symmetric Interactions, 2000
- [3] A. Arnold, H. Lange, P.F. Zweifel: A discrete-velocity, stationary Wigner equation, 1991
- [4] V.I. Arnol'd: Mathematische Methoden Der Klassischen Mechanik, 1988
- [5] K.E. Brenan, S.L. Campbell, L.R. Petzold: Numerical Solutions of Initial Value Problems in Differential Algebraic Equations, (1989)
- [6] K. Debrabant: Numerische Behandlung linearer und semilinearer partieller differentiell-algebraischer Systeme mit Runge-Kutta-Methoden, 2011
- [7] G. Engeln-Müllges, K. Niederdrenk, R. Wodicka: Numerik-Algorithmen: Verfahren, Beispiele, Anwendungen, 2005
- [8] L.C. Evans: Partial Differential Equations, 2010
- [9] J.W. Gibbs: Elementary Principles in Statistical Mechanics, 1902
- [10] E. Griepentrog, R. März: Differential Algebraic Equations and Their Numerical Treatment, 1986
- [11] P.A. Markowich, C.A. Ringhofer, C. Schmeiser: Semiconductor equations, 1990
- [12] R.J. LeVeque: Numerical Methods for Conservation Laws, 1992
- [13] M. Precht, K. Voit, R. Kraft: Funktionen - Folgen und Reihen - Differential- und Integralrechnung - Differentialgleichungen - Ordnung und Chaos, Band 2, 2005
- [14] L.F. Shampine, J. Kierzenka, M.W. Reichelt: Solving Boundary Value Problems for Ordinary Differential Equations in MATLAB with `bvp4c`, 2000

Abbildungsverzeichnis

1	Einströmrandbedingungen. Quelle: [1].	4
2	Das gegebene Potential $V(x)$ auf dem linken Bild erzeugt die Charakteristiken im rechten Bild. Die Einströmrandbedingungen werden entlang dieser transportiert und wie man sieht, können die geschlossenen Kurven um das Minimum nicht durch die Randbedingungen kontrolliert werden. Quelle: [1].	6
3	Die Grafik zeigt, wie man das Gitter wählen kann, um $w_{N+1} = 0$ garantieren zu können.	9
4	Das in Beispiel A gegebene Potential links und die dadurch bestimmten Charakteristiken rechts.	24
5	Fehlergrößen gegen exakte Lösung, Beispiel A.	26
6	Fehlergrößen der Erhaltung, Beispiel A.	27
7	Das in Beispiel B gegebene Potential links und die dadurch bestimmten Charakteristiken rechts.	27
8	Fehler im gesamten Gebiet, Beispiel B.	28
9	Fehler im unkontrollierten Bereich, Beispiel B.	29
10	Fehler im unkontrollierten Bereich, Beispiel C.	31
11	Fehler im unkontrollierten Bereich, Beispiel D.	31
12	Die numerische Lösung mittels dem zentralen Verfahren. Das linke Bild zeigt den Transport der Information in der (x, v) Ebene, während auf dem vergrößerten Ausschnitt rechts die Oszillationen gut zu sehen sind, Beispiel A.	34
13	Die numerischen Lösungen des Upwind Verfahrens (mit Null) in der (x, v) Ebene, Beispiel A.	34
14	Die Oszillationen der Verfahren werden durch die Höhenschichtlinien der Lösung in der (x, v) -Ebene sichtbar gemacht, Beispiel A.	35
15	Die numerischen Lösungen in der (x, v) -Ebene des zentralen und des Upwind Verfahrens, Beispiel B.	35
16	Die Problemstellen der Verfahren werden in diesen Grafiken hervorgehoben. Auf der linken Grafik liegt der Fokus auf dem unkontrollierten Bereich, indem das zentrale Verfahren Oszillationen zeigt. Das rechte Bild zeigt das Profil der Lösung in der $(x, w(x))$ -Ebene. Da Start- und Endpunkt der Charakteristik diesmal am selben Rand des Gebietes liegen, erkennt man die Glättung durch die Diffusion gut, Beispiel B.	36
17	Die numerischen Lösungen in der (x, v) -Ebene. Die Werte im unkontrollierten Bereich bilden ein eigenes Plateau in der Höhe von $\frac{f^+(0)+f^-(0)}{2} = 0.5$, Beispiel C.	36
18	Die Verfahren liefern hier Lösungen, die im unkontrollierten Bereich, die Werte der umschließenden Charakteristik, also $f^+(0) = 2$, annehmen, Beispiel D.	37

A 3-D Plots

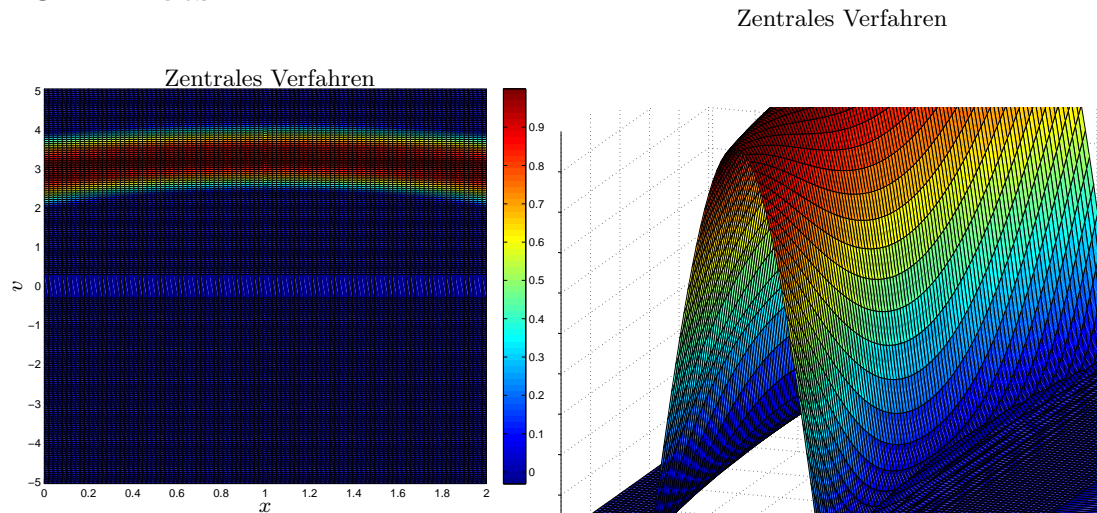


Abbildung 12: Die numerische Lösung mittels dem zentralen Verfahren. Das linke Bild zeigt den Transport der Information in der (x, v) Ebene, während auf dem vergrößerten Ausschnitt rechts die Oszillationen gut zu sehen sind, Beispiel A.

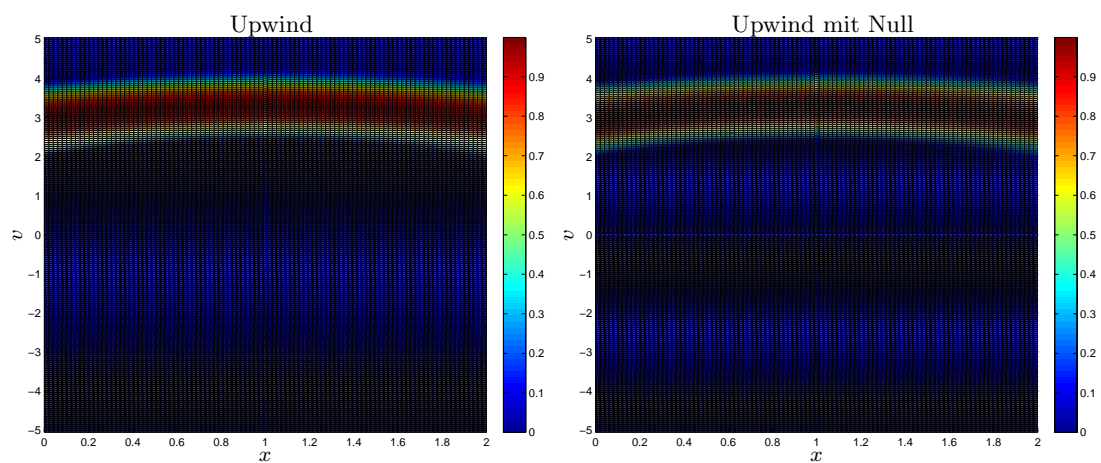


Abbildung 13: Die numerischen Lösungen des Upwind Verfahrens (mit Null) in der (x, v) Ebene, Beispiel A.

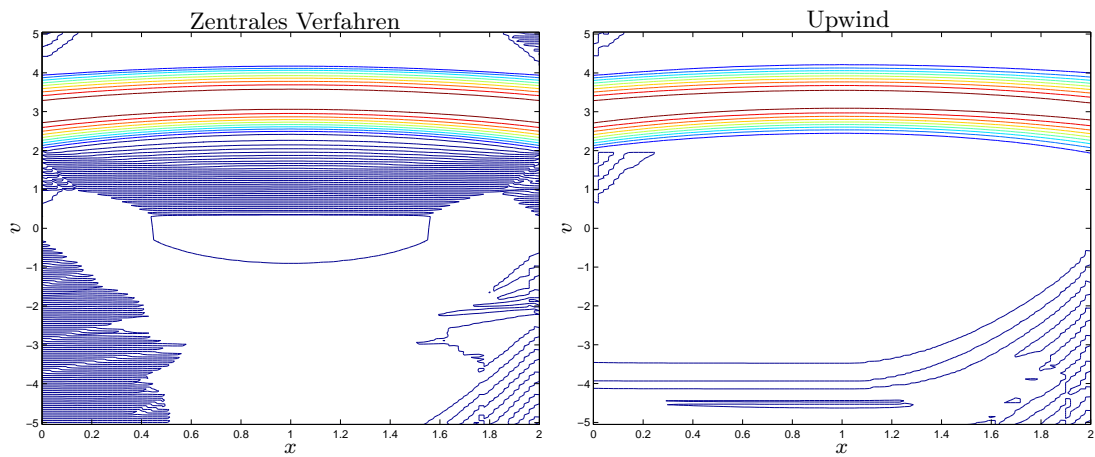


Abbildung 14: Die Oszillationen der Verfahren werden durch die Hörschichtlinien der Lösung in der (x, v) -Ebene sichtbar gemacht, Beispiel A.

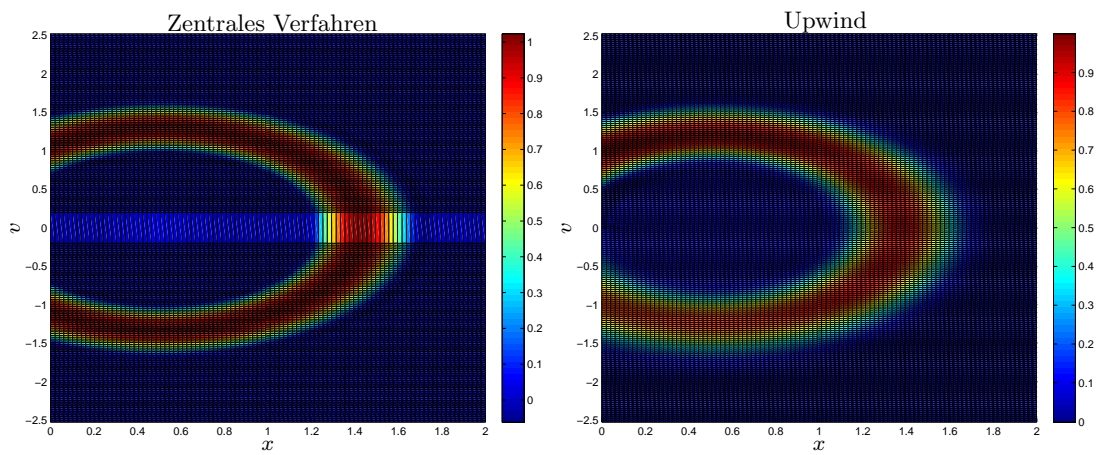


Abbildung 15: Die numerische Lösungen in der (x, v) -Ebene des zentralen und des Upwind Verfahrens, Beispiel B.

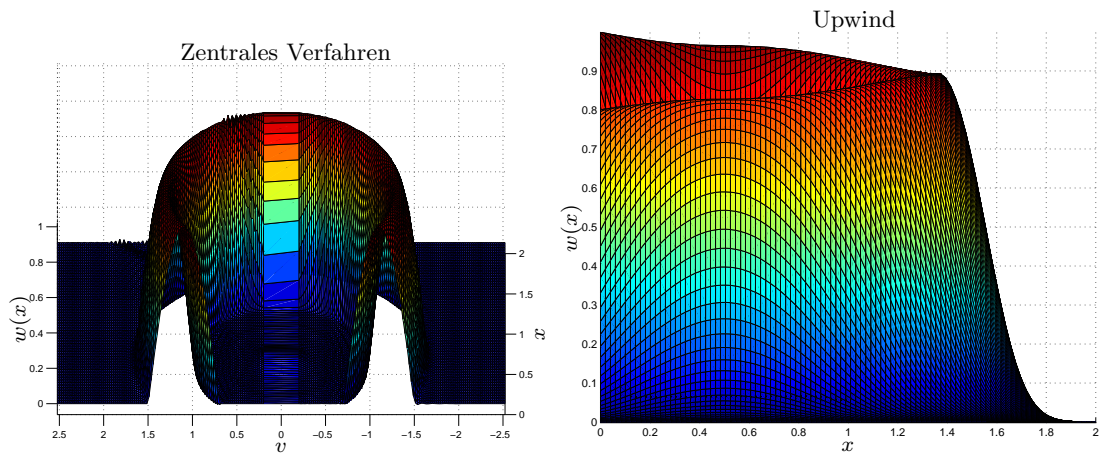


Abbildung 16: Die Problemstellen der Verfahren werden in diesen Grafiken hervorgehoben. Auf der linken Grafik liegt der Fokus auf dem unkontrollierten Bereich, indem das zentrale Verfahren Oszillationen zeigt. Das rechte Bild zeigt das Profil der Lösung in der $(x, w(x))$ -Ebene. Da Start- und Endpunkt der Charakteristik diesmal am selben Rand des Gebietes liegen, erkennt man die Glättung durch die Diffusion gut, Beispiel B.

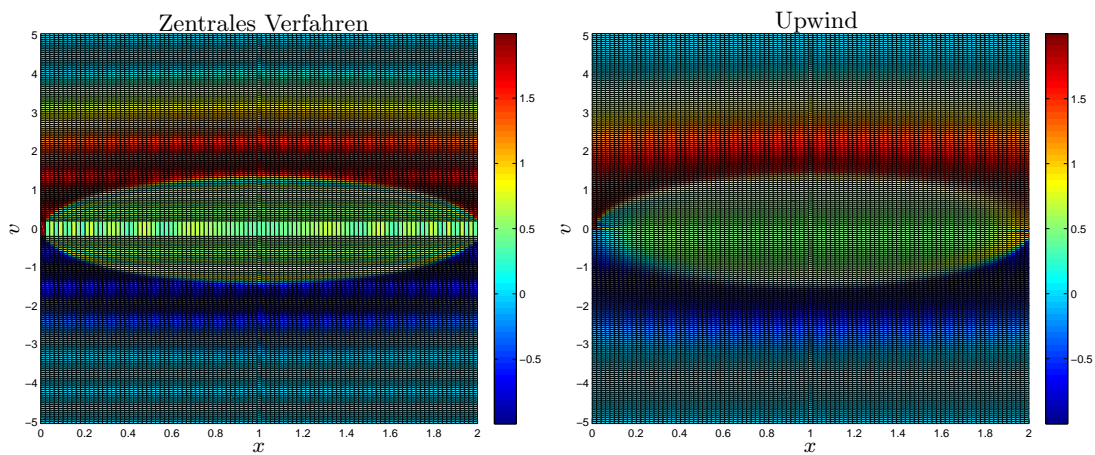


Abbildung 17: Die numerische Lösungen in der (x, v) -Ebene. Die Werte im unkontrollierten Bereich bilden ein eigenes Plateau in der Höhe von $\frac{f^+(0)+f^-(0)}{2} = 0.5$, Beispiel C.

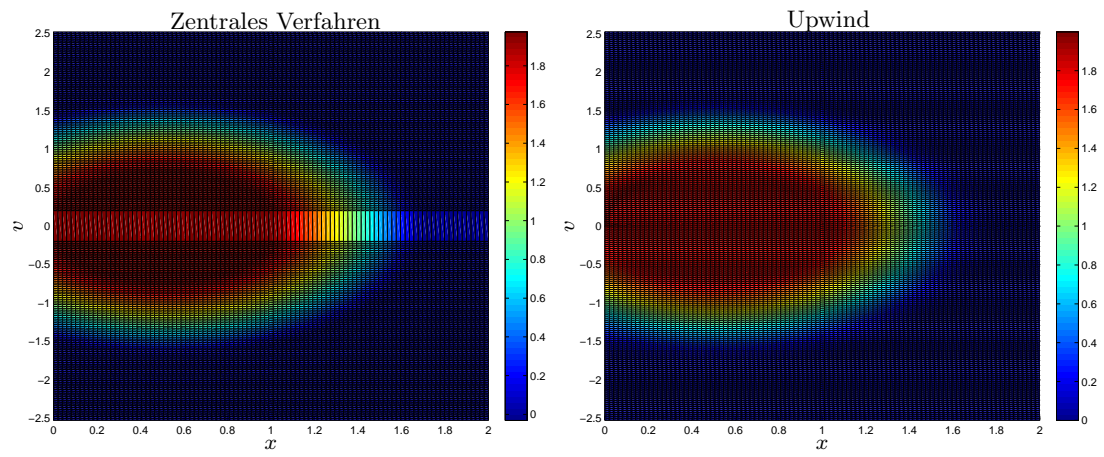


Abbildung 18: Die Verfahren liefern hier Lösungen, die im unkontrollierten Bereich, die Werte der umschließenden Charakteristik, also $f^+(0) = 2$, annehmen, Beispiel D.

B MATLAB Codes

Es werden nur die Programm Codes für die numerischen Verfahren, angewandt auf das Beispiel A, angehängt. Die Algorithmen für die anderen Beispiele funktionieren analog, nur mit verschiedenen Randbedingungen.

```
1 function[ N,delta_v, err_max, err_f, err_diff, err_I] = pl_zentral(N)
2 % Zentrale Semidiskretisierung und RWP Berechnung der stat. Liouville Gl.
3 %
4 % Semidiskretisiert die Liouville DGL
5 %     v*f_v - V'(x)*f_x = 0
6 %     Problem 1 Randbedingungen
7 % zuerst in v mittels Zentralem Schema
8 % und löst dann mittels Randwertsolver auf einem x-Gitter die semidiskrete
9 % Gleichung
10 %     T*w(x) - V'(x)*D*w(x) = 0.
11 % 0 als diskrete Geschwindigkeit wird ausgelassen
12 %
13 % Input Parameter:  N ... Hälfte der Anzahl der Semidiskretisierungspunkte in
14 % v
15 % Output Parameter: N
16 %     delta_v ... Schrittweite des v Gitters
17 %     Fehlergrößen
18
19 plot = 0; % = 1 um Plots zu erzeugen
20
21
22
23 % -----
24 % Gegebene Daten
25
26 % x in [0,L]
27 L = 2;
28
29 % Potential
30 V = @(x) (x-L/2).^2;
31 % Ableitung davon
32 Vx = @(x) 2*(x-L/2);
33
34 % Randbedingungen
35 % Für positive v
36 % Träger
37     supp_min_p = 2;
38     supp_max_p = 4;
39     function AB = f_0_p(v)
40
41         if (supp_min_p <= v) && (v <= supp_max_p)
42             AB = sin(pi/(supp_max_p - supp_min_p)*(v - supp_min_p));
43         else
44             AB = 0;
45         end
46     end
47 % Für negatives v, 0 Randbedingungen
48 % Träger
49     supp_min_n = 0;
50     supp_max_n = 0;
51     function AB = f_L_n(v)
52         AB = 0;
53     end
```

```

54
55
56
57
58
59
60
61
62
63 % -----
64
65 % Semidiskretisierung in v
66 % Semidiskrete Werte initialisieren
67
68
69 % Intervall für v
70 % muss mindesten den Träger plus Characteristik aus dem Träger umfassen
71 % Mindestgröße aus Anfangsdaten berechenbar
72 % v_max berechnen
73 % ist mindestens so groß dass Charakteristiken die von
74 % x = 0 bzw. L und v = 0 weglaufen in
75 % beiden Richtungen den anderen Rand erreichen
76 v_p = sqrt( (L/2)^2 + supp_max_p^2/2)*sqrt(2);
77 v_n = sqrt( (L/2)^2 + supp_max_n^2/2)*sqrt(2);
78 v_max = max( v_p, v_n);
79 % etwas Spielraum lassen
80 v_max = v_max + v_max/6;
81
82
83 % darf nicht zu klein sein (> 0.3) sonst treten Singularitäten beim Solver
84 % auf
85 v_min = 0.3;
86
87
88 % Schrittweite berechnen
89 % Für v_min = const
90 % geht mit N Schritten auf v_max +2*delta_v
91 delta_v = (v_max - v_min)/(N-3);
92
93 v(N) = v_min;
94 v(N+1) = - v_min;
95
96 for i = 1:2*N
97     if i < N
98         v(N-i) = v(N+1-i) + delta_v;
99     elseif i > N+1
100         v(i) = v(i-1) - delta_v;
101     end
102 end
103 % Matrix mit diskreten Geschwindigkeiten
104 T = zeros(2*N);
105 for i = 1:2*N
106     T(i,i) = v(i);
107 end
108
109
110 % Semidiskrete Randbedingungen
111 % an x=0
112 w_0_p = zeros(N,1);
113 w_L_n = zeros(N,1);

```

```

114 for i = 1:2*N
115     if i <= N
116         w_0_p(i) = f_0_p( v(i));
117     else
118         w_L_n(i - N) = f_L_n( v(i));
119     end
120 end
121
122
123
124 % Finite Differenzenmatrix
125 % Zentrale finite Differenzen
126 B = zeros(2*N,2*N);
127 B(1,2) = -1;
128 B(2*N-1,2*N) = 1;
129 for i = 2:2*N-1
130     B(i,i-1) = 1;
131     B(i,i+1) = -1;
132 end
133
134
135
136 % -----
137 % echte Lösung berechnen
138
139 % x-Gitter
140 x = linspace(0,L,100);
141
142 z = zeros(2*N,100);
143
144 % Echte Lösung auf dem Gitter auswerten
145 for i = 1:2*N
146     for j = 1:100
147
148         z(i,j) = realsol( x(j), v(i));
149
150     end
151 end
152
153
154 % -----
155 % BVP Lösen
156
157
158 % Anfangsschätzungen solinit( Gitter, Gestalt der Lsg)
159 x = linspace(0,L,100);
160 delta_x = L/100;
161
162 solinit = bvpinit( x, @bvpguess);
163
164 % Maximale Schrittzahl des Solvers setzen
165 % default = 1000/d, mit d = Anzahl der DGLs
166 options = bvpset( 'NMax', 100);
167
168
169 % Lösen
170 sol_z = bvp4c( @odefun_z, @bcfun, solinit,options);
171
172 % Werte der Lsg auf Gitter x ausgewertet
173 w_z = deval(sol_z,x);

```



```

174
175
176
177
178
179 % -----
180 % Fehlergrößen berechnen
181 %
182
183 % -----
184 % Gesamter Fehler
185 % % Maximumsnorm
186 err_max = max( max( abs( z - w_z)));
187 % L^2 alias Frobeniusm Norm im Diskreten
188 err_f = delta_x*delta_v* norm( z - w_z, 'fro');
189
190
191 % -----
192 % Fehler durch Diffusion des Upwinds
193 % vergleicht Maximum am linken Rand mit dem am Rechten
194 err_diff = abs( max( abs(w_z( :,1)) - abs( w_z(:,end)) ) ) );
195
196
197 % -----
198 % Testen ob der Strom = sum v_j w_j (x) konstant ist
199 I = zeros(1,100);
200 for j = 1:100
201     I(j) = dot( v, w_z( :,j));
202 end
203
204 err_I = max( abs( I(:) - mean(I)));
205
206
207
208
209 % -----
210 % graphische Ausgabe
211
212 if plot == 1
213
214
215 figure;
216 surf(x,v,w_z);
217 title('Zentrale Semidiskretisierung');
218 xlabel('x');
219 ylabel('v');
220 zlabel('w(x)');
221 axis([0 L v(2*N) v(1) min(min(w_z)) max(max(w_z))]);
222 view(0,90);
223 colorbar;
224
225 print('-r600','-depsc2', 'Grafiken/tmp_pl_zentral_1.eps');
226
227
228 figure;
229 surf(x,v,w_z);
230 title('Zentrale Semidiskretisierung');
231 xlabel('x');
232 ylabel('v');
233 zlabel('w(x)');

```

```

234 axis([0 L/2 v(N) v(1) min(min(w_z)) max(max(w_z))]);
235 view(-70,15);
236 zoom(1.7);
237
238 print('-r600','-depsc2', 'Grafiken/tmp_p1_zentral_2.eps');
239
240
241
242 figure;
243 contour(x,v,w_z);
244 title('Zentrale Semidiskretisierung');
245 xlabel('x');
246 ylabel('v');
247 zlabel('w(x)');
248
249 print('-r600','-depsc2', 'Grafiken/tmp_p1_zentral_3.eps');
250
251
252
253
254
255 figure;
256 surf(x,v,w_z);
257 title('Zentrale Semidiskretisierung');
258 xlabel('x');
259 ylabel('v');
260 zlabel('w(x)');
261 axis([0 L v(N) v(1) min(min(w_z)) max(max(w_z))]);
262 view(0,0);
263
264 print('-r600','-depsc2', 'Grafiken/tmp_p1_zentral_4.eps');
265
266
267 end
268
269
270
271
272
273
274
275 % -----
276 % Funktionen deklarieren zum Lösen der DGL
277
278 % Matrix
279     function Dx = D_z(x)
280         Dx = T^(-1)*Vx(x)/2/delta_v*B;
281     end
282 % DGL w'(x) = D_up(x)*w(x) für Upwind
283     function dwdx = odefun_z(x,w)
284         dwdx = D_z(x)*w;
285     end
286
287 % Randbedingungen
288     function res = bcfun(w_0, w_L)
289         res = [ w_0(1:N) - w_0_p
290               w_L(N+1:2*N) - w_L_n];
291     end
292
293 % Vermutung an Lösung

```

```

294     function winit = bvpguess(x)
295         winit = [w_0_p; zeros(N,1)];
296     end
297
298
299
300 % -----
301 % Funktionen deklarieren zum Berechnen der echten Lösung
302
303 % Bestimmt Charakteristikradius (C,sqrt(2)*C) zu (x,v)
304     function r = char(x,v)
305         r = sqrt( (x-L/2)^2 + v^2/2);
306     end
307
308 % Gibt echten Lösungswert von z(x,v) an
309     function z = realsol(x,v)
310         r = char(x,v);
311         % Gebiet ohne RB
312         if r < L/2
313             z = 0;
314             % Gebiet wo Chars von Links nach Rechts laufenn
315         elseif r > L/2 && v > 0
316             z = f_0_p( sqrt( 2*r^2 - 2*(L/2)^2));
317             % Gebiet wo Chars von Rechts nach Links laufen
318         elseif r > L/2 && v < 0
319             z = f_L_n( -sqrt( 2*r^2 - 2*(L/2)^2));
320         end
321     end
322
323
324
325
326
327
328
329
330
331
332 end

```

```

1 function[ N,delta_v, err_max, err_f, err_diff, err_I] = plupwind(N)
2 % Upwindsemidiskretisierung und RWP Berechnung der stat. Liouville Gl.
3 %
4 % Semidiskretisiert die Liouville DGL
5 %     v*f_v - V'(x)*f_x = 0
6 %     Problem 1 Randbedingungen
7 % zuerst in v mittels Upwind Schema
8 % und löst dann mittels Randwertsolver auf einem x-Gitter die semidiskrete
9 % Gleichung
10 %     T*w(x) - V'(x)*D*w(x) = 0.
11 % 0 als diskrete Geschwindigkeit wird ausgelassen
12 %
13 % Input Parameter: N ... Hälfte der Anzahl der Semidiskretisierungspunkte in
14 % v
15 % Output Parameter: N
16 %     delta_v ... Schrittweite des v Gitters
17 %     Fehlergrößen
18
19 plot = 0; % = 1 um Plots zu erzeugen

```

```

20
21
22
23
24 % -----
25 % Gegebene Daten
26
27 % x in [0,L]
28 L = 2;
29
30 % Potential
31 V = @(x) (x-L/2).^2;
32 % Ableitung des Potentials
33 Vx = @(x) 2*(x-L/2);
34
35
36 % Randbedingungen
37 % Für positive v
38 % Träger
39     supp_min_p = 2;
40     supp_max_p = 4;
41     function AB = f_0_p(v)
42
43         if (supp_min_p <= v) && (v <= supp_max_p)
44             AB = sin(pi/(supp_max_p - supp_min_p)*(v - supp_min_p));
45         else
46             AB = 0;
47         end
48     end
49 % Für negatives v, 0 Randbedingungen
50 % Träger
51     supp_min_n = 0;
52     supp_max_n = 0;
53     function AB = f_L_n(v)
54         AB = 0;
55     end
56
57
58
59
60
61
62
63
64
65 % -----
66
67 % Semidiskretisierung in v
68 % Semidiskrete Werte initialisieren
69
70
71 % Intervall für v
72 % muss mindesten den Träger plus Charakteristik aus dem Träger umfassen
73 % Mindestgröße aus Anfangsdaten berechenbar
74 % v_max berechnen: ist mindestens so groß dass Charakteristiken die von
75 % x = 0 bzw. L und v = 0 weglaufen in
76 % beiden Richtungen den anderen Rand erreichen
77 v_p = sqrt( (L/2)^2 + supp_max_p^2/2)*sqrt(2);
78 v_n = sqrt( (L/2)^2 + supp_max_n^2/2)*sqrt(2);
79 v_max = max( v_p, v_n);

```

```

80 % etwas Spielraum lassen
81 v_max = v_max + v_max/6;
82
83
84 % Schrittweite berechnen
85 % Für v_min = delta_v/2
86 % geht mit N Schritten auf v_max +2*delta_v
87 delta_v = 2*v_max/(2*N-5);
88
89 v(N) = delta_v/2;
90 v(N+1) = - delta_v/2;
91
92 for i = 1:2*N
93     if i < N
94         v(N-i) = v(N+1-i) + delta_v;
95     elseif i > N+1
96         v(i) = v(i-1) - delta_v;
97     end
98 end
99 % Matrix mit diskreten Geschwindigkeiten
100 T = zeros(2*N);
101 for i = 1:2*N
102     T(i,i) = v(i);
103 end
104
105
106 % Semidiskrete Randbedingungen
107 % an x=0
108 w_0_p = zeros(N,1);
109 w_L_n = zeros(N,1);
110 for i = 1:2*N
111     if i <= N
112         w_0_p(i) = f_0_p( v(i));
113     else
114         w_L_n(i - N) = f_L_n( v(i));
115     end
116 end
117
118
119
120 % Finite Differenzenmatrix
121 % Upwind
122 B_1 = zeros(2*N,2*N);
123 B_2 = zeros(2*N,2*N);
124
125 B_1(1,1) = 1;
126 B_1(1,2) = -1;
127 B_1(2*N,2*N) = 1;
128
129 B_2(1,1) = -1;
130 B_2(2*N,2*N-1) = 1;
131 B_2(2*N,2*N) = -1;
132
133 for i = 2:2*N-1
134     B_1(i,i) = 1;
135     B_1(i,i+1) = -1;
136
137     B_2(i,i) = -1;
138     B_2(i,i-1) = 1;
139 end

```

```

140
141 % je nach Vz von V' wird Richtung bestimmt
142 function B = B_up(x)
143     if Vx(x) <= 0
144         B = B_1;
145     else
146         B = B_2;
147     end
148 end
149
150
151 % -----
152 % echte Lösung berechnen
153
154 % x-Gitter
155 x = linspace(0,L,100);
156
157 z = zeros(2*N,100);
158
159 % Echte Lösung auf dem Gitter auswerten
160 for i = 1:2*N
161     for j = 1:100
162
163         z(i,j) = realsol( x(j), v(i));
164
165     end
166 end
167
168
169 % -----
170 % BVP Lösen
171
172
173 % Anfangsschätzungen solinit( Gitter, Gestalt der Lsg)
174 x = linspace(0,L,100);
175 delta_x = L/100;
176 solinit = bvpinit( x, @bvpguess);
177
178 % Maximale Schrittanzahl des Solvers setzen
179 % default = 1000/d, mit d = Anzahl der DGLs
180 options = bvpset('NMax', 100);
181
182
183 % Lösen
184 sol_up = bvp4c( @odefun_up, @bcfun, solinit,options);
185
186 % Werte der Lsg auf Gitter x ausgewertet
187 w_up = deval(sol_up,x);
188
189
190
191
192
193 % -----
194 % Fehlergrößen berechnen
195 %
196
197 % -----
198 % Gesamter Fehler
199 % % Maximumsnorm

```

```

200 err_max = max( max( abs( z - w_up)));
201 % L^2 alias Frobeniusm Norm im Diskreten
202 err_f = delta_x*delta_v* norm( z - w_up, 'fro');
203
204
205 % -----
206 % Fehler durch Diffusion des Upwinds
207 % vergleicht Maximum am linken Rand mit dem am Rechten
208 err_diff = abs( max( abs(w_up( :,1)) - abs( w_up(:,end)) ) ) );
209
210
211 % -----
212 % Testen ob der Strom = sum v_j w_j (x) konstant ist
213 I = zeros(1,100);
214 for j = 1:100
215     I(j) = dot( v, w_up( :,j));
216 end
217
218 err_I = max( abs( I(:) - mean(I)));
219
220
221
222
223 % -----
224 % graphische Ausgabe
225
226 if plot == 1
227
228 figure;
229 surf(x,v,w_up);
230 title('Upwind Differenzenquotienten, 0 keine disk. Geschw.');
```

```

231 xlabel('x');
232 ylabel('v');
233 zlabel('w(x)');
234 axis([0 L v(2*N) v(1) min(min(w_up)) max(max(w_up))]);
235 view(0,90);
236 colorbar;
237
238 print('-r600','-depsc2', 'Grafiken/tmp_p1_upwind_1.eps');
```

```

239
240
241 figure;
242 surf(x,v,w_up);
243 title('Upwind Differenzenquotienten, 0 keine disk. Geschw.');
```

```

244 xlabel('x');
245 ylabel('v');
246 zlabel('w(x)');
247 axis([0 L v(2*N) v(1) min(min(w_up)) max(max(w_up))]);
248 view(0,0);
249
250
251 print('-r600','-depsc2', 'Grafiken/tmp_p1_upwind_2.eps');
```

```

252
253
254
255 figure;
256 contour(x,v,w_up);
257 title('Upwind Differenzenquotienten, 0 keine disk. Geschw.');
```

```

258 xlabel('x');
259 ylabel('v');
```

```

260 xlabel('w(x)');
261
262 print('-r600','-depsc2', 'Grafiken/tmp_p1_upwind-3.eps');
263
264 end
265
266
267
268
269
270
271
272
273
274
275 % -----
276 % Funktionen deklarieren zum Lösen der DGL
277
278 % Matrix
279     function Dx = D_up(x)
280         Dx = T^(-1)*Vx(x)/delta_v*B_up(x);
281     end
282 % DGL w'(x) = D_up(x)*w(x) für Upwind
283     function dwdx = odefun_up(x,w)
284         dwdx = D_up(x)*w;
285     end
286
287 % Randbedingungen
288     function res = bcfun(w_0, w_L)
289         res = [ w_0(1:N) - w_0_p
290                w_L(N+1:2*N) - w_L_n];
291     end
292
293 % Vermutung an Lösung
294     function winit = bvpguess(x)
295         winit = [w_0_p; zeros(N,1)];
296     end
297
298
299
300 % -----
301 % Funktionen deklarieren zum Berechnen der echten Lösung
302
303 % Bestimmt Charakteristikradius (C,sqrt(2)*C) zu (x,v)
304     function r = char(x,v)
305         r = sqrt( (x-L/2)^2 + v^2/2);
306     end
307
308 % Gibt echten Lösungswert von z(x,v) an
309     function z = realsol(x,v)
310         r = char(x,v);
311         % Gebiet ohne RB
312         if r < L/2
313             z = 0;
314         % Gebiet wo Chars von Links nach Rechts laufenn
315         elseif r > L/2 && v > 0
316             z = f_0_p( sqrt( 2*r^2 - 2*(L/2)^2));
317         % Gebiet wo Chars von Rechts nach Links laufen
318         elseif r > L/2 && v < 0
319             z = f_L_n( -sqrt( 2*r^2 - 2*(L/2)^2));

```



```

320     end
321 end
322
323
324
325
326
327
328
329
330
331
332 end

```

```

1 function[ N,delta_v, err_max, err_f, err_diff, err_I] = pl.upwind_0(N)
2 % Upwindsemidiskretisierung und RWP Berechnung der stat. Liouville Gl.
3 %
4 % Semidiskretisiert die Liouville DGL
5 %     v*f_v - V'(x)*f_x = 0
6 %     Problem 1 Randbedingungen
7 % zuerst in v mittels Upwind Schema
8 % und löst dann mittels Randwertsolver auf einem x-Gitter die semidiskrete
9 % Gleichung
10 %     T*w(x) - V'(x)*D*w(x) = 0.
11 % 0 wird implizit als diskrete Geschwindigkeit verwendet.
12 %
13 % Input Parameter: N ... Hälfte der Anzahl der Semidiskretisierungspunkte in
14 % v
15 % Output Parameter: N
16 %     delta_v ... Schrittweite des v Gitters
17 %     Fehlergrößen
18
19 plot = 0; % = 1 um Plots zu erzeugen
20
21
22 % -----
23 % Gegebene Daten
24
25 % x in [0,L]
26 L = 2;
27
28 % Potential
29 V = @(x) (x-L/2).^2;
30 % Ableitung davon
31 Vx = @(x) 2*(x-L/2);
32
33 % Randbedingungen
34 % Für positive v
35 % Träger
36     supp_min_p = 2;
37     supp_max_p = 4;
38     function AB = f_0_p(v)
39
40         if (supp_min_p <= v) && (v <= supp_max_p)
41             AB = sin(pi/(supp_max_p - supp_min_p)*(v - supp_min_p));
42         else
43             AB = 0;
44         end
45     end

```

```

46 % Für negatives v, 0 Randbedingungen
47 % Träger
48     supp_min_n = 0;
49     supp_max_n = 0;
50     function AB = f_L_n(v)
51         AB = 0;
52     end
53
54
55
56
57
58
59
60
61
62 % -----
63
64 % Semidiskretisierung in v
65 % Semidiskrete Werte initialisieren
66
67
68 % Intervall für v
69 % muss mindesten den Träger plus Charakteristik aus dem Träger umfassen
70 % Mindestgröße aus Anfangsdaten berechenbar
71 % v_max berechnen
72 % ist mindestens so groß dass Charakteristiken die von
73 % x = 0 bzw. L und v = 0 weglafen in
74 % beiden Richtungen den anderen Rand erreichen
75 v_p = sqrt( (L/2)^2 + supp_max_p^2/2)*sqrt(2);
76 v_n = sqrt( (L/2)^2 + supp_max_n^2/2)*sqrt(2);
77 v_max = max( v_p, v_n);
78 % etwas Spielraum lassen
79 v_max = v_max + v_max/6;
80
81
82 % Schrittweite berechnen
83 % Für v_min = delta_v
84 % geht mit N Schritten auf v_max +2*delta_v
85 delta_v = v_max/(N-2);
86
87 v(N) = delta_v;
88 v(N+1) = - delta_v;
89
90 for i = 1:2*N
91     if i < N
92         v(N-i) = v(N+1-i) + delta_v;
93     elseif i > N+1
94         v(i) = v(i-1) - delta_v;
95     end
96 end
97 % Matrix mit diskreten Geschwindigkeiten
98 T = zeros(2*N);
99 for i = 1:2*N
100     T(i,i) = v(i);
101 end
102
103
104 % Semidiskrete Randbedingungen
105 % an x=0

```

```

106 w_0_p = zeros(N,1);
107 w_L_n = zeros(N,1);
108 for i = 1:2*N
109     if i <= N
110         w_0_p(i) = f_0_p( v(i));
111     else
112         w_L_n(i - N) = f_L_n( v(i));
113     end
114 end
115
116
117
118 % Finite Differenzenmatrix
119 % Upwind
120 B_1 = zeros(2*N,2*N);
121 B_2 = zeros(2*N,2*N);
122
123 B_1(1,1) = 1;
124 B_1(1,2) = -1;
125 B_1(2*N,2*N) = 1;
126
127 B_2(1,1) = -1;
128 B_2(2*N,2*N-1) = 1;
129 B_2(2*N,2*N) = -1;
130
131 for i = 2:2*N-1
132     B_1(i,i) = 1;
133     B_1(i,i+1) = -1;
134
135     B_2(i,i) = -1;
136     B_2(i,i-1) = 1;
137 end
138
139 % je nach Vz von V' wird Richtung bestimmt
140 function B = B_up(x)
141     if Vx(x) <= 0
142         B = B_1;
143     else
144         B = B_2;
145     end
146 end
147
148
149 % -----
150 % echte Lösung berechnen
151
152 % x-Gitter
153 x = linspace(0,L,100);
154
155 z = zeros(2*N,100);
156
157 % Echte Lösung auf dem Gitter auswerten
158 for i = 1:2*N
159     for j = 1:100
160
161         z(i,j) = realsol( x(j), v(i));
162
163     end
164 end
165

```

```

166
167 % -----
168 % BVP Lösen
169
170
171 % Anfangsschätzungen solinit( Gitter, Gestalt der Lsg)
172 x = linspace(0,L,100);
173 delta_x = L/100;
174
175 solinit = bvpinit( x, @bvpguess);
176
177 % Maximale Schrittzahl des Solvers setzen
178 % default = 1000/d, mit d = Anzahl der DGLs
179 options = bvpset('NMax', 100);
180
181
182 % Lösen
183 sol_up = bvp4c( @odefun_up, @bcfun, solinit,options);
184
185 % Werte der Lsg auf Gitter x ausgewertet
186 w_up = deval(sol_up,x);
187
188
189
190
191
192 % -----
193 % Fehlergrößen berechnen
194 %
195
196 % -----
197 % Gesamter Fehler
198 % % Maximumsnorm
199 err_max = max( max( abs( z - w_up)));
200 % L^2 alias Frobeniusm Norm im Diskreten
201 err_f = delta_x*delta_v* norm( z - w_up, 'fro');
202
203
204 % -----
205 % Fehler durch Diffusion des Upwinds
206 % vergleicht Maximum am linken Rand mit dem am Rechten
207 err_diff = abs( max( abs(w_up( :,1)) - abs( w_up(:,end)) ) ) );
208
209
210 % -----
211 % Testen ob der Strom = sum v_j w_j (x) konstant ist
212 I = zeros(1,100);
213 for j = 1:100
214     I(j) = dot( v, w_up( :,j));
215 end
216
217 err_I = max( abs( I(:) - mean(I)));
218
219
220
221
222
223
224 % -----
225 % graphische Ausgabe

```

```

226
227 if plot == 1
228
229 figure;
230 surf(x,v,w_up);
231 title('Upwind Differenzenquotienten, 0 implizit disk. Geschw.');
```

```

232 xlabel('x');
233 ylabel('v');
234 zlabel('w(x)');
235 axis([0 L v(2*N) v(1) min(min(w_up)) max(max(w_up))]);
236 view(0,90);
237 colorbar;
238 %
239
240 print('-r600','-depsc2', 'Grafiken/tmp_p1_upwind_0_1.eps');
```

```

241
242
243
244 figure;
245 surf(x,v,w_up);
246 title('Upwind Differenzenquotienten, 0 implizit disk. Geschw.');
```

```

247 xlabel('x');
248 ylabel('v');
249 zlabel('w(x)');
250 axis([0 L v(2*N) v(1) min(min(w_up)) max(max(w_up))]);
251 view(0,0);
252
253
254 print('-r600','-depsc2', 'Grafiken/tmp_p1_upwind_0_2.eps');
```

```

255
256
257 end
258
259
260
261
262
263
264
265
266
267
268 % -----
269 % Funktionen deklarieren zum Lösen der DGL
270
271 % Matrix
272 function Dx = D_up(x)
273     Dx = T^(-1)*Vx(x)/delta_v*B_up(x);
274 end
275 % DGL w'(x) = D_up(x)*w(x) für Upwind
276 function dwdx = odefun_up(x,w)
277     dwdx = D_up(x)*w;
278 end
279
280 % Randbedingungen
281 function res = bcfun(w_0, w_L)
282     res = [ w_0(1:N) - w_0_p
283           w_L(N+1:2*N) - w_L_n];
284 end
285
```

```

286 % Vermutung an Lösung
287     function winit = bvpguess(x)
288         winit = [w_0_p; zeros(N,1)];
289     end
290
291
292
293 % -----
294 % Funktionen deklarieren zum Berechnen der echten Lösung
295
296 % Bestimmt Charakteristikradius (C,sqrt(2)*C) zu (x,v)
297     function r = char(x,v)
298         r = sqrt( (x-L/2)^2 + v^2/2);
299     end
300
301 % Gibt echten Lösungswert von z(x,v) an
302     function z = realsol(x,v)
303         r = char(x,v);
304         % Gebiet ohne RB
305         if r < L/2
306             z = 0;
307         % Gebiet wo Chars von Links nach Rechts laufenn
308         elseif r > L/2 && v > 0
309             z = f_0_p( sqrt( 2*r^2 - 2*(L/2)^2));
310         % Gebiet wo Chars von Rechts nach Links laufen
311         elseif r > L/2 && v < 0
312             z = f_L_n( -sqrt( 2*r^2 - 2*(L/2)^2));
313         end
314     end
315
316
317
318
319
320
321
322
323
324
325 end

```
