

UNIVERSITÄT  
BAYREUTH

# Lineare Interpolation von Distanzfunktionen zur effizienteren Berechnung erreichbarer Mengen

Diplomarbeit

von

Eduard Leipi

FAKULTÄT FÜR MATHEMATIK, PHYSIK UND  
INFORMATIK

MATHEMATISCHES INSTITUT

Datum: 28. Februar 2012

Betreuung:  
Prof. Dr. Lars Grüne  
Dr. Robert Baier

# Danksagung

Zum Gelingen dieser Arbeit trugen viele Personen bei, bei denen ich mich an dieser Stelle herzlich bedanken möchte. Prof. Dr. Lars Grüne danke ich für die Möglichkeit, diese Arbeit anfertigen zu können. Dr. Robert Baier danke ich für die Betreuung und für seine ständige Hilfsbereitschaft. Vor allem die ausführlichen Betreuungsgespräche, die mir sehr geholfen haben, sind an dieser Stelle besonders hervorzuheben. Für die Berechnung der Referenzmengen und für wertvolle Tipps bei der Implementierung der Algorithmen danke ich Dipl.-Math. Thomas Jahn. Bedanken möchte ich mich auch bei Prof. Dr. Matthias Gerds, für den Zugang zum Rechenzentrum der Universität der Bundeswehr München. Schließlich möchte ich mich noch ganz besonders bei meiner Familie für die Unterstützung während der gesamten Studienzeit bedanken.

# Inhaltsverzeichnis

Symbolverzeichnis	ii
1 Einleitung	1
2 Theoretische Grundlagen	5
2.1 Kontrollsystem . . . . .	5
2.2 Differentialinklusion . . . . .	8
3 Approximation von Mengen durch Distanzfunktionen	11
3.1 Distanzfunktionen . . . . .	11
3.1.1 Vorzeichenlose Distanzfunktion . . . . .	11
3.1.2 Vorzeichenbehaftete Distanzfunktion . . . . .	14
3.2 Hausdorff-Abstand von Mengen . . . . .	22
3.3 Lineare Interpolation durch baryzentrische Koordinaten . . . . .	25
3.3.1 Fehleranalyse der stückweisen Interpolation . . . . .	27
4 Realisierung des mengenwertigen Euler-Verfahrens	38
4.1 Zeitdiskretisierung . . . . .	38
4.2 Zustandsdiskretisierung . . . . .	39
4.2.1 Aufblähung einer Menge . . . . .	41
4.3 Mengenwertiges Euler-Verfahren . . . . .	47
4.4 Realisierung mit Distanzfunktionen . . . . .	48
4.5 Realisierung mit Distanzfunktionen und linearer Interpolation . .	57
5 Berechnung erreichbarer Mengen	62
5.1 Lineare Differentialinklusion . . . . .	62
5.2 Rayleigh-Problem . . . . .	70
5.3 Bilineare Differentialinklusion . . . . .	76
6 Fazit und Ausblick	81
A Anhang	83
A.1 CD-Inhalt . . . . .	83
A.2 Beschreibung und Aufruf der Programme . . . . .	85
A.3 Implementierung von Algorithmus 2 . . . . .	90
Literaturverzeichnis	96

# Symbolverzeichnis

$A^c$	Komplement der Menge $A \subset \mathbb{R}^n$ , $\{y \in \mathbb{R}^n \mid y \notin A\}$
$\mathring{A}, \partial A$	Inneres bzw. Rand der Menge $A \subset \mathbb{R}^n$
$\bar{A}$	Abschluss der Menge $A \subset \mathbb{R}^n$ , $(\bar{A} = \mathring{A} \cup \partial A)$
$B_r(m)$	abgeschlossene euklidische Kugel im $\mathbb{R}^n$ mit Radius $r$ um den Mittelpunkt $m \in \mathbb{R}^n$
$\mathbb{Z}_\rho^n$	Gitter im $\mathbb{R}^n$ mit Diskretisierungsparameter $\rho^t = (\rho_1, \dots, \rho_n)$
$g_\rho$	Gitterpunkt in $\mathbb{Z}_\rho^n$
$d(y, A)$	vorzeichenlose Distanzfunktion des Punktes $y \in \mathbb{R}^n$ zur Menge $A \subset \mathbb{R}^n$
$b(y, A)$	vorzeichenbehaftete Distanzfunktion des Punktes $y \in \mathbb{R}^n$ zur Menge $A \subset \mathbb{R}^n$
$d(A, B)$	einseitiger Hausdorff-Abstand der Menge $A \subset \mathbb{R}^n$ zur Menge $B \subset \mathbb{R}^n$
$d_H(A, B)$	Hausdorff-Abstand der Menge $A$ und $B$ , $A, B \subset \mathbb{R}^n$
$X_{i+1}^h$	theoretisch bestimmte Erreichbarkeitsmenge nach dem $(i+1)$ -ten Schritt des mengenwertigen Euler-Verfahrens
$\tilde{X}_{i+1}^h$	Erreichbarkeitsmenge nach dem $(i+1)$ -ten Schritt des mengenwertigen Euler-Verfahrens ohne Aufblähung und Projektion der Menge im $(i+1)$ -ten Schritt
$X_{i+1}^{h,\rho}$	diskrete Erreichbarkeitsmenge nach dem $(i+1)$ -ten Eulerschritt mit Aufblähung und Projektion der Menge auf ein $\mathbb{Z}_\rho$ Gitter

---

$\mathcal{X}(T; t_0, X_0)$	Lösungstunnel einer vorgegebenen Differentialinklusion auf dem Intervall $[t_0, T]$ mit Startmenge $X_0$
$\mathcal{X}_N(T; t_0, X_0)$	Lösungstunnel einer Diskretisierung der Differentialinklusion (DDI) auf dem Intervall $[t_0, T]$ mit Startmenge $X_0$
$\mathcal{X}_{N,\rho}(T; t_0, \eta_0)$	diskreter Lösungstunnel einer vorgegebenen (DDI) auf $\mathbb{Z}_\rho$
$\mathcal{R}(T; t_0, X_0)$	erreichbare Menge einer vorgegebenen Differentialinklusion zur Zeit $T$ , mit Startmenge $X_0$ und Startzeit $t_0$
$\mathcal{R}_N(T; t_0, X_0)$	erreichbare Menge einer vorgegebenen (DDI) zur Zeit $T$
$\mathcal{R}_{N,\rho}(T; t_0, \eta_0)$	diskrete erreichbare Menge einer (DDI) auf $\mathbb{Z}_\rho$ zur Zeit $T$
$\delta^*(l, A)$	Stützfunktion der Menge $A \subset \mathbb{R}^n$ in Richtung $l$
$\mathcal{O}(h^k)$	Landau-Symbol, beschreibt eine Klasse von Funktionen $f: [0, \infty) \rightarrow \mathbb{R}$ deren Betrag oder Norm auf einem Intervall $[0, h_0]$ abgeschätzt werden kann durch $C \cdot h^k$ , $C \geq 0$



# 1 Einleitung

Das Ziel dieser Diplomarbeit ist die effizientere Berechnung erreichbarer Mengen von Kontrollsystemen. Kontrollsysteme können dabei als Differentialgleichungen (ODE-Systeme) der Form

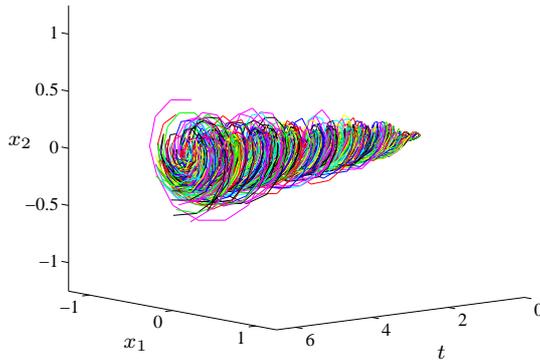
$$\frac{d}{dt}x(t) = f(t, x(t)) \tag{1.1}$$

mit einem zusätzlichen Parameter  $u(t)$ , der als Kontrollfunktion bezeichnet wird, und  $x(t)$  als Zustand des Kontrollsystems interpretiert werden. Die Kontrollfunktion  $u(t)$  kann dabei als Steuerung oder auch als Störung des Systems betrachtet werden, wodurch Kontrollsysteme in vielen Anwendungsgebieten auftreten. Typische Anwendungsbeispiele sind unter anderem Bevölkerungsmodelle, Ausweichmanöver im Straßenverkehr, Wetter- und Klimamodelle oder die Berechnung von Umlaufbahnen im Weltall [BGX11].

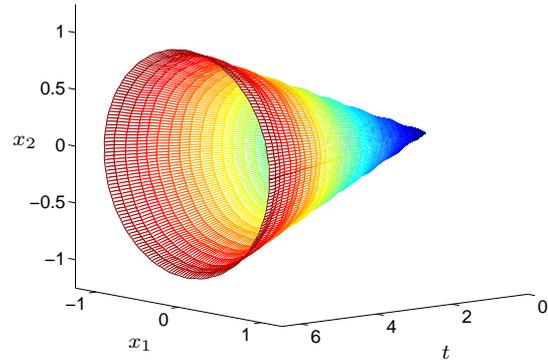
Ein möglicher Zugang zur Berechnung erreichbarer Mengen lässt sich beispielsweise durch numerische Simulationen formulieren, was ein einfacher Ansatz zur Überprüfung von aufwendigen Kontrollsystemen ist [Mal08]. Bei jeder Simulation wird die Kontrollfunktion  $u(t)$  beispielsweise als stückweise konstant angenommen und zu jedem Zeitpunkt nur zufällig aus den Punkten des Steuerbereichs  $U$  ausgewählt [spe08]. Dadurch erhält man, nach Auswertung des Kontrollsystems über numerische Integration, eine bestimmte Lösungstrajektorie für die zufälligen Werte der Kontrollfunktion und damit einen möglichen Zustand des Systems zur gewünschten Endzeit  $t_f = T$ .

Idealerweise würde man diese Simulationen nun für alle möglichen Werte der Kontrollfunktion durchführen wollen, was im Allgemeinen jedoch unendlich viele sind und daher praktisch nicht durchführbar ist. Die Berechnung erreichbarer Mengen folgt hier unmittelbar aus dem Anliegen das Kontrollsystem für alle zulässigen Kontrollfunktionen auszuwerten. Durch Analyse des Zustandsraums werden dabei für jeden einzelnen Zeitschritt alle möglichen Zustände, und damit die erreichbare Menge, des Kontrollsystems berechnet. Diese mengenwertige Berechnung ist zwar aufwendiger als die Simulation einer Lösungstrajektorie, liefert in der Regel jedoch einen besseren Einblick in das System und bessere Ergebnisse als die

vielfache Simulation einzelner Lösungen [Mal08]. Zur Veranschaulichung von numerischen Simulationen und der Berechnung erreichbarer Mengen, dienen folgende Abbildungen:



**Abbildung 1.1:** Simulation von 200 zufälligen Schwingungen



**Abbildung 1.2:** mengenwertige Berechnung der erreichbaren Menge

Abbildung 1.1 zeigt 200 zufällige Lösungen eines speziellen Kontrollproblems (eine angeregte Schwingung). Die Lösungen starten zum Zeitpunkt  $t = 0$  alle im Ursprung und sind zur besseren Unterscheidbarkeit in verschiedenen Farben gekennzeichnet. Selbst mit 20000 zufälligen Simulationen ergibt sich noch kein wesentlich genaueres Bild. Im Gegensatz dazu, sieht man in Abbildung 1.2 den Lösungstunnel der mengenwertigen Berechnung der erreichbaren Menge für dieses Kontrollsystem und erkennt, dass diese ein deutlich genaueres Ergebnis liefert [spe08].

Die in dieser Arbeit vorgestellte Realisierung der Berechnung erreichbarer Mengen führt unter anderem zur Diskretisierung des Zustandsraums. Dabei stellt sich jedoch heraus, dass die mengenwertige Berechnung erreichbarer Mengen umso zeitaufwendiger wird, je feiner der Zustandsraum diskretisiert wird. Betrachtet man für das ODE-System (1.1) das aus der Numerik bekannte Euler-Verfahren mit

$$x_{i+1} = x_i + h \cdot f(t_i, x(t_i)) \quad (i = 0, \dots, N - 1)$$

und der Schrittweite  $h$ , als ein einfaches Runge-Kutta-Verfahren der Ordnung 1, so wird für die Konvergenz des entsprechenden mengenwertigen Verfahrens bereits eine Zustandsraumdiskretisierung von  $\mathcal{O}(h^2)$  benötigt. Für feinere Schrittweiten wird die Berechnung erreichbarer Mengen somit sehr zeitaufwendig. Bei mengenwertigen Verfahren höherer Ordnung nimmt der Zeitaufwand für die Berechnung erreichbarer Mengen noch einmal erheblich zu, da hierfür eine noch feinere Diskretisierung des Zustandsraums verlangt wird. Hinsichtlich des Aufwands wird

---

in dieser Arbeit eine effizientere Methode zur Berechnung erreichbarer Mengen, durch lineare Interpolation von Distanzfunktionen, vorgestellt.

Distanzfunktionen sind ein häufig genutztes Hilfsmittel in der Numerik, beispielsweise für Konvergenz-Beweise von mengenwertigen Runge-Kutta-Verfahren [DF89]. In dieser Arbeit werden Distanzfunktionen unter anderem zur Approximation von Mengen benutzt. So lässt sich eine Menge  $A \subset \mathbb{R}^n$  auch implizit mittels Niveaumengen einer Distanzfunktion  $b$  darstellen:

$$A = \{y \in \mathbb{R}^n \mid b(y, A) \leq 0\}.$$

Für die diskrete Approximation von Mengen ist eine weitere Eigenschaft der Distanzfunktionen von großem Vorteil. So lassen sich die Werte der Distanzfunktionen interpolieren und damit diskrete Mengen genauer approximieren. Ferner lassen sich Mengenoperationen für die Berechnung erreichbarer Mengen mit Hilfe von Distanzfunktionen vereinfachen.

Neben Distanzfunktionen wird zur Berechnung erreichbarer Mengen das bereits erwähnte Euler-Verfahren in der mengenwertigen Version mit Distanzfunktionen eingeführt und verwendet. Die lineare Interpolation von Distanzfunktionen erlaubt es schließlich die Berechnung erreichbarer Mengen, unter Beibehaltung der Konvergenzeigenschaft des Euler-Verfahrens, auf einer gröberen Diskretisierung des Zustandsraums durchzuführen und damit den Zeitaufwand für die Berechnung der Mengen signifikant zu reduzieren.

Der Aufbau der Diplomarbeit ist dabei wie folgt gegliedert. In den ersten beiden Kapiteln werden die benötigten Grundlagen für die Berechnung von Mengen behandelt. Zu Beginn werden dazu in Kapitel 2 die grundlegenden Mengenoperationen definiert, Kontrollsysteme eingeführt und auf Differentialinklusionen verallgemeinert. Wichtige Sätze über die Existenz von Lösungen werden sowohl für Kontrollsysteme (Satz von Carathéodory) als auch für Differentialinklusionen, über die Existenz benachbarter Lösungen, (Filippov Theorem) angegeben.

Kapitel 3 führt zwei verschiedene Distanzfunktionen, die vorzeichenlose und die vorzeichenbehaftete, ein und untersucht die Approximation von Mengen mit Hilfe von Distanzfunktionen. Es zeigt sich, dass die vorzeichenlose Distanzfunktion zwar eine gute „äußere“ Beschreibung von Mengen liefert, für die lineare Interpolation im späteren Verlauf der Arbeit, aufgrund von Nichtdifferenzierbarkeitsstellen am Rand einer Menge, jedoch nicht geeignet ist. Zur Bewertung von berechneten Mengen wird anschließend der Hausdorff-Abstand definiert. Zum Abschluss des Kapitels wird die lineare Interpolation von Distanzfunktionen zur genaueren Approximation diskreter Mengen vorgestellt. Baryzentrische Koordinaten sind dabei

ein wichtiges Hilfsmittel für die räumliche Interpolation und werden daher zunächst definiert und anhand eines Beispiels veranschaulicht. Die anschließende Fehleranalyse der stückweisen Interpolation rechtfertigt letztlich die Verwendung der linearen Interpolation mit baryzentrischen Koordinaten, für eine genauere Approximation diskreter Mengen, im weiteren Verlauf der Arbeit.

Kapitel 4 widmet sich dem mengenwertigen Euler-Verfahren was ein einfaches Einschrittverfahren erster Ordnung ist. Als Erstes werden die für die Realisierung benötigten Diskretisierungen in Zeit und Ort erläutert. Die Diskretisierung des Zustandsraums zeigt dabei ein Problem auf, wodurch das Diskretisieren einer nicht leeren Menge zu einer leeren diskreten Menge führen kann. Angesichts dieser Schwierigkeit werden zwei mögliche Lösungen, die Verschiebung des Diskretisierungsgitters und die Aufblähung der Menge, für eine erfolgreiche Diskretisierung von Mengen vorgeschlagen. Für den weiteren Verlauf der Arbeit wird die Aufblähung einer Menge, als ein einfaches und wirkungsvolles Hilfsmittel gewählt. Die folgende Formulierung und Analyse des mengenwertigen Euler-Verfahrens mit Distanzfunktionen zeigt, wie Zeitaufwendig das klassische Verfahren in der mengenwertigen Version ist. Ein Ansatz zur effizienteren Berechnung, durch Vereinfachung der Diskretisierung auf Randpunkte einer Menge, wird nach kurzer Analyse verworfen, da dieser zu einer heuristischen Problemstellung führt. Die anschließende Formulierung des mengenwertigen Euler-Verfahrens mit linearer Interpolation von Distanzfunktionen führt letztlich zu einer deutlich effizienteren Berechnung erreichbarer Mengen.

In Kapitel 5 wird das vorgestellte Verfahren mit Interpolation der Distanzfunktion an drei Beispielen numerisch ausgewertet und mit der mengenwertigen Formulierung ohne Interpolation der Distanzfunktion verglichen. Die Resultate zeigen sowohl die Konvergenz des Verfahrens, als auch die deutlich höhere Effizienz bezüglich der Berechnungszeit. Sämtliche Berechnungen und Darstellungen von erreichbaren Mengen erfolgten dabei mit dem beiliegenden Programmpaket. Weitere Informationen zum Aufruf und Aufbau der einzelnen Programme findet man im Anhang A.

## 2 Theoretische Grundlagen

In diesem Kapitel werden grundlegende Mengenoperationen und die theoretischen Grundlagen von Kontrollsystemen und den daraus folgenden Differentialinklusionen behandelt. Neben Definitionen für Mengenoperationen, Kontrollsysteme und Differentialinklusionen werden wichtige Sätze, z.B. über die Existenz von Lösungen, angegeben.

Zu Beginn werden die Mengenoperationen, wie die Addition, Skalarmultiplikation und Verschiebung, die elementweise verstanden werden, definiert.

**Definition 2.1.** Seien  $A, B \subset \mathbb{R}^n$ ,  $v \in \mathbb{R}^n$  und  $\lambda \in \mathbb{R}$ . Dann definiert man die *Summe zweier Mengen* als

$$A + B := \{a + b \mid a \in A, b \in B\},$$

die *Multiplikation einer Menge mit einem Skalar* als

$$\lambda \cdot A := \{\lambda a \mid a \in A\}$$

und die *Verschiebung einer Menge* als

$$A + v := \{a + v \mid a \in A\}.$$

Für Rechenregeln verschiedener Mengenoperationen wird hier aus Zeitgründen auf [Bai95, Bemerkung 0.1.1.2] verwiesen.

**Definition 2.2.** Eine Menge  $A \subset \mathbb{R}^n$  heißt *konvex*, wenn die Verbindungsstrecke zweier Punkte von  $A$  ganz in der Menge liegt:

$$\lambda a + (1 - \lambda) \tilde{a} \in A \quad (a, \tilde{a} \in A, \lambda \in [0, 1])$$

### 2.1 Kontrollsystem

Kontrollsysteme entstehen aus Differentialgleichungen der Form  $\frac{d}{dt}x(t) = f(t, x(t))$  mit einem zusätzlichen Parameter  $u(t)$ , der als Steuerung oder auch als Störung

des Systems interpretiert werden kann.

**Definition 2.3.**

- i) Ein *Kontrollsystem* in *kontinuierlicher Zeit*  $t \in \mathbb{R}$  mit *Zustand*  $x \in \mathbb{R}^n$ ,  $n \in \mathbb{N}$ , ist gegeben durch die gewöhnliche Differentialgleichung

$$\frac{d}{dt}x(t) = f(t, x(t), u(t)) \quad (2.1)$$

wobei  $f: \mathbb{R} \times \mathbb{R}^n \times U \rightarrow \mathbb{R}^n$  ein *parameterabhängiges stetiges Vektorfeld* ist.

- ii) Die Menge  $U \subseteq \mathbb{R}^m$  heißt *Kontrollwertebereich* und definiert die Werte die die *Kontrolle* oder der *Kontrollwert*  $u(t)$  für  $t \in \mathbb{R}$  annehmen darf.

- iii) Mit  $\mathcal{U}$  bezeichnet man den *Raum der zulässigen Kontrollfunktionen*

$$\mathcal{U} := \left\{ u: \mathbb{R} \rightarrow U \mid u \text{ zulässig} \right\}.$$

Die Variable  $t \in \mathbb{R}$  wird stets als *Zeit* interpretiert und statt „ $\frac{d}{dt}x(t)$ “ wird kurz „ $\dot{x}(t)$ “ geschrieben. Als Klasse von zulässigen Kontrollfunktionen werden im Folgenden Lebesgue-messbare Funktionen definiert und der *Startwert* oder *Anfangswert* wird mit  $x_0 = x(t_0)$  bezeichnet.

**Definition 2.4.** Sei  $I = [a, b] \subset \mathbb{R}$  ein abgeschlossenes Intervall.

- i) Eine Funktion  $g: I \rightarrow \mathbb{R}^m$  heißt *stückweise konstant*, falls eine Zerlegung von  $I$  in endlich viele Teilintervalle  $I_j$ ,  $j = 1, \dots, n$  existiert, so dass  $g$  auf  $I_j$  konstant ist für alle  $j = 1, \dots, n$ .
- ii) Eine Funktion  $g: I \rightarrow \mathbb{R}^m$  heißt (*Lebesgue-*) *messbar*, falls eine Folge von stückweise konstanten Funktionen  $g_i: I \rightarrow \mathbb{R}^m$ ,  $i \in \mathbb{N}$  existiert mit  $\lim_{i \rightarrow \infty} g_i(x) = g(x)$  für fast alle  $x \in I$ , d.h. für alle  $x$  aus einer Menge  $J \subseteq I$  mit der Eigenschaft, dass  $I \setminus J$  eine Lebesgue-Nullmenge ist.
- iii) Eine Funktion  $g: \mathbb{R} \rightarrow \mathbb{R}^m$  heißt (*Lebesgue-*) *messbar*, falls für jedes abgeschlossene Teilintervall  $I = [a, b] \subset \mathbb{R}$  die Einschränkung  $g|_I$  messbar im Sinne von ii) ist.

Der folgende Satz von Carathéodory liefert mit der Wahl messbarer Kontrollfunktionen einen sinnvollen Lösungs begriff für das Kontrollsystem (2.1).

**Satz 2.5** (Satz von Carathéodory [Grü06, Satz 1.5]). *Betrachte ein Kontrollsystem mit folgenden Eigenschaften:*

i) *Der Raum der Kontrollfunktionen ist gegeben durch*

$$\mathcal{U} = L_\infty(\mathbb{R}, U) := \left\{ u: \mathbb{R} \rightarrow U \mid \begin{array}{l} u \text{ ist messbar und beschränkt außer-} \\ \text{halb einer Lebesgue-Nullmenge} \end{array} \right\}$$

ii) *Das Vektorfeld  $f: \mathbb{R} \times \mathbb{R}^n \times U \rightarrow \mathbb{R}^n$  ist stetig.*

iii) *Für jedes  $R > 0$  existiert eine Konstante  $L_R > 0$ , so dass die Abschätzung*

$$\|f(t, x_1, u) - f(t, x_2, u)\| \leq L_R \|x_1 - x_2\|$$

*für alle  $x_1, x_2 \in \mathbb{R}^n$  und alle  $u \in U$  mit  $\|x_1\|, \|x_2\|, \|u\| \leq R$  erfüllt ist.*

*Dann gibt es für jeden Punkt  $x_0 \in \mathbb{R}^n$  und jede Kontrollfunktion  $u \in \mathcal{U}$  ein (maximales) offenes Intervall  $I$  mit  $0 \in I$  und genau eine absolut stetige Funktion  $x(t)$ , die die Integralgleichung*

$$x(t) = x_0 + \int_0^t f(\tau, x(\tau), u(\tau)) \, d\tau$$

*für alle  $t \in I$  erfüllt.*

*Beweis.* Siehe [Son98, Anhang C]. ■

**Definition 2.6.** Die eindeutige Funktion  $x(t)$  aus Satz 2.5 heißt *Lösung* von (2.1) zum Anfangswert  $x_0 \in \mathbb{R}^n$  und zur *Kontrollfunktion*  $u \in \mathcal{U}$  und wird mit  $\varphi(t, x_0, u)$  bezeichnet.

Die folgende Beobachtung rechtfertigt diese Definition [Grü06, Seite 4]: Da  $\varphi(t, x_0, u)$  absolut stetig ist, ist diese Funktion für fast alle  $t \in I$  nach  $t$  differenzierbar. Insbesondere folgt aus dem Satz 2.5, dass  $\varphi(t, x_0, u)$  die Gleichung (2.1) für fast alle  $t \in I$  erfüllt und somit

$$\dot{\varphi}(t, x_0, u) = f(t, \varphi(t, x_0, u), u(t))$$

für fast alle  $t \in I$  gilt.

**Definition 2.7.** Betrachte ein Kontrollsystem (2.1). Die *Erreichbarkeitsmenge* (*reachable set*) des Kontrollsystems ausgehend von  $x_0$  zur Zeit  $t \geq 0$  ist gegeben durch

$$\mathcal{R}(t, t_0, x_0) := \left\{ \varphi(t, x_0, u) \mid u \in \mathcal{U} \right\}.$$

Die folgende Umformulierung der Kontrollsysteme in Differentialinklusionen kann später bei der Fehleranalyse der diskretisierten Kontrollsysteme von Vorteil sein und verdeutlicht zudem die Mengewertigkeit der Systeme für die nachfolgenden Untersuchungen.

## 2.2 Differentialinklusion

Differentialinklusionen entstehen durch Verallgemeinerung von gewöhnlichen Differentialgleichungen, indem die rechte Seite als Menge und nicht nur als Punkt im  $\mathbb{R}^n$  betrachtet wird.

**Definition 2.8.** Gegeben sei die *Differentialinklusion (DI)*

$$\dot{x}(t) \in F(t, x(t)) \quad \text{für fast alle } t \in [t_0, T], \quad (2.2)$$

$$x(t_0) \in X_0 \quad (2.3)$$

mit einer mengewertigen Abbildung

$$F: [t_0, T] \times \mathbb{R}^n \rightrightarrows \mathbb{R}^n,$$

die ein Element  $(t, x)$  auf eine Teilmenge des  $\mathbb{R}^n$  abbildet und einer Startmenge  $X_0 \subset \mathbb{R}^n$ .

Eine Funktion  $x: [t_0, T] \rightarrow \mathbb{R}^n$  heißt *Lösung der Differentialinklusion*, wenn sie absolut stetig ist, die Anfangsbedingung (2.3) und die Bedingung (2.2) erfüllt.

Unter dem *Lösungstunnel* der Differentialinklusion, ausgehend vom Zeitpunkt  $t_0$  und dem Anfangswert  $x_0$  bzw. der Anfangsmenge  $X_0 \subset \mathbb{R}^n$ , versteht man die Menge aller absolut stetigen Trajektorien, die die Differentialinklusion erfüllen, d.h.

$$\begin{aligned} \mathcal{X}(T; t_0, x_0) &:= \left\{ x: [t_0, T] \rightarrow \mathbb{R}^n \mid x(\cdot) \text{ löst die (DI) (2.2) mit } x(t_0) = x_0 \right\}, \\ \mathcal{X}(T; t_0, X_0) &:= \bigcup_{x_0 \in X_0} \mathcal{X}(T; t_0, x_0). \end{aligned}$$

Die Menge aller Endpunkte von Lösungen der (DI) bezeichnet man mit

$$\begin{aligned}\mathcal{R}(T, t_0, x_0) &:= \left\{ y \in \mathbb{R}^n \mid \exists x(\cdot) \in \mathcal{X}(T; t_0, x_0) \text{ mit } x(T) = y \right\} \\ \mathcal{R}(T, t_0, X_0) &:= \left\{ y \in \mathbb{R}^n \mid \exists x(\cdot) \in \mathcal{X}(T; t_0, X_0) \text{ mit } x(T) = y \right\}\end{aligned}$$

und heißt *erreichbare Menge* der Differentialinklusion ausgehend vom Zeitpunkt  $t_0$  und dem Anfangswert  $x_0$  bzw. der Anfangsmenge  $X_0 \subset \mathbb{R}^n$ .

*Bemerkung.* Für die erreichbare Menge einer (DI) und für die Erreichbarkeitsmenge eines Kontrollsystems wird dasselbe Symbol verwendet, da eine Lösung der (DI) unter einigen Voraussetzungen auch eine Lösung des Kontrollsystems und umgekehrt ist.

Wie man nach obiger Definition bemerkt, entstehen Differentialinklusionen z.B. aus Kontrollsystemen der Form

$$\begin{aligned}\dot{x}(t) &= f(t, x(t), u(t)) && (t \in [t_0, T]) \\ x(t_0) &= x_0 \\ u(t) &\in U(t, x(t)) && (t \in [t_0, T])\end{aligned}$$

mit der Kontrollfunktion  $u(\cdot)$ , die nur Werte aus  $U(t, x(t))$  zur Zeit  $t$  annehmen darf. Die entsprechende Differentialinklusion für dieses Kontrollsystem lautet

$$\begin{aligned}\dot{x}(t) \in F(t, x(t)) &:= \bigcup_{u(t) \in U(t, x(t))} \left\{ f(t, x(t), u(t)) \right\} && (t \in [t_0, T]), \\ x(t_0) &\in X_0 = \{x_0\}.\end{aligned}$$

Ähnlich zum Satz 2.5 (Satz von Carathéodory) für Kontrollsysteme, existiert auch für Differentialinklusionen eine Aussage über die Existenz einer benachbarten Lösung, die außerdem zu einer Fehlerabschätzung von Lösungen der (DI) führt.

**Theorem 2.9** (Filippov Theorem [Cha03, Theorem 2.1.1]). *Sei  $y: [t_0, T] \rightarrow \mathbb{R}^n$  eine absolut stetige Funktion,  $\beta$  eine positive Konstante und*

$$Q := \left\{ (t, x) \in [t_0, T] \times \mathbb{R}^n \mid \|x - y(t)\| \leq \beta \right\}.$$

*Weiterhin sei  $F$ , eine Abbildung die  $Q$  auf eine nichtleere und geschlossene Teilmenge des  $\mathbb{R}^n$  abbildet, stetig und erfülle*

$$d_{\text{H}}(F(t, x), F(t, z)) \leq k(t) \|x - z\|$$

für alle  $(t, x)$  und  $(t, z)$  aus  $Q$  mit  $k(\cdot) \in L_1([t_0, T])$ , wobei  $d_H(\cdot, \cdot)$  den Hausdorff-Abstand aus Kapitel 3.2 bezeichnet.

Zusätzlich wird angenommen, dass

$$\begin{aligned} \|y(t_0) - x_0\| &= \delta \leq \beta, \\ d(\dot{y}(t), F(t, y(t))) &\leq p(t) \quad (\text{für fast alle } t \in [t_0, T]), \end{aligned}$$

für ein  $p(\cdot) \in L_1([t_0, T])$ , so dass

$$\xi(t) := \delta e^{\int_{t_0}^t k(s) ds} + \int_{t_0}^t e^{\int_s^t k(\tau) d\tau} p(s) ds \leq \beta \quad (t \in [t_0, T]),$$

wobei  $d(\cdot, \cdot)$  die Distanzfunktion aus Kapitel 3.1.1 bezeichnet.

Dann existiert eine Lösung  $x(\cdot)$  der (DI), so dass

$$\|x(t) - y(t)\| \leq \xi(t) \quad (t \in [t_0, T]),$$

und

$$\|\dot{x}(t) - \dot{y}(t)\| \leq k(t)\xi(t) + p(t) \quad (\text{für fast alle } t \in [t_0, T]).$$

gilt.

*Beweis.* Siehe [AC84, Theorem 2.4.1]. ■

Das nächste Lemma stellt zum Abschluss dieses Kapitel die Grundlage der iterativen Berechnung erreichbarer Mengen der folgenden Kapitel dar.

**Lemma 2.10** ([Bai95, Lemma 2.1.2]). *Gegeben sei die Differentialinklusion (2.2), (2.3) mit einer mengenwertigen rechten Seite  $F: [t_0, T] \times \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ . Dann gilt für die erreichbare Menge die Halbgruppen-Eigenschaft, d.h für  $t$  und  $s$  mit  $t_0 \leq s \leq t \leq T$  gilt:*

$$\mathcal{R}(t, t_0, X_0) = \mathcal{R}(t, s, \mathcal{R}(s, t_0, X_0))$$

*Beweis.* Siehe [KF86]. ■

# 3 Approximation von Mengen durch Distanzfunktionen

In diesem Kapitel werden die Grundlagen für die Approximation von Mengen mittels Distanzfunktionen behandelt. Zunächst werden zwei verschiedene Distanzfunktionen, die *vorzeichenlose* und die *vorzeichenbehaftete*, definiert und einige wichtige Eigenschaften, Sätze und Unterschiede erläutert. Anschließend wird ein geeigneter Abstandsbegriff, der *Hausdorff-Abstand*, eingeführt um im späteren Verlauf der Arbeit Mengen vergleichen zu können. Zum Abschluss dieses Kapitels wird die lineare Interpolation von Distanzfunktionen durch baryzentrische Koordinaten als Hilfsmittel zur effizienteren Berechnung von erreichbaren Mengen, durch genauere Approximation diskreter Mengen, betrachtet.

## 3.1 Distanzfunktionen

Eine Möglichkeit um beliebige Mengen zu approximieren stellen Distanzfunktionen dar. Je nach Menge reichen bereits einige wenige Werte einer Distanzfunktion aus um die Menge sehr genau approximieren zu können. Bevor die Approximation an zwei Beispielen veranschaulicht wird, wird zunächst die vorzeichenlose Distanzfunktion definiert.

### 3.1.1 Vorzeichenlose Distanzfunktion

**Definition 3.1.** Sei  $A \subset \mathbb{R}^n$  gegeben. Die *vorzeichenlose Distanzfunktion* eines Punktes  $y \in \mathbb{R}^n$  zu  $A$  ist definiert als

$$d(y, A) = \begin{cases} \inf_{a \in A} \|y - a\|_2, & \text{falls } A \neq \emptyset \\ +\infty, & \text{falls } A = \emptyset. \end{cases} \quad (3.1)$$

Im Folgenden bezeichnet  $A^C = \{y \in \mathbb{R}^n \mid y \notin A\}$  das Komplement,  $\partial A$  den Rand,  $\overset{\circ}{A}$  das Innere und  $\overline{A} = \overset{\circ}{A} \cup \partial A$  den Abschluss einer Menge  $A$ . Als Norm wird im weiteren Verlauf stets die Euklidische Norm verwendet.

**Theorem 3.2** (Eigenschaften der vorzeichenlosen Distanzfunktion [DZ01, Theorem 2.1]).

Seien  $A$  und  $B$  abgeschlossene und nichtleere Teilmengen des  $\mathbb{R}^n$ . Für die vorzeichenlose Distanzfunktion gilt:

i) Die Abbildung  $y \mapsto d(y, A)$  ist gleichmäßig Lipschitz-stetig im  $\mathbb{R}^n$ ,

$$|d(z, A) - d(y, A)| \leq \|z - y\| \quad \forall y, z \in \mathbb{R}^n.$$

ii) Es existiert ein  $a \in \overline{A}$ , so dass  $d(y, A) = \|y - a\|$  und  $d(\cdot, A) = d(\cdot, \overline{A})$  in  $\mathbb{R}^n$ .

iii)  $\overline{A} = \{y \in \mathbb{R}^n \mid d(y, A) = 0\}$ .

iv)  $d(\cdot, A) = 0$  im  $\mathbb{R}^n \Leftrightarrow \overline{A} = \mathbb{R}^n$ .

v)  $\overline{A} \subset \overline{B} \Leftrightarrow d(\cdot, A) \geq d(\cdot, B)$ .

vi)  $d(\cdot, A)$  ist fast überall (Fréchet-) differenzierbar und

$$|\nabla d(y, A)| \leq 1 \quad \text{fast überall in } \mathbb{R}^n.$$

*Beweis.*

i) Die Lipschitz-Stetigkeit lässt sich mit Hilfe der Dreiecksungleichung zeigen. Für  $a \in A$  und  $y, z \in \mathbb{R}^n$  ist

$$\|z - a\| \leq \|z - y\| + \|y - a\|$$

und damit

$$\begin{aligned} d(z, A) &= \inf_{a \in A} \|z - a\| \\ &\leq \inf_{a \in A} \|z - y\| + \|y - a\| = \|z - y\| + \inf_{a \in A} \|y - a\| \\ &= \|z - y\| + d(y, A) \end{aligned}$$

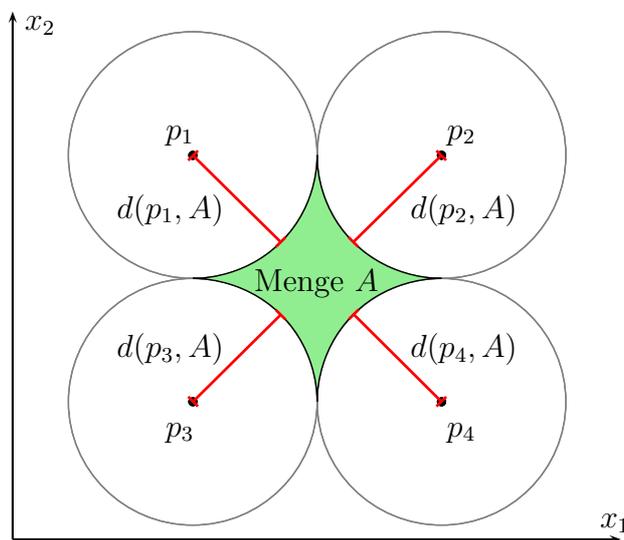
$$\Rightarrow d(z, A) - d(y, A) \leq \|z - y\|$$

Analog lässt sich  $d(y, A) - d(z, A) \leq \|y - z\|$  zeigen und damit folgt die Lipschitz-Stetigkeit.

Die Eigenschaften ii) - vi) werden hier aus Zeitgründen nicht bewiesen und es wird auf [DZ01, Seite 155] verwiesen. ■

Die Approximation einer Menge durch die vorzeichenlose Distanzfunktion wird in Abbildung 3.1 und 3.2 veranschaulicht. Dabei wird die Distanz eines Punktes  $p$  zu der Menge  $A \subset \mathbb{R}^2$  (*grün*) durch einen Kreis mit Mittelpunkt  $p$  und Radius  $d(p, A)$  dargestellt.

Die konstruierte Menge in Abbildung 3.1 lässt sich beispielsweise bereits durch vier bekannte Distanzwerte an den Punkten  $p_1, \dots, p_4$  sehr gut approximieren, während die Kreismenge in Abbildung 3.2 viel mehr Distanzwerte benötigt um nur annähernd gut approximiert werden zu können.



**Abbildung 3.1:** sehr gute Approximation der sternförmigen Menge  $A$

In Abbildung 3.2 stellt man außerdem fest, dass je mehr Punkte mit den zugehörigen Distanzwerten bekannt sind, desto besser wird die Menge approximiert. Man bemerkt außerdem, dass die Wahl der Punkte, für die die Distanzfunktion ausgewertet wird, einen Einfluss auf die Approximation der Menge hat. So könnte man in Abbildung 3.2 die auszuwertenden Punkte kreisförmig um die Menge anordnen, um damit eine bessere Approximation zu erhalten. Mit Ausblick auf die Diskretisierung von beliebigen Mengen in Kapitel 4.2 werden die Punkte der Einfachheit halber stets gitterförmig, wie in den Abbildungen 3.1 und 3.2, angeordnet.

Vorzeichenlose Distanzfunktionen liefern eine gute „äußere“ Beschreibung von Mengen. Innerhalb einer Menge haben diese jedoch immer den Wert 0, was zu Nichtdifferenzierbarkeitsstellen der Distanzfunktion am Rand einer Menge führt.

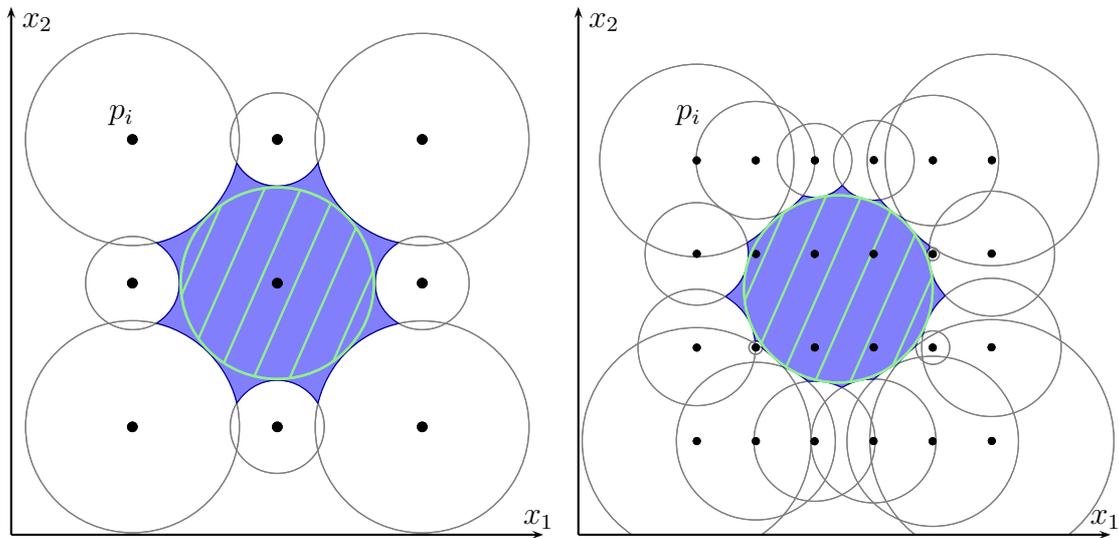


Abbildung 3.2: Approximation einer Kreismenge (grün schraffiert)

Zusätzlich wird man in Kapitel 3.3 feststellen, dass die lineare Interpolation der vorzeichenlosen Distanzfunktion ungeeignet für den weiteren Verlauf der Arbeit ist. Durch eine Verallgemeinerung der vorzeichenlosen Distanzfunktion lässt sich die Anzahl der Nichtdifferenzierbarkeitsstellen jedoch reduzieren und man erhält zudem eine geeignete Funktion für die lineare Interpolation in Kapitel 3.3.

### 3.1.2 Vorzeichenbehaftete Distanzfunktion

Die vorzeichenbehaftete Distanzfunktion basiert auf der vorzeichenlosen Distanzfunktion und erweitert diese durch zusätzliche Informationen für Punkte im Inneren einer Menge.

**Definition 3.3.** Sei  $A \subset \mathbb{R}^n$  gegeben. Die *vorzeichenbehaftete Distanzfunktion* eines Punktes  $y \in \mathbb{R}^n$  zu  $A$  ist definiert als

$$b(y, A) := d(y, A) - d(y, A^C) \quad \forall y \in \mathbb{R}^n.$$

Für eine nichtleere Teilmenge  $D$  des  $\mathbb{R}^n$  bezeichnet

$$C_b(D) := \{b(\cdot, A) \mid A \subset \overline{D} \text{ und } \partial A \neq \emptyset\}$$

die *Familie der vorzeichenbehafteten Distanzfunktionen*.

*Bemerkung* ([DZ01, Seite 207]). Die Funktion  $b(\cdot, A)$  ist endlich genau dann, wenn

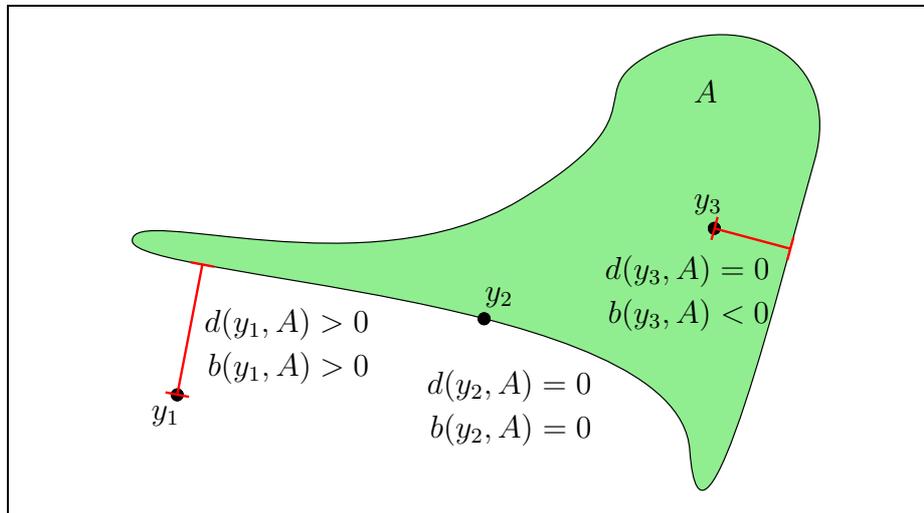
$\emptyset \neq A \neq \mathbb{R}^n$ , denn nach Definition 3.1 ist

$$b(\cdot, A) = \begin{cases} d(\cdot, A) = +\infty & \text{falls } A = \emptyset, \\ -d(\cdot, A^C) = -\infty & \text{falls } A^C = \emptyset \text{ bzw. } A = \mathbb{R}^n. \end{cases}$$

Diese Bedingung ist somit äquivalent zu der Bedingung  $\partial A \neq \emptyset$ , wodurch die vorzeichenbehaftete Distanzfunktion auch als „algebraischer Abstand“ zum Rand (distance to boundary) von  $A$  betrachtet werden kann:

$$b(y, A) = \begin{cases} d(y, A) = d(y, \partial A) & \text{falls } y \notin A, \\ 0 & \text{falls } y \in \partial A, \\ -d(y, A^C) = -d(y, \partial A) & \text{falls } y \in \overset{\circ}{A}. \end{cases}$$

Abbildung 3.3 veranschaulicht den Unterschied zwischen der vorzeichenlosen und der vorzeichenbehafteten Distanzfunktion. Für einen Punkt  $y_1$  außerhalb der Menge  $A$  sind beide Distanzfunktionswerte, dargestellt durch eine rote Linie, positiv. Für einen Punkt  $y_2$  am Rand der Menge  $A$  haben beide Distanzfunktionen den Wert 0 und für einen Punkt  $y_3$  innerhalb der Menge  $A$  unterscheiden sich die Distanzfunktionen. Die vorzeichenlose Distanzfunktion hat für  $y_3 \in \overset{\circ}{A}$  den Wert 0, während die vorzeichenbehaftete Distanzfunktion, als Abstand zum Rand innerhalb einer Menge, einen negativen Wert annimmt.



**Abbildung 3.3:** vorzeichenlose und vorzeichenbehaftete Distanzfunktion

**Theorem 3.4** (Eigenschaften der vorzeichenbehafteten Distanzfunktion [DZ01, Theorem 2.1]).

Seien  $A$  und  $B$  Teilmengen des  $\mathbb{R}^n$ . Es gilt:

$$i) \quad A \neq \emptyset \text{ und } A^C \neq \emptyset \Leftrightarrow \partial A \neq \emptyset$$

ii) Für vorzeichenbehaftete Distanzfunktionen  $b(\cdot, A)$  und  $b(\cdot, B)$  in  $C_b(D)$  gilt:

$$A \supset B \Rightarrow b(\cdot, A) \leq b(\cdot, B),$$

$$A = B \Rightarrow b(\cdot, A) = b(\cdot, B),$$

$$b(\cdot, A) \leq b(\cdot, B) \text{ in } D \Leftrightarrow \overline{B} \subset \overline{A} \text{ und } \overline{A^C} \subset \overline{B^C},$$

$$b(\cdot, A) = b(\cdot, B) \text{ in } D \Leftrightarrow \overline{B} = \overline{A} \text{ und } \overline{A^C} = \overline{B^C} \Leftrightarrow \overline{B} = \overline{A} \text{ und } \partial A = \partial B.$$

Insbesondere gilt

$$b(\cdot, \overline{A}) \leq b(\cdot, A) \text{ und } b(\cdot, \overline{A}) = b(\cdot, A) \Leftrightarrow \partial \overline{A} = \partial A.$$

$$iii) \quad |b(\cdot, A)| = d(\cdot, A) + d(\cdot, A^C) = \max \{d(\cdot, A), d(\cdot, A^C)\} = d(\cdot, \partial A) \\ \text{und } \partial A = \{y \in \mathbb{R}^n \mid b(y, A) = 0\}.$$

$$iv) \quad b(\cdot, A) \geq 0 \Leftrightarrow \overline{A^C} \supset \partial A \supset \overline{A} \Leftrightarrow \partial A = \overline{A}.$$

$$v) \quad b(\cdot, A) = 0 \Leftrightarrow \overline{A^C} = \partial A = \overline{A} \Leftrightarrow \partial A = \mathbb{R}^n.$$

vi) Falls  $\partial A \neq \emptyset$ , ist die Funktion  $b(\cdot, A)$  gleichmäßig Lipschitz-stetig im  $\mathbb{R}^n$  und

$$|b(y, A) - b(z, A)| \leq \|y - z\| \quad \forall y, z \in \mathbb{R}^n. \quad (3.2)$$

Darüber hinaus ist  $b(\cdot, A)$  fast überall (Fréchet-) differenzierbar und

$$|\nabla b(y, A)| \leq 1 \quad \text{fast überall in } \mathbb{R}^n.$$

*Beweis.* Siehe [DZ01, Seite 208 f.] ■

**Beispiel 3.5** ([DZ94, Example 5.2]). Betrachte die vorzeichenlose Distanzfunktion  $d(\cdot, A)$  und die vorzeichenbehaftete Distanzfunktion  $b(\cdot, A)$  im  $\mathbb{R}^2$  für einen Ball um den Ursprung mit Radius  $r > 0$ , also

$$A = \{y \in \mathbb{R}^2 \mid |y| \leq r\}, \quad \partial A = \{y \in \mathbb{R}^2 \mid |y| = r\}.$$

Die vorzeichenlose Distanzfunktion für einen Punkt  $z \in \mathbb{R}^2$  lautet

$$d(z, A) = \begin{cases} \|z\| - r, & \text{falls } z \notin A \\ 0, & \text{falls } z \in A \end{cases}$$

und die vorzeichenbehaftete Distanzfunktion lautet

$$b(z, A) = \|z\| - r.$$

Abbildung 3.4 und 3.5 verdeutlichen nochmal grafisch den Unterschied zwischen der vorzeichenlosen und der vorzeichenbehafteten Distanzfunktion für  $r = 1$ , d.h.  $A$  entspricht dem Einheitskreis.

Man erkennt, dass die vorzeichenlose Distanzfunktion am Rand der Menge Nichtdifferenzierbarkeitsstellen hat, während die vorzeichenbehaftete Distanzfunktion, durch die negativen Distanzwerte innerhalb der Menge, die Nichtdifferenzierbarkeitsstellen auf den Ursprung reduziert. Weitere Untersuchungen zum Gradienten und Glattheit der Distanzfunktion am Rand einer Menge können hier aus Zeitgründen nicht durchgeführt werden und es wird daher auf [DZ94, DZ01] verwiesen.

Im weiteren Verlauf der Arbeit werden Distanzfunktionen verwendet, um erreichbare Mengen von Kontrollsystemen zu berechnen. Die dazu benötigten Mengenoperationen für Distanzfunktionen werden in den folgenden Sätzen behandelt.

**Satz 3.6** (Abstand zum Komplement [Rod10, Satz 2.5]).

Sei  $y \in \mathbb{R}^n$  und  $A \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Für die vorzeichenbehaftete Distanzfunktion gilt:

$$b(y, A^C) = -b(y, A) \tag{3.3}$$

*Beweis.*

$$b(y, A^C) \stackrel{\text{def}}{=} d(y, A^C) - d(y, A) = -\left(d(y, A) - d(y, A^C)\right) = -b(y, A)$$

■

**Satz 3.7** (Verschiebung der Menge [Rod10, Satz 2.3]).

Sei  $y, v \in \mathbb{R}^n$  und  $A \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Für Distanzfunktionen gilt:

$$d(y, A + v) = d(y - v, A) \quad \text{resp.} \quad b(y, A + v) = b(y - v, A) \tag{3.4}$$

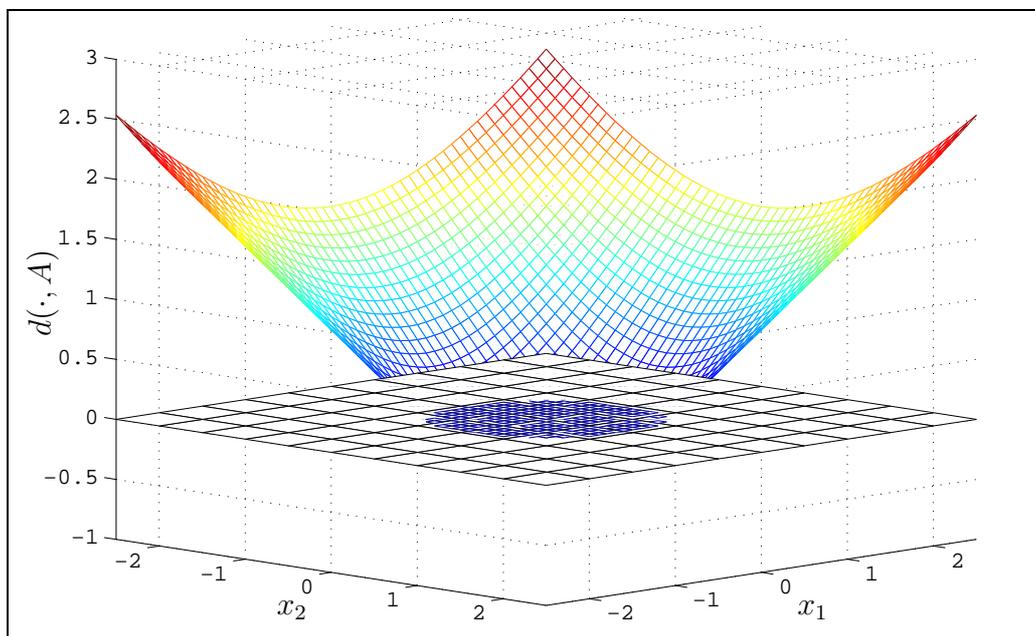


Abbildung 3.4: vorzeichenlose Distanzfunktion für den Einheitskreis

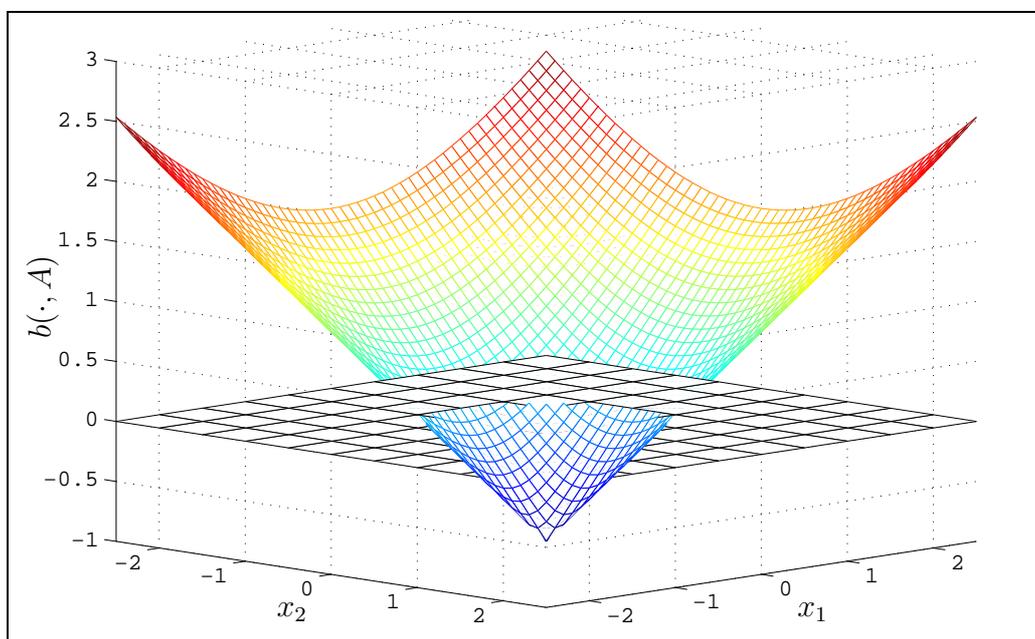


Abbildung 3.5: vorzeichenbehaftete Distanzfunktion für den Einheitskreis

*Beweis.* Die Aussage wird zunächst für die vorzeichenlose und anschließend für die vorzeichenbehaftete Distanzfunktion bewiesen.

i)

$$\begin{aligned} d(y, A + v) &= \inf_{z \in (A+v)} \|y - z\| \\ &\stackrel{(\star)}{=} \inf_{a \in A} \|y - a - v\| = \inf_{a \in A} \|y - v - a\| \\ &= d(y - v, A) \end{aligned}$$

( $\star$ ) Für  $z \in (A + v)$  existiert ein  $a \in A$ , so dass  $z = a + v$ .

ii)

$$\begin{aligned} b(y, A + v) &= d(y, A + v) - d(y, (A + v)^C) \\ &\stackrel{(\Delta)}{=} d(y, A + v) - d(y, A^C + v) \\ &\stackrel{i)}{=} d(y - v, A) - d(y - v, A^C) \\ &= b(y - v, A) \end{aligned}$$

( $\Delta$ ) Durch Nachrechnen lässt sich zeigen, dass  $(A + v)^C = A^C + v$  ist. ■

**Satz 3.8** (Skalares Vielfaches [Rod10, Satz 2.4]).

Sei  $y \in \mathbb{R}^n$ ,  $\lambda > 0$  und  $A \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Dann gilt:

$$d(y, \lambda \cdot A) = \lambda \cdot d\left(\frac{1}{\lambda} \cdot y, A\right) \quad \text{resp.} \quad b(y, \lambda \cdot A) = \lambda \cdot b\left(\frac{1}{\lambda} \cdot y, A\right) \quad (3.5)$$

*Beweis.* Die Aussage wird wie oben zuerst für die vorzeichenlose und dann für die vorzeichenbehaftete Distanzfunktion bewiesen.

i)

$$\begin{aligned} d(y, \lambda \cdot A) &= \inf_{z \in \lambda \cdot A} \|y - z\| \\ &\stackrel{(\star)}{=} \inf_{a \in A} \|y - \lambda \cdot a\| \\ &= \inf_{a \in A} \left\| \lambda \cdot \left( \frac{1}{\lambda} \cdot y - a \right) \right\| = \lambda \cdot \inf_{a \in A} \left\| \frac{1}{\lambda} \cdot y - a \right\| \\ &= \lambda \cdot d\left(\frac{1}{\lambda} \cdot y, A\right) \end{aligned}$$

( $\star$ ) Für  $z \in (\lambda \cdot A)$  existiert ein  $a \in A$ , so dass  $z = \lambda \cdot a$ .

ii)

$$\begin{aligned} b(y, \lambda \cdot A) &= d(y, \lambda \cdot A) - d(y, (\lambda \cdot A)^C) \\ &\stackrel{(\Delta)}{=} d(y, \lambda \cdot A) - d(y, \lambda \cdot A^C) \\ &\stackrel{i)}{=} \lambda \cdot \left( d\left(\frac{1}{\lambda} \cdot y, A\right) - d\left(\frac{1}{\lambda} \cdot y, A^C\right) \right) \\ &= \lambda \cdot b\left(\frac{1}{\lambda} \cdot y, A\right) \end{aligned}$$

( $\Delta$ ) Durch Nachrechnen lässt sich zeigen, dass  $(\lambda \cdot A)^C = \lambda \cdot A^C$  ist. ■

**Satz 3.9** (Durchschnitt von Mengen [Har96, Theorem 3]).

Seien  $A, B \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Die Distanz eines Punktes  $y \in \mathbb{R}^n$  zum Durchschnitt der Mengen  $A$  und  $B$  ist begrenzt durch

$$b(y, A \cap B) \geq \max \{b(y, A), b(y, B)\}. \quad (3.6)$$

Diese Aussage gilt gleichermaßen für die vorzeichenlose Distanzfunktion.

*Beweis.* Es werden die vier möglichen Fälle (I:  $y \in A \cap B$ ), (II:  $y \in A, x \notin B$ ), (III:  $y \notin A, x \in B$ ) und (IV:  $y \notin A \cup B$ ), wie in Abbildung 3.6 veranschaulicht, untersucht.

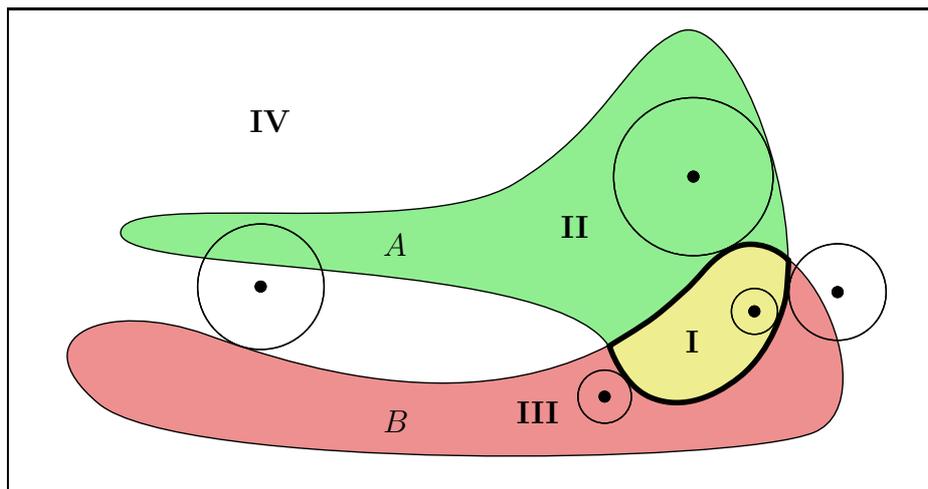


Abbildung 3.6: Skizze zum Beweis von Satz 3.9

- I: Sei  $y \in A \cap B$ . Beide vorzeichenbehaftete Distanzfunktionen  $b(y, A)$  und  $b(y, B)$  haben einen negativen Wert und der größere der beiden Werte ist gerade die Distanz zum nächsten Randpunkt des Durchschnitts.
- II: Sei  $y \in A$  und  $y \notin B$ . Dann ist  $b(y, A)$  negativ und  $b(y, B)$  positiv und daher der größere der beiden Werte. Der nächste Punkt aus  $B$  zu  $y$  muss zwar nicht im Durchschnitt liegen, es kann aber auch kein Punkt aus dem Durchschnitt näher an  $y$  sein.
- III: Sei  $y \notin A$  und  $y \in B$ . Analog zum 2. Fall mit vertauschten Rollen von  $A$  und  $B$ .
- IV: Sei  $y \notin A \cup B$ . Der nächste Punkt aus dem Durchschnitt kann nicht näher an  $y$  liegen als der nächste Punkt aus  $A$  oder aus  $B$ .

Der Beweis funktioniert analog für die vorzeichenlose Distanzfunktion, indem man alle negativen Werte von  $b(y, \cdot)$  mit  $d(y, \cdot) = 0$  ersetzt. ■

**Satz 3.10** (Vereinigung von Mengen [Rod10, Satz 2.7]).

Sei  $y \in \mathbb{R}^n$  und  $A, B \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Für die vorzeichenlose Distanzfunktion gilt:

$$d(y, A \cup B) = \min\{d(y, A), d(y, B)\} \quad (3.7)$$

*Beweis.* Betrachte die vier möglichen Fälle (I:  $y \in A \cap B$ ), (II:  $y \in A, x \notin B$ ), (III:  $y \notin A, x \in B$ ) und (IV:  $y \notin A \cup B$ ).

- I: Sei  $y \in A \cap B$ . Beide Distanzfunktionen  $d(y, A)$  und  $d(y, B)$  haben den Wert 0 und damit auch das Minimum beider Werte. Andererseits ist  $d(y, A \cup B) = 0$  da  $A \cap B \subseteq A \cup B$  ist.
- II: Sei  $y \in A$  und  $y \notin B$ . Dann ist  $d(y, A) = 0$  und  $d(y, B) > 0$ . Das Minimum hat daher den Wert 0 und  $d(y, A \cup B) = 0$  da  $A \subseteq A \cup B$  ist.
- III: Sei  $y \notin A$  und  $y \in B$ . Analog zum Fall (II) mit vertauschten Rollen von  $A$  und  $B$ .
- IV: Sei  $y \notin A \cup B$ . Ist  $d(y, A) < d(y, B)$  dann ist nach Definition der vorzeichenlosen Distanzfunktion  $d(y, A \cup B) = d(y, A)$ . Ist  $d(y, B) < d(y, A)$  dann ist  $d(y, A \cup B) = d(y, B)$  und damit ist  $d(y, A \cup B) = \min\{d(y, A), d(y, B)\}$ . ■

**Satz 3.11** (Vereinigung von Mengen).

Sei  $y \in \mathbb{R}^n$  und  $A, B \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Für die vorzeichenbehaftete Distanzfunktion gilt:

$$b(y, A \cup B) \leq \min\{b(y, A), b(y, B)\} \quad (3.8)$$

*Beweis.*

$$\begin{aligned} b(y, A \cup B) &= d(y, A \cup B) - d(y, (A \cup B)^C) \\ &\stackrel{(\diamond)}{=} d(y, A \cup B) - d(y, A^C \cap B^C) \\ &\stackrel{(3.6)}{\leq} \min\{d(y, A), d(y, B)\} - \max\{d(y, A^C), d(y, B^C)\} \\ &\stackrel{(\star)}{\leq} \begin{cases} d(y, A) - d(y, A^C) & \text{falls } d(y, A) \leq d(y, B) \\ d(y, B) - d(y, B^C) & \text{falls } d(y, B) < d(y, A) \end{cases} \\ &= \begin{cases} b(y, A) & \text{falls } d(y, A) \leq d(y, B) \\ b(y, B) & \text{falls } d(y, B) < d(y, A) \end{cases} \\ &= \min\{b(y, A), b(y, B)\} \end{aligned}$$

( $\diamond$ ) nach De Morgan gilt:  $(A \cup B)^C = A^C \cap B^C$

( $\star$ ) mit  $\max\{d(y, A^C), d(y, B^C)\} \geq d(y, A^C)$  und ebenso  $\max\{d(y, A^C), d(y, B^C)\} \geq d(y, B^C)$ . ■

Mit Hilfe dieser Sätze wird im nächsten Kapitel die Berechnung erreichbarer Mengen durch Distanzfunktionen realisiert. Um jedoch Aussagen über die Ergebnisse treffen zu können, wird zunächst eine Methode zum Vergleich von Mengen benötigt.

## 3.2 Hausdorff-Abstand von Mengen

Um berechnete Mengen im weiteren Verlauf der Arbeit vergleichen zu können, braucht man einen geeigneten Abstands begriff für Mengen. In diesem Abschnitt wird dazu der Hausdorff-Abstand definiert.

**Definition 3.12.** Gegeben seien zwei nichtleere Mengen  $A, B \subset \mathbb{R}^n$ . Dann heißt

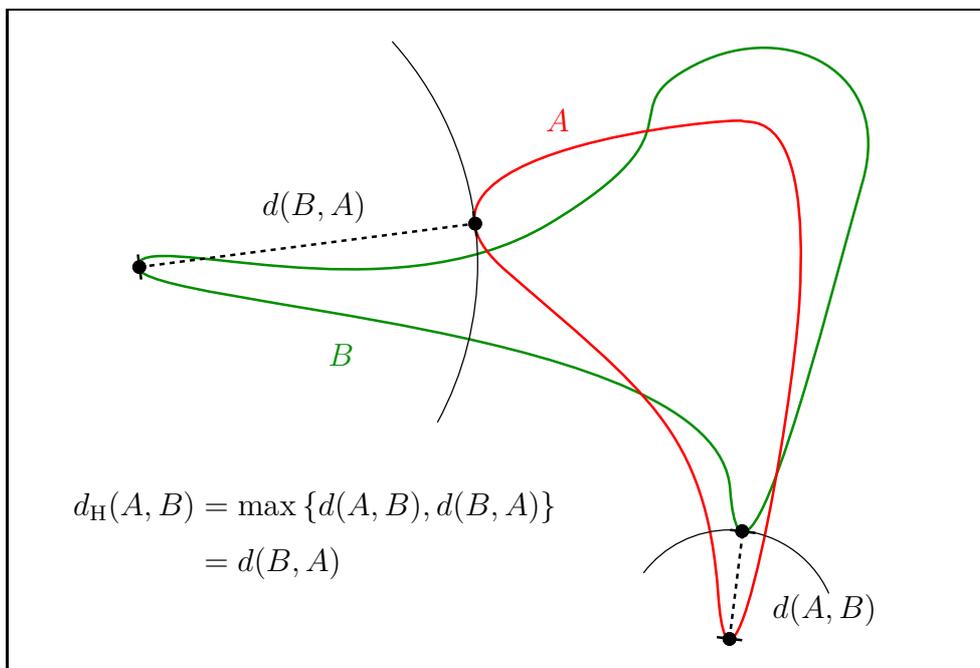
$$d_H(A, B) := \max\{d(A, B), d(B, A)\} \in [0, +\infty] \quad (3.9)$$

der *Hausdorff-Abstand* von  $A$  und  $B$ , wobei

$$d(A, B) := \sup_{a \in A} d(a, B) \in [0, +\infty]$$

den *einseitigen (Hausdorff-)Abstand* zwischen den Mengen  $A$  und  $B$ , und  $d(a, B)$  die vorzeichenlose Distanzfunktion (3.1) von Punkt  $a$  zur Menge  $B$  bezeichnet. Für  $B = \{0_{\mathbb{R}^n}\}$  erhält man aus (3.9) die *Norm* einer kompakten Menge  $A$

$$\|A\| := \sup_{a \in A} \|a\|_2 \in [0, +\infty].$$



**Abbildung 3.7:** Hausdorff-Abstand zwischen den Mengen  $A$  und  $B$

Der nächste Satz motiviert die Verwendung des Abstandsbegriffes bei späteren Konvergenzuntersuchungen und beschreibt eine besondere Eigenschaft des Haus-

dorff-Abstandes.

**Satz 3.13** ([Bai95, Satz 0.1.2.3]). *Die beiden Mengen*

$$\begin{aligned}\mathcal{C}(\mathbb{R}^n) &:= \{A \subset \mathbb{R}^n \mid A \text{ nichtleer, kompakt}\}, \\ \mathcal{CC}(\mathbb{R}^n) &:= \{C \subset \mathbb{R}^n \mid C \text{ nichtleer, kompakt und konvex}\}\end{aligned}$$

*bilden zusammen mit dem Hausdorff-Abstand metrische Räume.*

*Beweis.* Vgl. [Kur66, CH. II, §21, VII., Theorem, p. 214] und [Mar77, XII.1, Lemma 1, S. 220]. ■

**Lemma 3.14** (Eigenschaften des Hausdorff-Abstandes [Bai95, Lemma 0.1.2.4]). *Gegeben seien die nichtleeren, abgeschlossenen Mengen  $A, B, C \subset \mathbb{R}^n$ . Dann gilt:*

- i)  $d_{\text{H}}(A, B) = d_{\text{H}}(B, A)$*
- ii)  $d(A, B) = 0 \Leftrightarrow A \subset B$*
- iii)  $d_{\text{H}}(A, B) = 0 \Leftrightarrow A = B$*
- iv)  $d(A, B) \leq \delta \Leftrightarrow A \subset B + \delta B_n$ ,  
wobei  $B_n := \{y \in \mathbb{R}^n \mid \|y\| \leq 1\}$  die abgeschlossene Einheitskugel im  $\mathbb{R}^n$  bezeichnet.*
- v)  $d_{\text{H}}(A, C) \leq d_{\text{H}}(A, B) + d_{\text{H}}(B, C)$*
- vi)  $d_{\text{H}}(A + C, B + D) \leq d_{\text{H}}(A, B) + d_{\text{H}}(C, D)$*

*Beweis.* Für *i) - v)* vgl. [Fil88, Seite 65], *vi)* kann durch Nachrechnen gezeigt werden. ■

Wie man zu Beginn des Kapitels in den Abbildungen 3.1 und 3.2 gesehen hat, lassen sich Mengen, mit Hilfe von Distanzfunktionen, durch endlich viele Punkte approximieren. Ein weiterer Vorteil besteht in der Möglichkeit die Werte der Distanzfunktion zu interpolieren und dadurch diskrete Mengen, ohne Auswertung der Distanzfunktion an weiteren Punkten, genauer zu approximieren. Im Folgenden wird dazu die stückweise lineare Interpolation mittels baryzentrischen Koordinaten eingeführt.

### 3.3 Lineare Interpolation durch baryzentrische Koordinaten

Eine Möglichkeit, um diskrete Mengen genauer zu approximieren, ist die Auswertung der Distanzfunktion an weiteren Punkten, siehe Abbildung 3.2. Wie man später jedoch sehen wird hat die Anzahl der Auswertungen einen großen Einfluss auf die Berechnungsdauer der erreichbaren Mengen. In diesem Abschnitt wird daher die lineare Interpolation von Distanzfunktionen behandelt, die es ermöglicht Distanzfunktionswerte weiterer Punkte näherungsweise zu berechnen, ohne die Distanzfunktion in diesen Punkten auszuwerten. Bei der Realisierung der stückweise linearen Interpolation werden baryzentrische Koordinaten verwendet, die wie folgt definiert sind.

**Definition 3.15.** Seien  $\{z_i\}_{i=0,1,\dots,m}$  Punkte im  $\mathbb{R}^n$  und  $\{\lambda_i\}_{i=0,1,\dots,m}$  skalare Größen, so dass  $\lambda_i = 0$  bis auf endlich viele Indizes und  $\sum_{i=0}^m \lambda_i = 1$ . Der *Schwerpunkt* (das *Baryzentrum*) der Punkte  $z_i$  mit den Gewichten  $\lambda_i$  ist der Punkt  $y$  mit

$$y = \sum_{i=0}^m \lambda_i \cdot z_i.$$

**Satz 3.16** ([Ber04, Proposition 3.6.2]). *Sei  $\{z_0, z_1, \dots, z_n\}$  ein Simplex im  $\mathbb{R}^n$ . Für jeden Punkt  $y \in \mathbb{R}^n$  existieren  $\lambda_i \in \mathbb{R}$ , ( $i = 0, 1, \dots, n$ ), so dass*

$$\sum_{i=1}^n \lambda_i = 1,$$

$$y = \sum_{i=1}^n \lambda_i \cdot z_i.$$

Die skalaren Größen  $\lambda_i$  sind eindeutig und heißen *baryzentrische Koordinaten* von  $y$  bezüglich des Simplex  $\{z_i\}_{i=0,1,\dots,n}$ .

*Beweis.* Siehe [Ber04, Seite 81]. ■

Der Einfachheit halber sei im weiteren Verlauf dieses Kapitels  $n = 2$ , man betrachte also die stückweise lineare Interpolation durch baryzentrische Koordinaten und die zugehörige Fehleranalyse im  $\mathbb{R}^2$ .

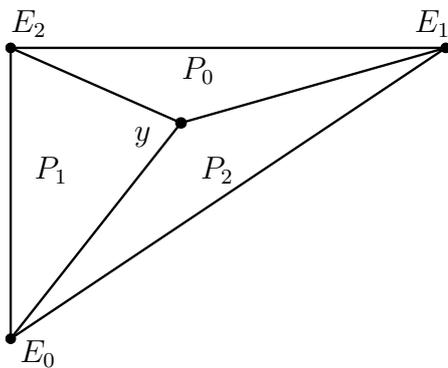
**Beispiel 3.17.** *Betrachte als Simplex des  $\mathbb{R}^2$  ein Dreieck mit den Ecken  $E_0, E_1$  und  $E_2$ , siehe Abbildung 3.8. Jeder Punkt  $y \in \mathbb{R}^2$  kann nach Satz 3.16 mit den baryzentrischen Koordinaten  $\lambda_i$ , ( $i = 0, 1, 2$ ), bezüglich des Dreiecks  $\triangle E_0 E_1 E_2$  als*

$$y = \lambda_0 \cdot E_0 + \lambda_1 \cdot E_1 + \lambda_2 \cdot E_2$$

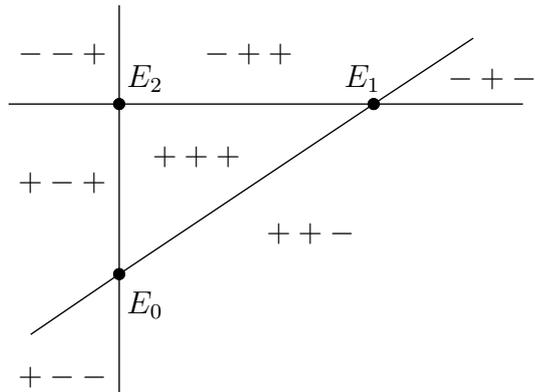
dargestellt werden. Der Punkt  $y$  ist dabei genau dann innerhalb des Dreiecks  $\triangle E_0 E_1 E_2$ , wenn  $0 \leq \lambda_i \leq 1$ , ( $i = 0, 1, 2$ ), siehe Abbildung 3.9. Geometrisch betrachtet entsprechen die baryzentrischen Koordinaten in diesem Fall

$$\lambda_i = \frac{P_i}{P} \quad (i = 0, 1, 2),$$

wobei  $P$  die Fläche des Dreiecks  $\triangle E_0 E_1 E_2$  und  $P_i$  die gerichtete Fläche des  $i$ -ten Teildreiecks bezeichnet. Die Richtung der Fläche wird dabei durch die Orientierung des Teildreiecks bestimmt.



**Abbildung 3.8:** geometrische Interpretation der baryzentrischen Koordinaten im  $\mathbb{R}^2$  mit  $\lambda_i = \frac{P_i}{P}$  ( $i = 0, 1, 2$ )



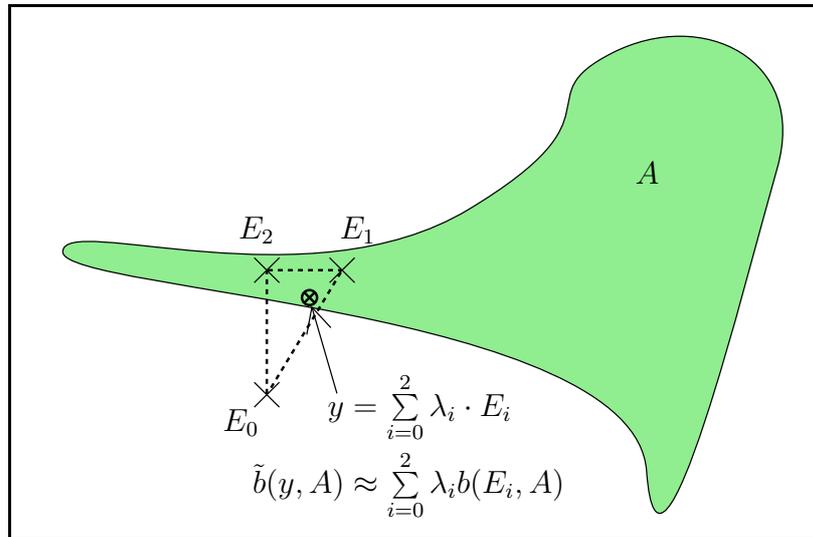
**Abbildung 3.9:** Vorzeichen der baryzentrischen Koordinaten  $\lambda_0, \lambda_1$  und  $\lambda_2$  in den verschiedenen Gebieten

Mit bekannten Distanzfunktionswerten in den Punkten  $E_0, E_1$  und  $E_2$  lässt sich somit für jeden beliebigen Punkt  $y$  im Dreieck  $\triangle E_0 E_1 E_2$  der Distanzfunktionswert interpolieren und dabei der Rechenaufwand für die Auswertung der Distanzfunktion in diesem Punkt einsparen. Für die Distanzfunktion zu einer Menge  $A$  gilt dabei die Näherung

$$\tilde{b}(y, A) = \tilde{b}\left(\sum_{i=0}^2 \lambda_i \cdot E_i, A\right) \approx \sum_{i=0}^2 \lambda_i b(E_i, A).$$

Anhand der Abbildung 3.10, wird die stückweise lineare Interpolation von Distanzfunktionen durch baryzentrische Koordinaten nochmal veranschaulicht und die Wahl der vorzeichenbehafteten Distanzfunktion, für den weiteren Verlauf der Arbeit, begründet.

*Bemerkung.* Für eine bessere Approximation diskreter Mengen, durch lineare Interpolation der Distanzfunktion, muss die vorzeichenbehaftete Distanzfunktion



**Abbildung 3.10:** stückweise lineare Interpolation von Distanzfunktionen durch baryzentrische Koordinaten

gewählt werden. Betrachtet man beispielsweise die vorzeichenlose Distanzfunktion in Abbildung 3.10, dann erhält man die Werte  $d(E_0, A) > 0$  und  $d(E_1, A) = d(E_2, A) = 0$ . Damit ist für jeden Punkt  $y$  der innerhalb des Dreiecks  $\triangle E_0 E_1 E_2$  und nicht auf der Geraden  $\overrightarrow{E_1 E_2}$  liegt

$$\tilde{d}(y, A) = \tilde{d}\left(\sum_{i=0}^2 \lambda_i \cdot E_i, A\right) \approx \sum_{i=0}^2 \lambda_i \cdot d(E_i, A) > 0.$$

Somit liegt der Punkt nicht in der Menge und man erhält, bei Verwendung der vorzeichenlosen Distanzfunktion, keine genauere Approximation der Menge.

### 3.3.1 Fehleranalyse der stückweisen Interpolation

Für die Fehleranalyse im  $\mathbb{R}^2$  werden Punkte, für die die Distanzfunktion ausgewertet wird, gitterförmig im Raum verteilt. Betrachte, wie in Abbildung 3.10, drei dieser Punkte als Eckpunkte eines rechtwinkligen Dreiecks. In diesem Dreieck wird zunächst der Fehler der stückweise konstanten Interpolation und anschließend der stückweise linearen Interpolation unter verschiedenen Bedingungen analysiert.

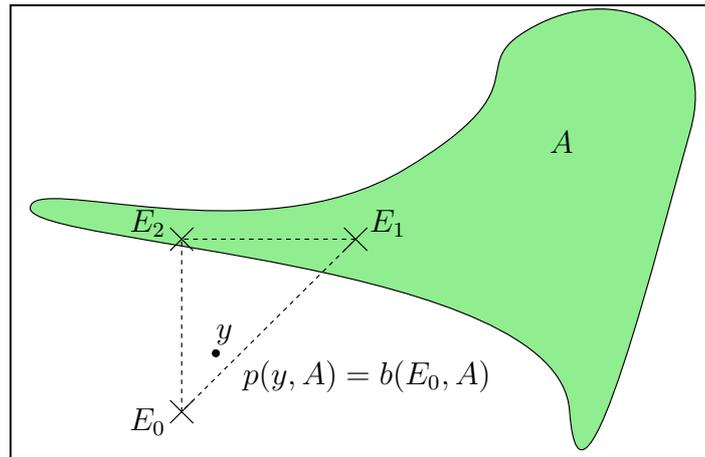
**Satz 3.18** (Stückweise konstante Interpolation).

Gegeben sei ein Dreieck im  $\mathbb{R}^2$  mit den Eckpunkten  $E_0$ ,  $E_1$  und  $E_2$  und den zugehörigen Distanzfunktionswerten  $b(E_i, A)$ , ( $i = 0, 1, 2$ ). Ferner sei  $y \in \triangle E_0 E_1 E_2$

und

$$p(y, A) = b(E_i, A),$$

wobei  $p(y, A)$  den interpolierten Distanzfunktionswert für den Punkt  $y$  zur Menge  $A$  darstellt und  $E_i$  als beliebiger Eckpunkt des Dreiecks  $\triangle E_0E_1E_2$ , beispielsweise als der nächste Eckpunkt zu  $y$ , gewählt werden kann, siehe Abbildung 3.11.



**Abbildung 3.11:** stückweise konstante Interpolation der Distanzfunktion

Dann kann der Fehler für den interpolierten Distanzfunktionswert bei der stückweise konstanten Interpolation abgeschätzt werden durch

$$|p(y, A) - b(y, A)| \leq \text{diam}(\triangle E_0E_1E_2),$$

wobei  $\text{diam}(\triangle E_0E_1E_2)$  den Umkreisdurchmesser des Dreiecks  $\triangle E_0E_1E_2$  bezeichnet.

*Bemerkung.* Für gitterförmig verteilte Eckpunkte mit  $|\overrightarrow{E_0E_2}| = |\overrightarrow{E_1E_2}| = \rho$  liegt der Umkreismittelpunkt genau zwischen den Punkten  $E_0$  und  $E_1$ . Folglich beträgt der Durchmesser des Umkreises

$$\text{diam}(\triangle E_0E_1E_2) = \sqrt{2}\rho.$$

*Beweis von Satz 3.18.*

$$\begin{aligned} |p(y, A) - b(y, A)| &= |b(E_i, A) - b(y, A)| \\ &\leq \|E_i - y\| \\ &\leq \text{diam}(\triangle E_0E_1E_2) \end{aligned}$$



**Satz 3.19** (Stückweise lineare Interpolation).

Gegeben sei ein Dreieck im  $\mathbb{R}^2$  mit den Eckpunkten  $E_0, E_1$  und  $E_2$  und den zugehörigen Distanzfunktionswerten  $b(E_i, A)$ , ( $i = 0, 1, 2$ ). Ferner sei  $y \in \triangle E_0 E_1 E_2$ , dann existieren nach Satz 3.16 baryzentrische Koordinaten von  $y$  bezüglich des Dreiecks  $\triangle E_0 E_1 E_2$  mit  $\sum_{i=0}^2 \lambda_i = 1$  und  $y = \sum_{i=0}^2 \lambda_i \cdot E_i$ . Der interpolierte Distanzfunktionswert von  $y$  zur Menge  $A$  sei

$$p(y, A) = \sum_{i=0}^2 \lambda_i b(E_i, A).$$

Dann kann der Fehler bei der stückweise linearen Interpolation mittels baryzentrischer Koordinaten abgeschätzt werden durch

$$|p(y, A) - b(y, A)| \leq \text{diam}(\triangle E_0 E_1 E_2).$$

*Beweis.*

$$\begin{aligned} |p(y, A) - b(y, A)| &= \left| \sum_{i=0}^2 \lambda_i b(E_i, A) - b(y, A) \right| \\ &= \left| \sum_{i=0}^2 \lambda_i b(E_i, A) - \underbrace{\sum_{i=0}^2 \lambda_i}_{=1} b(y, A) \right| \\ &\leq \sum_{i=0}^2 \lambda_i |b(E_i, A) - b(y, A)| \\ &\leq \underbrace{\sum_{i=0}^2 \lambda_i}_{=1} \|E_i - y\| \\ &\leq \text{diam}(\triangle E_0 E_1 E_2) \end{aligned}$$



Unter einer zusätzlichen Voraussetzung an den Gradienten der Distanzfunktion, lässt sich eine bessere Fehlerabschätzung herleiten.

**Satz 3.20** (Stückweise lineare Interpolation).

Sei  $y$  und  $p(y, A)$  wie im Satz 3.19. Zusätzlich sei der Gradient der Distanzfunktion lokal Lipschitzstetig, d.h.

$$\|\nabla b(x, A) - \nabla b(y, A)\| \leq L \|x - y\| \quad \forall x, y \in \triangle E_0 E_1 E_2.$$

Dann kann der Fehler für den interpolierten Distanzfunktionswert bei der stückweise linearen Interpolation durch baryzentrische Koordinaten abgeschätzt werden durch

$$|p(y, A) - b(y, A)| \leq \frac{1}{2}L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2.$$

*Beweis.* Da im Folgenden die Distanzfunktion stets zur Menge  $A$  betrachtet wird, wird die Notation vereinfacht, indem  $b(y)$  für  $b(y, A)$  und  $p(y)$  für  $p(y, A)$  geschrieben wird. Betrachte zunächst die Funktion  $\phi$  zwischen einem Eckpunkt  $E_i$  und dem Punkt  $y$ , definiert durch

$$\phi(\vartheta) := y + \vartheta \cdot (E_i - y) \quad \text{für } \vartheta \in [0, 1],$$

mit  $\phi(1) = E_i$  und  $\phi(0) = y$ . Nach Einsetzen von  $\phi(\vartheta)$  in die Distanzfunktion  $b(\cdot)$ , lässt sich die Differenz  $(b(E_i) - b(y))$  wie folgt abschätzen:

$$\begin{aligned} b(\phi(\vartheta)) &= b(\phi(0)) + \int_0^\vartheta \frac{d}{ds} (b \circ \phi)(s) ds \\ &= b(\phi(0)) + \int_0^\vartheta \nabla b(\phi(s)) \cdot \nabla \phi(s) ds \\ &= b(\phi(0)) + \int_0^\vartheta \nabla b(\phi(0)) \cdot \nabla \phi(s) ds \\ &\quad + \int_0^\vartheta (\nabla b(\phi(s)) - \nabla b(\phi(0))) \cdot \nabla \phi(s) ds \end{aligned}$$

und für  $\vartheta = 1$  erhält man:

$$\begin{aligned} b(E_i) &= b(y) + \int_0^1 \nabla b(y) \cdot (E_i - y) ds \\ &\quad + \int_0^1 (\nabla b(y + s \cdot (E_i - y)) - \nabla b(y)) \cdot (E_i - y) ds \\ \Rightarrow b(E_i) - b(y) &= \nabla b(y) \cdot (E_i - y) \\ &\quad + \int_0^1 (\nabla b(y + s \cdot (E_i - y)) - \nabla b(y)) \cdot (E_i - y) ds \end{aligned}$$

$$\begin{aligned}
\Rightarrow |b(E_i) - b(y)| &\leq |\nabla b(y) \cdot (E_i - y)| \\
&\quad + \int_0^1 \underbrace{\|\nabla b(y + s \cdot (E_i - y)) - \nabla b(y)\|}_{\substack{\leq L \|s \cdot (E_i - y)\| \\ \leq sL \|E_i - y\| \\ \leq sL \cdot \text{diam}(\triangle E_0 E_1 E_2)}} \cdot \underbrace{\|E_i - y\|}_{\leq \text{diam}(\triangle E_0 E_1 E_2)} \, ds \\
&\leq |\nabla b(y) \cdot (E_i - y)| + L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2 \cdot \int_0^1 s \, ds \\
&= |\nabla b(y) \cdot (E_i - y)| + \frac{1}{2} L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2
\end{aligned}$$

Damit lässt sich nun die Fehlerabschätzung bei der stückweise linearen Interpolation durch baryzentrische Koordinaten unter der Voraussetzung eines lipschitzstetigen Gradienten der Distanzfunktion beweisen. Es gilt:

$$\begin{aligned}
|p(y) - b(y)| &= \left| \sum_{i=0}^2 \lambda_i b(E_i) - \underbrace{\sum_{i=0}^2 \lambda_i}_{=1} b(y) \right| \\
&= \left| \sum_{i=0}^2 \lambda_i (b(E_i) - b(y)) \right| \\
&\leq \left| \sum_{i=0}^2 \lambda_i \left( |\nabla b(y) \cdot (E_i - y)| + \frac{1}{2} L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2 \right) \right| \\
&\leq \left| \nabla b(y) \cdot \sum_{i=0}^2 \lambda_i (E_i - y) \right| + \underbrace{\sum_{i=0}^2 \lambda_i}_{=1} \frac{1}{2} L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2 \\
&= \left| \nabla b(y) \cdot \left( \underbrace{\sum_{i=0}^2 \lambda_i E_i}_{=y} - \underbrace{\sum_{i=0}^2 \lambda_i y}_{=1} \right) \right| + \frac{1}{2} L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2 \\
&= \frac{1}{2} L \cdot \text{diam}(\triangle E_0 E_1 E_2)^2
\end{aligned}$$

■

**Satz 3.21.** *Betrachte die stückweise lineare Interpolation wie im Satz 3.20 mit  $\text{diam}(\triangle E_0 E_1 E_2) = \sqrt{2}\rho$  und*

$$|p(y, A) - b(y, A)| \leq L\rho^2. \quad (3.10)$$

Ferner sei

$$A := \{y \in \mathbb{R}^n \mid b(y, A) \leq 0\},$$

$$A_\rho := \{y \in \mathbb{R}^n \mid p(y, A) \leq \tilde{L}\rho^2\}$$

mit  $\tilde{L} \geq L$  gegeben. Dann gilt:

- i)  $A_\rho \subset A + \hat{L}\rho^2 \cdot B_1(0) \quad \forall \rho > 0,$   
mit  $\hat{L} = \tilde{L} + L$  und  $B_1(0)$  als Einheitskreis um den Ursprung.
- ii)  $A \subset A_\rho,$   
und insbesondere:  $d_{\text{H}}(A, A_\rho) \leq \hat{L}\rho^2.$

*Beweis.*

- i) Sei  $y \in A_\rho$ :

$$\Rightarrow p(y, A) \leq \tilde{L}\rho^2$$

$$\Rightarrow b(y, A) \stackrel{(3.10)}{\leq} p(y, A) + L\rho^2 \leq \tilde{L}\rho^2 + L\rho^2 = \hat{L}\rho^2$$

- a) falls  $y \in A$ :  $\Rightarrow d(y, A) = 0$

- b) falls  $y \notin A$ :

$$\Rightarrow b(y, A) = d(y, A) - \underbrace{d(y, A^{\text{C}})}_{=0} \leq \hat{L}\rho^2$$

$$\Rightarrow d(y, A) \leq \hat{L}\rho^2$$

Aus a) und b) folgt

$$d(A_\rho, A) \leq \hat{L}\rho^2,$$

mit anderen Worten:  $A_\rho \subset A + \hat{L}\rho^2 \cdot B_1(0).$

- ii) Sei  $y \in A$ :

$$\Rightarrow b(y, A) = -d(y, A^{\text{C}}) \leq 0$$

$$\Rightarrow p(y, A) \stackrel{(3.10)}{\leq} b(y, A) + L\rho^2 \leq L\rho^2$$

$$\Rightarrow y \in A_\rho$$

$$\Rightarrow d(y, A_\rho) = 0$$

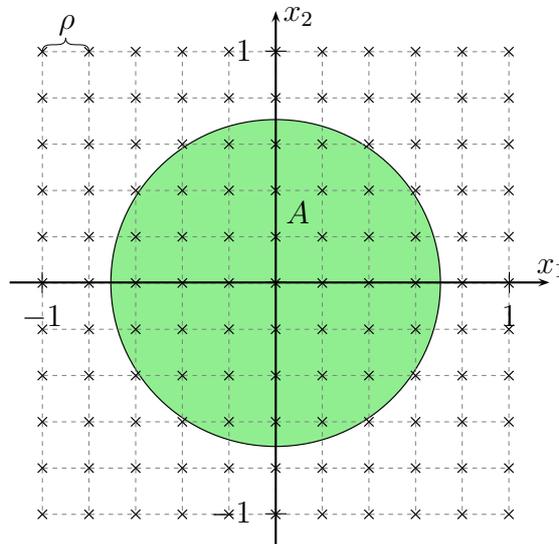
$$\Rightarrow d(A, A_\rho) = 0, \quad \text{d.h.} \quad A \subset A_\rho.$$

Zusammen mit i) folgt somit

$$d_H(A, A_\rho) \leq \hat{L}\rho^2.$$

■

**Beispiel 3.22.** Betrachte einen Kreis  $A$  im  $\mathbb{R}^2$  um den Ursprung mit Radius  $1/\sqrt{2}$ . Für die Berechnung der Mengen durch Distanzfunktionen wird ein Gitter der Größe  $[-1, 1] \times [-1, 1]$  mit der Gitterweite  $\rho = 0.2$  angelegt und an den Gitterpunkten die vorzeichenbehaftete Distanzfunktion ( $b(y, A) = \|y\| - 1/\sqrt{2}$ ) ausgewertet, siehe Abbildung 3.12.

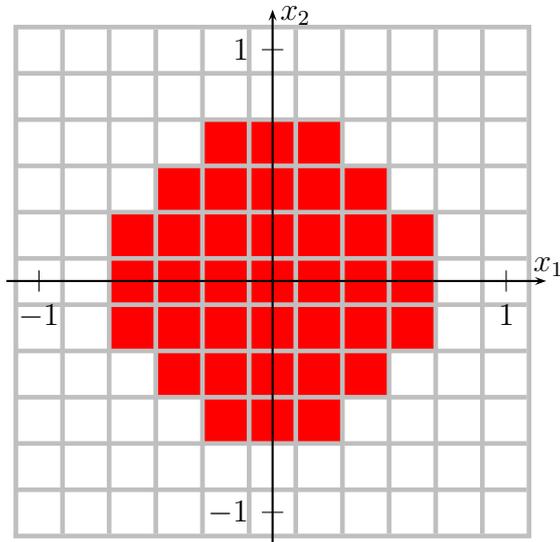


**Abbildung 3.12:** Auswertung der Distanzfunktion findet nur an den  $11 \times 11$  Gitterpunkten statt

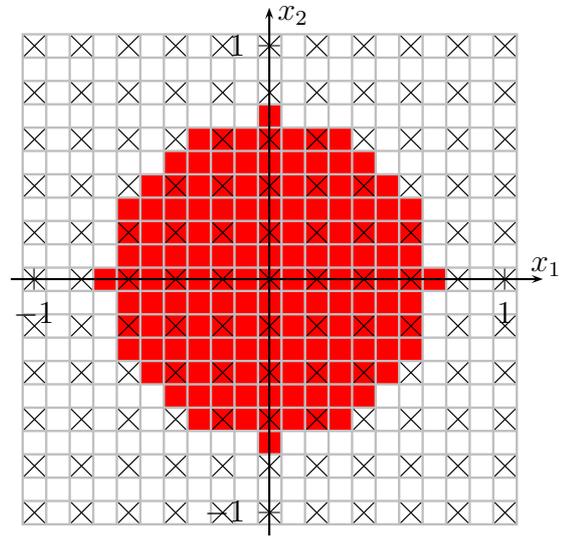
Es ergeben sich  $11 \times 11$  Punkte mit den zugehörigen Werten der Distanzfunktion. Zur Darstellung der Ergebnisse wird in den folgenden Abbildungen jeder dieser Punkte durch ein Pixel bzw. Quadrat der Breite  $\rho_{\text{limit}}$  repräsentiert. Für Punkte mit  $b(\cdot, A) \leq 0$  wird das Quadrat farbig markiert und gehört somit zur Menge  $A$ . Auf diese Weise erhält man eine diskrete Approximation der Menge, die später in Kapitel 4.2 noch genauer untersucht wird.

Abbildung 3.13 zeigt die diskrete Approximation der Menge  $A$  durch  $11 \times 11$  Punkte an denen die Distanzfunktion ausgewertet wurde. Die Abbildung 3.14 veranschaulicht die diskrete Approximation der Menge nach Halbierung der Gitterweite  $\rho_{\text{limit}}$  (auf  $\rho_{\text{limit}} = 0.1$ ), also durch  $21 \times 21$  Punkte. Rechtecke, die mit „x“ markiert sind, stellen dabei die ursprünglichen  $11 \times 11$  Punkte dar, an denen die Distanzfunktion

bereits ausgewertet wurde. Die Distanzfunktionswerte der restlichen Punkte wurde mit linearer Interpolation durch baryzentrische Koordinaten berechnet.



**Abbildung 3.13:** diskrete Approximation der Menge  $A$  durch  $11 \times 11$  Gitterpunkte ( $\rho = \rho_{\text{linint}} = 0.2$ )



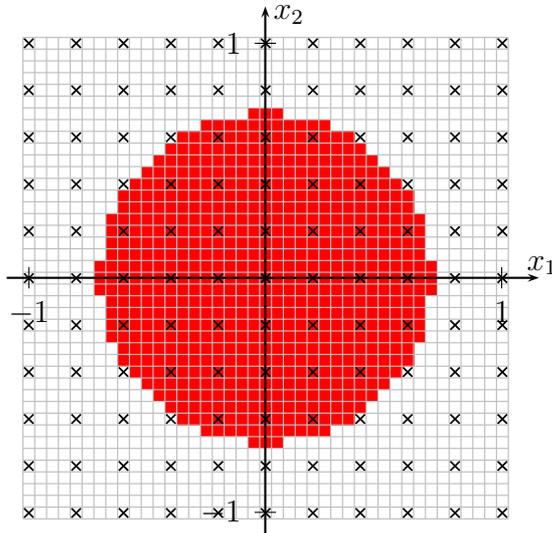
**Abbildung 3.14:** diskrete Approximation der Menge nach Interpolation auf  $21 \times 21$  Punkte ( $\rho_{\text{linint}} = 0.1$ )

Die Abbildungen 3.15 - 3.18 zeigen die diskrete Approximation der Menge nach weiteren Halbierungen der Gitterweite  $\rho_{\text{linint}}$ . In diesen Abbildungen fand wie zuvor keine weitere Auswertung der Distanzfunktion statt und die Distanzfunktionswerte für die neuen Gitterpunkte wurden mit linearer Interpolation durch baryzentrische Koordinaten berechnet.

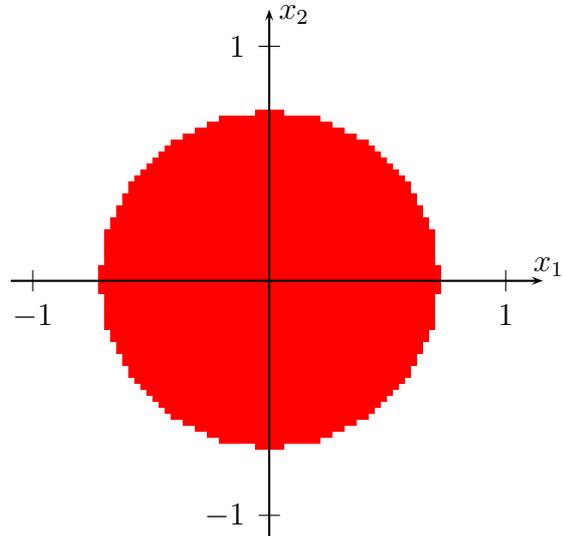
Wie man sieht wird die diskrete Approximation der Menge genauer, je kleiner die Gitterweite  $\rho_{\text{linint}}$  wird. Die zugehörigen Hausdorff-Abstände zwischen den approximierten Mengen  $\tilde{A}$  und dem Kreis  $A$  mit Radius  $1/\sqrt{2}$  sind der Tabelle 3.1 zu entnehmen.

Man erkennt, dass die Menge durch lineare Interpolation nicht beliebig genau approximiert werden kann, da der zugehörige Hausdorff-Abstand ab einer bestimmten Gitterweite  $\rho_{\text{linint}}$  nicht mehr abnimmt. Dies ist auch klar, wenn man bedenkt, dass durch die lineare Interpolation keine neuen Informationen über die Menge entstehen, sondern nur die vorhandenen Informationen aus den ursprünglichen  $11 \times 11$  Gitterpunkten, für die die Distanzfunktion ausgewertet wurde, verarbeitet werden.

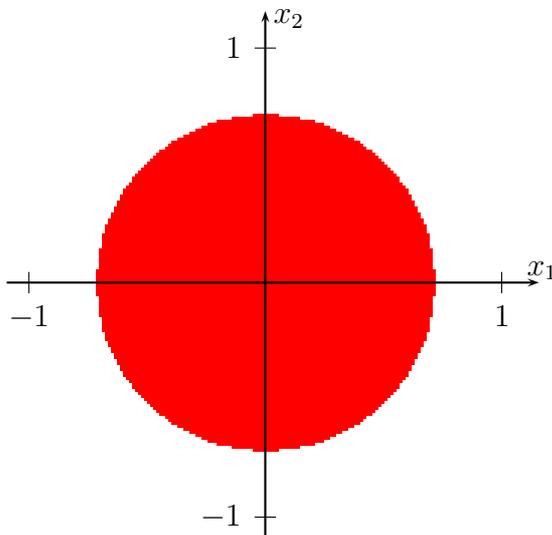
In Tabelle 3.2 sind die Hausdorff-Abstände der approximierten Mengen, berechnet durch lineare Interpolation, für verschiedene Gitterweiten  $\rho$  aufgelistet. Die Werte wurden dabei so gewählt, dass bei der nächsten Halbierung der Gitterweite  $\rho_{\text{linint}}$



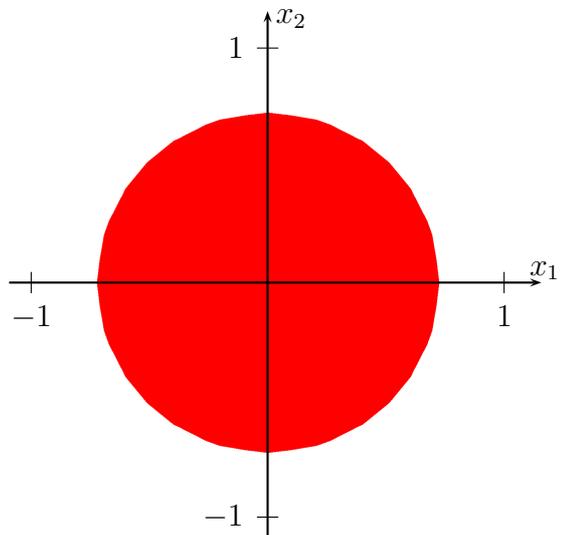
**Abbildung 3.15:** Interpolation auf  $41 \times 41$  Punkte ( $\rho_{\text{linint}} = 0.05$ )



**Abbildung 3.16:** Interpolation auf  $81 \times 81$  Punkte ( $\rho_{\text{linint}} = 0.025$ )



**Abbildung 3.17:** Interpolation auf  $161 \times 161$  Punkte ( $\rho_{\text{linint}} = 0.0125$ )



**Abbildung 3.18:** Interpolation auf  $1281 \times 1281$  Punkte ( $\rho_{\text{linint}} = 0.0015625$ )

$\rho$	$\rho_{\text{linint}}$	$d_{\text{H}}(\tilde{A}, A)$
0.2	0.2	1.41421356e-01
0.2	0.1	7.07106781e-02
0.2	0.05	3.53553391e-02
0.2	0.025	1.76776695e-02
0.2	0.0125	8.83883476e-03
0.2	0.006125	6.32969087e-03
0.2	0.003125	5.82144973e-03
0.2	0.0015625	5.57497819e-03
0.2	0.00078125	5.57497819e-03

**Tabelle 3.1:** Hausdorff-Abstand zwischen den approximierten Mengen  $\tilde{A}$ , mit den Gitterweiten  $\rho_{\text{linint}}$ , und der Menge  $A$

der Hausdorff-Abstand zur Menge  $A$  nicht mehr halbiert wurde. In der Tabelle 3.1 für  $\rho = 0.2$  ist dies bei  $\rho_{\text{linint}} = 0.0125$  der Fall.

$\rho$	$\rho_{\text{linint}}$	$d_{\text{H}}(\tilde{A}, A)$
0.2	0.0125	8.83883476e-03
0.1	0.003125	2.20970869e-03
0.05	0.00078125	5.52427173e-04
0.025	0.0001953125	1.38106793e-04
0.0125	0.000048828125	3.45266983e-05

**Tabelle 3.2:** Hausdorff-Abstand zwischen den approximierten Mengen  $\tilde{A}$ , berechnet durch lineare Interpolation, und der Menge  $A$

Man erkennt, dass die numerischen Ergebnisse gut mit den Resultaten aus der Fehleranalyse übereinstimmen und bei Halbierung der Gitterweite  $\rho$  der Hausdorff-Abstand etwa ein viertel des vorherigen beträgt. Für die Distanzfunktion eines Kreises lässt sich dies auch durch folgende theoretische Untersuchungen des Gradienten begründen.

Der Gradient für die vorzeichenbehaftete Distanzfunktion ( $b(y, A) = \|y\| - r$ ) eines beliebigen Kreises um den Ursprung mit Radius  $r$  lautet

$$\nabla b(y, A) = \begin{pmatrix} \frac{y_1}{\|y\|} \\ \frac{y_2}{\|y\|} \end{pmatrix} \quad (\|y\| \neq 0).$$

Hier erkennt man auch den Ursprung als einzige Nichtdifferenzierbarkeitsstelle

dieser Distanzfunktion. Um die Notation zu vereinfachen wird im Folgenden  $b(y)$  für  $b(y, A)$  geschrieben. Betrachte nun die Funktion  $\phi$  definiert durch

$$\phi(\vartheta) := x + \vartheta(y - x) \quad \text{für } \vartheta \in [0, 1],$$

mit  $\phi(0) = x$ ,  $\phi(1) = y$ , ( $x, y \in \mathbb{R}^2$ ) und  $\nabla\phi(\vartheta) = (y - x)$ . Damit lässt sich die Differenz zweier Gradienten wie folgt abschätzen:

$$\begin{aligned} \nabla b(\phi(\vartheta)) &= \nabla b(\phi(0)) + \int_0^\vartheta \frac{d}{ds} (\nabla b \circ \phi)(s) ds \\ &= \nabla b(\phi(0)) + \int_0^\vartheta \nabla^2 b(\phi(s)) \cdot \underbrace{\nabla\phi(s)}_{=(y-x)} ds \\ \Rightarrow \nabla b(\phi(\vartheta)) - \nabla b(\phi(0)) &= \int_0^\vartheta \nabla^2 b(\phi(s)) ds \cdot (y - x) \end{aligned}$$

und für  $\vartheta = 1$  erhält man:

$$\begin{aligned} \nabla b(y) - \nabla b(x) &= \int_0^1 \nabla^2 b(\phi(s)) ds \cdot (y - x) \\ \Rightarrow \|\nabla b(y) - \nabla b(x)\| &\leq \int_0^1 \|\nabla^2 b(\phi(s))\| ds \cdot \|y - x\| \end{aligned} \quad (3.11)$$

Die zugehörige Hesse-Matrix der Distanzfunktion lautet

$$\nabla^2 b(z) = H(b(z)) = \begin{pmatrix} \frac{1}{\|z\|} - \frac{z_1^2}{\|z\|^3} & -\frac{z_1 z_2}{\|z\|^3} \\ -\frac{z_1 z_2}{\|z\|^3} & \frac{1}{\|z\|} - \frac{z_2^2}{\|z\|^3} \end{pmatrix} \quad (\|z\| \neq 0).$$

Mit  $\|z\| \geq \tilde{\delta}$  (für ein  $\tilde{\delta} > 0$ ), lässt sich durch eine einfache aber längliche Rechnung die Frobeniusnorm der Matrix in einer beschränkten Umgebung von  $z$  durch

$$\|H(b(z))\| \leq \frac{1}{\tilde{\delta}} \cdot C$$

abschätzen. Mit  $L = \frac{1}{\tilde{\delta}} \cdot C$  erhält man somit aus (3.11)

$$\|\nabla b(y) - \nabla b(x)\| \leq L \cdot \|y - x\|,$$

und damit die Voraussetzungen für Satz 3.20 und 3.21.

# 4 Realisierung des mengenwertigen Euler-Verfahrens

Es gibt nur wenige Kontrollsysteme, für die die erreichbare Menge analytisch berechnet werden kann, dies sind meist eindimensionale oder lineare Kontrollsysteme mit additivem Kontrollterm. Somit muss im Allgemeinen die erreichbare Menge numerisch berechnet werden [Häc96, Preface]. Um dies realisieren zu können, ist jedoch eine Diskretisierung des Zeitparameters und des Zustandsraums nötig. Als erstes wird in diesem Kapitel daher die Zeit und Zustandsdiskretisierung betrachtet und die auftretenden Probleme, vor allem bei der Diskretisierung des Zustandsraums, untersucht. Anschließend wird das mengenwertige Euler-Verfahren hergeleitet und die Realisierung mit Distanzfunktionen dargestellt.

## 4.1 Zeitdiskretisierung

Die Diskretisierung des Zeitparameters  $t$  erfolgt durch das Auswechseln der Differentialinklusion (2.2), (2.3) gegen eine Folge von Differenzeninklusionen. Der Einfachheit halber werden diese als Diskretisierung der Differentialinklusion bezeichnet.

**Definition 4.1.** Betrachte die Differentialinklusion (2.2), (2.3). Gegeben sei  $N \in \mathbb{N}$  und

$$h := \frac{T - t_0}{N},$$
$$t_k := t_0 + kh, \quad (k = 0, \dots, N),$$

dann lautet die *Diskretisierung der Differentialinklusion (DDI)*

$$\eta(t_{k+1}) = \eta_{k+1} \in G_k^N(h; \eta_k) \quad (k = 0, \dots, N - 1), \quad (4.1)$$

$$\eta_0 = X_0, \quad (4.2)$$

mit einer mengenwertigen Abbildung für das diskrete System

$$G_k^N : [0, T - t_0] \times \mathbb{R}^n \Rightarrow \mathbb{R}^n.$$

Jede Folge von  $N + 1$  Vektoren  $(\eta_k)_{k=0, \dots, N} \subset \mathbb{R}^n$ , die (4.1) und (4.2) erfüllt, heißt *Lösung der (DDI)*. Entsprechend dieser Notation bezeichnet man die *Menge aller Lösungen der (DDI)* mit

$$\mathcal{X}_N(T; t_0, \eta_0) := \left\{ (\eta_k)_{k=0, \dots, N} \subset \mathbb{R}^n \mid (\eta_k)_{k=0, \dots, N} \text{ erfüllt (4.1) und (4.2)} \right\}$$

und die *erreichbare Menge der (DDI)* mit

$$\mathcal{R}_N(T, t_0, \eta_0) := \left\{ y \in \mathbb{R}^n \mid \exists (\eta_k)_{k=0, \dots, N} \in \mathcal{X}_N(T; t_0, \eta_0) \text{ mit } y = \eta_N \right\}.$$

Auf diese Weise lassen sich durch Diskretisierung der Differentialinklusion verschiedene numerische Verfahren realisieren. Für das explizite Euler-Verfahren erhält man beispielsweise [LV98, Seite 154]

$$G_k^N(h; \eta_k) = \eta_k + hF(t_k, \eta_k)$$

und für das Euler-Cauchy-Verfahren [LV98, Seite 180]

$$G_k^N(h; \eta_k) = \eta_k + \frac{h}{2} \left\{ \xi + F(t_{k+1}, \eta_k + h\xi) \mid \xi \in F(t_k, \eta_k) \right\}.$$

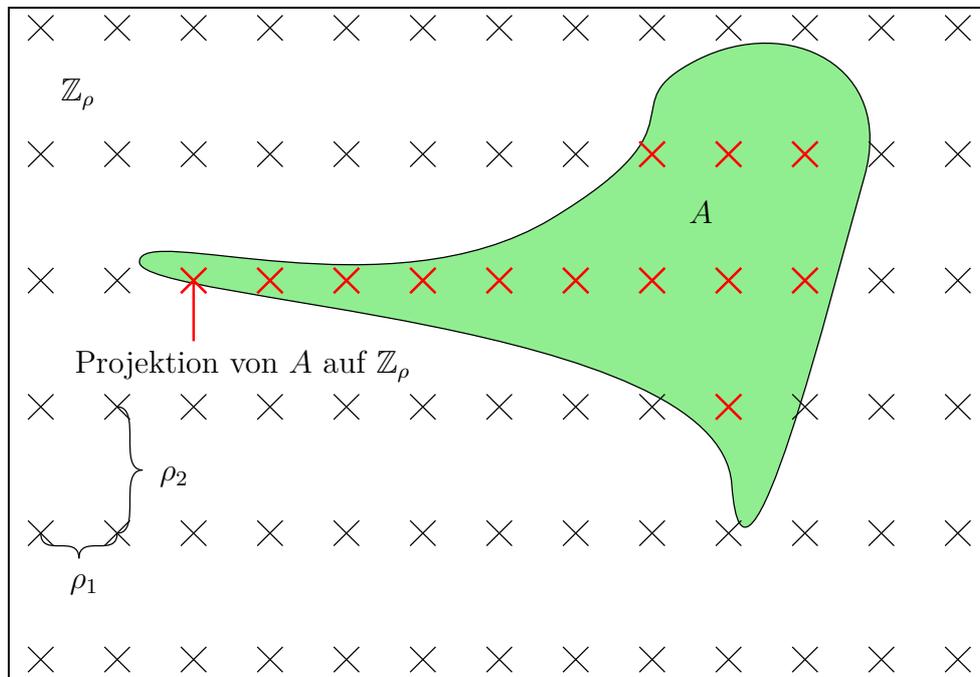
Entsprechend lassen sich (DDIs) für weitere Runge-Kutta-Verfahren formulieren.

Um erreichbare Mengen mit solchen Verfahren numerisch berechnen, darstellen und abspeichern zu können, stößt man jedoch schnell auf das Problem, dass die Mengen im Allgemeinen aus unendlich vielen Punkten bestehen. Daher ist eine Approximation der Menge durch endlich viele Punkte nötig, die durch Diskretisierung des Zustandsraums erreicht wird.

## 4.2 Zustandsdiskretisierung

Die Diskretisierung des Zustandsraums  $Y$  erfolgt durch Projektion der Teilmengen von  $Y$  auf  $\mathbb{Z}_\rho^n$ . Die Menge  $\mathbb{Z}_\rho^n$  besteht dabei aus Gitterpunkten des  $\mathbb{R}^n$  zu einem entsprechenden Parameter  $\rho \in \mathbb{R}^n$ . Mit  $\rho^t = (\rho_1, \rho_2, \dots, \rho_n)$  bezeichnet man den Diskretisierungsparameter des Zustandsraums, d.h.  $\rho_i$  ist der Diskretisierungsparameter der jeweiligen Komponente des Koordinatensystems. Abbildung 4.1 dient

zur Veranschaulichung der Projektion einer Menge  $A \subset \mathbb{R}^2$  auf ein Gitter  $\mathbb{Z}_\rho^2$  mit  $\rho^t = (\rho_1, \rho_2)$ , wobei rot markierte Gitterpunkte die diskrete Approximation der Menge  $A$  darstellen.



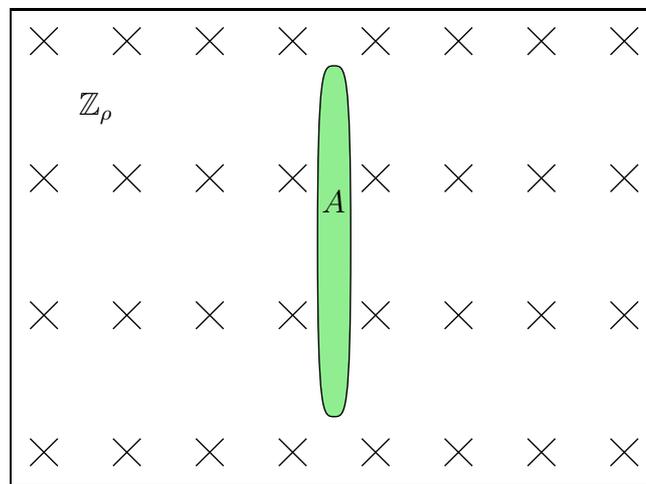
**Abbildung 4.1:** Diskretisierung des Zustandsraums

Jede nichtleere, beschränkte Teilmenge von  $Y$  kann somit, mit Hilfe der Projektion auf  $\mathbb{Z}_\rho^n$ , durch endlich viele Punkte approximiert werden. Die Notation aus Definition 4.1 wird entsprechend erweitert und man schreibt im Folgenden  $\mathcal{X}_{N,\rho}$  für die Menge aller diskreten Lösungen und  $\mathcal{R}_{N,\rho}$  für die diskrete erreichbare Menge der (DDI).

Für die Berechnung erreichbarer Mengen von Kontrollsystemen gibt es nun verschiedene Ansätze. Beispielsweise der „Boxansatz“, bei dem für jeden Gitterpunkt aus  $\mathbb{Z}_\rho^n$ , interpretiert als Box der Breite  $\rho$ , in der Menge  $A$  ein zuvor gewähltes numerisches Verfahren angewendet wird, um damit Gitterpunkte zu bestimmen, die im nächsten Iterationsschritt zur Menge gehören, siehe auch [BR07]. Ein anderer Ansatz, der hier weiter verfolgt wird, ist die Berechnung erreichbarer Mengen durch Distanzfunktionen.

### 4.2.1 Aufblähung einer Menge

Durch Diskretisierung des Zustandsraums tritt bei der Berechnung erreichbarer Mengen allerdings ein Problem auf, das in Abbildung 4.2 visualisiert ist. Die Projektion einer Menge auf ein vorher bestimmtes Gitter  $\mathbb{Z}_\rho$  liefert unter Umständen eine leere Menge. Die Ursache dafür ist, dass die Menge zwischen den Gitterpunkten liegt, auf die sie projiziert werden soll. Die iterative Berechnung der erreichbaren Menge durch ein numerisches Verfahren, das auf den Gitterpunkten der diskreten Approximation basiert, liefert in diesem Fall fälschlicherweise eine leere Menge als Ergebnis.



**Abbildung 4.2:** Projektion der Menge  $A$  auf  $\mathbb{Z}_\rho$  liefert leere Menge

Eine Möglichkeit dieses Problem zu lösen wird in Abbildung 4.3 dargestellt. Durch eine geeignete Verschiebung des Gitters  $\mathbb{Z}_\rho$  liegen einige Gitterpunkte innerhalb der Menge und die Projektion auf das Gitter liefert keine leere Menge mehr.

Eine weitere Möglichkeit, die in dieser Arbeit näher untersucht und angewendet wird, ist das sogenannte *Aufblähen* einer Menge, siehe Abbildung 4.4. Dabei wird die Menge künstlich so weit vergrößert, bis die Projektion auf das Gitter  $\mathbb{Z}_\rho$  keine leere Menge mehr ergibt.

**Satz 4.2** (Aufblähung einer Menge [BGX11, Lemma 2.4]). *Sei  $\varrho > 0$  und  $A \subset \mathbb{R}^n$  abgeschlossen und nicht leer. Die diskrete Approximation einer Menge  $A$  auf dem Gitter  $\mathbb{Z}_\rho$  sei gegeben durch*

$$A_\varrho := (A + \varrho \cdot B_1(0)) \cap \mathbb{Z}_\rho,$$

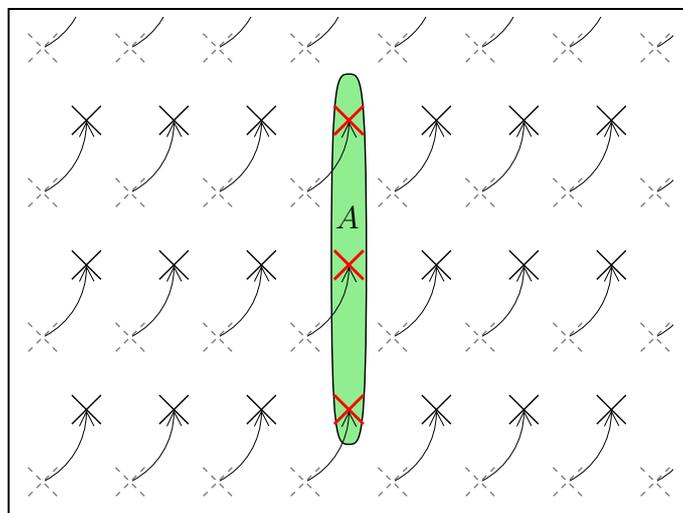


Abbildung 4.3: Verschiebung des Gitters  $\mathbb{Z}_\rho$

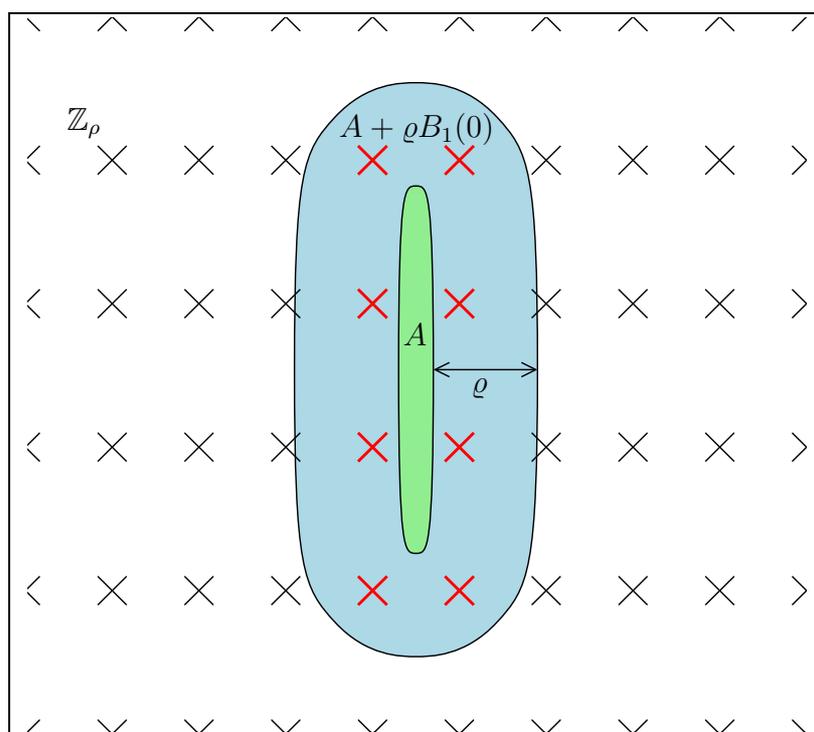


Abbildung 4.4: Aufblähen der Menge  $A$  mit Aufblähungsparameter  $\rho$

wobei  $\varrho$  den Aufblähungsparameter und  $B_1(0)$  die Einheitskugel um den Ursprung bezeichnet. Dann gilt die folgende Abschätzung:

$$d_H(A, A_\varrho) \leq \varrho$$

*Beweis.* Siehe [BGX11, Seite 8]. ■

Das Aufblähen einer nicht leeren Menge vor der Projektion auf das Gitter  $\mathbb{Z}_\rho$  stellt also sicher, dass die diskrete Approximation berechneter Mengen nicht leer ist. Das Aufblähen lässt sich dabei mit Hilfe von Distanzfunktionen stark vereinfachen.

**Satz 4.3** (Vgl. [Rod10, Seite 20]). *Sei  $\varrho > 0$  und  $A \subset \mathbb{R}^n$  abgeschlossen und nichtleer. Für die Distanzfunktion eines Punktes  $y \in \mathbb{R}^n$  zur aufgeblähten Menge  $A + \varrho \cdot B_1(0)$  gilt:*

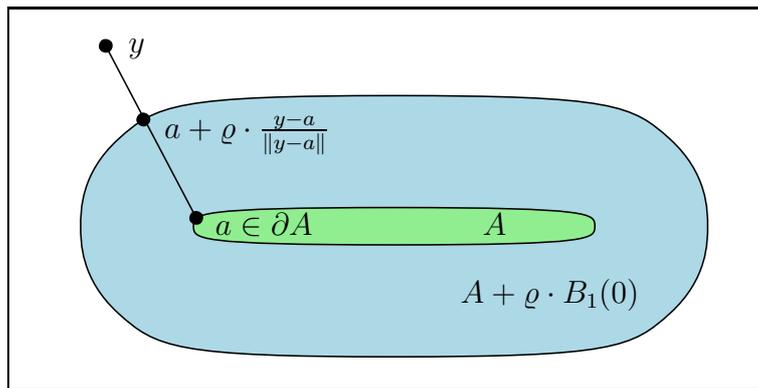
$$b(y, A + \varrho \cdot B_1(0)) \leq b(y, A) - \varrho \quad (4.3)$$

Falls  $y$  außerhalb der aufgeblähten Menge liegt oder  $A$  konvex ist, dann gilt sogar:

$$b(y, A + \varrho \cdot B_1(0)) = b(y, A) - \varrho \quad (4.4)$$

*Beweis.* Betrachte die drei möglichen Fälle (I:  $y \notin A + \varrho \cdot B_1(0)$ ), (II:  $y \in A + \varrho \cdot B_1(0), y \notin A$ ), und (III:  $y \in A + \varrho \cdot B_1(0), y \in A$ ).

I: Sei  $y$  außerhalb der aufgeblähten Menge  $A + \varrho \cdot B_1(0)$  und  $a \in \partial A$  der nächste Punkt zu  $y$ , wie in Abbildung 4.5 für  $n = 2$ .



**Abbildung 4.5:** Skizze zum Beweis von Satz 4.3 für  $y \notin (A + \varrho \cdot B_1(0))$

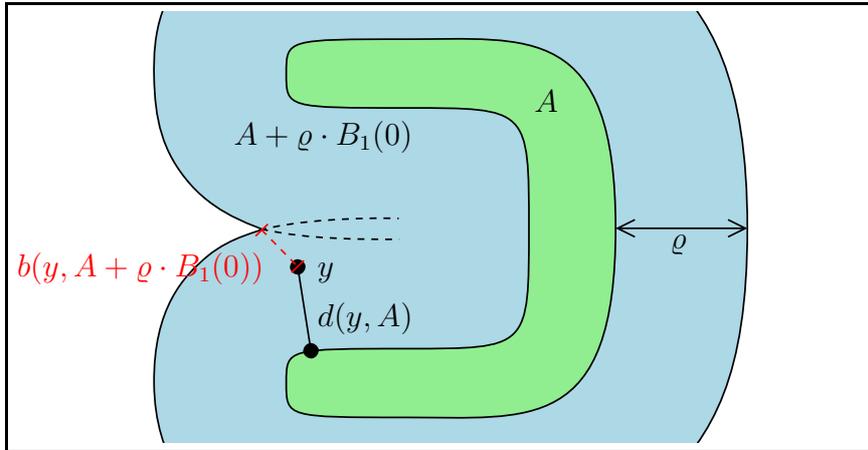
Der Abstand von  $y$  zur aufgeblähten Menge ist

$$\begin{aligned}
 b(y, A + \varrho \cdot B_1(0)) &= d(y, A + \varrho \cdot B_1(0)) - \underbrace{d\left(y, (A + \varrho \cdot B_1(0))^C\right)}_{=0} \\
 &= \left\| y - \left( a + \varrho \cdot \frac{y - a}{\|y - a\|} \right) \right\| \\
 &= \left\| \underbrace{\left( 1 - \frac{\varrho}{\|y - a\|} \right)}_{\geq 0} (y - a) \right\| \\
 &= \left( 1 - \frac{\varrho}{\|y - a\|} \right) \|y - a\| \\
 &= \|y - a\| - \varrho \\
 &= d(y, A) - \varrho \\
 &= d(y, A) - \underbrace{d(y, A^C)}_{=0} - \varrho \\
 &= b(y, A) - \varrho.
 \end{aligned}$$

II: i) Sei  $y \in (A + \varrho \cdot B_1(0))$  und  $y \notin A$ . Der Abstand von  $y$  zur aufgeblähten Menge beträgt

$$\begin{aligned}
 b(y, A + \varrho \cdot B_1(0)) &= \underbrace{d(y, A + \varrho \cdot B_1(0))}_{=0} - \underbrace{d\left(y, (A + \varrho \cdot B_1(0))^C\right)}_{\geq \varrho - d(y, A)} \\
 &\leq d(y, A) - \varrho \\
 &= d(y, A) - \underbrace{d(y, A^C)}_{=0} - \varrho \\
 &= b(y, A) - \varrho.
 \end{aligned}$$

ii) Sei  $A$  konvex,  $y \in A + \varrho \cdot B_1(0)$  und  $y \notin A$ . Ferner sei  $a$  der nächste Punkt zu  $y$  aus dem Rand von  $A$  und  $\left( a + \varrho \cdot \frac{y - a}{\|y - a\|} \right)$  der nächste Punkt zu  $y$  aus dem Abschluss von  $(A + \varrho \cdot B_1(0))^C$ . Der Abstand von  $y$  zur



**Abbildung 4.6:** Skizze zum Beweis von Satz 4.3 für  $y \in (A + \varrho \cdot B_1(0))$ ,  $y \notin A$

aufgeblähten Menge beträgt

$$\begin{aligned}
 b(y, A + \varrho \cdot B_1(0)) &= \underbrace{d(y, A + \varrho \cdot B_1(0))}_{=0} - d(y, (A + \varrho \cdot B_1(0))^C) \\
 &= - \left\| y - \left( a + \varrho \cdot \frac{y - a}{\|y - a\|} \right) \right\| \\
 &= - \left\| \underbrace{\left( 1 - \frac{\varrho}{\|y - a\|} \right)}_{\leq 0} (y - a) \right\| \\
 &= \left( 1 - \frac{\varrho}{\|y - a\|} \right) \|y - a\| \\
 &= d(y, A) - \underbrace{d(y, A^C)}_{=0} - \varrho \\
 &= b(y, A) - \varrho.
 \end{aligned}$$

III: i) Sei  $y \in (A + \varrho \cdot B_1(0))$  und  $y \in A$ . Der Abstand von  $y$  zur aufgeblähten Menge beträgt

$$\begin{aligned}
 b(y, A + \varrho \cdot B_1(0)) &= \underbrace{d(y, A + \varrho \cdot B_1(0))}_{=0} - \underbrace{d(y, (A + \varrho \cdot B_1(0))^C)}_{\geq \varrho + d(y, A^C)} \\
 &\leq -d(y, A^C) - \varrho
 \end{aligned}$$

$$\begin{aligned}
&= \underbrace{d(y, A)}_{=0} - d(y, A^C) - \varrho \\
&= b(y, A) - \varrho.
\end{aligned}$$

- ii) Sei  $A$  konvex,  $y \in A + \varrho \cdot B_1(0)$  und  $y \in A$ . Ferner sei  $a$  der nächste Punkt zu  $y$  aus dem Abschluss von  $A^C$  und  $\left(a + \varrho \cdot \frac{a-y}{\|a-y\|}\right)$  der nächste Punkt zu  $y$  aus dem Abschluss von  $(A + \varrho \cdot B_1(0))^C$ . Für den Abstand von  $y$  zur aufgeblähten Menge gilt:

$$\begin{aligned}
b(y, A + \varrho \cdot B_1(0)) &= \underbrace{d(y, A + \varrho \cdot B_1(0))}_{=0} - d\left(y, (A + \varrho \cdot B_1(0))^C\right) \\
&= -\left\|y - \left(a + \varrho \cdot \frac{a-y}{\|a-y\|}\right)\right\| \\
&= -\left\|-(a-y) \underbrace{\left(1 + \frac{\varrho}{\|a-y\|}\right)}_{\geq 0}\right\| \\
&= -\left(1 + \frac{\varrho}{\|a-y\|}\right) \|a-y\| \\
&= -\|a-y\| - \varrho \\
&= -\|y-a\| - \varrho \\
&= \underbrace{d(y, A)}_{=0} - d(y, A^C) - \varrho \\
&= b(y, A) - \varrho.
\end{aligned}$$

■

*Bemerkung.* Zur einfacheren Notation wird im Folgenden stets  $n = 2$  und ein quadratisches Gitter, also  $\rho_1 = \rho_2$ , betrachtet. Ferner wird die Gitterweite  $\rho$  auch als Aufblähungsparameter benutzt, wodurch  $A_\rho$  nun die aufgeblähte Menge  $A$ , projiziert auf das Gitter  $\mathbb{Z}_\rho$ , darstellt. Ein Gitter mit Gitterweite  $\rho$ , wird im Folgenden als  $\rho$ -Gitter bezeichnet, und für iterativ berechnete Mengen, bei denen der jeweilige Iterationsschritt im Index steht, wird der Aufblähungsparameter als oberer Index geschrieben, d.h.  $X_i^{h,\rho}$  bezeichnet die iterativ berechnete diskrete Menge nach dem ( $i$ )-ten Schritt mit Aufblähung und Projektion auf das  $\rho$ -Gitter.

## 4.3 Mengenwertiges Euler-Verfahren

Ein einfaches numerisches Einschrittverfahren zum Lösen von Differentialgleichungen ist das aus der Numerik bekannte Euler-Verfahren. In diesem Abschnitt wird das mengenwertige Euler-Verfahren und die zugehörige Konvergenzaussage formuliert.

Anhand der Euler-Diskretisierung für Kontrollsysteme [BG10, Abschnitt 2]

$$\begin{aligned}\eta_{i+1} &= \eta_i + h \cdot f(t_i, \eta_i, u_i) & (i = 0, \dots, N-1), \\ u_i &\in U, \\ \eta_0 &= x_0\end{aligned}$$

und entsprechender Euler-Diskretisierung für Differentialinklusionen

$$\begin{aligned}\eta_{i+1} &\in \eta_i + h \cdot F(t_i, \eta_i) & (i = 0, \dots, N-1), \\ F(t_i, \eta_i) &= \bigcup_{u_i \in U} \{f(t_i, \eta_i, u_i)\}, \\ \eta_0 &\in X_0 = \{x_0\},\end{aligned}$$

jeweils mit der Schrittweite  $h = \frac{T-t_0}{N}$ ,  $N \in \mathbb{N}$  und dem diskretisiertem Zeitparameter  $t_i = t_0 + i \cdot h$ , lässt sich die mengenwertige Formulierung des Euler-Verfahrens herleiten:

$$\begin{aligned}X_{i+1}^h &= \bigcup_{\eta_i \in X_i^h} \{\eta_i + h \cdot F(t_i, \eta_i)\} & (i = 0, \dots, N-1), \\ X_0^h &= \{x_0\}.\end{aligned}$$

Die erreichbare Menge des Euler-Verfahrens wird mit

$$\mathcal{R}_N(T, t_0, X_0) = X_N^h$$

bezeichnet.

**Satz 4.4** (Konvergenz des mengenwertigen Euler-Verfahrens [BGX11, Proposition 3.3]). *Sei  $F: I \times \mathbb{R}^n \Rightarrow \mathbb{R}^n$  Lipschitz in  $(t, x)$  mit kompakten, konvexen und nicht leeren Bildern. Dann gilt für das mengenwertige Euler-Verfahren und für alle  $N \in \mathbb{N}$  mit  $h = \frac{T-t_0}{N}$  die folgende Konvergenzaussage:*

$$d_H(\mathcal{R}(T, t_0, X_0), \mathcal{R}_N(T, t_0, X_0)) \leq C \cdot h$$

*Beweis.* Vgl. [DF89, Section 1, Theorem]. ■

## 4.4 Realisierung mit Distanzfunktionen

Zur Realisierung des mengenwertigen Euler-Verfahrens werden Mengenoperationen, wie skalares Vielfaches, Verschiebung und Vereinigung von Mengen, benötigt. Für Distanzfunktionen wurden diese Operationen bereits in Kapitel 3.1.2 eingeführt und bewiesen. Im Folgenden wird das mengenwertige Euler-Verfahren daher mit Hilfe von Distanzfunktionen realisiert.

Betrachte zunächst einen Eulerschritt der mengenwertigen Formulierung:

$$X_{i+1}^h = \bigcup_{\eta_i \in X_i^h} \{ \eta_i + h \cdot F(t_i, \eta_i) \}$$

Die erreichbare Menge  $X_{i+1}^h$  lässt sich gleichermaßen implizit mittels Niveaumengen der Distanzfunktion darstellen:

$$X_{i+1}^h = \{ y \in \mathbb{R}^n \mid b(y, X_{i+1}^h) \leq 0 \}$$

Die Iterationsvorschrift zur Berechnung der Menge  $X_{i+1}^h$  lässt sich nun mit Hilfe von Distanzfunktionen und den bekannten Mengenoperationen stark vereinfachen:

$$\begin{aligned} d(y, X_{i+1}^h) &= d\left(y, \bigcup_{\eta_i \in X_i^h} \{ \eta_i + h \cdot F(t_i, \eta_i) \}\right) \\ &\stackrel{(3.7)}{=} \min_{\eta_i \in X_i^h} d(y, \eta_i + h \cdot F(t_i, \eta_i)) \\ &\stackrel{(3.4)}{=} \min_{\eta_i \in X_i^h} d(y - \eta_i, h \cdot F(t_i, \eta_i)) \\ &\stackrel{(3.5)}{=} h \cdot \min_{\eta_i \in X_i^h} d\left(\frac{y - \eta_i}{h}, F(t_i, \eta_i)\right) \end{aligned}$$

bzw.

$$b(y, X_{i+1}^h) \leq h \cdot \min_{\eta_i \in X_i^h} b\left(\frac{y - \eta_i}{h}, F(t_i, \eta_i)\right)$$

bei Verwendung der vorzeichenbehafteten Distanzfunktion.

*Bemerkung.* Trotz der vereinfachten Iterationsvorschrift, lässt sich die erreichbare Menge am Computer auf diese Weise jedoch nicht berechnen. Das Problem stellt, wie bereits erwähnt, die Menge  $X_i^h$  dar, die am Computer praktisch nicht durchlaufen werden kann, da sie im Allgemeinen einer Menge mit unendlich vielen Elementen entspricht. Es ist daher erforderlich den Zustandsraum, wie in Kapitel 4.2, auf endlich viele Punkte zu reduzieren. Nach Diskretisierung des Zustandsraums entspricht ein Schritt des mengenwertigen Euler-Verfahrens somit den Auswertungen der Distanzfunktion an den Gitterpunkten.

Die praktische Umsetzung des mengenwertigen Euler-Verfahrens mit Distanzfunktionen, Projektion und  $\rho$ -Aufblähung von Mengen lautet nun wie folgt:

---

**Algorithmus 1** : mengenwertiges Euler-Verfahren mit Distanzfunktionen

---

1.  $X_0^{h,\rho} = \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, X_0) \leq \rho\}$ .
2. Für  $i = 0, 1, \dots, N - 1$ :
  - a) Berechne für jeden Gitterpunkt  $g_\rho \in \mathbb{Z}_\rho$  den Abstand zur Menge  $\tilde{X}_{i+1}^h$  durch

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right),$$

wobei  $\tilde{X}_{i+1}^h$  die Menge nach dem  $(i + 1)$ -ten Eulerschritt, ohne Aufblähung und Projektion auf das Gitter  $\mathbb{Z}_\rho$ , bezeichnet.

- b) Die diskrete Menge  $X_{i+1}^{h,\rho}$  erhält man nach Aufblähen und Projektion der Menge  $\tilde{X}_{i+1}^h$  durch

$$X_{i+1}^{h,\rho} = \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, \tilde{X}_{i+1}^h) \leq \rho\}.$$

3. Die diskrete erreichbare Menge lautet

$$\mathcal{R}_{N,\rho}(T, t_0, X_0) = X_N^{h,\rho}.$$


---

*Bemerkung.* Unter Berücksichtigung, dass für Gitterpunkte innerhalb der aufgeblähten Menge

$$b(g_\rho, \tilde{X}_{i+1}^h) \leq h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right)$$

gilt, wird im weiteren Verlauf der Arbeit und in der Implementierung des men-

genwertigen Euler-Verfahrens aus heuristischen Gründen der Einfachheit halber, wie in Schritt 2a) von Algorithmus 1,

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right)$$

verwendet.

Bevor eine Fehlerabschätzung für die Diskretisierung des Zustandsraums im mengenwertigen Euler-Verfahrens angegeben und bewiesen wird, werden die auftretenden Fehler in den einzelnen Schritten des Algorithmus angegeben:

$$1) \quad X_0^{h,\rho} = \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, X_0) \leq \rho\} \quad \text{Fehler: } \mathcal{O}(\rho) \quad (\text{Satz 4.2})$$

für  $i = 0, 1, \dots, N - 1$ :

$$2a) \quad b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right)$$

$$2b) \quad X_{i+1}^{h,\rho} = \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, \tilde{X}_{i+1}^h) \leq \rho\} \quad \text{Fehler: } \mathcal{O}(\rho) \quad (\text{Satz 4.2})$$

Durch theoretische Untersuchungen des Hausdorff-Abstandes lässt sich somit der folgende Satz, für den [Rod10, Kapitel 4.2.4] als Ausgangspunkt diente, formulieren.

**Satz 4.5.** *Der Fehler im Algorithmus 1, der durch Diskretisierung des Zustandsraums entsteht, lässt sich abschätzen durch*

$$d_H(X_i^h, X_i^{h,\rho}) \leq e^{L(T-t_0)} N \cdot \rho,$$

wobei  $X_i^h$  die theoretisch bestimmte Menge und  $X_i^{h,\rho}$  die diskrete erreichbare Menge, berechnet durch das mengenwertige Euler-Verfahren, bezeichnet.

*Beweis.* Der Beweis erfolgt per Induktion und der Dreiecksungleichung für den Hausdorff-Abstand. Dabei werden folgende Mengen betrachtet:

- theoretisch bestimmte Erreichbarkeitsmenge

$$X_{i+1}^h = \bigcup_{\eta_i \in X_i^h} \{\eta_i + h \cdot F(t_i, \eta_i)\}$$

$$X_0^h = \{x_0\}$$

- Erreichbarkeitsmenge nach dem  $(i + 1)$ -ten Schritt des mengenwertigen Euler-Verfahrens ohne Aufblähung und Projektion der Menge im  $(i + 1)$ -ten Schritt

$$\begin{aligned}\tilde{X}_{i+1}^h &= \bigcup_{g_i \in X_i^{h,\rho}} \{g_i + h \cdot F(t_i, g_i)\} \\ X_0^{h,\rho} &= \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, X_0) \leq \rho\}\end{aligned}$$

- diskrete Erreichbarkeitsmenge nach dem  $(i + 1)$ -ten Eulerschritt mit Aufblähung und Projektion auf ein  $\mathbb{Z}_\rho$ -Gitter

$$\begin{aligned}X_{i+1}^{h,\rho} &= \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, \tilde{X}_{i+1}^h) \leq \rho\} \\ X_0^{h,\rho} &= \{g_\rho \in \mathbb{Z}_\rho \mid b(g_\rho, X_0) \leq \rho\}\end{aligned}$$

**Induktionsanfang** ( $i = 0$ ):

$$d_H(X_0^h, X_0^{h,\rho}) \leq \rho,$$

denn  $X_0^{h,\rho}$  ist gerade die aufgeblähte und auf  $\mathbb{Z}_\rho$  projizierte Menge  $X_0^h$ , siehe Satz 4.2.

**Induktionsschritt** ( $i \rightarrow i + 1$ ):

$$d_H(X_{i+1}^h, X_{i+1}^{h,\rho}) \leq d_H(X_{i+1}^h, \tilde{X}_{i+1}^h) + d_H(\tilde{X}_{i+1}^h, X_{i+1}^{h,\rho})$$

- Betrachte  $d_H(\tilde{X}_{i+1}^h, X_{i+1}^{h,\rho})$ :

$$d_H(\tilde{X}_{i+1}^h, X_{i+1}^{h,\rho}) \leq \rho,$$

denn  $X_{i+1}^{h,\rho}$  ist gerade die aufgeblähte und auf  $\mathbb{Z}_\rho$  projizierte Menge  $\tilde{X}_{i+1}^h$ .

- Betrachte  $d_H(X_{i+1}^h, \tilde{X}_{i+1}^h)$ :

$$d_H(X_{i+1}^h, \tilde{X}_{i+1}^h) \stackrel{(3.9)}{=} \max \left\{ d(X_{i+1}^h, \tilde{X}_{i+1}^h), d(\tilde{X}_{i+1}^h, X_{i+1}^h) \right\}$$

1. Betrachte  $d(X_{i+1}^h, \tilde{X}_{i+1}^h)$ :

$$d(X_{i+1}^h, \tilde{X}_{i+1}^h) = d\left(\bigcup_{\eta_i \in X_i^h} \{\eta_i + h \cdot F(t_i, \eta_i)\}, \bigcup_{g_i \in X_i^{h,\rho}} \{g_i + h \cdot F(t_i, g_i)\}\right)$$

Wähle ein  $\eta_i \in X_i^h$ , dann gilt:

$$d(X_{i+1}^h, \tilde{X}_{i+1}^h) = d\left(\{\eta_i + h \cdot F(t_i, \eta_i)\}, \bigcup_{g_i \in X_i^{h,\rho}} \{g_i + h \cdot F(t_i, g_i)\}\right)$$

und für jedes  $g_i \in X_i^{h,\rho}$ :

$$\begin{aligned} d(X_{i+1}^h, \tilde{X}_{i+1}^h) &\leq d\left(\{\eta_i + h \cdot F(t_i, \eta_i)\}, \{g_i + h \cdot F(t_i, g_i)\}\right) \\ &\stackrel{(3.4)}{=} d\left(\{\eta_i - g_i + h \cdot F(t_i, \eta_i)\}, \{h \cdot F(t_i, g_i)\}\right) \\ &\stackrel{(*)}{\leq} \|\eta_i - g_i\| + d\left(\{h \cdot F(t_i, \eta_i)\}, \{h \cdot F(t_i, g_i)\}\right) \\ &\stackrel{(3.5)}{=} \|\eta_i - g_i\| + h \cdot \underbrace{d(F(t_i, \eta_i), F(t_i, g_i))}_{\leq L\|\eta_i - g_i\|} \\ &\leq (1 + hL) \|\eta_i - g_i\| \end{aligned}$$

$$\begin{aligned} (*) \quad d(v + A, B) &= \sup_{a \in A} d(v + a, B) \\ &= \sup_{a \in A} \inf_{b \in B} \|v + a - b\| \\ &\leq \sup_{a \in A} \inf_{b \in B} \|v\| + \|a - b\| \\ &= \|v\| + d(A, B) \end{aligned}$$

Wähle nun  $g_i \in X_i^{h,\rho}$  so, dass  $\|\eta_i - g_i\| = d(\eta_i, X_i^{h,\rho})$ . Dann gilt:

$$\begin{aligned} \|\eta_i - g_i\| &= d(\eta_i, X_i^{h,\rho}) \\ &\leq d_H(X_i^h, X_i^{h,\rho}) \end{aligned}$$

und damit

$$d\left(\left\{\eta_i + h \cdot F(t_i, \eta_i)\right\}, \tilde{X}_{i+1}^h\right) \leq (1 + hL) \cdot d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right)$$

für alle  $\eta_i \in X_i^h$ .

$$\Rightarrow d\left(X_{i+1}^h, \tilde{X}_{i+1}^h\right) \leq (1 + hL) \cdot d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right)$$

2. Betrachte  $d\left(\tilde{X}_{i+1}^h, X_{i+1}^h\right)$ :

Die Abschätzung verläuft analog zum vorherigen Fall mit vertauschten Rollen von  $\eta_i$  und  $g_i$ . Man erhält:

$$\begin{aligned} d\left(\tilde{X}_{i+1}^h, X_{i+1}^h\right) &\leq (1 + hL) \cdot d_{\text{H}}\left(X_i^{h,\rho}, X_i^h\right) \\ &= (1 + hL) \cdot d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right) \end{aligned}$$

$$\Rightarrow d_{\text{H}}\left(X_{i+1}^h, \tilde{X}_{i+1}^h\right) \leq (1 + hL) \cdot d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right)$$

Aus den Abschätzungen für  $d_{\text{H}}\left(X_{i+1}^h, \tilde{X}_{i+1}^h\right)$  und  $d_{\text{H}}\left(\tilde{X}_{i+1}^h, X_{i+1}^{h,\rho}\right)$  folgt:

$$d_{\text{H}}\left(X_{i+1}^h, X_{i+1}^{h,\rho}\right) \leq (1 + hL) \cdot d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right) + \rho$$

Für die Gesamtabschätzung erhält man somit:

$$\begin{aligned} d_{\text{H}}\left(X_i^h, X_i^{h,\rho}\right) &\leq \sum_{j=0}^i (1 + hL)^j \cdot \rho \\ &\leq \sum_{j=0}^i (1 + hL)^N \cdot \rho \\ &= \sum_{j=0}^i \left(1 + \frac{T - t_0}{N} L\right)^N \cdot \rho \\ &\leq \sum_{j=0}^i e^{L(T-t_0)} \cdot \rho \\ &\leq \sum_{j=0}^N e^{L(T-t_0)} \cdot \rho \\ &= e^{L(T-t_0)} N \cdot \rho \end{aligned}$$

■

Nach Satz 4.5 ist die Wahl der Gitterweite  $\rho$  somit entscheidend für die Konvergenz des Verfahrens. Betrachtet man den Hausdorff-Abstand für die gesamte Realisierung des mengenwertigen Euler-Verfahrens, erhält man aufgrund der Diskretisierung des Zeitparameters die Abschätzung

$$d_H(\mathcal{R}(T, t_0, X_0), \mathcal{R}_N(T, t_0, X_0)) \leq C_1 \cdot h, \quad (\text{vgl. Satz 4.4})$$

und aufgrund der Diskretisierung des Zustandsraums

$$d_H(\mathcal{R}_N(T, t_0, X_0), \mathcal{R}_{N,\rho}(T, t_0, X_0)) \leq C_2 \cdot \rho N. \quad (\text{vgl. Satz 4.5})$$

Um insgesamt Konvergenz der Ordnung 1, also

$$d_H(\mathcal{R}(T, t_0, X_0), \mathcal{R}_{N,\rho}(T, t_0, X_0)) \leq C_3 \cdot h,$$

für das mengenwertige Euler-Verfahren zu erhalten, muss bei der Diskretisierung des Zustandsraums

$$\rho = h^2 = \left( \frac{T - t_0}{N} \right)^2$$

gewählt werden.

Das Gitter mit der Gitterweite ( $\rho = h^2$ ) besteht jedoch, bei sinnvoller Wahl der Schrittweite ( $h < 1$ ), aus einer deutlich größeren Anzahl an Gitterpunkten als mit der Gitterweite ( $\rho = h$ ). Aufgrund des feineren ( $\rho = h^2$ )-Gitters müssen in jedem Eulerschritt mehr Distanzfunktionen zur Berechnung der erreichbaren Menge ausgewertet werden, was wiederum eine deutlich längere Berechnungszeit zur Folge hat.

Tabelle 4.1 stellt die Berechnungszeit bezüglich der Gitterweiten ( $\rho = h$ ) und ( $\rho = h^2$ ) für zwei Beispiele aus dem nächsten Kapitel gegenüber. Die Berechnung wurde mit dem beiliegenden Programm "`set_valued_Euler_v2`" auf einem Rechner mit  $2 \times$  Intel® Xeon® E5640 Prozessor (12MB Cache, 2.66 GHz, 4C/8T), 24GB RAM und dem Betriebssystem openSUSE 11.3 durchgeführt. Man erkennt eine erhebliche Zeitdifferenz für die verschiedenen Gitterweiten. Im weiteren Verlauf der Arbeit werden daher effizientere Methoden zur Berechnung erreichbarer Mengen, basierend auf dem mengenwertigen Euler-Verfahren mit Distanzfunktionen und unter Beibehaltung der Konvergenzeigenschaft, untersucht.

N	Beispiel (5.1)		Beispiel (5.4)	
	$(\rho = h)$	$(\rho = h^2)$	$(\rho = h)$	$(\rho = h^2)$
10	$\leq 1s$	$\leq 1s$	$\leq 1s$	$\leq 1s$
20	$\leq 1s$	35s	$\leq 1s$	5m 46s
25	$\leq 1s$	4min 1s	5s	50m 0s
30	$\leq 1s$	19min 40s	11s	4h 3m 25s
35	$\leq 1s$	1h 16min 2s	23s	16h 8m 7s
40	$\leq 1s$	4h 5min 50s	29s	40h 50m 29s
50	$\leq 1s$	29h 37min 27s	2m 9s	-

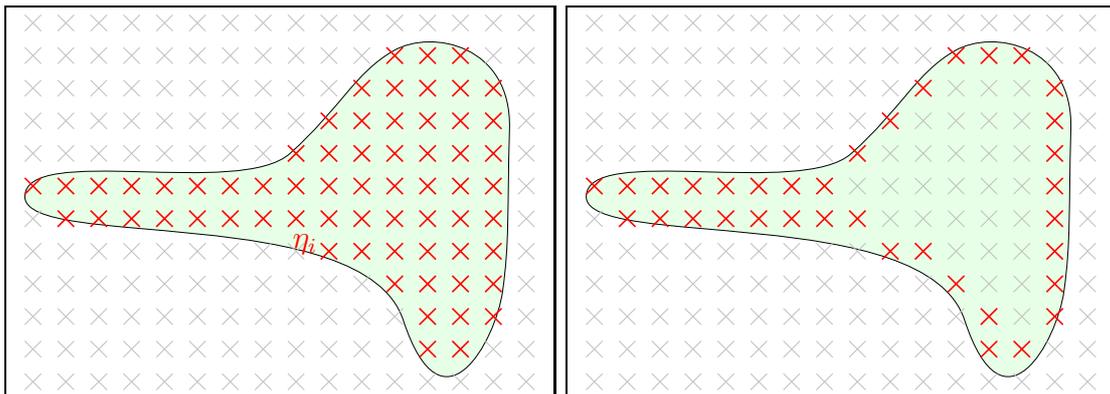
**Tabelle 4.1:** Berechnungsdauer für Gitterweite  $(\rho = h)$  und  $(\rho = h^2)$

Betrachtet man den rechenintensivsten Schritt in Algorithmus 1:

$$2a) \quad b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right),$$

so stellt man fest, dass die Laufzeit maßgeblich von der Anzahl der Gitterpunkte  $g_\rho \in \mathbb{Z}_\rho$  und  $\eta_i \in X_i^{h,\rho}$  beeinflusst wird.

Eine Möglichkeit, zur effizienteren Berechnung erreichbarer Mengen, ist die Reduzierung der Gitterpunkte  $\eta_i$ . Betrachtet man im  $(i+1)$ -ten Iterationsschritt die diskrete erreichbare Menge  $X_i^{h,\rho}$  des vorherigen Eulerschritts, so lässt sich die Menge auch durch Reduktion auf ihre Randpunkte approximieren, siehe Abbildung 4.7.

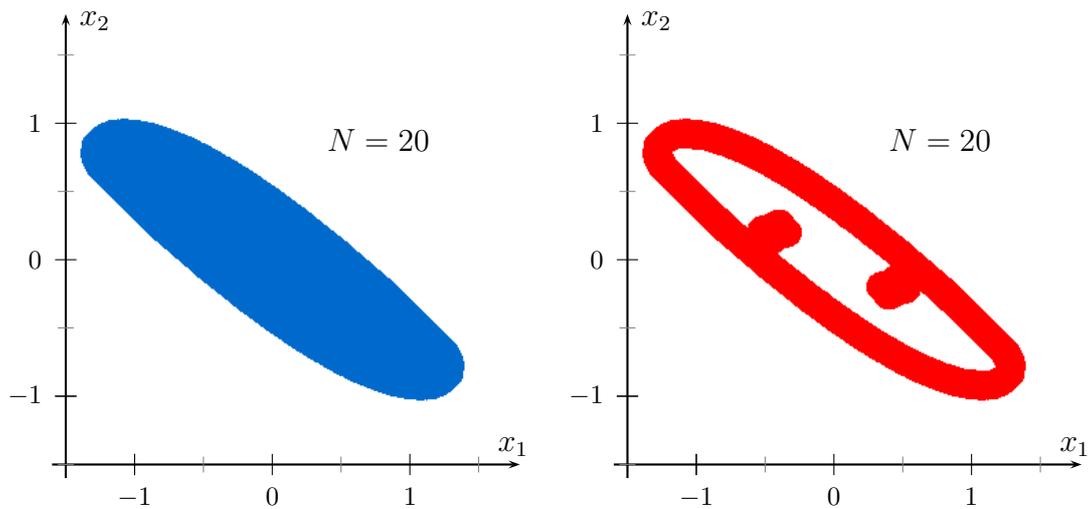


**Abbildung 4.7:** diskrete Approximation einer Menge durch Gitterpunkte  $\eta_i$  (*links*) und nach Reduktion auf ihre Randpunkte (*rechts*)

Im aktuellen Eulerschritt müsste das Minimum in Schritt 2a) von Algorithmus 1

somit nicht mehr über alle Punkte  $\eta_i \in X_i^{h,\rho}$  berechnet werden, sondern nur noch für Punkte die den Rand der Menge  $X_i^{h,\rho}$  repräsentieren.

Abbildung 4.8 zeigt die erreichbare Menge aus Beispiel 5.1 für  $N = 20$ , berechnet mit und ohne Approximation der Menge durch ihre Randpunkte nach jedem Eulerschritt. Man erkennt, dass die erreichbare Menge mit Reduktion auf Randpunkte, nach Rekonstruktion der inneren Punkte, mit der erreichbaren Menge ohne Reduktion auf Randpunkte übereinstimmt.

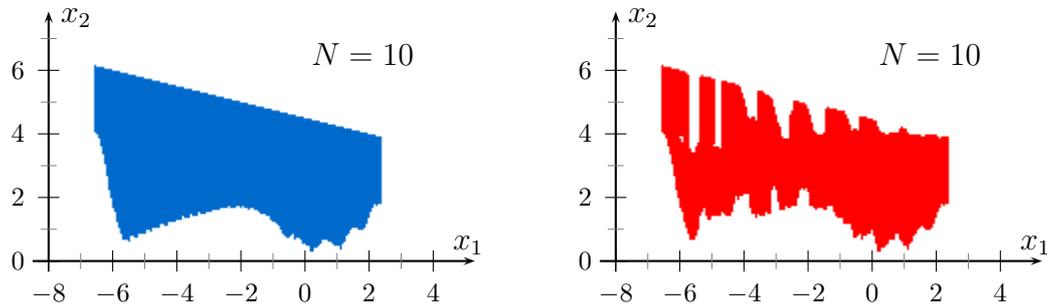


**Abbildung 4.8:** erreichbare Menge aus Beispiel 5.1, berechnet ohne Reduzierung (*links*) und mit Reduzierung auf Randpunkte nach jedem Eulerschritt (*rechts*)

Betrachtet man jedoch Abbildung 4.9, die die entsprechenden erreichbaren Mengen aus Beispiel 5.4 für  $N = 10$  darstellt, so stellt man fest, dass die Rekonstruktion der inneren Punkte zu einem neuen Problem führt und die erreichbare Menge sich nur durch Kenntnis der berechneten Menge ohne Reduktion auf Randpunkte erahnen lässt.

Bei linearen Kontrollsystemen, wie das Beispiel 5.1, entstehen Randpunkte erreichbarer Mengen nach jedem Eulerschritt aus Randpunkten des vorherigen Eulerschritts, weshalb sich die Rekonstruktion der inneren Punkte ohne weiteres durchführen lässt. Bei nichtlinearen Systemen, wie das Beispiel 5.4, werden Randpunkte im Allgemeinen nach einem Iterationsschritt nicht wieder auf Randpunkte abgebildet, wodurch die Rekonstruktion der inneren Punkte beliebig erschwert werden kann. Aus diesem Grund wird der Ansatz an dieser Stelle nicht weiter verfolgt.

Eine andere Möglichkeit, zur effizienteren Berechnung erreichbarer Mengen, ist die



**Abbildung 4.9:** erreichbare Menge aus Beispiel 5.4, berechnet ohne Reduzierung (*links*) und mit Reduzierung auf Randpunkte nach jedem Eulerschritt (*rechts*)

Reduzierung der Gitterpunkte  $g_\rho \in \mathbb{Z}_\rho$ , auf denen die Auswertungen der Distanzfunktion stattfinden. Betrachtet man die Berechnungszeiten in Tabelle 4.1, so wäre es wünschenswert ( $\rho \approx h$ ) zu wählen. Nach obiger Bemerkung kann die Konvergenz des mengenwertigen Euler-Verfahrens dadurch jedoch nicht mehr sichergestellt werden, da hierfür eine Gitterweite ( $\rho = h^2$ ) benötigt wird. Die Idee ist nun den Fehler, der bei Diskretisierung des Zustandsraums mit ( $\rho = h$ ) entsteht, mit Hilfe der linearen Interpolation soweit zu verringern, dass man trotz der größeren Gitterweite die Konvergenz des Verfahrens erhält. Erreicht wird dies, indem die Auswertungen der Distanzfunktion auf einem ( $\rho = h$ )-Gitter stattfinden und die berechnete Menge, durch lineare Interpolation mit baryzentrischen Koordinaten, stets auf einem ( $\rho = h^2$ )-Gitter approximiert wird, was einer Zustandsraumdiskretisierung mit ( $\rho = h^2$ ) entspricht.

## 4.5 Realisierung mit Distanzfunktionen und linearer Interpolation

In diesem Abschnitt wird die Realisierung des mengenwertigen Euler-Verfahrens mit Distanzfunktionen aus dem vorherigen Kapitel mit linearer Interpolation erweitert, um eine effizientere Berechnung der erreichbaren Menge zu erreichen. Im Folgenden werden zwei Gitter mit den Gitterweiten ( $\rho = h$ ) und ( $\tilde{\rho} = h^2$ ) betrachtet. Die Umsetzung des mengenwertigen Euler-Verfahrens mit linearer Interpolation der Distanzfunktion wird in Algorithmus 2 dargestellt.

---

**Algorithmus 2** : mengenwertiges Euler-Verfahren mit linearer Interpolation der Distanzfunktion

---

I. Berechne die Startmenge auf dem  $\tilde{\rho}$ -Gitter durch

$$X_0^{h,\tilde{\rho}} = \{g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid b(g_{\tilde{\rho}}, X_0) \leq \tilde{\rho}\}.$$

II. Für  $i = 0, 1, \dots, N - 1$ :

a) Berechne für jeden Gitterpunkt des  $\rho$ -Gitters den Abstand zur Menge  $\tilde{X}_{i+1}^h$  durch

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\tilde{\rho}}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right),$$

wobei  $\tilde{X}_{i+1}^h$  die Menge nach dem  $(i + 1)$ -ten Eulerschritt, ohne Aufblähung und Projektion, bezeichnet.

b) Das Aufblähen der Menge mit Aufblähungsparameter  $r$  erfolgt durch

$$b(g_\rho, \tilde{X}_{i+1}^{h,r}) = b(g_\rho, \tilde{X}_{i+1}^h) - r,$$

wobei  $\tilde{X}_{i+1}^{h,r}$  die aufgeblähte Menge  $\tilde{X}_{i+1}^h$  bezeichnet.

c) Berechne für jeden Gitterpunkt des  $\tilde{\rho}$ -Gitters, mit linearer Interpolation, die Werte der Distanzfunktion zur Menge  $\tilde{X}_{i+1}^{h,r}$

$$\tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,r}) = \tilde{b}\left(\sum_{j=0}^2 \lambda_j \cdot g_j, \tilde{X}_{i+1}^{h,r}\right) \approx \sum_{j=0}^2 \lambda_j \cdot b(g_j, \tilde{X}_{i+1}^{h,r}),$$

wobei  $\lambda_j$  die baryzentrischen Koordinaten des Gitterpunktes  $g_{\tilde{\rho}}$  bezüglich der drei benachbarten Gitterpunkte  $g_j$  des  $\rho$ -Gitters bezeichnen.

d) Die diskrete Menge  $X_{i+1}^{h,\tilde{\rho}}$  erhält man nun nach Projektion auf das  $\tilde{\rho}$ -Gitter durch

$$X_{i+1}^{h,\tilde{\rho}} = \{g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,r}) \leq 0\}.$$

III. Die diskrete erreichbare Menge lautet

$$\mathcal{R}_{N,\tilde{\rho}}(T, t_0, X_0) = X_N^{h,\tilde{\rho}}.$$


---

*Bemerkung.* Wie bereits in Schritt 2a) von Algorithmus 1 wird unter Berücksichtigung, dass innerhalb einer aufgeblähten Menge

$$b(g_\rho, \tilde{X}_{i+1}^{h,r}) \leq b(g_\rho, \tilde{X}_{i+1}^h) - r$$

gilt, im weiteren Verlauf der Arbeit und in Schritt II.b) von Algorithmus 2 der Einfachheit halber

$$b(g_\rho, \tilde{X}_{i+1}^{h,r}) = b(g_\rho, \tilde{X}_{i+1}^h) - r$$

verwendet. Zur Implementierung sei an dieser Stelle noch angemerkt, dass die Berechnung der Distanzen innerhalb einer Menge, aufgrund der Vereinfachung, durch einen zusätzlichen Methodenaufruf nach dem jeweiligen Eulerschritt durchgeführt werden kann.

Betrachtet man die Größenordnung der Fehler in den einzelnen Schritten von Algorithmus 2:

$$\text{I.)} \quad X_0^{h,\tilde{\rho}} = \left\{ g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid b(g_{\tilde{\rho}}, X_0) \leq \tilde{\rho} \right\} \quad \text{Fehler: } \mathcal{O}(\tilde{\rho}) \quad (\text{Satz 4.2})$$

für  $i = 0, 1, \dots, N - 1$ :

$$\text{II.a)} \quad b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\tilde{\rho}}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right)$$

$$\text{II.b)} \quad b(g_\rho, \tilde{X}_{i+1}^{h,r}) = b(g_\rho, \tilde{X}_{i+1}^h) - r \quad \text{Fehler: } \mathcal{O}(r) \quad (\text{Satz 4.2})$$

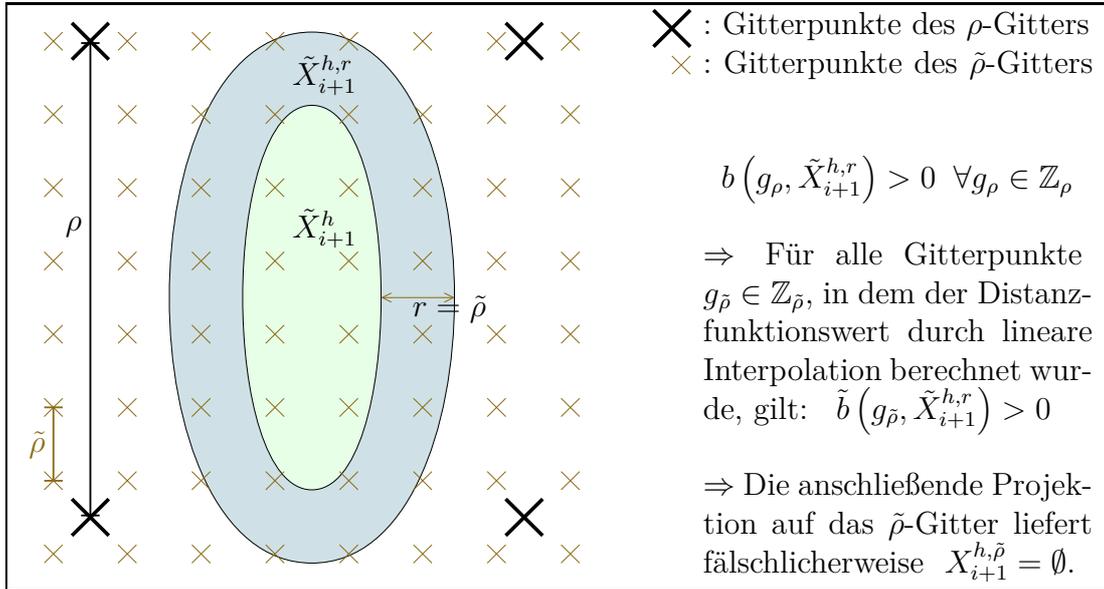
$$\text{II.c)} \quad \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,r}) \approx \sum_{j=0}^2 \lambda_j \cdot b(g_j, \tilde{X}_{i+1}^{h,r}) \quad \text{Fehler: } \mathcal{O}(\tilde{\rho}) \quad (\text{Satz 3.21})$$

$$\text{II.d)} \quad X_{i+1}^{h,\tilde{\rho}} = \left\{ g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,r}) \leq 0 \right\} \quad \text{Fehler: } \mathcal{O}(\tilde{\rho}) \quad (\tilde{\rho}\text{-Gitter})$$

so stellt man fest, dass der Gesamtfehler von der Wahl des Aufblähungsparameters  $r$  in Schritt II.b) abhängt.

Um Konvergenz der Ordnung 1 zu erhalten, muss in Schritt II.b) ( $r = \tilde{\rho}$ ) gewählt werden. Durch die Wahl des Aufblähungsparameters ( $r = \tilde{\rho}$ ) und den Auswertungen der Distanzfunktion auf den Gitterpunkten des  $\rho$ -Gitters, kann es jedoch passieren, dass durch lineare Interpolation in Schritt II.c) fälschlicherweise eine leere Menge berechnet wird. Die Ursache dafür ist, wie zuvor in Kapitel 4.2.1, die Größe des Aufblähungsparameters  $r$ . Mit ( $r = \tilde{\rho}$ ) ist dieser zu klein für das  $\rho$ -Gitter, wodurch die aufgeblähte Menge unter Umständen zwischen den Gitterpunkten liegt, auf denen die Distanzfunktion ausgewertet wird.

Abbildung 4.10 stellt die Situation nochmal grafisch dar. In diesem Fall haben alle Gitterpunkte  $g_\rho$ , nach Auswertung der Distanzfunktion, einen positiven Abstand



**Abbildung 4.10:** Problem bei der Wahl des Aufblähungsparameters ( $r = \tilde{\rho}$ )

zur Menge und man erhält, nach linearer Interpolation der Distanzfunktion, für alle Gitterpunkte des  $\tilde{\rho}$ -Gitters ebenfalls einen positiven Abstand zur Menge. Die anschließende Projektion der Menge auf das  $\tilde{\rho}$ -Gitter in Schritt II.d) liefert somit eine leere Menge, wodurch alle folgenden Iterationsschritte des Euler-Verfahrens fälschlicherweise ebenfalls eine leere Menge liefern.

Um bei der Berechnung erreichbarer Mengen nicht fälschlicherweise eine leere Menge zu erhalten, muss der Aufblähungsparameter  $r$  entsprechend dem  $\rho$ -Gitter, auf dem die Distanzfunktion ausgewertet wird, auf ( $r = \rho$ ) gesetzt werden. Die Konvergenz des Verfahrens lässt sich nun durch einen zusätzlichen Schritt, bei dem das Aufblähen mit ( $r = \rho$ ) korrigiert wird, wiederherstellen:

$$\text{II.c}^*) \quad \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,\tilde{\rho}}) = \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,r}) + (r - \tilde{\rho})$$

Durch die Addition von  $(r - \tilde{\rho})$  in Schritt II.c\*) wird das Aufblähen mit ( $r = \rho$ ), dass für die lineare Interpolation in II.c) notwendig ist, rückgängig gemacht und zugleich eine Aufblähung mit dem Parameter  $\tilde{\rho}$  durchgeführt. Aufgrund der  $\tilde{\rho}$ -Aufblähung wird sichergestellt, dass die anschließende Projektion der Menge auf das  $\tilde{\rho}$ -Gitter keine leere Menge liefert. Für das korrigierte Verfahren erhält man im Einzelnen somit

$$\text{I.)} \quad X_0^{h,\tilde{\rho}} = \left\{ g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid b(g_{\tilde{\rho}}, X_0) \leq \tilde{\rho} \right\} \quad \text{Fehler: } \mathcal{O}(\tilde{\rho})$$

für  $i = 0, 1, \dots, N - 1$ :

$$\text{II.a)} \quad b(g_{\rho}, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\tilde{\rho}}} b\left(\frac{g_{\rho} - \eta_i}{h}, F(t_i, \eta_i)\right)$$

$$\left. \begin{array}{l} \text{II.b)} \quad b(g_{\rho}, \tilde{X}_{i+1}^{h,\rho}) = b(g_{\rho}, \tilde{X}_{i+1}^h) - r \\ \text{II.c)} \quad \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,\rho}) \approx \sum_{j=0}^2 \lambda_j \cdot b(g_j, \tilde{X}_{i+1}^{h,\rho}) \\ \text{II.c*)} \quad \tilde{b}(g_{\tilde{\rho}}, X_{i+1}^{h,\tilde{\rho}}) = \tilde{b}(g_{\tilde{\rho}}, \tilde{X}_{i+1}^{h,\rho}) + r - \tilde{\rho} \end{array} \right\} \text{ Fehler: } \mathcal{O}(\tilde{\rho})$$

$$\text{II.d)} \quad X_{i+1}^{h,\tilde{\rho}} = \left\{ g_{\tilde{\rho}} \in \mathbb{Z}_{\tilde{\rho}} \mid \tilde{b}(g_{\tilde{\rho}}, X_{i+1}^{h,\tilde{\rho}}) \leq 0 \right\} \quad \text{Fehler: } \mathcal{O}(\tilde{\rho})$$

und damit auch mit  $(r = \rho)$  einen Gesamtfehler  $\mathcal{O}(\tilde{\rho})$  für einen Eulerschritt aus Algorithmus 2, was einer Zustandsraumdiskretisierung mit  $(\tilde{\rho} = h^2)$  entspricht.

# 5 Berechnung erreichbarer Mengen

In diesem Kapitel wird die effizientere Berechnung erreichbarer Mengen mit linearer Interpolation durch baryzentrische Koordinaten aus dem vorherigen Kapitel an drei Beispielen numerisch ausgewertet. Sowohl die Berechnung und Darstellung der erreichbaren Mengen, als auch die Berechnung der entsprechenden Hausdorff-Abstände erfolgten dabei gänzlich mit dem beiliegendem Programmpaket, das im Anhang A näher erläutert wird. Zu Beginn wird eine zweidimensionale lineare Differentialinklusion betrachtet.

## 5.1 Lineare Differentialinklusion

**Beispiel 5.1** (vgl. [Bai95, Beispiel 3.3.1]).

Gegeben sei die lineare Differentialinklusion

$$\begin{aligned} \dot{x}(t) &\in F(t, x(t)) = A(t)x(t) + B(t)U, & (\text{für fast alle } t \in I := [t_0, T]), \\ x(t_0) &\in X_0 \end{aligned}$$

mit

$$A(t) = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix}, \quad B(t) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

dem Intervall  $I = [0, 2]$ ,  $U = B_1(0) := \{y \in \mathbb{R}^2 : \|y\|_2 \leq 1\}$  und  $X_0 = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix} \right\}$ .

Für dieses Beispiel werden im Folgenden einige Sätze zur effizienteren Berechnung des Hausdorff-Abstandes bereitgestellt.

**Satz 5.2** ([BBCG07, Proposition 2.3]). *In einem linearen Kontrollsystem ist die erreichbare Menge  $\mathcal{R}(t, t_0, x_0)$  konvex, kompakt und nicht leer für jedes  $t \in [t_0, T]$ .*

*Beweis.* Siehe [SvV65, Theorem 1]. ■

*Bemerkung.* Das Resultat aus Satz 5.2 wird im weiteren Verlauf der Arbeit zur Reduzierung des Rechenaufwands bei der Berechnung des Hausdorff-Abstandes für lineare Kontrollsysteme verwendet. Nach Kapitel 3.2 wird der Hausdorff-Abstand zwischen zwei nichtleeren Mengen  $A$  und  $B$  durch

$$d_{\text{H}}(A, B) = \max\{d(A, B), d(B, A)\}$$

berechnet. In der Implementierung wird also für jeden Punkt aus  $A$  der Abstand zur Menge  $B$  und für jeden Punkt aus  $B$  der Abstand zur Menge  $A$  berechnet. Für lineare Kontrollsysteme, dessen erreichbare Menge konvex ist, lässt sich der Rechenaufwand mit Hilfe von Stützfunktionen signifikant reduzieren. Der folgende Satz liefert eine Vorschrift zur Berechnung des Hausdorff-Abstandes durch Stützfunktionen, wobei hier aus Zeitgründen nicht näher auf Stützfunktionen eingegangen und auf [Bai95] verwiesen wird.

**Satz 5.3** ([Bai95, Satz 0.1.3.5]).

*Sind  $A, B$  nichtleere, kompakte, konvexe Mengen des  $\mathbb{R}^n$ , so folgt:*

$$d_{\text{H}}(A, B) = \sup_{l \in B_1(0)} |\delta^*(l, A) - \delta^*(l, B)|,$$

wobei  $\delta^*(\cdot, \cdot)$  die Stützfunktion der jeweiligen Menge bezeichnet.

*Beweis.* Vgl. [Bai95, Seite 13]. ■

Für die Implementierung des mengenwertigen Euler-Verfahren wird die Distanzfunktion in Schritt II.a) des Algorithmus 2, unter Verwendung der Sätze 3.7 (Verschiebung der Menge), 3.8 (Skalares Vielfaches) und der Zerlegung

$$F(t, x(t)) = p(t, x(t)) + q(t, x(t)) \cdot U,$$

in einen Anteil unabhängig von der Kontrolle und einen Anteil abhängig von der Kontrolle  $U$ , noch weiter umgeformt:

$$\begin{aligned} b(g_\rho, \tilde{X}_{i+1}^h) &= \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b(g_\rho, \eta_i + h \cdot F(t_i, \eta_i)) \\ &= \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b\left(g_\rho - \eta_i, h \cdot \left(p(t_i, \eta_i) + q(t_i, \eta_i) \cdot U\right)\right) \\ &= \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b\left(g_\rho - \eta_i, h \cdot p(t_i, \eta_i) + h \cdot q(t_i, \eta_i) \cdot U\right) \\ &= \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b\left(g_\rho - \eta_i - h \cdot p(t_i, \eta_i), h \cdot q(t_i, \eta_i) \cdot U\right) \end{aligned}$$

$$= h \cdot \min_{\eta_i \in X_i^{h,\bar{\rho}}} b\left(\frac{g_\rho - \eta_i - h \cdot p(t_i, \eta_i)}{h}, q(t_i, \eta_i) \cdot U\right).$$

Mit  $p(t_i, \eta_i) = A(t_i)\eta_i$  und  $q(t_i, \eta_i) = B(t_i)$  erhält man für die lineare Differentia-linkklusion 5.1

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\bar{\rho}}} b\left(\frac{g_\rho - \eta_i - h \cdot A(t_i)\eta_i}{h}, B(t_i)U\right),$$

wobei  $B(t_i)U$  dem Einheitskreis im  $\mathbb{R}^2$  entspricht. Für die Berechnung der erreichbaren Menge wird somit die Distanzfunktion zum Einheitskreis implementiert. Die Aufblähung und lineare Interpolation nach jedem Eulerschritt, kann wie in Kapitel 4.5 beschrieben, ohne weitere Umformungen implementiert werden.

Abbildungen 5.1 und 5.2 zeigen die erreichbare Menge (*rot*), berechnet ohne und mit linearer Interpolation der Distanzfunktion, für  $N = 5, 10, 20$  und  $40$ . Die grün dargestellte Menge repräsentiert die Referenzmenge mit  $N = 1\,000\,000$ . Die Referenzmenge wurde durch die iterierte Trapezregel für lineare Systeme mit Berechnung des mengenwertigen Aumann Integrals unter Kenntnis des Fundamentalsystems berechnet. Für weitere Einzelheiten zur Berechnung der Referenzmenge sei hier auf [Bai95, Satz 2.1.3] und [Bai95, Beispiel 1.3.3] oder allgemein auf [Bai95, Kapitel 2.2] verwiesen.

Tabelle 5.1 zeigt die Hausdorff-Abstände der berechneten Mengen auf einem ( $\rho = h$ )-Gitter, während Tabelle 5.2 die Hausdorff-Abstände der entsprechenden Mengen mit ( $\rho = h^2$ ) zeigt. Zum Vergleich wurde neben der Referenzmenge mit  $N = 1\,000\,000$  auch eine Referenzmenge mit  $N = 160$ , berechnet durch Algorithmus 1 auf einem ( $\rho = h^2$ )-Gitter, verwendet. Während man in Abbildung 5.1 auch für die berechneten Mengen mit ( $\rho = h$ ) eine lineare Konvergenz vermuten könnte, wird in Tabelle 5.1 deutlich, dass dies nicht der Fall ist.

In Tabelle 5.3 werden die Hausdorff-Abstände der erreichbaren Mengen, berechnet mit linearer Interpolation der Distanzfunktion, aufgelistet. In allen drei Tabellen erkennt man eine erhöhte geschätzte Konvergenz bezüglich der Referenzmenge mit  $N = 160$ . Dies lässt sich jedoch durch die Nähe der Schrittweiten zur Referenzschrittweite erklären, da diese nicht sehr präzise bezüglich der gewählten Schrittweiten für den Vergleich ist. Die geschätzte Konvergenzordnung  $p$  wird dabei mit

dem Ansatz

$$d_{\text{H}}(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}}) \approx \tilde{C} \cdot h^p \quad | \log$$

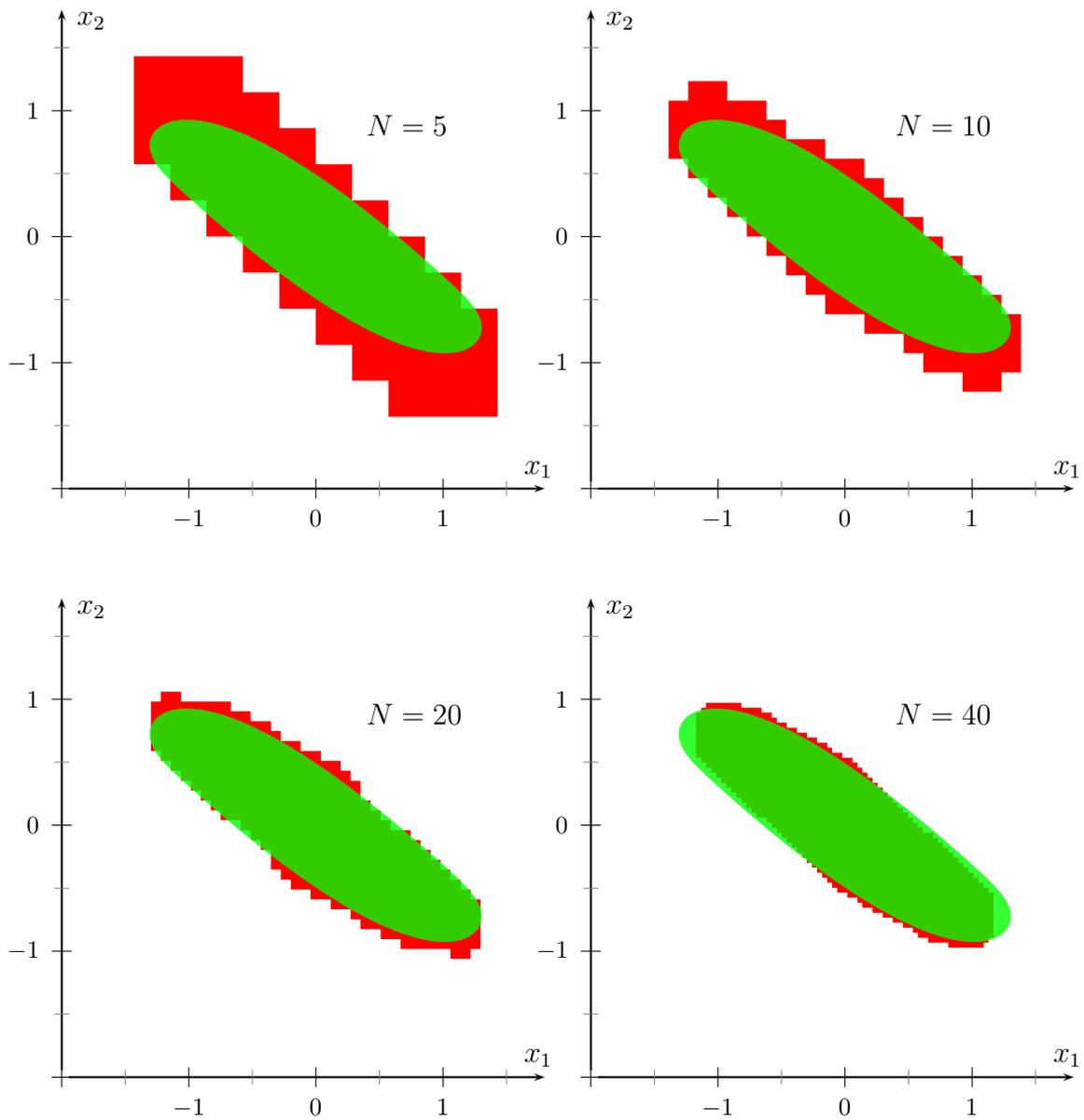
$$\log(d_{\text{H}}(h)) \approx \underbrace{\log(\tilde{C})}_C + p \log(h),$$

und durch Lösen eines linearen Ausgleichsproblems zur Bestimmung von  $C$  und  $p$  berechnet.

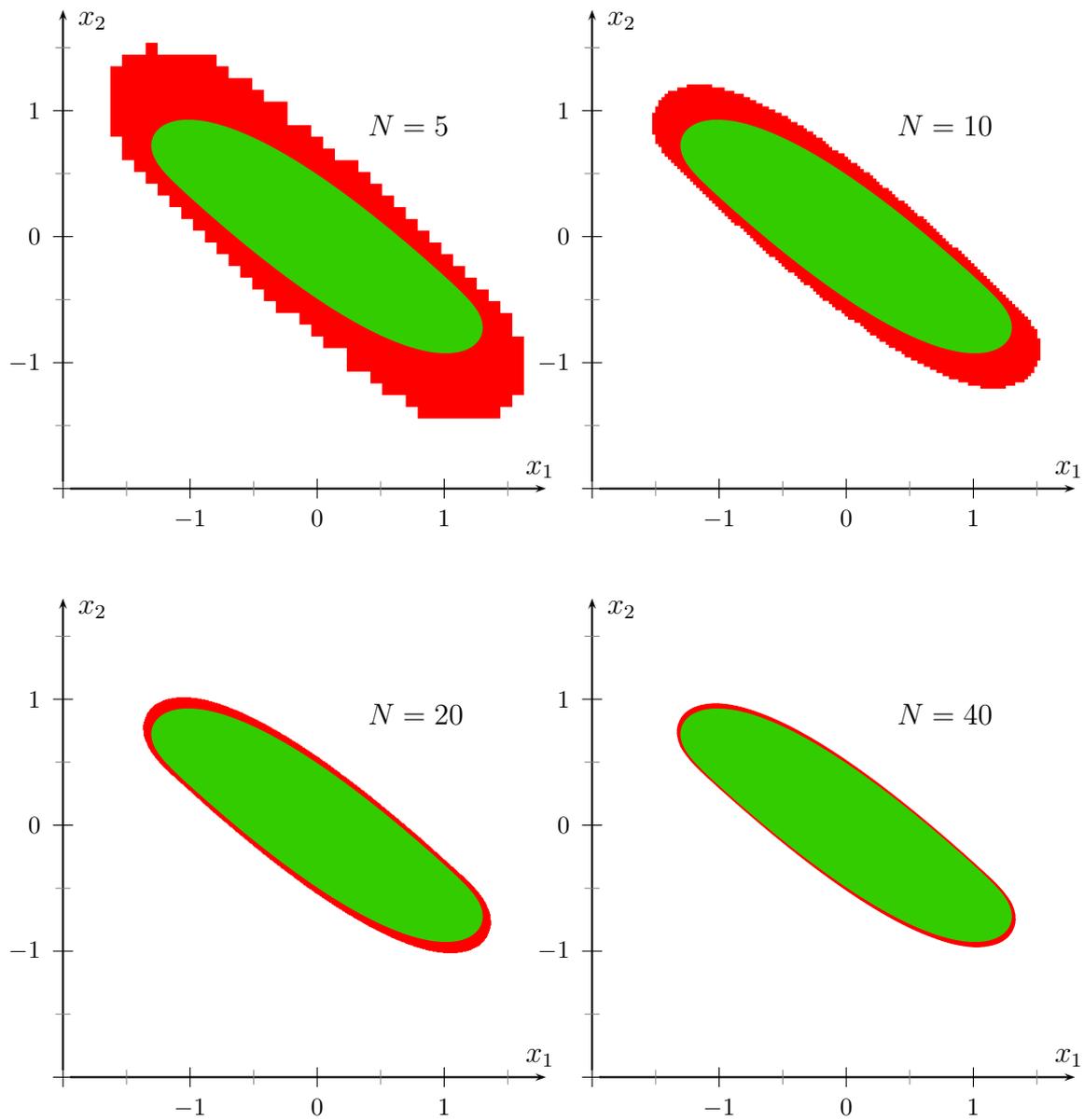
Die Effizienz der Berechnungen wird schließlich in Tabelle 5.4 mit den Berechnungen auf einem ( $\rho = h^2$ )-Gitter verglichen. In Tabelle 5.5 wird zum Abschluss dieses Beispiels die Berechnungszeit für einige Schrittweiten detaillierter aufgelistet.

N	h	$d_{\text{H}}(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 160$	$d_{\text{H}}(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 1\,000\,000$
5	0.4	9.82736547e-01	9.99837479e-01
10	0.2	6.68914967e-01	6.85783793e-01
15	0.13333333	4.81395488e-01	4.98420875e-01
20	0.1	4.95259413e-01	5.12421871e-01
25	0.08	3.78183966e-01	3.95167180e-01
30	0.06666667	3.11846478e-01	3.28895077e-01
35	0.05714286	3.62391648e-01	3.79488628e-01
40	0.05	3.33514435e-01	3.50375410e-01
50	0.04	2.75624867e-01	2.92606926e-01
60	0.03333333	2.96247709e-01	3.13107938e-01
70	0.02857143	2.85600073e-01	3.02460101e-01
80	0.025	2.89205470e-01	3.06192917e-01
geschätzte Konvergenzordnung		0.4558	0.4397

**Tabelle 5.1:** mengenwertiges Euler-Verfahren auf einem ( $\rho = h$ )-Gitter



**Abbildung 5.1:** erreichbare Menge des Beispiels 5.1, berechnet mit dem mengenwertigen Euler-Verfahren und der Gitterweite ( $\rho = h$ )



**Abbildung 5.2:** erreichbare Menge des Beispiels 5.1, berechnet mit dem mengenwertigen Euler-Verfahren und linearer Interpolation der Distanzfunktion

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 160$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 1\,000\,000$
5	0.4	6.47076714e-01	6.62485089e-01
10	0.2	2.55234030e-01	2.72497992e-01
15	0.13333333	1.50294186e-01	1.67399295e-01
20	0.1	1.06229704e-01	1.22618252e-01
25	0.08	8.76733622e-02	1.04916364e-01
30	0.06666667	7.01664181e-02	8.74272772e-02
35	0.05714286	5.79868905e-02	7.50672947e-02
40	0.05	4.95416940e-02	6.67086787e-02
50	0.04	3.70637479e-02	5.43067499e-02
geschätzte Konvergenzordnung		1.2189	1.0703

**Tabelle 5.2:** mengenwertiges Euler-Verfahren auf einem  $(\rho = h^2)$ -Gitter

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 160$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 1\,000\,000$
5	0.4	6.47570121e-01	6.63646051e-01
10	0.2	3.14988989e-01	3.32145674e-01
15	0.13333333	2.28461799e-01	2.45655092e-01
20	0.1	8.44817056e-02	1.01594044e-01
25	0.08	9.40723308e-02	1.11348708e-01
30	0.06666667	8.68355692e-02	1.04104377e-01
35	0.05714286	3.20737788e-02	4.84909941e-02
40	0.05	2.92655483e-02	4.58441584e-02
50	0.04	3.50376663e-02	5.22267076e-02
60	0.03333333	2.36501641e-02	4.06462410e-02
70	0.02857143	1.07033928e-02	2.67682951e-02
80	0.025	6.07669890e-03	1.95264250e-02
geschätzte Konvergenzordnung		1.6060	1.2498

**Tabelle 5.3:** mengenwertiges Euler-Verfahren auf einem  $(\rho = h)$ -Gitter mit linearer Interpolation der Distanzfunktion

N	$(\rho = h^2)$ -Gitter	$(\rho = h)$ -Gitter mit linearer Interpolation
5	$\leq$ 1s	$\leq$ 1s
10	$\leq$ 1s	$\leq$ 1s
15	3s	$\leq$ 1s
20	35s	3s
25	4min 1s	9s
30	19min 40s	17s
35	1h 16min 2s	2min 6s
40	4h 5min 50s	3min 48s
50	29h 37min 27s	9min 53s
60	-	21min 7s
70	-	3h 32min 36s
80	-	6h 10min 12s

**Tabelle 5.4:** Berechnungsdauer der erreichbaren Mengen

N	Zeit für Eulerschritte	Zeit für linearer Interpolation	Gesamtzeit
20	2s	1s	3s
30	12s	4s	17s
40	1min 55s	1min 52s	3min 48s
50	5min 39s	4min 11s	9min 53s
60	13min 5s	7min 58s	21min 7s
70	1h 45min 36s	1h 46min 40s	3h 32min 36s
80	3h 16min 21s	2h 53min 17s	6h 10min 12s

**Tabelle 5.5:** detaillierte Berechnungsdauer der erreichbaren Mengen für Gitterweite  $(\rho = h)$  mit linearer Interpolation der Distanzfunktion

## 5.2 Rayleigh-Problem

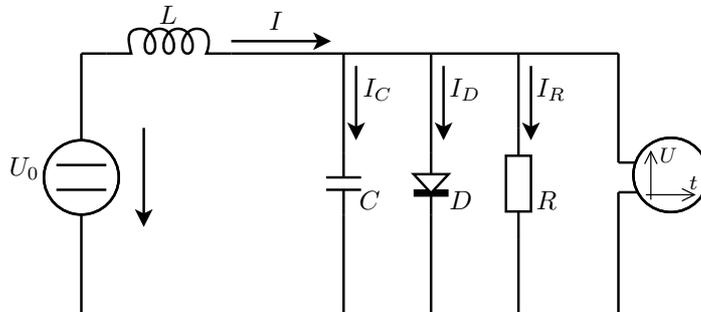


Abbildung 5.3: Tunnel diode Oszillator

Betrachte den elektrischen Schaltkreis (Tunnel diode Oszillator) in Abbildung 5.3, wobei  $L$  die Induktivität,  $C$  die Kapazität,  $R$  den Widerstand,  $I$  die elektrische Stromstärke,  $U_0$  die Spannung am Generator und  $D$  die Diode darstellt. Die Zustandsvariable  $x(t)$  stellt die elektrische Stromstärke  $I$  zur Zeit  $t$  dar. Die Kontrollfunktion wird über die Spannung  $U_0(t)$  am Generator definiert. Nach einer geeigneten Transformation von  $U_0(t)$  erhält man die folgende Rayleigh-Gleichung mit einer Kontrollfunktion  $u(t)$  und einem Parameter  $s \in \mathbb{R}$ , siehe [BM98, Abschnitt 4.1]:

$$\ddot{x}(t) = -x(t) + \dot{x}(t) \left(1.4 - s\dot{x}(t)^2\right) + 4u(t). \quad (5.1)$$

Der Parameter wird als  $s = 0.14$  gewählt. Ziel ist nun die Schwingungen der elektrischen Stromstärke an der Tunnel diode, durch die anliegende Spannung am Generator  $U_0$ , zu kontrollieren. Daraus ergibt sich unmittelbar die Fragestellung, ob ein bestimmter Wert oder Wertebereich an der Diode  $D$  erreicht werden kann, was wiederum zur Berechnung der erreichbaren Menge des entsprechenden Kontrollsystems führt.

Mit  $u(t) \in [-1, 1]$  und der Endzeit  $t_f = 2.5$  erhält man, nach Umformung von (5.1) in ein System erster Ordnung, folgende Differentialinklusion:

**Beispiel 5.4** (vgl. [Cha03, Example 5.2.5]).

Betrachte folgende nichtlineare Differentialinklusion

$$\begin{aligned} \dot{x}(t) &\in F(t, x(t)) = p(t, x(t)) + q(t, x(t))U \quad (\text{für fast alle } t \in I := [t_0, T]), \\ x(t_0) &\in X_0 \end{aligned}$$

mit

$$p(t, x(t)) = \begin{pmatrix} x_2 \\ -x_1 + x_2(1.4 - 0.14x_2^2) \end{pmatrix}, \quad q(t, x(t)) = \begin{pmatrix} 0 \\ 4 \end{pmatrix},$$

dem Intervall  $I = [0, 2.5]$ ,  $U = [-1, 1]$  und  $X_0 = \left\{ \begin{pmatrix} -5 \\ -5 \end{pmatrix} \right\}$ .

Die auszuwertende Distanzfunktion für dieses Beispiel lautet nach Umformungen wie im vorherigen Beispiel

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b\left(\frac{g_\rho - \eta_i - h \cdot p(t_i, \eta_i)}{h}, q(t_i, \eta_i) \cdot U\right).$$

Für die Berechnung der erreichbaren Menge wird somit die Distanzfunktion zur Menge

$$q(t_i, \eta_i) \cdot U = \{0\} \times [-4, 4]$$

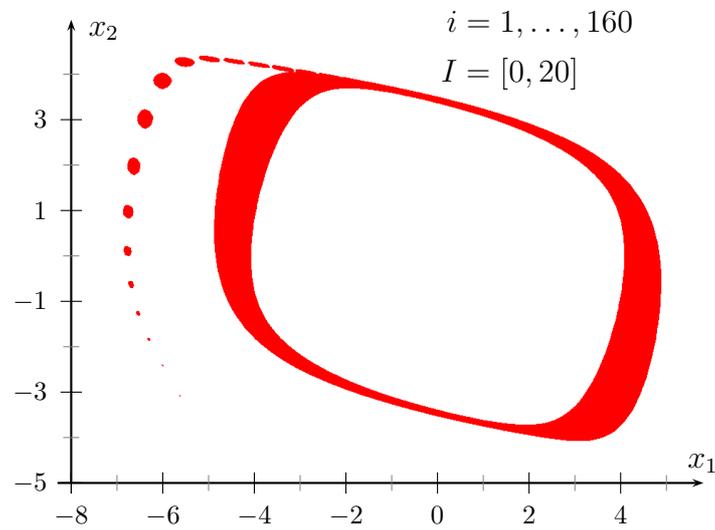
implementiert.

Abbildung 5.5 zeigt die erreichbare Menge (*rot*), berechnet mit linearer Interpolation der Distanzfunktion, für  $N = 10, 15, 20, 30, 40$  und  $60$ . Die grün dargestellte Menge repräsentiert die Referenzmenge mit  $N = 80$ . Die Referenzmenge wurde dabei durch ein Verfahren höherer Ordnung, basierend auf *barrier methods*, berechnet. Für weitere Einzelheiten zur Berechnung der Referenzmenge sei hier auf [GJ12] verwiesen. Ergebnisse für ( $\rho = h$ ) ohne lineare Interpolation werden aufgrund fehlender Konvergenz nicht weiter verglichen.

Tabelle 5.7 bestätigt die Konvergenz des vorgestellten Verfahrens durch die Hausdorff-Abstände für verschiedene Schrittweiten  $h$ . Zum Vergleich werden in Tabelle 5.6 die entsprechenden Hausdorff-Abstände der erreichbaren Mengen, berechnet auf einem ( $\rho = h^2$ )-Gitter, dargestellt. Als Referenzmenge wird in beiden Tabellen einerseits die erreichbare Menge, berechnet durch Algorithmus 1 auf einem ( $\rho = h^2$ )-Gitter für  $N = 120$ , und andererseits die oben erwähnte erreichbare Menge für  $N = 80$  verwendet.

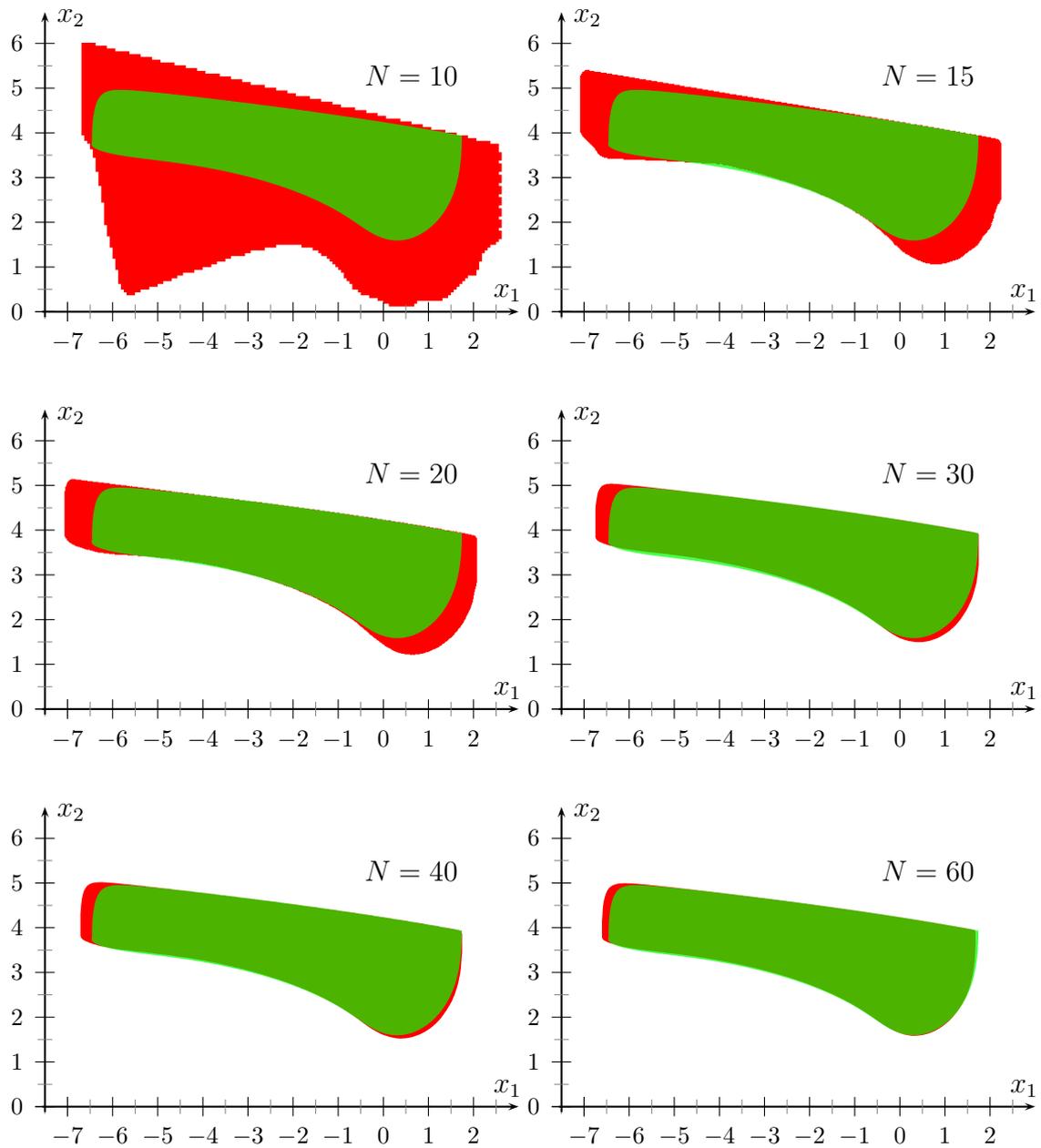
*Bemerkung.* Die hohe Konvergenzordnung für dieses Beispiel wurde bereits 2003 von I. A. Chahma in [Cha03, Tabelle 5.11] festgestellt. Eine mögliche Ursache dafür lässt sich durch folgende Untersuchung finden. Betrachtet man das Beispiel 5.4 mit  $I = [0, 20]$  und  $U = B_1(0)$ , so stellt man einen Grenzyklus im System fest, wodurch Fehler oder Störungen des Systems gewissermaßen abgedämpft werden, und damit eine Konvergenzordnung größer als eins erklären. Abbildung 5.4

zeigt den Verlauf der berechneten Mengen für 160 Eulerschritte des modifizierten Beispiels.



**Abbildung 5.4:** Grenzzyklus im System aus Beispiel 5.4

Die Dauer für die Berechnung der erreichbaren Mengen mit Interpolation der Distanzfunktion werden schließlich in Tabelle 5.8 mit den Berechnungen auf einem  $(\rho = h^2)$ -Gitter verglichen und in Tabelle 5.9 für einige Schrittweiten detailliert aufgelistet.



**Abbildung 5.5:** erreichbare Menge des Beispiels 5.4, berechnet mit dem mengenwertigen Euler-Verfahren und linearer Interpolation der Distanzfunktion

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 120$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 80$
5	0.5	1.22241422e+01	1.22339088e+01
10	0.25	2.81998492e+00	2.79575124e+00
15	0.16666667	7.61178311e-01	8.46413093e-01
20	0.125	4.80548531e-01	5.68758585e-01
25	0.1	3.66869532e-01	4.55329551e-01
30	0.08333333	2.86641421e-01	3.76684386e-01
35	0.07142857	2.22940645e-01	3.12241633e-01
40	0.0625	1.84429634e-01	2.73821026e-01
geschätzte Konvergenzordnung		2.0296	1.8315

**Tabelle 5.6:** mengenwertiges Euler-Verfahren auf einem  $(\rho = h^2)$ -Gitter

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 120$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 80$
5	0.5	1.23873172e+01	1.23958753e+01
10	0.25	2.91306576e+00	2.88962476e+00
15	0.16666667	8.81831338e-01	9.69010449e-01
20	0.125	6.90456349e-01	7.80405794e-01
25	0.1	3.64527877e-01	4.53501998e-01
30	0.08333333	2.97263404e-01	3.86758812e-01
35	0.07142857	2.57545296e-01	3.47189564e-01
40	0.0625	2.35203413e-01	3.24452137e-01
50	0.05	1.20459404e-01	2.10257469e-01
60	0.04166667	7.76325587e-02	1.85347021e-01
70	0.03571429	7.76325587e-02	1.67545333e-01
80	0.03125	6.76531009e-02	1.56831574e-01
geschätzte Konvergenzordnung		1.8571	1.5336

**Tabelle 5.7:** mengenwertiges Euler-Verfahren auf einem  $(\rho = h)$ -Gitter mit linearer Interpolation der Distanzfunktion

N	$(\rho = h^2)$ -Gitter			$(\rho = h)$ -Gitter mit linearer Interpolation		
5	$\leq$ 1s			$\leq$ 1s		
10	$\leq$ 1s			$\leq$ 1s		
15	39s			4s		
20	5min	46s		10s		
25	50min	0s		1min	50s	
30	4h	3min	25s	4min	3s	
35	16h	8min	7s	8min	8s	
40	40h	50min	29s	12min	7s	
50	-			3h	4min	33s
60	-			7h	11min	7s
70	-			14h	53min	28s
80	-			22h	28min	36s

**Tabelle 5.8:** Berechnungsdauer der erreichbaren Mengen

N	Zeit für Eulerschritte			Zeit für linearer Interpolation			Gesamtzeit		
20	6s			4s			10s		
30	2min	58s		1min	5s		4min	3s	
40	9min	12s		2min	54s		12min	7s	
50	2h	15min	12s	49min	12s		3h	4min	33s
60	5h	32min	17s	1h	38min	26s	7h	11min	7s
70	11h	53min	35s	2h	59min	14s	14h	53min	28s
80	17h	28min	13s	4h	59min	21s	22h	28min	36s

**Tabelle 5.9:** detaillierte Berechnungsdauer der erreichbaren Mengen für Gitterweite  $(\rho = h)$  mit linearer Interpolation der Distanzfunktion

### 5.3 Bilineare Differentialinklusion

Das letzte Beispiel enthält einen bilinearen Ausdruck in der Dynamik.

**Beispiel 5.5** (vgl. [BGX11, Abschnitt 5.2]).

*Betrachte die folgende Differentialinklusion*

$$\dot{x}(t) \in F(t, x(t)) = \bigcup_{u(t) \in U} \{f(t, x(t), u(t))\} \quad (\text{für fast alle } t \in I := [t_0, T]),$$

$$x(t_0) \in X_0$$

mit

$$f(t, x(t), u(t)) = \begin{pmatrix} \pi x_2(t) \\ -\pi u(t)x_1(t) \end{pmatrix}$$

dem Intervall  $I = [0, 1]$ ,  $U = [0, 1]$  und  $X_0 = \left\{ \begin{pmatrix} -1 \\ 0 \end{pmatrix} \right\}$ .

Die auszuwertende Distanzfunktion für dieses Beispiel lautet

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h, \tilde{\rho}}} b\left(\frac{g_\rho - \eta_i - h \cdot p(t_i, \eta_i)}{h}, q(t_i, \eta_i) \cdot U\right)$$

mit  $p(t_i, \eta_i) = \begin{pmatrix} \pi(\eta_i)_2 \\ 0 \end{pmatrix}$  und  $q(t_i, \eta_i) = \begin{pmatrix} 0 \\ -\pi(\eta_i)_1 \end{pmatrix}$ . Für die Berechnung der erreichbaren Menge wird somit die Distanzfunktion zur Menge

$$q(t_i, \eta_i) \cdot U = \{0\} \times [0, -\pi(\eta_i)_1]$$

implementiert.

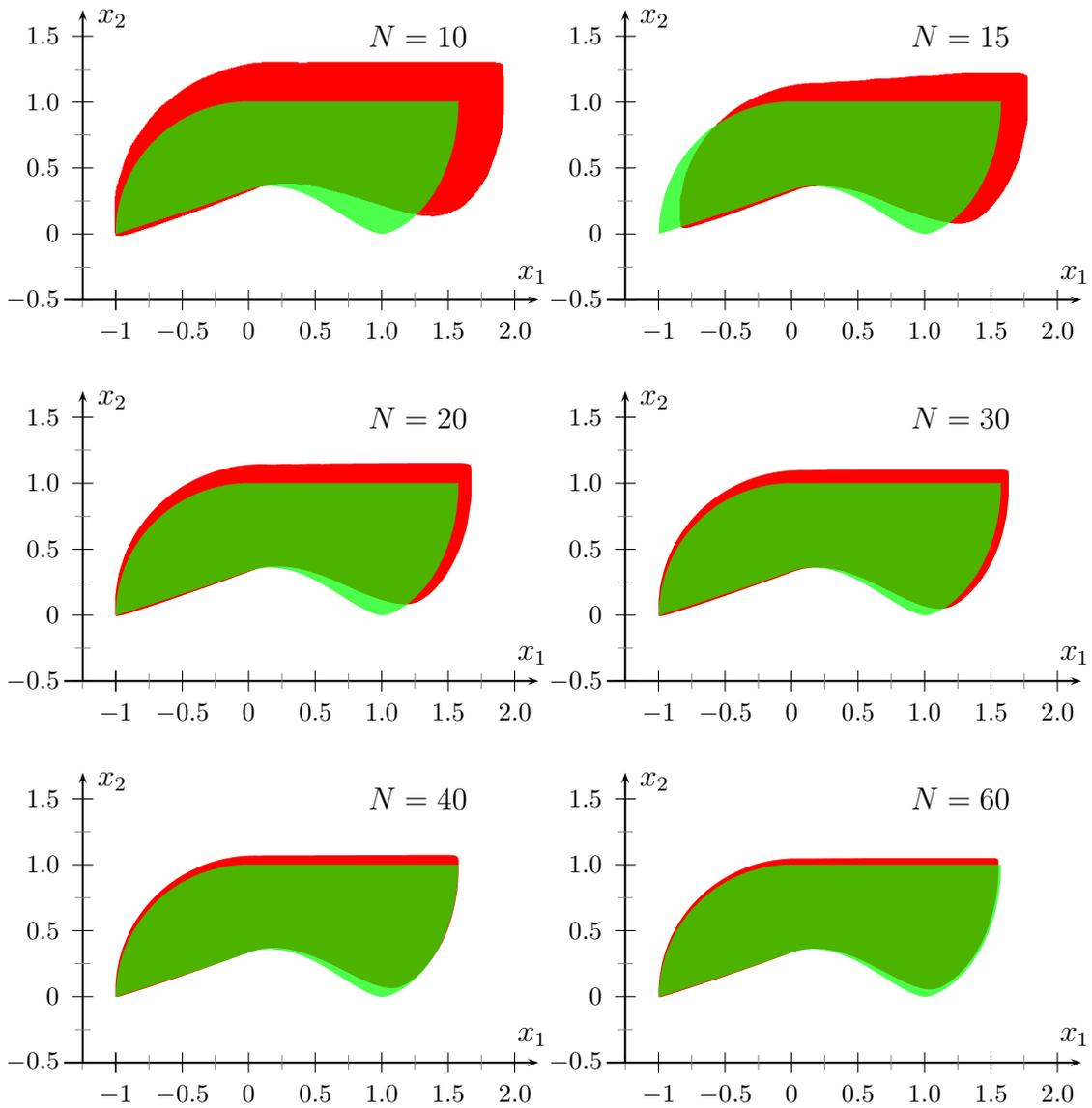
*Bemerkung.* Im Gegensatz zu den vorherigen Beispielen, geht in diesem Beispiel ein Teil des ODE-Systems in die Distanzfunktion mit ein. Das Auswerten der Distanzfunktion wird dadurch aufwendiger, da diese nun auch vom Zustand  $\eta_i$  abhängt und für jedes  $\eta_i$  die Distanz zu einer anderen Menge berechnet wird. Man stellt somit fest, dass eine Auswertung der Distanzfunktion im Allgemeinen sehr kompliziert werden kann, was wiederum zur Reduzierung der Anzahl motiviert.

Wie in den Beispielen zuvor wird in Abbildung 5.6 zunächst die erreichbare Menge (*rot*), berechnet mit linearer Interpolation der Distanzfunktion, für  $N = 10, 15, 20, 30, 40$  und  $60$  dargestellt. Die grün dargestellte Menge repräsentiert die Referenzmenge mit  $N = 80$ . Die Referenzmenge wurde, wie im Beispiel zuvor, durch

ein Verfahren höherer Ordnung, basierend auf *barrier methods* [GJ12], berechnet. Die Konvergenz des Verfahrens ist auch hier bereits anhand der Darstellungen zu erkennen.

Tabelle 5.11 bestätigt die Konvergenz des Verfahrens durch die Hausdorff-Abstände der erreichbaren Mengen, berechnet mit linearer Interpolation der Distanzfunktion, während Tabelle 5.10 nochmal die Konvergenz des mengenwertigen Euler-Verfahrens auf dem  $(\rho = h^2)$ -Gitter aufzeigt. Als Referenzmenge wurde die oben erwähnte Referenzmenge für  $N = 80$  und zum Vergleich, die erreichbare Menge, berechnet durch Algorithmus 1 auf einem  $(\rho = h^2)$ -Gitter, für  $N = 80$  verwendet. Wie bei den vorherigen Beispielen bemerkt man, dass die Schrittweite der zuletzt genannten Referenzmenge zu nahe an den Schrittweiten der berechneten Mengen liegt und die Ergebnisse dementsprechend mit Bedacht zu bewerten sind.

Tabelle 5.12 vergleicht schließlich die Effizienz des Verfahrens mit den Berechnungen auf einem  $(\rho = h^2)$ -Gitter und Tabelle 5.13 listet die ausführlichen Berechnungszeiten für einige Schrittweiten auf.



**Abbildung 5.6:** erreichbare Menge des Beispiels 5.5, berechnet mit dem mengenwertigen Euler-Verfahren und linearer Interpolation der Distanzfunktion

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 80$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 80$
5	0.2	8.72441498e-01	9.32319701e-01
10	0.1	4.46509138e-01	5.08747482e-01
15	0.06666667	2.65365487e-01	3.27582980e-01
20	0.05	1.91933012e-01	2.54293629e-01
25	0.04	1.25297381e-01	1.87263138e-01
30	0.03333333	1.00942941e-01	1.63314587e-01
35	0.02857143	8.34904757e-02	1.45865092e-01
geschätzte Konvergenzordnung		1.2323	0.9791

**Tabelle 5.10:** mengenwertiges Euler-Verfahren auf einem ( $\rho = h^2$ )-Gitter

N	h	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}_{\text{Alg.1}}^{\text{ref}})$ mit $N_{\text{ref}} = 80$	$d_H(\mathcal{R}_{N,\rho}, \mathcal{R}^{\text{ref}})$ mit $N_{\text{ref}} = 80$
5	0.2	8.94645141e-01	9.56067572e-01
10	0.1	3.74820921e-01	4.36165393e-01
15	0.06666667	2.17631242e-01	2.77376398e-01
20	0.05	1.21565010e-01	1.65273554e-01
25	0.04	2.13084386e-01	2.06761373e-01
30	0.03333333	7.00895385e-02	1.07185498e-01
35	0.0285714	1.12316796e-01	1.09611982e-01
40	0.025	6.95440648e-02	7.68088257e-02
50	0.02	6.06765245e-02	6.31696624e-02
60	0.01666667	7.27740048e-02	6.13052445e-02
70	0.01428571	9.05074559e-02	6.16333152e-02
geschätzte Konvergenzordnung		0.9699	1.1044

**Tabelle 5.11:** mengenwertiges Euler-Verfahren auf einem ( $\rho = h$ )-Gitter mit linearer Interpolation der Distanzfunktion

N	$(\rho = h^2)$ -Gitter	$(\rho = h)$ -Gitter mit linearer Interpolation
5	$\leq$ 1s	$\leq$ 1s
10	1s	$\leq$ 1s
15	3min 34s	5s
20	42min 34s	54s
25	4h 59min 9s	2min 12s
30	24h 36min 33s	4min 21s
35	100h 37min 6s	43min 26s
40	-	1h 14min 14s
50	-	3h 9min 28s
60	-	6h 52min 49s
70	-	77h 47min 15s

**Tabelle 5.12:** Berechnungsdauer der erreichbaren Mengen

N	Zeit für Eulerschritte	Zeit für linearer Interpolation	Gesamtzeit
20	21s	32s	54s
30	2min 15s	2min 4s	4min 21s
40	31min 18s	42min 45s	1h 14min 14s
50	1h 31min 6s	1h 37min 52s	3h 9min 28s
60	3h 39min 21s	3h 12min 42s	6h 52min 49s
70	30h 15min 26s	47h 27min 30s	77h 47min 15s

**Tabelle 5.13:** detaillierte Berechnungsdauer der erreichbaren Mengen für Gitterweite  $(\rho = h)$  mit linearer Interpolation der Distanzfunktion

## 6 Fazit und Ausblick

Im Rahmen dieser Arbeit wurde gezeigt, dass die Berechnung erreichbarer Mengen mit linearer Interpolation von Distanzfunktionen durch baryzentrische Koordinaten deutlich effizienter ist als das mengenwertige Euler-Verfahren auf einem Gitter mit der Gitterweite ( $\rho = h^2$ ). Die lineare Konvergenz des mengenwertigen Euler-Verfahrens bleibt dabei aufgrund der linearen Interpolation, trotz der Auswertungen der Distanzfunktion auf einem ( $\rho = h$ )-Gitter, erhalten.

Bei der Implementierung des Verfahrens zeigten sich jedoch einige unvorhersehbare Schwierigkeiten, die jedoch durch leichte Einbußen bei der Effizienz behoben werden konnten. So stellte sich beispielsweise bei den Berechnungen in Beispiel 5.4 (Rayleigh-Problem) heraus, dass die Aufblähung der Menge in jedem Eulerschritt durchzuführen ist und eine einmalige Aufblähung im ersten Iterationsschritt dazu führen kann, dass man fälschlicherweise eine leere Menge als erreichbare Menge erhält.

Beispiel 5.5 (bilineares Beispiel) zeigte zusammen mit dem Rayleigh-Problem, dass nicht nur die Wahl des Gitters durch Vorgabe der Bounding-Box, auf der die erreichbare Menge berechnet wird, sondern auch der Aufblähungsparameter in den jeweiligen Eulerschritten vor der linearen Interpolation der Distanzfunktion entscheidend für die Berechnung der erreichbaren Menge ist. Es stellte sich heraus, dass ein weiterer Rechenschritt für die Korrektur der Aufblähung nach der linearen Interpolation benötigt wird.

Betrachtet man die ausführliche Laufzeit für die Berechnung der erreichbaren Mengen mit linearer Interpolation, so stellt man fest, dass die lineare Interpolation, je nach Beispiel, etwa genau so lange benötigt wie der durchgeführte Eulerschritt zuvor. Die Implementierung der linearen Interpolation besteht dabei aus zwei Schritten:

1. Die eigentliche lineare Interpolation bei der die Distanzfunktionswerte auf dem ( $\rho \leq h^2$ )-Gitter durch baryzentrische Koordinaten interpoliert werden.
2. Das Korrigieren des Aufblähungsparameters aus dem vorherigen Eulerschritt.

Nähere Analysen zeigen, dass beinahe die gesamte Zeit eines Interpolationsschrittes für die Korrektur des Aufblähungsparameters benötigt wird, da hierfür zunächst die Distanzen innerhalb der Menge neu berechnet werden müssen. Die heuristische Annahme in Kapitel 4.4 und 4.5, bei der Implementierung des Verfahrens der Einfachheit halber

$$b(g_\rho, \tilde{X}_{i+1}^h) = h \cdot \min_{\eta_i \in X_i^{h,\rho}} b\left(\frac{g_\rho - \eta_i}{h}, F(t_i, \eta_i)\right)$$

und

$$b(g_\rho, \tilde{X}_{i+1}^{h,r}) = b(g_\rho, \tilde{X}_{i+1}^h) - r$$

zu verwenden, wird an dieser Stelle mit einer erhöhten Berechnungszeit erkaufte.

Die Korrektur des Aufblähungsparameters bzw. die Aufblähung einer Menge bietet somit noch Raum für weitere Untersuchungen, um die Effizienz des vorgestellten Verfahrens noch zusätzlich zu steigern. Das Ausrichten des Gitters an der Startmenge wäre zudem eine interessante Erweiterung der Implementierung.

Weiterhin bietet der Ansatz, Methoden der Interpolation zu nutzen, um mengenwertige Berechnungen auf einem größeren Gitter durchzuführen, die Möglichkeit für künftige Forschungsarbeiten. In Bezug auf die erreichte Genauigkeit sind Verfahren höherer Konvergenzordnung mit größeren Schrittweiten im Allgemeinen besser als Verfahren mit niedriger Konvergenzordnung und feineren Schrittweiten, wie man beispielsweise in [Cha03, Tabelle 5.1] und [Cha03, Tabelle 5.2] vergleichen kann. Eine gröbere Schrittweite führt darüber hinaus zu einer verminderten Anzahl an Auswertungen der Distanzfunktion, was sich wiederum deutlich in der Rechenzeit bemerkbar macht.

Mengenwertige Runge-Kutta-Verfahren höherer Ordnung könnten demnach mit Interpolationsmethoden höherer Ordnung, beispielsweise Spline-Interpolation oder auch Kriging (stochastische Interpolation) [Heb05], verknüpft werden, um die Verfahren zur Berechnung erreichbarer Mengen effizienter zu realisieren.

# A Anhang

## A.1 CD-Inhalt

Ordner: "Beispiele"

Enthält die Berechnungen sämtlicher Beispiele der Diplomarbeit. Die Datei "Beispiele/Hausdorff-distances.ods" beinhaltet alle berechneten Hausdorff-Abstände der erreichbaren Mengen und die geschätzten Konvergenzordnungen, für die Beispiele aus Kapitel 5. Die Datei ist im *OpenDocument-Spreadsheet* Dateiformat mit *OpenOffice.org*<sup>TM</sup> 3.2 erstellt worden.

Die berechneten Mengen des jeweiligen Beispiels sind in den Unterordner

- "example\_1",
- "example\_rayleigh" und
- "example\_bilinear"

zu finden. Die Unterordner innerhalb der Beispiele enthalten wiederum die folgenden Berechnungen:

- Unterordner "0(h)" enthält die Berechnungen der erreichbaren Menge auf einem ( $\rho = h$ )-Gitter ohne Interpolation der Distanzfunktion.
- Unterordner "0(hh)" enthält die Berechnungen auf einem ( $\rho = h^2$ )-Gitter ohne Interpolation der Distanzfunktion.
- Unterordner "linint\_0(hh)" enthält die Berechnungen der erreichbaren Menge auf einem ( $\rho = h$ )-Gitter mit linearer Interpolation der Distanzfunktion auf ein ( $\rho = h^2$ )-Gitter.
- Unterordner "linint\_keep\_gridpoints" enthält die Berechnungen auf einem ( $\rho = h$ )-Gitter mit linearer Interpolation der Distanzfunktion auf ein ( $\rho \leq h^2$ )-Gitter, siehe Bemerkung auf Seite 90.

Darstellungen der berechneten Mengen findet man, für einige ausgewählte Schrittweiten, in den Unterordner "png". Die vom Programm "set\_valued\_Euler\_v2" erstellten BMP-Dateien, zur Veranschaulichung der Berechnungen, wurden dabei aufgrund der Größe mit Hilfe der Programme "convert" und "mogrify" aus dem freien Softwarepaket *ImageMagick*<sup>®</sup> in PNG-Dateien konvertiert.

#### Ordner: "Programme"

Enthält das vorkompilierte Programmpaket für eine 64-bit-Linux-Plattform mit den Einlesedateien für verschiedene Beispiele. Die Parallelisierung der Anwendungen wurde mit Hilfe der Bibliothek "libboost\_thread" aus der freien C++-Bibliothek: *Boost C++ Libraries* ([www.boost.org](http://www.boost.org)) in der Version 1.48.0 realisiert. Diese wird im Ordner "Programme/lib64" mitgeliefert.

Zu Demonstrationszwecken findet man im Unterordner "Programme/Demonstration" verschiedene vorkompilierte Anwendungen, mit denen die Berechnung erreichbarer Mengen, aufgrund bereits erwähnter Schwierigkeiten, nicht erfolgreich durchgeführt werden kann. So ist beispielsweise im Ordner "Programme/Demonstration/einmahliges Aufblaehen" eine Version mit einmaliger Aufblähung der Menge im ersten Eulerschritt abgelegt. Startet man diese Version mit der Einlesedatei "example\_rayleigh" für  $N = 15$  auf einem  $(\rho = h^2)$ -Gitter, so stellt man fest, dass die Berechnung der erreichbaren Menge nach dem achten Eulerschritt abbricht, da ohne Aufblähung der Menge, wie in Kapitel 4.2.1 beschrieben, die berechnete Menge zwischen den Gitterpunkten des  $(\rho = h^2)$ -Gitters liegt. Die Übergabeparameter zum Starten der Programme werden in Anhang A.2 näher erläutert.

#### Ordner: "Quellcode"

Enthält den jeweiligen C++-Quellcode der beigefügten Anwendungen. Das Kompilieren einer Anwendung erfolgt im "Release" Unterordner durch Aufruf des Build-Management-Tools "make".

## A.2 Beschreibung und Aufruf der Programme

Programm: "Hausdorff\_distance\_2d"

Berechnet den Hausdorff-Abstand für zwei nichtleere und kompakte Mengen  $A$  und  $B$ . Die Berechnung erfolgt dabei approximativ durch den Hausdorff-Abstand der Randpunkte beider Mengen:

$$d_H(A, B) \approx d_H(\partial A, \partial B)$$

Diese Approximation lässt sich nicht für beliebige Mengen rechtfertigen, sondern nur für konvexe Mengen und für Mengen mit gleicher Struktur, was beim Vergleich von berechneten Mengen in Kapitel 5 zur Referenzmenge erfüllt ist, siehe dazu auch [Cha03, Seite 103].

Der Aufruf erfolgt durch:

```
"Hausdorff_distance_2d [FILENAME_OF_SET_A] [FILENAME_OF_SET_B]"
```

wobei "[FILENAME\_OF\_SET\_A]" bzw. "[FILENAME\_OF\_SET\_B]" für die Einlese-dateien der Mengen  $A$  und  $B$  steht, berechnet und abgespeichert durch das Programm "set\_valued\_Euler\_v\*".

Programm: "Hausdorff\_distance\_2d\_convex"

Berechnet den Hausdorff-Abstand für zwei nichtleere, kompakte und konvexe Mengen  $A$  und  $B$ . Die Berechnung erfolgt dabei, nach Satz 5.3, mit Hilfe von Stützfunktionen  $\delta^*(l, \cdot)$ , durch:

$$d_H(A, B) = \sup_{l \in B_1(0)} |\delta^*(l, A) - \delta^*(l, B)|$$

Der Aufruf erfolgt durch:

```
"Hausdorff_distance_2d_convex [FILENAME_OF_SET_A] [FILENAME_OF_SET_B]"
```

wobei "[FILENAME\_OF\_SET\_A]" bzw. "[FILENAME\_OF\_SET\_B]" für die Einlese-dateien der Mengen  $A$  und  $B$ , im DSB-Dateiformat, steht. Zu diesem Zweck lassen sich die berechneten Mengen aus dem Programm "set\_valued\_Euler\_v\*" mit Hilfe des Programm "set2dsb" in das DSB-Dateiformat konvertieren.

*Bemerkung.* Bei der Berechnung der Hausdorff-Abstände in Beispiel 5.1 wurden die Ergebnisse sowohl mit 200 Richtungen als auch mit 20000 Richtungen der Stützfunktion ausgewertet. Dabei wurden keine nennenswerte Unterschiede der Hausdorff-Abstände festgestellt. Damit zeigt sich, dass die Berechnung

der Hausdorff-Abstände in Beispiel 5.1 selbst mit sehr präzisen Mengen, wie der Referenzmenge mit  $N = 1\,000\,000$ , kaum von der Anzahl der Richtungen abhängt und bereits mit 200 Richtungen gute Resultate liefert.

Programm: "Hausdorff\_distance\_2d\_full"

Berechnet den Hausdorff-Abstand für zwei nichtleere und kompakte Mengen  $A$  und  $B$ . Die Berechnung erfolgt dabei, nach Definition 3.12, durch:

$$d_H(A, B) = \max\{d(A, B), d(B, A)\}$$

Der Aufruf erfolgt durch:

```
"Hausdorff_distance_2d_full [FILENAME_OF_SET_A] [FILENAME_OF_SET_B]"
```

wobei "[FILENAME\_OF\_SET\_A]" bzw. "[FILENAME\_OF\_SET\_B]" für die Einlese-dateien der Mengen  $A$  und  $B$  steht, berechnet und abgespeichert durch das Programm "set\_valued\_Euler\_v\*".

Programm: "set2bmp"

Konvertiert, zur Veranschaulichung der berechneten Menge, eine SET-Datei aus dem Programm "set\_valued\_Euler\_v\*" in eine BMP-Datei. Der Aufruf erfolgt durch:

```
"set2bmp [FILENAME_OF_SET] [SAVE_AS_FILENAME]"
```

wobei "[FILENAME\_OF\_SET]" für eine SET-Datei, abgespeichert durch das Programm "set\_valued\_Euler\_v\*" und "[SAVE\_AS\_FILENAME]" für den Dateinamen der BMP-Datei steht.

Programm: "set2dsb"

Konvertiert, zur Berechnung des Hausdorff-Abstands von konvexen Mengen, eine SET-Datei aus dem Programm "set\_valued\_Euler\_v\*" in eine DSB-Datei. Der Aufruf erfolgt durch:

```
"set2dsb [FILENAME_OF_SET]"
```

wobei "[FILENAME\_OF\_SET]" für eine SET-Datei des Programms "set\_valued\_Euler\_v\*" steht.

Programm: "mat2set.m"

MATLAB-Programm zum Konvertieren von MAT-Dateien, berechnet durch das Verfahren höherer Ordnung beschrieben in [GJ12], in SET-Dateien.

Programm: "set\_valued\_Euler\_v1"

Berechnet die erreichbare Menge eines Kontrollsystems mit dem mengenwertigen Euler-Verfahren nach Algorithmus 1 (Seite 49). Der Aufruf erfolgt durch:

```
"set_valued_Euler_v1 [OPTION]... [FILENAME]"
```

mit den Optionen

- "--debug\_off": schaltet die Ausgabe erweiterter Informationen aus.
- "--save\_bmp": speichert nach jedem Eulerschritt die berechnete Menge, zur Veranschaulichung, als BMP-Datei ab.
- "--save\_distances": speichert nach jedem Eulerschritt die Distanzfunktionswerte der berechneten Menge als MATLAB-Datei ab.
- "--adaptive\_grid": führt die Berechnung der erreichbaren Menge auf einem adaptiven Gitter mit Gittererweiterung und (DTSP) durch.

und "[FILENAME]" als Einlesedatei des entsprechenden Kontrollsystems, für die Berechnung der erreichbaren Menge.

Bei der Berechnung auf einem adaptiven Gitter wird die Menge in jedem Eulerschritt nicht auf dem gesamten Gitter, vorgegeben durch eine Bounding-Box in der Einlesedatei, berechnet, sondern nur auf einem Teil des Gitters. Der Teil des Gitters auf dem die Distanzfunktion ausgewertet wird, wird dabei durch die berechnete Menge des vorherigen Eulerschritts bestimmt. Falls das Teil-Gitter zu klein für die Menge im aktuellen Eulerschritt ist, wird es solange in die entsprechende Richtung erweitert, bis es größer ist als die Menge bzw. bis zum Rand der vorgegebenen Bounding-Box. Für weitere Informationen zur Gittererweiterung sei hier auf die Kommentare im Quellcode der Methode "euler()" hingewiesen.

*Bemerkung.* Die Berechnung der erreichbaren Menge auf einem adaptiven Gitter kann nur für zusammenhängende Mengen durchgeführt werden, da bei nicht zusammenhängenden Mengen im Allgemeinen nur ein Teil der Menge durch das adaptive Gitter erfasst wird.

Darüber hinaus wird bei der Übergabe des Parameters "`--adaptive_grid`" das sogenannte „DTSP“ (use distance to skip points), zur effizienteren Implementierung von Algorithmus 1, verwendet. Dabei werden nach der Auswertung der Distanzfunktion  $b(g_\rho, \cdot)$  für einen Gitterpunkt  $g_\rho$ , alle benachbarten Gitterpunkte, die innerhalb des Balls um  $g_\rho$  mit Radius  $b(g_\rho, \cdot)$  liegen, markiert und bei der Auswertung der Distanzfunktion übersprungen. (DTSP) beschleunigt die Berechnung der erreichbaren Menge durch Algorithmus 1 zwar spürbar, ist jedoch in der implementierten Form nicht für Algorithmus 2 geeignet, da die Distanzfunktion für sehr viele Gitterpunkte nicht ausgewertet wird und diese somit nicht für die lineare Interpolation der Distanzfunktion zur Verfügung stehen.

Die Einlesedatei für die Programme "`set_valued_Euler_v1`" und "`set_valued_Euler_v2`", die auch zum Abspeichern der berechneten Mengen verwendet wird, hat folgendes Format:

```
example_1
```

1	10 1
2	0.0 2.0
3	0 0.0 0.0 0.0 0.0 0.0 0.0
4	-2.5 2.5
5	-2.5 2.5
6	0 0 1
7	1 0.0 0.0 1.0 0.0 0.0 0.0
8	1
9	- - - - - stored set - - - - -

Zeile: Beschreibung:

- 
- 1 Enthält die Anzahl der Eulerschritte ( $N$ ) und der Eulerschritt mit dem die Berechnung gestartet wird. Ist der zweite Wert größer als 1, dann wird die abgespeicherte Menge des vorherigen Eulerschritts ab Zeile 10 eingelesen und die Berechnung fortgesetzt.
  - 2 Besteht aus der Startzeit  $t_0$  und der Endzeit  $T$ .
  - 3 Beinhaltet den Funktionszeiger für die Startmenge, gefolgt von sechs Parameter für diese Funktion. Der Wert 0 steht beispielsweise für einen Startpunkt mit den Koordinaten in den ersten beiden Parameter (restliche Parameter werden ignoriert). Im obigen Beispiel wäre die Startmenge ein Punkt mit den Koordinaten  $(0, 0)$ .
  - 4 Gibt die Grenzen des Gitters in  $x$ -Richtung an.
  - 5 Gibt die Grenzen des Gitters in  $y$ -Richtung an.

- 6 Setzt sich aus der Anzahl der Gitterpunkte in  $x$ -Richtung und  $y$ -Richtung zusammen. Sind beide Werte 0 so wird durch den dritten Wert in der Zeile die Anzahl der Gitterpunkte automatisch bestimmt. Ist der dritte Wert 0, so wird ein  $\mathcal{O}(h)$ -Gitter erstellt. Ist der Wert, wie im obigen Beispiel, 1 so wird ein  $\mathcal{O}(h^2)$ -Gitter erstellt.
- 7 Beinhaltet den Funktionszeiger für die Distanzfunktion, gefolgt von sechs Parameter für diese Funktion. Der Wert 1 steht für die Distanz zu einem Kreis mit Mittelpunkt und Radius in den folgenden drei Parameter. Im obigen Beispiel wird somit die Distanzfunktion zu einem Kreis mit Mittelpunkt  $(0, 0)$  und Radius 1 (also dem Einheitskreis) gesetzt.
- 8 Enthält den Funktionszeiger für das ODE-System. Der Wert 1 steht dabei für das ODE-System aus Beispiel 5.1.
- 9 Dient als Abgrenzung zum Abspeichern der berechneten Menge nach einem Eulerschritt.

Programm: "set\_valued\_Euler\_v2"

Berechnet die erreichbare Menge eines Kontrollsystems mit dem mengenwertigen Euler-Verfahren und Interpolation der Distanzfunktion nach Algorithmus 2 (Seite 58). Der Aufruf erfolgt durch:

```
"set_valued_Euler_v2 [OPTION]... [FILENAME]"
```

mit den Optionen

- "--debug\_off": schaltet die Ausgabe erweiterter Informationen aus.
- "--save\_bmp": speichert nach jedem Eulerschritt die berechnete Menge, zur Veranschaulichung, als BMP-Datei ab.
- "--save\_distances": speichert nach jedem Eulerschritt die Distanzfunktionswerte der berechneten Menge als MATLAB-Datei ab.
- "--linint": führt nach jedem Eulerschritt die lineare Interpolation der Distanzfunktion mit baryzentrischen Koordinaten auf einem  $(\rho = h^2)$ -Gitter durch.
- "--linint\_keep\_gridpoints": führt nach jedem Eulerschritt die lineare Interpolation der Distanzfunktion mit baryzentrischen Koordinaten auf einem  $(\rho \leq h^2)$ -Gitter, unter Beibehaltung der ursprünglichen Gitterpunkte mit zugehörigen Distanzfunktionswerten, durch.

- `--adaptive_grid`: führt die Berechnung der erreichbaren Menge auf einem adaptiven Gitter mit Gittererweiterung durch.

und `[FILENAME]` als Einlesedatei des entsprechenden Kontrollsystems, für die Berechnung der erreichbaren Menge.

*Bemerkung.* Bei den Berechnungen der erreichbaren Menge in Beispiel 5.5 stellt man fest, dass die lineare Interpolation der Distanzfunktion auf einem  $(\rho = h^2)$ -Gitter zu einer schlechteren Konvergenz des Verfahrens führt. Die Ursache liegt in der Wahl des  $(\rho = h)$ -Gitters auf dem die Distanzfunktion ausgewertet wird. Die Grenzen des Gitters, vorgegeben durch die Bounding-Box in der Einlesedatei des Programms, werden auch bei der linearen Interpolation beibehalten. Im Allgemeinen werden dadurch jedoch die Werte der bereits ausgewerteten Distanzfunktionen bei der linearen Interpolation auf ein  $(\rho = h^2)$ -Gitter verworfen, da die Gitterpunkte nicht mit denen des  $(\rho = h)$ -Gitters übereinstimmen. Das Programm wurde deshalb um den Parameter `--linint_keep_gridpoints` erweitert, wodurch die lineare Interpolation auf einem  $(\rho \leq h^2)$ -Gitter durchgeführt wird, sodass die ursprünglichen Gitterpunkte des  $(\rho = h)$ -Gitters mit den zugehörigen Distanzwerten beibehalten werden. Die Berechnungen mit linearer Interpolation der Distanzfunktion in Kapitel 5 wurden deshalb mit dem Parameter `--linint_keep_gridpoints` durchgeführt.

### A.3 Implementierung von Algorithmus 2

Zur Berechnung erreichbarer Mengen von Kontrollsystemen wurde die Klasse `CSystem_2d` erstellt. Im folgenden werden einige Eigenschaften der Klasse näher erläutert und die wichtigsten Methoden zur Berechnung erreichbarer Mengen vorgestellt. Zu Beginn werden einige Attribute der Klasse aufgelistet.

## CSystem\_2d.h

```

32 class CSystem_2d {
33
34     public:
35         number steps; // Anzahl der Eulerschritte
36         number current_step; // aktueller Eulerschritt
37         std::string filename; // Name der Einlesedatei
38         grid_2d grid; // Klasse 2D-Gitter
39         real ** ptr_distances; // Matrix fuer Distanzfunktionswerte
40
41     private:
42         const static std::string DEF_string; // default String
43         real t_0, t_max; // Startzeit t_0 und Endzeit t_max
44         real (*init_values)(const real *, parameters &); // Funktionszeiger auf
           Startmenge
45         parameters init_func_param; // Parameter fuer Startmenge-Funktion
46         real h; // Schrittweite h
47         real rho; // Gitterweite
48         real expand_param; // Gitteraufblachungsparameter = rho/sqrt(2)
49
50         real(*dist)(const real *, parameters &, const real *, const real *); //
           Funktionszeiger auf Distanzfunktion
51         parameters dist_func_param; // Parameter der Distanzfunktion
52         void (*ode_system)(real *, const real *, const real &); // Funktionszeiger
           auf ODE-System
53
54         set_2d set; // Klasse 2D-Menge
55         set_2d set_copy;
56         ...

```

Die berechnete Menge wird nach jedem Eulerschritt in der Membervariable "set" der Klasse "set\_2d" abgespeichert. Diese besteht aus Gitterinformationen der Menge, einem Vektor von Indizes aller Gitterpunkte  $g_\rho$ , mit  $g_\rho \in \mathcal{R}_{N,\rho}$ , und den minimalen und maximalen Index-Werten. Die minimalen und maximalen Index-Werte werden z.B. bei der Berechnung erreichbarer Mengen auf einem adaptiven Gitter verwendet.

*Bemerkung.* Der Zugriff auf die Vektorelemente, in denen Gitterpunkte von diskretisierter Menge abgespeichert werden, erfolgt in der Implementierung, aus Performance-Gründen, bewußt über den Array-Zugriffoperator "[]". Der Zugriff über die Member-Methode "vector::at()" ist zwar sicherer, da diese bei fehlerhaften Zugriffen eine Ausnahmebedingung (exception) wirft, jedoch auch spürbar langsamer, so dass die Berechnung erreichbarer Mengen noch länger benötigt. Darüber hinaus wurden bei der Implementierung der Algorithmen, unter anderem, soweit wie möglich dynamische Arrays benutzt und alle rechenintensiven Methoden parallelisiert, um eine möglichst effiziente Implementierung zu erreichen. Für weitere Einzelheiten zur effizienten Implementierung wird hier auf die Kommentare im Quellcode der Programme verwiesen.

## set\_2d.h

```

21 class set_2d {
22     public:
23
24         std::vector<xy> component;
25         number MAXx, MAXy, MINx, MINy;
26         grid_2d grid;
27
28         set_2d();
29         virtual ~set_2d();
30
31         void clear(void);
32         void push_back(const number &x_index, const number &y_index);
33
34 };

```

Für das Gitter wurde wiederum eine Klasse erstellt, bestehend aus der vorgegebenen Bounding-Box der Einlesedatei, der Anzahl der Gitterpunkte in  $x$ - und  $y$ -Richtung und dem Abstand zwischen den Gitterpunkten.

## grid\_2d.h

```

13 class grid_2d {
14
15     public:
16         // ----- 2D-Gitter -----
17         real start_x, end_x; // Gitterinitialisierung in x-Richtung
18         real start_y, end_y; // Gitterinitialisierung in y-Richtung
19         real delta_x, delta_y; // Entfernung zwischen Gitterpunkten
20         number points_x, points_y; // Anzahl der Gitterpunkte
21
22         grid_2d();
23         virtual ~grid_2d();
24
25         void set(double const &s_x, double const &e_x, double const &s_y, double
                const &e_y, number const &p_x,
26                 number const &p_y);
27         void set_points_x(number const &p_x);
28         void set_points_y(number const &p_y);
29 };

```

Die wichtigsten Methoden der Klasse "CSystem\_2d", für die Berechnung erreichbarer Mengen, lauten:

## CSystem\_2d.h

```

94     ...
95     void init_euler(bool use_hh_grid, bool keep_old_gridpoints);
96     void euler(bool adaptive_grid);
97     void linear_interpolation(bool use_hh_grid, const number _new_points_x, const
                number _new_points_y, bool keep_old_gridpoints = false);
98     void calc_dist_complement(const unsigned int &dim_x, const unsigned int &
                dim_y, real ** distances, bool calc_full_grid = false);
99     ...

```

Die Implementierung von Algorithmus 2 zur Berechnung erreichbarer Mengen wird nun an einem Beispiel vorgeführt. Das Programm "set\_valued\_Euler\_v2" wird dafür mit folgenden Parameter gestartet:

```
">./set_valued_Euler_v2 --linint_keep_gridpoints example_rayleigh"
```

Der Name der Einlesedatei "example\_rayleigh" wird im main-Programm in die Variable "filename" gespeichert und damit ein Objekt der Klasse "CSystem\_2d" angelegt:

main.cpp

```
99  ...
100  CSystem_2d example_ray(filename);
101  ...
```

Der entsprechende Konstruktor der Klasse liest nun die Werte aus der Einlesedatei ein und setzt die Werte des Objekts entsprechend. Schritt I.) im Algorithmus 2 erfolgt nun durch Aufruf der Methode "init\_euler()":

main.cpp

```
108  ...
109  example_ray.init_euler(linint , linint_keep_gridpoints);
110  ...
```

Da beim Aufruf des Programms der Parameter `--linint_keep_gridpoints` übergeben wurde, wird die Startmenge in der Methode "init\_euler()" bereits auf einem  $(\rho \leq h^2)$ -Gitter erzeugt.

In einer Schleife werden nun,  $N$  mal, folgende Methoden aufgerufen: Die Berechnung der Distanzfunktionswerte für alle Gitterpunkte des vorgegeben Gitters aus der Einlesedatei, erfolgt in der Methode "euler()". Dabei wird der berechnete Distanzfunktionswert bereits um den Aufblähungsparameter reduziert und somit Schritt II.a) und II.b) von Algorithmus 2 zusammengefasst und innerhalb einer Methode berechnet.

main.cpp

```
137  ...
138  example_ray.euler(adaptive_grid);
139  ...
```

*Bemerkung.* Die Aufblähung in Schritt II.b) sollte einerseits so klein wie möglich gewählt werden, um den Fehler durch Aufblähung der Menge klein zu halten, andererseits muss er jedoch groß genug sein, damit die Projektion der Menge auf das

$\rho$ -Gitter keine leere Menge liefert. Eine sinnvolle Wahl für den Aufblähungsparameter ist somit

$$\varrho = \frac{1}{\sqrt{2}} \cdot \rho.$$

In der Implementierung wird dazu die Membervariable "expand\_param" als Aufblähungsparameter benutzt.

Die lineare Interpolation der Distanzfunktion mit baryzentrischen Koordinaten findet nun in der Methode "linear\_interpolation()" statt.

main.cpp

```

159     ...
160     example_ray.linear_interpolation(linint , 0, 0, linint_keep_gridpoints);
161     ...

```

Wie bereits im Fazit (Kapitel 6) erwähnt, führt die Methode "linear\_interpolation()" sowohl die eigentliche Interpolation der Distanzfunktion (Schritt II.c)), als auch die Korrektur des Aufblähungsparameters (Schritt II.c\*)), durch Aufruf der Methode "calc\_dist\_complement()", durch. Die anschließende Projektion der Menge auf das entsprechende Gitter (Schritt II.d)), erfolgt ebenfalls in der Methode "linear\_interpolation()". Damit erhält man nach dem ausführen der Methode bereits die diskrete Menge mit Interpolation der Distanzfunktion im aktuellen Eulerschritt auf einem ( $\rho \leq h^2$ )-Gitter und damit auch die Voraussetzung für den nächsten Iterationsschritt.

*Bemerkung.* Für die Berechnung der baryzentrischen Koordinaten bei der Interpolation der Distanzfunktion muss ein lineares Gleichungssystem gelöst werden. Aufgrund der äquidistanten Anordnung der Gitterpunkte, kann dies jedoch im Vorfeld erfolgen und damit das wiederholte lösen desselben linearen Gleichungssystems entfallen. Bei der Implementierung der Methode "linear\_interpolation()" wurde deshalb das entsprechende Gleichungssystem einmal per Hand gelöst, sodass die baryzentrischen Koordinaten innerhalb der Methode, für jeden Gitterpunkt, ohne das Lösen eines linearen Gleichungssystems direkt berechnet werden können.

Das Abspeichern der berechneten Menge erfolgt anschließend durch die Methoden:

main.cpp

```

180     ...
181     example_ray.store_state();
182
183     example_ray.save_bmp();
184     ...

```

Die Methode `"save_bmp()"` erstellt eine BMP-Datei, zur Veranschaulichung der berechneten Menge.

Die Methode `"store_state()"` erstellt eine SET-Datei mit sämtlichen Informationen des Kontrollsystems und der berechneten Menge. Die SET-Datei kann anschließend z.B. zur Berechnung des Hausdorff-Abstands durch das Programm `"Hausdorff_distance_2d"` verwendet werden. Die Menge wird dabei, durch eine einfache Lauflängenkodierung des Gitters, komprimiert abgespeichert. Für weitere Einzelheiten zur Komprimierung der Menge wird hier auf die Kommentare im Quellcode der Methode `"store_state()"` verwiesen.

# Literaturverzeichnis

- [AC84] AUBIN, J.-P. und A. CELLINA: *Differential Inclusions*. Springer-Verlag, Berlin Heidelberg New York Tokyo, Grundlehren der mathematischen Wissenschaften 264 Auflage, 1984.
- [Bai95] BAIER, R.: *Mengenwertige Integration und die diskrete Approximation erreichbarer Mengen*.  
Doktorarbeit, University of Bayreuth, Bayreuther Mathematische Schriften 50, 1995.
- [BBCG07] BAIER, R., C. BÜSKENS, I. A. CHAHMA und M. GERDTS: *Approximation of reachable sets by direct solution methods for optimal control problems*.  
Optimisation Methods and Software, 22(3):433–452, 2007.
- [Ber04] BERGER, M.: *Geometry I*.  
Springer-Verlag, 2004.
- [BG10] BAIER, R. und M. GERDTS: *Approximation of Reachable Sets using Optimal Control Algorithms and Distance Functions*.  
Vortrag am 23.06.2010 im Gemeinsamen Seminar des Internationalen Doktorandenkollegs im Elite-Netzwerk Bayern „Identifikation, Optimierung und Steuerung für technische Anwendungen“ Universität Würzburg, 2010.
- [BGX11] BAIER, R., M. GERDTS und I. XAUSA: *Approximation of Reachable Sets using Optimal Control Algorithms*.  
University of Bayreuth, Oktober 2011.
- [BM98] BÜSKENS, C. und H. MAURER: *Sensitivity Analysis and Real-Time Control of Nonlinear Optimal Control Systems via Nonlinear Programming Methods*.  
International Series of Numerical Mathematics, 124:185–196, 1998.
- [BR07] BEYN, W.-J. und J. RIEGER: *Numerical fixed grid methods for differential inclusions*.  
Computing, 81(1):91–106, 2007.
- [Bra05] BRADLEY, C. J.: *Challenges in Geometry: for Mathematical Olympians past and present*.  
Oxford University Press, 2005.

- [Cha03] CHAHMA, I. A.: *Set-Valued Discrete Approximation of State-Constrained Differential Inclusions*.  
Doktorarbeit, University of Bayreuth, Bayreuther Mathematische Schriften 67, 2003.
- [DF89] DONTCHEV, A. L. und E. M. FARKHI: *Error Estimates for Discretized Differential Inclusions*.  
Computing, 41(4):349–358, 1989.
- [DZ94] DELFOUR, M. C. und J. P. ZOLÉSIO: *Shape Analysis via Oriented Distance Functions*.  
Journal of Functional Analysis, 123(1):129–201, 1994.
- [DZ01] DELFOUR, M. C. und J. P. ZOLÉSIO: *Shapes and Geometries - Analysis, Differential Calculus, and Optimization*.  
SIAM, 2001.
- [Fil88] FILIPPOV, A. F.: *Differential Equations with Discontinuous Right-hand Sides*.  
Mathematics and Its Applications (Soviet Series), Kluwer Academic Publishers, 1988.
- [GJ12] GRÜNE, L. und T. JAHN: *Computing reachable sets via barrier methods on SIMD architectures*.  
Mathematical Institute, University of Bayreuth, 2012.
- [Grü06] GRÜNE, L.: *Mathematische Kontrolltheorie II: Nichtlineare Systeme*.  
Universität Bayreuth Vorlesungsskript, Sommersemester 2006.
- [Grü09] GRÜNE, L.: *Mathematische Kontrolltheorie I: Lineare Systeme*.  
Universität Bayreuth Vorlesungsskript, Wintersemester 2008/2009.
- [Häc96] HÄCKL, G.: *Reachable Sets, Control Sets and Their Computation*.  
Doktorarbeit, University of Augsburg, Augsburger Mathematisch-Naturwissenschaftliche Schriften 7, 1996.
- [Han95] HANDSCOMB, D. C.: *Errors of Linear Interpolation on a Triangle*.  
Oxford University Computing Laboratory, 1995.
- [Har96] HART, J. C.: *Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces*.  
The Visual Computer, 12:527–545, 1996.
- [Heb05] HEBELER, F.: *Interpolationsmethoden in ArcGIS 9*.  
[http://www.geo.unizh.ch/~fhebeler/interpolation\\_gis2.pdf](http://www.geo.unizh.ch/~fhebeler/interpolation_gis2.pdf),  
Vorlesung GIS2, Wintersemester 2004/2005.
- [HP97] HU, S. und N. S. PAPAGEORGIOU: *Handbook of Multivalued Analysis: Theory*.  
Kluwer Academic Publishers, 1997.
- [KF86] KURZHANSKI, A. B. und T. F. FILIPPOVA: *On the set-valued calculus in problems of viability and control for dynamic processes: the evolution equation*.

- Working Paper WP-88-091, Institute for Applied Systems Analysis, Laxenburg, Austria, 1986.
- [Kur66] KURATOWSKI, K.: *Topology Volume I*. Academic Press, 1966.
- [Lem98] LEMPIO, F.: *Numerische Mathematik II: Methoden Der Analysis*. Bayreuther Mathematische Schriften, 55, 1998.
- [LV98] LEMPIO, F. und V. VELIOV: *Discrete Approximations of Differential Inclusions*. Bayreuther Mathematische Schriften, 54:149–232, 1998.
- [Mal08] MALER, O.: *Computing Reachable Sets: An Introduction*. Technischer Bericht, French National Center of Scientific Research, <http://www-verimag.imag.fr/~maler/Papers/reach-intro.pdf>, 2008.
- [Mar77] MARTI, J. T.: *Konvexe Analysis*, Band Mathematische Reihe 54. Birkhäuser, Basel, 1977.
- [Nik87] NIKOL'SKII, M. S.: *Approximation of the attainability set for a controlled process*. Mathematical Notes, 41(1):44–48, 1987.
- [Rod10] RODLER, MATTHIAS: *Mengenwertige Runge-Kutta-Verfahren und Distanzfunktionen*. Seminararbeit im Hauptseminar: Numerische Mathematik, Dezember 2010. Universität Bayreuth.
- [Ska08] SKALA, V.: *Barycentric coordinates computation in homogeneous coordinates*. Computers & Graphics, 32(1):120–127, 2008.
- [Son98] SONTAG, E.D.: *Mathematical Control Theory*. Springer New York, 1998.
- [spe08] *Spektrum - das Wissenschaftsmagazin der Universität Bayreuth*, 1/2008. [www.uni-bayreuth.de](http://www.uni-bayreuth.de).
- [SvV65] SONNEBORN, L.M. und F. S. VAN VLECK: *The bang-bang principle for linear control problems*. SIAM J. Control, Ser. A, 2(2):151–159, 1965.

# Erklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keinem anderen Prüfungsverfahren vorgelegt.

Bayreuth, den 28. Februar 2012

---

EDUARD LEIPI