

DEUTSCHE GEODÄTISCHE KOMMISSION
bei der Bayerischen Akademie der Wissenschaften

Reihe C

Dissertationen

Heft Nr. 575

Andrzej Borkowski

Modellierung von Oberflächen mit Diskontinuitäten

München 2004

Verlag der Bayerischen Akademie der Wissenschaften
in Kommission beim Verlag C. H. Beck

ISSN 0065-5325

ISBN 3 7696 5014 X

Summary

The laser scanning provides very dense information about the surface to be modelled in the form of a irregularly distributed points cloud $\{x, y, z\} \subset \mathbb{R}^3$. Such dense information makes possible an efficient modelling of characteristic structures of the terrain surface like discontinuities, which are necessary for high-qualitative description of the surface. Simultaneously, the points not belonging to the modelled surface (for example: reflexes from buildings, trees etc.) stand a very important influence on the obtained data. During the modelling process, such data should be effectively filtered from the whole data set.

The laser scanning data can be efficiently elaborated by the use of deformable models of curves and surfaces. These models base on the physical principle of energy-minimizing and are presented as the solution of variational problem. The total energy consists both of internal and external energy. The external energy, depending on a context is generated by the data; in most cases it describes a deviation between the data and a model. The internal energy describes geometrical properties of curves and is characterized by elasticity and viscosity. Both terms are mutually weighted by the local control parameters α and β . Varying the parameters makes it possible to stretch the curves to a geometrical shapes. The snake-approximation is used for a profiled modelling of surfaces. Due to that, a formulation of external energy was proposed making possible a robust modelling of profiles: during an iterative process, gross errors can be filtered, measuring errors can be smoothed and discontinuities can be preserved. Fitting the snakes-models to the data runs iteratively, however the control parameters depending on the data are being spread.

By generalizing the snakes, the model is introduced by sufficient smoothes, energy-charged pieces of a surface and furthermore described by flakes. The internal energy within the flakes model consists of a membrane and a thin-plate kernel which describes the inclination and curvature properties of the modelled terrain surface. The energy pieces will furthermore be weighted by the local control parameters α and β . A minimizing of the total flakes energy leads again to the variational problem which had been differently solved. By formulate the Euler equations and their further diskretizing by finite differences, the flakes-model stands for regular data. The previous variational problem will also be solved by the use of so-called Ritz method. The improved flakes model was developed for regular data by using a linear base function. However, for the irregular data the flakes model was modified by the use of a Gaussian function. The modelling of the data by flakes runs iteratively. By using the flakes model for regular data it is possible to reject the gross errors, also to smooth the noise by simultaneous preserving the form of edges.

In many applications the information about spatial location of terrain edges is needed. To present such spatial location description of edges in a vector format based on irregular points cloud $\{x, y, z\}$ obtained during laser scanning, it was proposed to describe a gross-errors-free data by surface functions and to average the edges as a intersection of two surfaces $z_i = f_i(x, y); i = 1, 2$. To this purpose, all the data should furthermore be ordered in separate pieces of the surface. This problem can be solved by using the standard methods of image processing. The projection $\{x(s), y(s)\}$ of edges is found in the xy coordinate plane and the z -coordinates consist of $f_i(x(s), y(s))$. To the intersection of two surfaces relates: $f_1(x(s), y(s)) = f_2(x(s), y(s)) = 0$. Based on this condition, two approaches of intersection curve identification were developed. The line-tracking algorithm relies on numerical integration of differential equations relative to the particular problem. For the numerical integration there is a starting point needed. Due to that, a seeking-approach was proposed. Opposite to the local algorithm it was presented a global approach using a snakes-method with a proper definition of external energy. Both algorithms make it possible a reliable, high-accurate identification of terrain edges basing on irregular points cloud.

The algorithms and approaches developed in this work have been tested on real data sets obtained by a laser scanning. Furthermore, a qualitative consideration of a modelling has been given. Finally, some hints for user according the steering and operating of the approaches have been presented.

Inhaltsverzeichnis

Verzeichnis der Tabellen	7
Verzeichnis der Abbildungen	7
1 Einführung und Motivation	9
2 Grundlagen digitaler Geländemodelle	11
2.1 Allgemeines	11
2.2 Datengewinnung	11
2.3 Datenorganisation	12
3 Verfahren zur Oberflächenmodellierung	13
3.1 Approximationsmethoden - eine Übersicht	13
3.1.1 Nächstgelegener-Nachbar-Interpolation	14
3.1.2 Polynomapproximation	14
3.1.3 Shepard-Methoden	15
3.1.4 Multiquadratische Methode	16
3.1.5 Spline minimaler Krümmung	16
3.1.6 Interpolation nach kleinsten Quadraten und Kriging	17
3.1.7 Berücksichtigung von Diskontinuitäten	19
3.2 Robuste Verfahren	20
3.2.1 Allgemeines und M-Schätzer	20
3.2.2 Verfahren für Höhendaten	21
4 Snakes	25
4.1 Snakes - das Modell für deformierbare Kurven	25
4.1.1 Lösungsmöglichkeiten der Energieminimierung	25
4.1.2 Variationsproblem und Euler-Gleichungen	26
4.1.3 Diskretisierung mit finiten Differenzen	26
4.2 Snakes mit lokaler Steuerung	28
4.3 Snakes mit Nebenbedingungen	28
4.4 Iterationsmöglichkeiten	29
4.5 Snakes-Varianten	30
4.6 Filtereigenschaften von Snakes	32
5 Flakes	35
5.1 Variationsproblem und Lösungsmöglichkeiten	35
5.2 Euler-Gleichungen	35
5.2.1 Diskretisierung mit finiten Differenzen	36
5.2.2 Ansatzverfahren	38
5.3 Direkte Lösung des Variationsproblems	39
5.3.1 Linearer Finite-Elemente-Ansatz für reguläre Daten	40
5.3.2 Quasi-Finite-Elemente Ansatz für irreguläre Daten	44
5.4 Filtereigenschaften von Flakes	46
6 Approximation von Profilen mit Snakes	48
6.1 Glatte Profile	48
6.2 Profile mit Diskontinuitäten	50
7 Experimente zur Filterung von Laserscanner-Daten	54
7.1 Hilfsalgorithmen für lineare Gleichungssysteme	54
7.1.1 Iterative Erweiterung des Gleichungssystems	54
7.1.2 Formel von Sherman-Morrison	55
7.1.3 Formel von Woodbury	55
7.2 Unterteilungsalgorithmus für große Datensätze	55
7.3 TPS basiertes Verfahren	56
7.4 Filterung mit Flakes	59

8	Modellierung von Diskontinuitäten in Laserscanner-Daten	60
8.1	Bisherige Ansätze	60
8.2	Approximation mit Flakes	60
8.2.1	Testbeispiele	60
8.2.2	Genauigkeit der Approximation	65
8.3	Kante als Schnitt zweier Oberflächen	66
8.3.1	Einordnung der Daten in die Teiloberflächen	67
8.3.2	Ein Verfolgungsalgorithmus	68
8.3.3	Numerische Ausführung	68
8.3.4	Ein Verfolgungsalgorithmus in Parameterdarstellung	69
8.3.5	Startpunktsuche	70
8.3.6	Kantensuche mit Snakes	71
8.3.7	Genauigkeitsabschätzung	72
8.3.8	Testbeispiele	73
9	Zusammenfassung	80
	Literatur	82
	Anhang	88
A	Ableitungen für die Euler-Gleichungen	88
B	Approximation der Ableitungen mit finiten Differenzen	89
C	Auswertung der Integrale vom Abschnitt 5.3.1	90
D	Auswertung der Integrale vom Abschnitt 5.3.2	91

Verzeichnis der Tabellen

3.1-1	Zusammenstellung der Interpolationsmethoden für irreguläre Daten.	14
3.1-2	Vergleich der Interpolation nach kleinsten Quadraten und der Kriging-Interpolation.	19
5.2-1	Koeffizienten der Euler-Gleichungen mit variablen Steuerparametern.	38
6.2-1	Ausführungsvarianten und Steuerparameter zu den Tests der Abbildung 6.2-1	51
8.3-1	Testbeispiel 1: Genauigkeitsparameter der identifizierten Kanten [in m]	75
8.3-2	Testbeispiel 2: Genauigkeitsparameter der identifizierten Kanten [in m]	76

Verzeichnis der Abbildungen

2.2-1	Datenorganisation in DGM. Gittermodell (links), Dreiecksmodell (Mitte) und hybrides Modell (rechts).	12
3.1-1	Ein Beispiel zur Interpolation nach dem nächstgelegenen Nachbarn. Voronoi-Mosaik mit Datenpunkten und Interpolationsfläche.	15
3.1-2	Eine Kovarianzfunktion (links) und das zugehörige Variogramm (rechts).	18
3.1-3	Zur Approximation von Oberflächen mit Kanten. Messdaten (Punkte) und approximierende Oberflächen F_1 und F_2 . Vektordaten zur Definition der 3D-Geländekante: Messpunkte (große Kreise) und zwischeninterpolierte Punkte (kleine Kreise).	20
3.2-1	Polynomiale Profilapproximation (Ausschnitt) nach kleinsten Quadraten und nach einem robusten Verfahren	21
3.2-2	Die Gewichtsfunktion nach Huber	21
3.2-3	Hierarchische robuste Interpolation (nach BRIESE et al. (2002b)).	23
4.1-1	Irreguläre Verteilung der Stützstellen.	27
4.5-1	Zusammenstellung der Snakes-Varianten	31
4.6-1	Stationäre Durchlasscharakteristik von Snakes mit $\alpha = const, \beta = const$	33
4.6-2	Instationäre Durchlasscharakteristik von Snakes mit dem eingetragenen Verlauf von $\beta, \alpha = const$	33
4.6-3	Phasenverschiebung.	34
5.2-1	Rechteckiges Gitter für die Diskretisierung mit finiten Differenzen.	37
5.3-1	Lineare Basisfunktionen und ihre Ableitungen.	41
5.3-2	Zweidimensionale Basisfunktion für Finite-Elemente-Flakes.	42
5.3-3	Struktur der schwach besetzten Systemmatrix.	44
5.3-4	Zweidimensionale Gaußsche Basisfunktion	44
5.4-1	Stationäre Durchlasscharakteristik von Flakes mit $\alpha = const, \beta = const$	47
5.4-2	Durchlasscharakteristik von Flakes mit variablem $\beta, \alpha = const$	47
6.1-1	Die Ableitung der externen Energie	49
6.1-2	Robuste Profilapproximation mit Snakes - Test 1. (Steuerparameter: $\alpha = 1, \beta = 1, \sigma^2 = 3.5$)	49
6.1-3	Robuste Profilapproximation mit Snakes - Test 2. (Steuerparameter: $\alpha = 5, \beta = 5, \sigma^2 = 0.2$)	49
6.2-1	Snakes-Approximation einer Böschungsfolge, einer Sprung- und einer Dreieckfunktion mit Nachbildung der Diskontinuitäten. Einzelheiten in der Tabelle 6.2-1 sowie im Text	52
6.2-2	Robuste Snakes-Approximation mit Erhaltung von Diskontinuitäten	53
7.2-1	Unterteilung des Approximationsgebietes. Kernbereich und Überlappungsbereiche.	56
7.3-1	Testgebiet Stockholm	57
7.3-2	Testgebiet Stockholm: Aus Bodenpunkten interpolierte Oberfläche	57
7.3-3	Untergebiete und Startpunkte	58
7.3-4	Bodenpunkte	58
7.4-1	Ausschnitt aus dem Testgebiet Stockholm	59
7.4-2	Approximation mit Flakes	59
7.4-3	Bodenpunkte zum Testgebiet von Abb. 7.4-1	59
8.1-1	Zu modellierende Geländekante mit Patches (links) und approximierende Ebenenpaare mit Schnittgeraden und repräsentativen Punkten im \mathbb{R}^3 (rechts) (nach KRAUS und PFEIFER (2001))	61
8.2-1	Synthetisches Testbeispiel: Daten mit überlagertem Rauschen (A), stationäre Approximation mit Flakes (B) und unterschiedliche Varianten der instationären Flakes-Approximation mit Erhaltung der Kante (C), (D), (E) und (F); Einzelheiten im Text	62
8.2-2	Testbeispiel Tagebau	63
8.2-3	Delaunay-Triangulation der Daten mit markierten Kantenpunkten (*)	64
8.2-4	Residuen nach der Filterung ohne Kantenerhaltung	64
8.2-5	Residuen nach der Filterung mit Kantenerhaltung	65

8.3-1	Geländekante als Schnitt zweier Oberflächen	67
8.3-2	Schematische Darstellung der Startpunktsuche	70
8.3-3	Identifikation der Schnittkurve K zweier Oberflächen mit Snake S	71
8.3-4	Modellrechnung: Höhengenaugigkeit einer Schnittlinie für unterschiedliche Abstände dieser Linie von den Datenpunkten	72
8.3-5	Datenpunkte, Startschlange (dünne Linie) und die identifizierte Kante	73
8.3-6	Zwei Oberflächen mit snakes-identifizierter Schnittlinie	73
8.3-7	Testbeispiel 1: Datenpunkte und die identifizierten Kanten	74
8.3-8	Testbeispiel 1: Lage- und Höhenfehler der identifizierten Kanten; Einzelheiten im Text	75
8.3-9	Zur Abschätzung der Lagegenauigkeit; die gemessene (dicke Linie) und die modellierte Kante (dünne Linie)	75
8.3-10	Testbeispiel 2: Datenpunkte und die identifizierten Kanten	77
8.3-11	Testbeispiel 2: Lage- und Höhenfehler der identifizierten Kanten; Einzelheiten im Text	77
8.3-12	Testbeispiel 1: Die innere Höhengenaugigkeit der identifizierten Kanten [in m]	78
8.3-13	Testbeispiel 2: Die innere Höhengenaugigkeit der identifizierten Kanten [in m]	78
8.3-14	Testbeispiel Tagebau mit identifizierten Kanten	79

1 Einführung und Motivation

Das Problem der Oberflächenmodellierung tritt in vielen Bereichen der Geowissenschaften auf. Über Jahrzehnte wurde eine Fülle von analytischen Methoden, vor allem in der Mathematik, entwickelt und an konkrete geowissenschaftliche Aufgabenstellungen angepasst. Die analytischen Modelle erzeugen vorzugsweise kontinuierliche und glatte Oberflächen. Denkt man indessen an die geomorphologische Gestalt der Erdoberfläche, das Relief, so können diese Eigenschaften nur stückweise nachgewiesen werden: Es treten *Diskontinuitäten* natürlicher Herkunft und solche, die künstlich erstanden sind, auf.

Die Diskontinuitäten können sowohl mathematisch als auch morphologisch definiert werden. Aus der mathematischen Sicht, die wichtig für die Modellbildung ist, handelt es sich um Nichtdifferenzierbarkeitsstellen: Die Oberfläche kann sowohl stetig und nicht differenzierbar als auch unstetig und nicht differenzierbar sein. Morphologisch gesehen sind es (Gelände-)Kanten und Formlinien.

Die korrekte Nachbildung solcher *Formlinien* ist von großer Bedeutung für die praktische Nutzung der digitalen Oberflächenmodelle. In Überschwemmungsgebieten beispielsweise ist die Bereicherung der digitalen Gelände-Modelle mit detaillierter Information über Kanten an Böschungen, Dämmen etc. unentbehrlich für das effektive Krisenmanagement. Mit analytischen Modellen, welche sich durch konstante Approximationseigenschaften auf dem gesamten Definitionsbereich auszeichnen, werden solche Kanten mehr oder weniger verschliffen. Der Ausweg hierzu ist, die Modellierung lokal anzuwenden.

Die Diskontinuitäten werden während der Oberflächenerfassung in einem getrennten Vorgang als Vektordaten erfasst und dann als fest zu haltende Randbedingung in den Modellierungsprozess eingeführt. Die Diskontinuitäten teilen das gesamte Gebiet in Teilgebiete auf, die separat mit Berücksichtigung der Randbedingungen modelliert werden. Indessen liefern moderne Verfahren der Datengewinnung, wie *Laserscanning* beispielsweise, sehr dichte diskrete Informationen über die abgetastete Oberfläche. Die Verdichtung der Abtastpunkte in der Lage beträgt gegenwärtig von wenigen Punkten/ m^2 beim flugzeuggetragenen Laserscanner bis zu 1000 Punkte/ m^2 beim terrestrischen Laser in der Nähe des Sensors. Aus so dichter, regellos, aber gleichmäßig in der Ebene verteilter Punktwolke $\{x, y, z\} \subset \mathbb{R}^3$, kann versucht werden, die Diskontinuitäten direkt zu modellieren. Dieses Problem erscheint uns gegenwärtig sehr aktuell und offen zu sein. Versuche, aus den ursprünglichen dreidimensionalen Laserscanner-Daten durch Interpolation und Grauwertbildung ein zweidimensionales "Höhenbild" zu erzeugen und daraus mit Hilfe von Standardmethoden der digitalen Bildverarbeitung Geländekanten zu detektieren, halten wir als unzufriedenstellend. Eine echte dreidimensionale Modellierung von Geländekanten kann nur aufgrund der Originaldaten erfolgen.

Die modernen Messtechniken zeichnen sich durch hohe Zuverlässigkeit und Genauigkeit aus; Zufallsfehler sind gering. Im Gegensatz dazu stellen die systematischen und groben Fehler ein Problem im Datenbearbeitungsprozess dar. Um beim Laserscanning zu bleiben, zählen zu den systematischen Fehlern sowohl flächenhafte Reflexionen am Bewuchs oder an der untereren Vegetation im Walde statt am festen Boden als auch Verschiebung in der Orbitbestimmung. Die letzteren können durch ein geeignetes Modell behoben werden, die ersteren durch entsprechende Datenmodellierungsmethoden. Die Hauptquelle der Grobfehler sind Reflexionen an Gebäuden und an der Vegetation. Solche Ausreißer im Beobachtungsmaterial lassen sich mit Hilfe von robusten Methoden automatisch beseitigen bzw. unschädlich machen. Auch der Einfluss von systematischen Fehlern lässt sich durch robuste Modellierung vermindern, wenn nur genügend viele Oberflächenpunkte vorhanden sind. Im Modellierungsprozess der Oberfläche müssen somit robuste Verfahren berücksichtigt werden. Aus der breiten Palette der robusten Methoden sind die M-Schätzer bzw. deren Verwandte, wenn auch heuristisch begründet, von besonderer Bedeutung. Sie lassen sich relativ einfach iterativ ausführen und leicht an vorhandene Algorithmen adaptieren. Die Bedeutung der robusten Verfahren im Zeitalter der automatischen Messung oder Datengewinnung wie Laserscanning, automatische Messung von Stereomodellen, CCD-Sensoren, InSAR, GPS u.a. wird auch deshalb immer größer, weil die Datensätze riesig sind. Sie beinhalten mehrere Millionen von Daten und mehr, und sind nicht mehr überschaubar. Der automatischen Datengewinnung muss daher auch eine weitgehende automatische und zuverlässige Datenbearbeitung zur Seite stehen.

Obwohl die in dieser Arbeit vorgestellten Methoden und Algorithmen zur Modellierung von unterschiedlichen Oberflächen in einer funktionalen Form, $z = z(x, y)$, geeignet sind, wird die Anwendung an erster Stelle an die Erdoberfläche fokussiert. Speichermedium und Informationsträger über die Erdoberfläche sind heutzutage digitale Geländemodelle, deren Qualität vor allem von den Daten und von den Interpolationsalgorithmen abhängig ist. Im Abschnitt 2 geben wir einen stichwortartigen Überblick über die digitalen Geländemodelle, um diese Abhandlung in die geowissenschaftliche Aufgabenstellung einzuordnen. Danach folgt eine Übersicht der

existierenden Approximationsmethoden, die zur Geländemodellierung bevorzugt werden.

Im Mittelpunkt dieser Arbeit stehen die deformierbaren Modelle der Kurven und Oberflächen, die als Lösung von energieminimierenden Variationsproblemen entstehen, und welche wir als Alternative zu den klassischen Approximationsverfahren sehen. Während die letzteren die gleichen Approximationseigenschaften auf dem gesamten Definitionsbereich besitzen, können bei den deformierbaren Modellen die Approximationseigenschaften lokal durch variable Steuerung kontrolliert werden. Durch eine geeignete Wahl der Steuerparameter lassen sich mit Hilfe dieser Modelle Diskontinuitäten modellieren. An die energieminimierenden Verfahren greifen wir auch deshalb gern zurück, weil das diesen Verfahren zugrunde liegende Prinzip ein universelles ist und die kontinuierliche Realisierung der Methode der kleinsten Quadrate darstellt. Im Kapitel 4 geben wir einen Überblick über *Snakes*, das energieminimierende Modell für Kurven, und verallgemeinern dann dieses Modell im Kapitel 5 zu *Flakes*, dem entsprechenden Modell für Oberflächen, und zwar für Daten unterschiedlicher Struktur. Für die energieminimierenden Variationsprobleme können keine exakte Lösungen angegeben werden. Diese Probleme lassen sich lediglich auf dem numerischen Wege lösen. Für die Ableitung von Flakes-Modellen werden wir unterschiedliche Vorgehensweisen, wie die Ritz-Methode oder Diskretisierung von Euler-Gleichungen, anwenden und diskutieren. Die abgeleiteten Modelle werden ferner an unterschiedlichen Datensätzen getestet.

Obwohl im Vordergrund dieser Arbeit die Modellierung von Oberflächen steht, lassen wir die eindimensionalen Modelle wie Snakes nicht völlig außer Acht, weil manche Verfahren die Oberfläche profilweise abtasten. Vertretungsweise sei das Echolotverfahren oder die dynamische Registrierung von Fahrbahnoberflächen (vgl. CASPARY (2002)) genannt. Eine weitere Motivation zur profilweisen Approximation ergibt sich aus der praktischen Nutzung von Höhenprofilen; beispielsweise wird das Durchflussverhalten in Flussgebieten zum Zwecke von Hochwasservorhersagen (im einfachsten Modell) in Querprofilen analysiert bzw. dynamisch modelliert. Nicht zuletzt ist auch die pragmatische Vorgehensweise wichtig: Im Gegensatz zur zweidimensionalen ist die eindimensionale Modellierung leichter ausführbar und überschaubarer. Die Erfahrung, Erkenntnisse und teilweise auch Ergebnisse, welche man bei der eindimensionalen Modellierung gesammelt hat, lassen sich im gewissen Umfang auf die zweidimensionale Modellierung übertragen.

Ein wichtiger Teil dieser Abhandlung stellen Methoden und Algorithmen zur dreidimensionalen Modellierung von Diskontinuitäten dar. Mit den deformierbaren Modellen lassen sich Diskontinuitäten im Modellierungsprozess von Oberflächen weitgehend erhalten bzw. nachbilden. Sie geben aber keine vektorielle Information zum räumlichen Verlauf von Diskontinuitäten. Die vektorielle Information über Geländekanten ist aber unentbehrlich für die Erstellung von hochqualitativen digitalen Geländemodellen. Methoden zur Ableitung von dreidimensionalen Geländekanten aufgrund der unstrukturierten diskreten Information (Laserscanner-Daten) über die zu modellierenden Oberflächen werden im Kapitel 8 ausführlich dargestellt, diskutiert und an praktischen Beispielen getestet.

Die Arbeit schließt mit der Zusammenfassung von wichtigen Ergebnissen und Erkenntnissen.

2 Grundlagen digitaler Geländemodelle

2.1 Allgemeines

Digitale Geländemodelle beschreiben mit einer gewünschten Genauigkeit die Erdoberfläche durch entsprechend geordnete Punkte mit Höhenwerten und geomorphologischen Informationen. Aus den ursprünglichen Punkten im \mathbb{R}^3 wird meistens ein Modell in der Funktionsform

$$z = z(x, y) \quad (2.1-1)$$

interpoliert (sog. 2.5D-Modell). Zu DGM gehören auch die Interpolationsmethoden, welche die Oberfläche modellieren.

Im Aufbauprozess und in der Anwendung digitaler Geländemodelle können drei, in der Regel nacheinander realisierte Phasen unterschieden werden (EBNER, 1990; REINHARDT, 1991):

- Erfassung und Aufbereitung der Primärdaten;
- DGM-Generierung, Datenorganisation;
- Ableitung und Ausgabe unterschiedlicher Produkte von DGM.

Die vorliegende Arbeit ist in die zweite und teilweise in die erste Phase einzuordnen. Nachfolgend wird auf diese Thematik kurz eingegangen. Die Ableitung der Folgeprodukte und damit verbundene Probleme werden im weiteren nicht betrachtet. Stattdessen verweisen wir auf die umfangreiche Literatur. Vertretungsweise seien genannt: FRITSCH (1991); REINHARDT (1991); KRAUS (2000).

2.2 Datengewinnung

Der Ausgangspunkt für die digitalen Geländemodelle sind Primärdaten, die mit unterschiedlichen Verfahren gewonnen werden können:

- Durch Digitalisieren von topographischen Karten. Dabei werden sowohl Höhenlinien als auch markante Höhenpunkte erfasst.
- Durch Messung in photogrammetrischen Stereomodellen. Hier können die Daten semiautomatisch und direkt als Gittermodell, mit Optimierung der Gitterweite, erfasst werden.
- Durch Fernerkundung. Hier stehen die Daten im Rasterformat zur Verfügung.
- Durch direkte Erfassung im Gelände mittels tachymetrischer Aufnahme. Die Messung kann sowohl mit Hilfe von modernen Tachymetern als auch von GPS (RTK-Messung) durchgeführt werden.
- Durch die Geländeabtastung mit flugzeuggetragenen (seltener terrestrischen) Laserentfernungsmessern (Laserscanning).

Aus der Sicht dieser Arbeit ist das letztgenannte Verfahren von besonderem Interesse. Die Daten werden mit einer sehr hohen Dichte, etwa 1 Punkt pro Quadratmeter, erfasst. Dieses erlaubt die Feinstrukturen von Oberflächen zu modellieren, insbesondere in bewaldeten oder anderen besonderen Gebieten, wo das Laserscanning eigentlich keine Konkurrenz hat. Beim Abtasten der Erdoberfläche mit dem Laser entstehen auch Reflexionen an Bäumen und künstlichen Gebilden. Solche Daten, die als Grobfehler angesehen werden können, sind im Zuge der Bearbeitung zwecks DGM-Erstellung, zu beseitigen.

Für die Daten, die im Gitter- beziehungsweise Rasterformat vorliegen, sind die Lösungen, die solchen Datenstrukturen angepasst sind, zu bevorzugen.

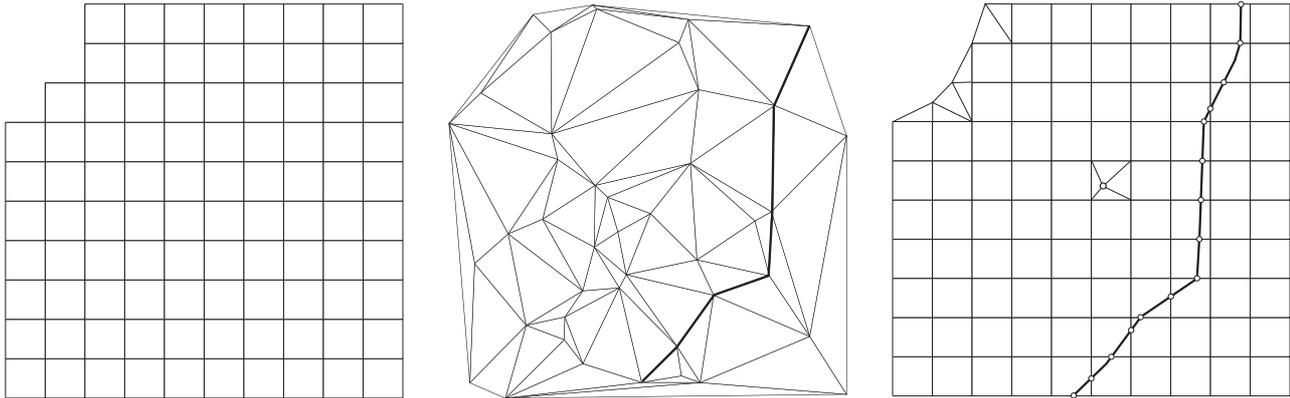


Abbildung 2.2-1: Datenorganisation in DGM. Gittermodell (links), Dreiecksmodell (Mitte) und hybrides Modell (rechts).

2.3 Datenorganisation

In den digitalen Geländemodellen werden die Höhendaten in einem von drei Modellen organisiert und verwaltet (vgl. Abb. 2.2-1):

- **Gittermodell.** Die Primärhöhendaten werden auf ein reguläres, meistens quadratisches Gitter interpoliert und in Matrixform gespeichert. Die Vorteile dieses Modells liegen auf der Hand, sowohl in der DGM-Verwaltung als auch bei der Ableitung der Folgeprodukte. In diesem Modell können wichtige morphologische Informationen verloren gehen.
- **Dreiecksmodell.** Die Grundstruktur dieses Modells bilden die Primärdaten. Sie werden in ein irreguläres Dreiecksnetz vermascht (TIN), wobei die Vernetzung nach unterschiedlichen Kriterien erfolgen kann. In der Praxis hat sich die Delaunay-Triangulation (BARTELME, 1995) durchgesetzt; eine Alternative dazu stellt die Triangulation mit minimalem Gewicht (KOCH, 1985) dar. In diesem Modell können Geländekanten exakt berücksichtigt werden.
- **Hybrides Modell.** Die Grundstruktur dieses Modells bildet ein reguläres Gitter, das mit zusätzlichen geomorphologischen Informationen, wie Geländekanten, markante Höhenpunkte etc. bereichert wird. Die zusätzlichen vektoriellen Informationen sind auf das Gitter bezogen. In diesem Modell kann auch die Dreiecksvermaschung integriert werden.

3 Verfahren zur Oberflächenmodellierung

3.1 Approximationsmethoden - eine Übersicht

Es gibt eine Fülle von Verfahren zur Oberflächenmodellierung, vor allem im Bereich der Mathematik bzw. der Computergraphik. Umso umfangreicher ist die Literatur zu dieser Thematik. Wir nennen nur drei Standardtextbücher (HOSCHEK und LASSER, 1992; FARIN, 1993; KICIAK, 2000), die auch sehr umfangreiche Literaturlisten beinhalten. Eine Übersicht und Zusammenstellung von relativ aktuellen Erkenntnissen, auch zu den wohl bekannten Methoden und Algorithmen ist bei LODHA und FRANKE (1999) zu finden.

Im Textbuch von KRAUS (2000) sind populäre Approximationsmethoden, insbesondere im Bezug auf Anwendung in den Topographischen Informationssystemen in einer bildhaften, anwendungsnahen Art dargestellt. Daher wird nachfolgend die Beschreibung der Polynomapproximation, der multiquadratischen Methoden und der linearen Prädiktion knapp gehalten.

Die Methoden zur Oberflächenmodellierung können nach unterschiedlichen Gesichtspunkten betrachtet werden. Die Wahl des Verfahrens ist jeweils mit einer konkreten Aufgabenstellung, der Form der zu modellierenden Oberflächen, dem numerischen Aufwand, der Datendichte etc. verbunden. Entscheidend ist aber die *Datenstruktur*, d.h. die Datenverteilung im \mathbb{R}^2 , die regelmäßig oder unregelmäßig (sog. scattered data) sein kann. Unter der regulären Datenstruktur versteht man Daten, die an regulären Vierecken, Dreiecken oder $(2n + 1)$ -Ecken vorliegen. Für solche Daten werden unterschiedliche Splines, insbesondere die eindimensionalen Splines in Tensorproduktform bevorzugt. In den Geowissenschaften spielen solche Daten, als Rohdaten, eher eine marginale Rolle, sollen aber nicht völlig außer Acht gelassen werden, denn in manchen Anwendungen, z.B. in der digitalen Bildverarbeitung, können sie nützlich sein. Manche moderne Messverfahren, wie terrestrische Laserabtastung beispielsweise, können auch reguläre Punktwolken von Messwerten liefern.

In den Geowissenschaften dominieren Daten mit einer irregulären Struktur. Ausgewählte Interpolationsmethoden sind für solche Daten in der Tabelle 3.1-1 zusammengestellt. Sie sind nach dem meist benutzten Lokalkriterium geordnet. Zu den lokalen Methoden gehören solche, die die Informationen oft aus der direkten Umgebung des zu interpolierenden Punktes verwenden. Bei den globalen Methoden beeinflussen alle Messwerte den zu interpolierenden Wert. In digitalen Modellen werden, auf verschiedenen Aufbauetappen, sowohl lokale als auch globale Verfahren verwendet. Darüber hinaus können, bei großen Datensätzen globale Methoden lokal, gebietsweise angesetzt werden.

Zu den lokalen Methoden zählen sowohl einfache Verfahren, wie die gewichtete Mittelwertbildung, z.B. aus natürlicher Nachbarschaft (SIBSON, 1981), als auch die fortgeschrittenen Modelle wie die Finite-Elemente-Methode, die zur Interpolation von digitalen Geländemodellen u.a. von EBNER et al. (1980) und REISS (1985) benutzt wurde. Die lokalen Methoden werden im Weiteren, mit Ausnahme der Nächstegelegener-Nachbar-Interpolation, außer Acht gelassen.

Zu den radialen Basisfunktionsmethoden gehören solche der Gestalt

$$z(x, y) = \sum_{i=1}^n \alpha_i R(r_i(x, y)) + \sum_{j=1}^m \beta_j p_j(x, y), \quad (3.1-1)$$

wobei α_i und β_j die zu bestimmenden Parameter sind. $R(r_i(x, y))$ ist die sog. radiale Funktion, welche nur von der euklidischen Distanz $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ abhängig ist. Die $p_j(x, y)$ stellen eine Basis für den Raum P_k der Polynome vom Grad k dar und es ist $m = k(k + 1)/2$. Dies bedeutet, dass für $k = 0, 1, 2$ kein Polynom, eine Konstante oder ein lineares Polynom zur Interpolationsfunktion addiert wird.

An dieser Stelle wollen wir noch eine Bemerkung zu Interpolation und Approximation machen. Aus der mathematischen Sicht ist die Unterscheidung der beiden Begriffe klar. In der geowissenschaftlichen Literatur verwendet man auch Begriffe wie exakte und nicht exakte Interpolation oder Interpolation und Interpolation mit Filterung. Diese Bezeichnungen sind als Synonyme für die Approximation und die Interpolation zu verstehen.

Tabelle 3.1-1: Zusammenstellung der Interpolationsmethoden für irreguläre Daten.

■	Lokale Methoden
	<ul style="list-style-type: none"> • Gewichtete Mittelwertbildung • Nächstgelegener-Nachbar-Interpolation • Gleitende Polynomflächen • Finite-Elemente-Methode
■	Globale Methoden
	<ul style="list-style-type: none"> • Polynomflächen • Shepard-Methoden • Radiale Basisfunktionsmethoden <ul style="list-style-type: none"> – Multiquadratische Methoden – Spline minimaler Krümmung – Spline minimaler Krümmung mit Steifigkeit • Interpolation nach kleinsten Quadraten • Kriging

3.1.1 Nächstgelegener-Nachbar-Interpolation

Diese spezielle Interpolationsmethode wird bei kartographischer Modellierung und Satellitenbildanalyse angewendet. Die Idee der Methode beruht auf der Ausbreitung des gemessenen Wertes in einem Punkt P auf eine bestimmte Fläche $F \subset \mathbb{R}^2$ des Skalarfeldes. Mit anderen Worten: Man geht davon aus, dass das Skalarfeld in einer Umgebung von P die gleichen Werte wie im Punkt P hat. Zur Bestimmung sind bei dieser Methode die Grenzen zwischen Teilflächen, welche zu den Messdaten gehören. Daher wird auch diese Methode gelegentlich Grenzmethode genannt. Die Grenzlinien werden meistens als *Thiessen(Voronoi)-Polygone* gezogen. Sie entstehen, wenn man auf halber Strecke aller direktbenachbarten Punkte eine Senkrechte zu der Strecke zieht und den \mathbb{R}^2 mosaikartig zerlegt (*Voronoi- oder Thiessen-Mosaik*). Diese Methode wird in einem Beispiel in der Abbildung 3.1-1 veranschaulicht.

Zur Modellierung von Oberflächen, insbesondere im Hinblick auf topographische Anwendung, ist diese Interpolationsmethode nicht geeignet. Wir benutzen sie im praktischen Teil der Arbeit zur Regularisierung von Laserscanning-Daten, d.h. zur Übertragung der Daten von gleichmäßig-irregulär in der Ebene verteilten Punkten auf ein reguläres Gitter.

3.1.2 Polynomapproximation

Die polynomiale Approximation bzw. Interpolation gehört zu den wohlbekanntesten Standardverfahren. Um eine Polynomoberfläche vom Grad n in der Form

$$z(x, y) = \sum_{i,j} a_{ij} x^i y^j = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{02}y^2 + a_{11}xy + \dots \quad (3.1-2)$$

zu konstruieren, werden $\binom{n+2}{2}$ Messdaten in den Punkten $P(x_i, y_i); i = 1, 2, \dots, n$ benötigt. Die Zusammensetzung von (3.1-2) hängt jedoch auch von der geometrischen Konfiguration der Punktmenge ab. Manche Polynome können verschwinden, wenn Punkte z.B. auf einer Geraden oder einem Kreis liegen. DE BOOR und RON (1992) haben einen Algorithmus für die Konstruktion eines Interpolationspolynoms minimalen Grades entwickelt. Dabei wird die Existenz und Eindeutigkeit der Interpolationsfunktion garantiert.

Die Polynominterpolation eines höheren Grades hat bekanntlich schlechte Eigenschaften. Insbesondere in den Randbereichen kommt es in der Regel zu großen Ausschwingungen. Aus diesem Grunde werden Polynome niederen Grades als Approximationsfunktion für die Modellierung von Grobstrukturen der Oberfläche (sog. Trendabsplattung) benutzt. Die Parameter a_{ij} können nach der Methode der kleinsten Quadrate geschätzt werden:

$$\mathbf{a} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{w}, \quad (3.1-3)$$

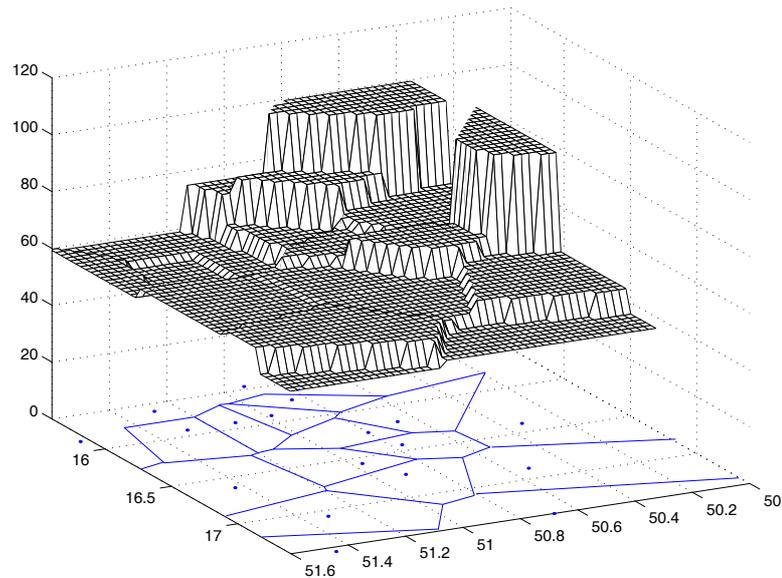


Abbildung 3.1-1: Ein Beispiel zur Interpolation nach dem nächstgelegenen Nachbarn. Voronoi-Mosaik mit Datenpunkten und Interpolationsfläche.

wobei \mathbf{w} der Vektor der Messwerte ist und \mathbf{A} die Matrix

$$\mathbf{A} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & y_1^2 & x_1 y_1 & \dots \\ 1 & x_2 & y_2 & x_2^2 & y_2^2 & x_2 y_2 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_n^2 & y_n^2 & x_n y_n & \dots \end{bmatrix}.$$

Bei Bedarf kann in (3.1-3) die Gewichtsmatrix $\mathbf{P} = \text{diag}\{p_1, p_2, \dots, p_n\}$ und somit die individuelle Genauigkeit der Daten berücksichtigt werden:

$$\mathbf{a} = (\mathbf{A}^\top \mathbf{P} \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{P} \mathbf{w}. \quad (3.1-4)$$

Die Polynomapproximation zeichnet sich durch einen geringen Rechenaufwand aus und besitzt noch bei Polynomen vom niedrigen Grad relativ gute Interpolationseigenschaften. Um diese Eigenschaften auszunutzen und gleichzeitig eine akzeptable Annäherung der Funktion an die Daten zu erreichen, liegt es nahe, das Einzugsgebiet zu begrenzen und das Polynom lokal anzuwenden. Ein solches Verfahren wird als Interpolation mittels *gleitender Polynomflächen* bezeichnet. An jeder zu interpolierenden Stelle wird in diesem Verfahren eine eigene Polynomfläche gebildet, und zwar nur aus den Punkten, die in der direkten Nachbarschaft der Interpolationsstelle liegen. Die Anzahl der berücksichtigten Punkte kann gleich der Anzahl der Parameter der Polynomfunktion oder größer sein. In dem letzten Fall sind die Parameter a_{ij} nach (3.1-3) oder (3.1-4) zu bestimmen. KRAUS (2000) weist darauf hin, dass bei dieser Methode Unstetigkeitsstellen entstehen können und empfiehlt, die lokale Polynomapproximation zur Lokalisierung grober Datenfehler zu verwenden.

Die Polynome niedrigen Grades werden auch als Grundfunktionen für unterschiedliche Spline-Modelle benutzt.

3.1.3 Shepard-Methoden

Die in der Literatur nach SHEPARD (1968) benannten Methoden entsprechen der in der Geodäsie gut bekannten gewichteten Mittelwertbildung:

$$z(x, y) = \sum_{i=1}^n p_i(x, y) w_i. \quad (3.1-5)$$

Die Gewichtsfunktion p_i besitzt die Eigenschaften

- $p_i(x_j, y_j) = \delta_{ij}$, d.h. $z(x_i, y_i) = w_i$,
- $p_i > 0$ und $\sum_{i=1}^n p_i = 1$,
- C^0 Stetigkeit.

Nach dem ursprünglichen Ansatz werden die Gewichtsfunktionen zu

$$p_i(x, y) = \frac{\sigma_i(x, y)}{\sum_{j=1}^n \sigma_j(x, y)} \quad (3.1-6)$$

ermittelt, wobei $\sigma = 1/r^\mu$, $\mu > 0$, proportional zu Inversen des Punktabstandes r mit $r^2 = (x-x_i)^2 + (y-y_i)^2$ ist. Die Shepard-Methode ist ein globales Interpolationsverfahren, das lokalisiert werden kann durch Multiplikation der Gewichtsfunktion mit einer Dämpfungsfunktion, welche außerhalb einer Kreisscheibe mit einem gewählten Radius verschwindet (HOSCHEK und LASSER, 1992). Diese Methode wurde gründlich untersucht und es sind mehrere Varianten und Modifikationen entstanden (FRANKE und NIELSON, 1980; FRANKE, 1982).

3.1.4 Multiquadratische Methode

Die Multiquadratische Methode, die in der Literatur auch als Flächensummationsmethode bekannt ist, geht auf HARDY (1971), vgl. auch (HARDY, 1972, 1990), zurück. Sie basiert auf der Vorstellung, dass sich eine irreguläre, analytisch nicht definierbare Oberfläche als Summe von regulären, mathematisch definierbaren Elementarflächen darstellen lässt:

$$z(x, y) = \sum_{i=1}^n c_i q(x_i, y_i; x, y). \quad (3.1-7)$$

Die Elementarfläche entsteht durch Drehung der Kern- bzw. Basisfunktion $q(x_i, y_i; x, y)$, die im Punkt P_i ($i = 1, 2, \dots, n$) verankert und mit dem Koeffizient c_i skaliert ist. Die Koeffizienten lassen sich aus dem linearen Gleichungssystem

$$\mathbf{c} = \mathbf{Q}^{-1} \mathbf{w} \quad (3.1-8)$$

mit dem Stützwertvektor \mathbf{w} und $Q_{ij} = q(x_i, y_i; x_j, y_j)$ bestimmen. Die Basisfunktionen sind Abstandsfunktionen und können in der allgemeinen Form

$$q(x_i, y_i; x, y) = [p^2 + (x - x_i)^2 + (y - y_i)^2]^{\mu/2}, \quad \mu \neq 0 \quad (3.1-9)$$

aufgefasst werden. Für $\mu = 1$ bekommt man die multiquadratische und für $\mu = -1$ die inverse multiquadratische Methode. Über den geeignet zu wählenden Parameter p^2 kann die Glattheit der Interpolation gesteuert werden.

Die multiquadratische Methode wurde an mehreren Datensätzen unterschiedlicher Art gründlich getestet und gehört zu den effizientesten Methoden. Beispiele sind in der zahlreichen Literatur zu finden. In WILD (1983), vgl. auch (KRAUS, 2000), wird über ausgezeichnete Interpolationseigenschaften dieser Methode berichtet, wenn man lokal für jede Stützstelle individuelle Basisfunktionen benutzt. In CARLSON und FOLEY (1991) oder HOSCHEK und LASSER (1992) ist ein Algorithmus für die Bestimmung der nahezu optimalen p^2 -Werte zu finden.

Die multiquadratische Methode gehört zu den Methoden der radialen Basisfunktionen, wobei in der Gleichung (3.1-1) $m = 0$ zu setzen ist.

3.1.5 Spline minimaler Krümmung

Der Spline minimaler Krümmung wurde durch DUCHON (1976) als die Lösung des Variationsproblems

$$\Phi(z(x, y)) = \iint_{\Omega} (z_{xx}^2 + 2z_{xy}^2 + z_{yy}^2) dx dy + \sum_{i=1}^n \alpha (z(x_i, y_i) - w_i)^2 \rightarrow \min \quad (3.1-10)$$

mit $z_{xx} := \partial^2 z / \partial x^2$, $z_{xy} := \partial^2 z / \partial x \partial y$, $z_{yy} := \partial^2 z / \partial y^2$ eingeführt. Der erste Term bedeutet Minimierung der Biegeenergie einer an den Datenpunkten befestigten dünnen elastischen Platte über einem ebenen Bereich $\Omega \in \mathbb{R}^2$, daher auch in der Literatur als *thin plate spline* (TPS) bekannt. Durch den zweiten Term wird die Forderung ausgedrückt, dass die Platte $z(x, y)$ nicht weit von den Messwerten w_i ($i = 1, 2, \dots, n$) verlaufen soll. Bildhaft gesprochen wird durch diesen Term die "Befestigungskraft", ausgedrückt. Die beiden Terme werden gegenseitig durch den Parameter α gewichtet. Die analytische Lösung des obigen Variationsproblems ergibt die Splinefunktion

$$z(x, y) = \frac{1}{2} \sum_{i=1}^n \lambda_i r_i^2 \ln r_i^2 + \nu_{00} + \nu_{10}x + \nu_{01}y \quad (3.1-11)$$

mit $r_i^2 = (x - x_i)^2 + (y - y_i)^2$, den Parametern der Basisfunktionen λ_i und des Polynoms ν . Aus (3.1-11) ist es ersichtlich, dass TPS eine radiale Basisfunktionsmethode mit $R = r^2 \ln r$ und $m = 3$ ist. In der Spline-Theorie

wird gezeigt, dass die Parameter λ orthogonal zu dem Polynom sein sollten. Somit können die $n + 3$ Parameter aus dem Gleichungssystem

$$\begin{bmatrix} \mathbf{A} + \alpha \mathbf{P}^{-1} & \mathbf{T} \\ \mathbf{T}^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \lambda \\ \nu \end{bmatrix} = \begin{bmatrix} \mathbf{w} \\ \mathbf{0} \end{bmatrix} \quad (3.1-12)$$

ermittelt werden, wobei

$$\mathbf{A} = \begin{bmatrix} 0 & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & 0 & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & 0 & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & 0 \end{bmatrix}, \quad \mathbf{T} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \\ \vdots & \vdots & \vdots \\ 1 & x_n & y_n \end{bmatrix},$$

$$\lambda = [\lambda_1 \ \lambda_2 \ \lambda_3 \ \dots \ \lambda_n]^\top, \quad \nu = [\nu_{00} \ \nu_{10} \ \nu_{01}]^\top, \quad \mathbf{w} = [w_1 \ w_2 \ w_3 \ \dots \ w_n]^\top$$

und $a_{ij} = a_{ji} = r_{ij}^2 \ln r_{ij}$. Durch die Einführung der Gewichtsmatrix

$$\mathbf{P} = \text{diag}\{1/\sigma_1^2, 1/\sigma_2^2, \dots, 1/\sigma_n^2\} \quad (3.1-13)$$

kann die Anpassung der Funktion z an die Messwerte w_i in Abhängigkeit von der individuellen Messgenauigkeit σ_i^2 durchgeführt werden. Ist die Messgenauigkeit für alle Messwerte konstant, so ist die Matrix \mathbf{P} durch die Einheitsmatrix zu ersetzen. Der Parameter α steuert die Glättungseigenschaften des Approximationsplines. Dabei treten zwei Sonderfälle auf:

- Für $\alpha = 0$ verschwindet der zweite Term in (3.1-10) bzw. $\alpha \mathbf{P}^{-1}$ in (3.1-12) und der Approximationspline geht in einen reinen Interpolationsspline über; die Messwerte werden exakt wiedergegeben.
- Für genügend großes α , speziell $\alpha \rightarrow \infty$ spielt der zweite Term in (3.1-10) die dominierende Rolle und der Approximationspline tendiert gegen eine ausgleichende Ebene.

3.1.6 Interpolation nach kleinsten Quadraten und Kriging

Die in diesem Abschnitt beschriebenen Methoden sind stochastisch begründet und werden auch als Optimalfilterung bezeichnet. Es liegt ihnen die Forderung zugrunde, die Schätzfehlervarianz zu minimieren. Die Filtervorschrift, welche diese Eigenschaft hat, wird auch als Kolmogorow-Wiener-Filter bezeichnet (vgl. KOCH (1987); MEIER und KELLER (1990)). Kriging und die Interpolation nach kleinsten Quadraten können als zwei Varianten der praktischen Realisierung des Kolmogorow-Wiener-Filters angesehen werden. Wegen dieser gemeinsamen Grundlage behandeln wir kurz die beiden Interpolationsmethoden gemeinsam.

Die Interpolationsvorschrift ähnelt der gewichteten Mittelwertbildung und kann in der Form

$$z(x, y) = \sum_{i=1}^n \lambda_i w_i \quad (3.1-14)$$

aufgefasst werden, mit dem Unterschied, dass die Gewichte λ_i aufgrund der geostatistischen Analyse des zu interpolierenden Feldes gewonnen werden.

Für die Interpolation nach kleinsten Quadraten werden die Gewichte aus der Beziehung

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} C(PP_1) & C(PP_2) & \dots & C(PP_n) \end{bmatrix} \begin{bmatrix} C(P_1P_1) & C(P_1P_2) & \dots & C(P_1P_n) \\ C(P_2P_1) & C(P_2P_2) & \dots & C(P_2P_n) \\ \vdots & \vdots & \ddots & \vdots \\ C(P_nP_1) & C(P_nP_2) & \dots & C(P_nP_n) \end{bmatrix}^{-1} \quad (3.1-15)$$

ermittelt. Der Vektor auf der rechten Seite der Beziehung (3.1-15) enthält die Kovarianzwerte $C(PP_1), \dots, C(PP_n)$ zwischen der zu interpolierenden Stelle und allen anderen Stützstellen, wo die Messwerte w_i gegeben sind. Die Matrix enthält die Kovarianzen $C(P_iP_j)$ der Stützstellen miteinander in allen Kombinationen. Alle Kovarianzen werden anhand der gleichen Kovarianzfunktion bestimmt, welche vom Abstand $d = \overline{P_iP_j}$ zwischen den

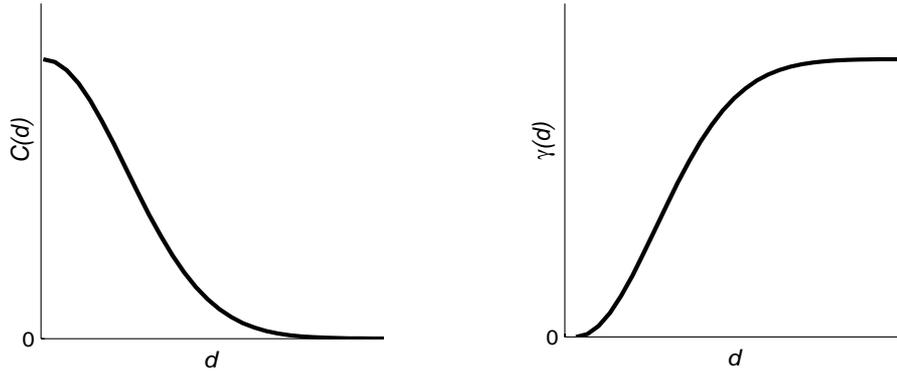


Abbildung 3.1-2: Eine Kovarianzfunktion (links) und das zugehörige Variogramm (rechts).

Stützstellen abhängig ist. Die Kovarianzfunktion wird durch Approximation der empirischen, aus dem Datensatz geschätzten Kovarianzwerte gebildet. Durch die Kovarianzfunktion wird die räumliche Variation des zu interpolierenden Feldes ausgedrückt. Daher ist die Wahl eines geeigneten Modells von großer Bedeutung bei dieser Methode. Eine Zusammenstellung der möglichen Modelle ist z.B. bei MEIER und KELLER (1990) zu finden. Wichtig ist auch die Bestimmung der Parameter der jeweiligen Kovarianzfunktion. So lassen sich in der Varianz $C(0) = C(P_i P_i)$ sowohl die empirische Signalvarianz als auch die Fehlervarianz (Residuenvarianz) berücksichtigen und damit im Zuge der Interpolation auch die Messfehler filtern (minimieren). Zu diesem Zweck macht man den Ansatz $C(0) = C_{zz}(0) - \sigma^2$, mit der geschätzten Varianz $C_{zz}(0)$ der zentrierten Stützwerte und der Genauigkeit σ^2 des Messverfahrens. Beim Kriging wird die Filterung durch Einbeziehung der sogenannten Nugget-Varianz, die hauptsächlich die Messgenauigkeit bedeutet, realisiert.

In der Kriging-Methode verwendet man meistens anstelle der Kovarianzfunktion das sogenannte Variogramm $\gamma(P_i P_j)$. Beide Funktionen sind zueinander komplementär, ausgedrückt durch die Beziehung

$$\gamma(P_i P_j) = C(0) - C(P_i P_j), \quad (3.1-16)$$

welche bei gewissen stochastischen Voraussetzungen gilt. In der Abbildung 3.1-2 ist ein Modell für die beiden Funktionen graphisch veranschaulicht.

Beim Kriging stellt man, neben der Minimierung der Schätzfehlervarianz, eine zusätzliche Forderung: Die Schätzung soll erwartungstreu bzw. biasfrei sein. Die Forderung kann erfüllt werden, indem man an die Parameter λ_i die zusätzliche Bedingung

$$\sum_{i=1}^n \lambda_i = 1 \quad (3.1-17)$$

stellt. Unter dieser Bedingung lassen sich die Parameter aus dem Gleichungssystem

$$\sum_{i=1}^n \lambda_i \gamma(P_i P_j) + \mu = \gamma(P_j P) \quad \text{für alle } j \quad (3.1-18)$$

bestimmen, wobei μ der Lagrange-Multiplikator ist. Die Gleichungen (3.1-17) und (3.1-18) lassen sich in Matrixschreibweise formulieren:

$$\begin{bmatrix} \lambda \\ \mu \end{bmatrix} = \mathbf{A}^{-1} \mathbf{b}, \quad (3.1-19)$$

wobei

$$\mathbf{A} = \begin{bmatrix} \gamma(P_1 P_1) & \gamma(P_1 P_2) & \dots & \gamma(P_1 P_n) & 1 \\ \gamma(P_2 P_1) & \gamma(P_2 P_2) & \dots & \gamma(P_2 P_n) & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \gamma(P_n P_1) & \gamma(P_n P_2) & \dots & \gamma(P_n P_n) & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \gamma(P P_1) \\ \gamma(P P_2) \\ \vdots \\ \gamma(P P_n) \\ 1 \end{bmatrix}.$$

In der Tabelle 3.1-2 sind einige Merkmale der in diesem Abschnitt diskutierten Interpolationsmethoden gegenübergestellt. Die empirischen Schätzwerte sowohl der Kovarianzfunktion als auch des Variogramms werden

Tabelle 3.1-2: Vergleich einiger Merkmale der Interpolation nach kleinsten Quadraten und der Kriging-Interpolation.

Interpolation nach kleinsten Quadraten	Kriging
<ul style="list-style-type: none"> • Im Zuge der Vorverarbeitung ist eine Trendabspaltung nötig: $u = w - w_t$, so dass $\mathbf{E}u = 0$ (zentrierte Werte) • Empirische Kovarianzfunktion aus Absolutwerten: 	<ul style="list-style-type: none"> • Keine Vorverarbeitung nötig, Trendabspaltung während der Interpolation (in einem Schritt) • Empirisches Variogramm aus Relativwerten:
$C(d_j) = \frac{1}{n(d_j)} \sum_{i=1}^{n(d_j)} u_{1,i} u_{2,i}$	$\gamma(d_j) = \frac{1}{2n(d_j)} \sum_{i=1}^{n(d_j)} (w_{1,i} - w_{2,i})^2$

aus den $n(d_j)$ der im Abstand d_j liegenden Paare $u_{1,i}$, $u_{2,i}$ bzw. $w_{1,i}$, $w_{2,i}$ ermittelt. Eine ausführliche Gegenüberstellung der beiden Methoden ist bei MENZ und PILZ (1990); CHEN (1996); KRAUS (1998) zu finden. Es lässt sich zeigen, dass diese Methoden vom theoretischen Standpunkt her identisch sind. Kriging hat aber gewisse Vorteile in der praktischen Anwendung: Der Rechenaufwand ist geringer, der Algorithmus schneller und numerisch stabiler.

Die Interpolation nach kleinsten Quadraten, wegen ihre Eigenschaften auch als (lineare) Prädiktion oder Optimalfilterung bezeichnet, stellt nur einen Sonderfall der allgemeinen, vorrangig in der physikalischen Geodäsie angewendeten Kollokation, vgl. z.B. MORITZ (1989), dar. Für die Anwendung dieser Interpolationsmethode bei der Topographiemodellierung hat Professor KRAUS, Technische Universität Wien, gesorgt. In einer praxisorientierten Forschung wurden über Jahrzehnte von ihm oder seinen Mitarbeitern viele spezielle Probleme gelöst und in die Anwendung umgesetzt. Der Höhepunkt dieser Entwicklung stellt SCOP, das Software-Paket zum Aufbau von digitalen Geländemodellen dar. Die Erkenntnisse und Ergebnisse der Forschung wurden auch im Lehrbuch von KRAUS (2000) präsentiert. Eine interessante Verbindung der linearen Prädiktion mit gleitender Polynomfläche (gleitender Schrägebene), welche auf die Homogenität verzichtet und die Kovarianzfunktion für jeden zu interpolierenden Punkt getrennt bestimmt, hat KOCH (1973) vorgeschlagen.

Die Kriging-Interpolation ist vor allem die Domäne der Geostatistik; vgl. z.B. CRESSIE (1993). Sie ist auch in populären Computerprogrammen zur Oberflächenmodellierung, wie z.B. Surfer, implementiert. Dadurch wird diese Interpolationsmethode immer populärer und gewinnt an Bedeutung außerhalb der Geostatistik. Sowohl bei der Kollokation als auch beim Kriging treten fortgeschrittene Modelle auf, die zueinander äquivalent sind. Darüber hinaus lassen die beiden Methoden eine vollständige Genauigkeitsanalyse zu.

3.1.7 Berücksichtigung von Diskontinuitäten

Die vorgestellten analytischen Modellierungsverfahren erzeugen kontinuierliche und glatte Oberflächen. Die Diskontinuitäten können im Modellierungsprozess berücksichtigt werden, indem man entlang der letzteren nicht interpoliert. Die Kanten müssen in Form von Vektordaten

$$\{x_1, y_1, z_1; x_2, y_2, z_2; \dots; x_n, y_n, z_n\}$$

vorgegeben, vorher sorgfältig mit entsprechenden Verfahren, getrennt von Massenpunkten, erfasst werden. Die Diskontinuitätslinien teilen im Grundriss das gesamte Gebiet in Untergebiete, die getrennt der Modellierung unterworfen werden. Zunächst aber werden Höhen in Stützstellen entlang der Geländekante dicht interpoliert (1D-Interpolation). Diese Punkte gehen dann in die Oberflächeninterpolation als fest zu haltende Punkte, was durch genügend große Gewichte erreicht wird (KRAUS, 2000), ein. Damit werden die Kanten im Modellierungsprozess quasi als Zusatzrestriktionen behandelt. Die Modellierung von Oberflächen mit Kanten ist schematisch in der Abbildung 3.1-3 dargestellt.

In technischen Einzelheiten werden Diskontinuitäten verschiedenartig in unterschiedlichen Softwarepaketen behandelt, so stellt in SCOP eine Kante die Barriere dar, die keine Informationsflut für die Interpolation ermöglicht. In SURFER hingegen stellt nur eine Verwerfung die Barriere dar. Bei der Interpolation im Kantenbereich werden Höheninformationen, die auf beiden Seiten der Kante liegen, auf spezielle Weise benutzt.

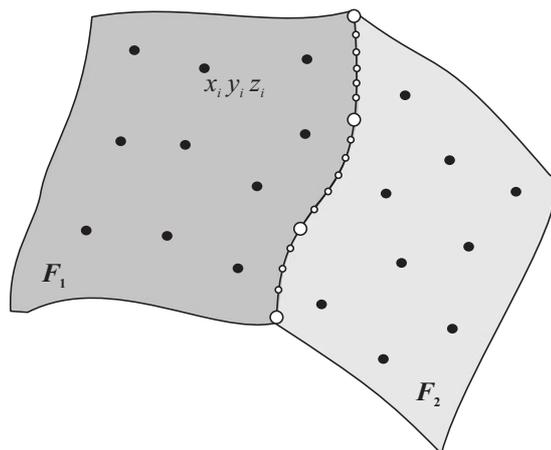


Abbildung 3.1-3: Zur Approximation von Oberflächen mit Kanten. Messdaten (Punkte) und approximierende Oberflächen F_1 und F_2 . Vektordaten zur Definition der 3D-Geländekante: Messpunkte (große Kreise) und zwischeninterpolierte Punkte (kleine Kreise).

3.2 Robuste Verfahren

3.2.1 Allgemeines und M-Schätzer

Die Anpassung der vorgestellten Approximationsmodelle an Messdaten geschieht meist nach der Methode der kleinsten Quadrate. Die Parameter werden so geschätzt, dass die Quadratsumme der Residuen minimal wird:

$$\sum_i v_i^2 \rightarrow \min. \quad (3.2-1)$$

Die Methode der kleinsten Quadrate ist sehr sensitiv gegenüber Abweichungen von der zugrunde liegenden Annahme der Normalverteilung der Beobachtungsfehler. Schon wenige grobfehlerbehaftete Messwerte können das Ergebnis der Schätzung deutlich verfälschen. Gefragt werden also Auswerteverfahren, welche möglichst hohe Effizienz aufweisen und gleichzeitig die Grobfehler unschädlich machen können bzw. deren Einfluss minimieren. Zur Behandlung von Ausreißern im Beobachtungsmaterial gibt es eine Reihe von Verfahren und Methoden, welche sowohl eine mehr heuristische Herkunft, als auch solche, die gute statistische Fundierung aufweisen. Zu den letzteren gehört die wohl populärste Technik der robusten Verfahren, die auf HUBER (1964) zurückgeht, die M-Schätzer. Statt (3.2-1) versucht man hier eine andere Zielfunktion, die sogenannte Verlustfunktion $\rho(v)$, zu minimieren:

$$\sum_i \rho(v_i) \rightarrow \min. \quad (3.2-2)$$

Der Vorteil der M-Schätzer liegt darin, dass sich die Minimalaufgabe (3.2-2) auf die Methode der kleinsten Quadrate zurückführen lässt, statt das Problem direkt zu lösen. Der M-Schätzer der unbekannt Parameter $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$, der die Forderung (3.2-2) erfüllt, ergibt sich aus dem Gleichungssystem

$$\sum_i \psi(v_i) \frac{\partial v_i}{\partial x_j} = 0, \quad j = 1, 2, \dots, n, \quad (3.2-3)$$

mit der Funktion

$$\psi(v) := \frac{d\rho(v)}{dv}, \quad (3.2-4)$$

die als Einflussfunktion bezeichnet wird. Definiert man nun die Gewichtsfunktion

$$p(v) := \frac{\psi(v)}{v}, \quad (3.2-5)$$

so entspricht das Gleichungssystem

$$\sum_i p(v_i) v_i \frac{\partial v_i}{\partial x_j} = 0, \quad j = 1, 2, \dots, n, \quad (3.2-6)$$

der iterativen Lösung der kleinsten Quadrate-Schätzung

$$\sum_i p(v_i)^{(k-1)} v_i^2 \rightarrow \min. \quad (3.2-7)$$

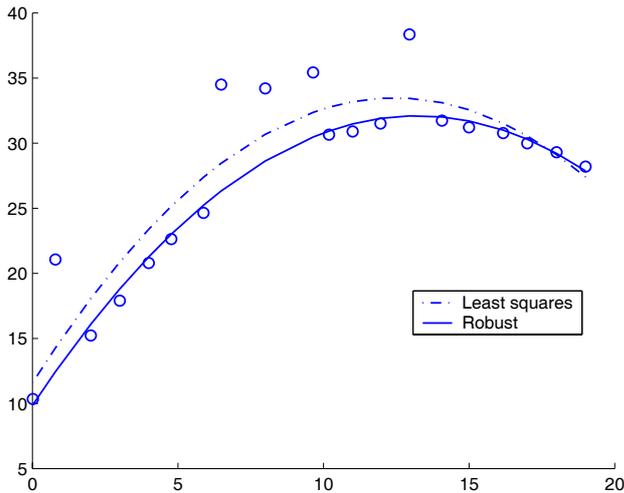


Abbildung 3.2-1: Polynomiale Profilapproximation (Ausschnitt) nach kleinsten Quadraten und nach einem robusten Verfahren

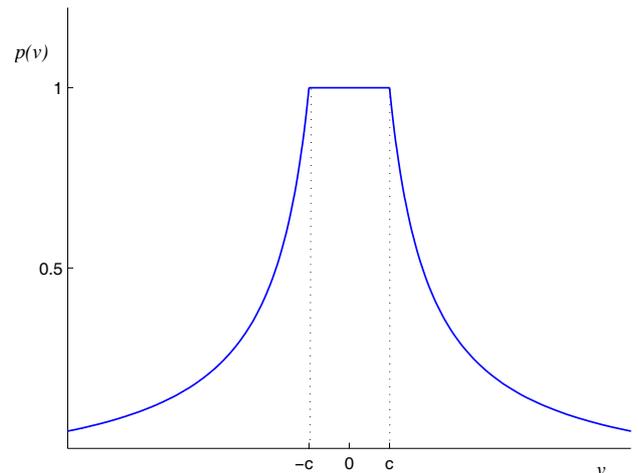


Abbildung 3.2-2: Die Gewichtsfunktion nach Huber

Die Gewichte werden in jedem Iterationsschritt k aufgrund der Residuen aus dem vorhergehenden $(k - 1)$ ten Iterationsschritt neu berechnet.

In dieser Darstellung haben wir versucht, lediglich die Idee der M-Schätzer zu skizzieren. Weitere Einzelheiten samt statistischer Begründung kann in der umfangreichen einschlägigen Literatur gefunden werden. Stellvertretend nennen wir HUBER (1981), HAMPEL et al. (1986) und KOCH (1996).

Die Effizienz der M-Schätzer wollen wir nun an dem in der Abbildung 3.2-1 dargestellten Beispiel demonstrieren. Ein Ausschnitt aus einem Geländeprofil wurde mit einem Polynom approximiert. Die Schätzung der Parameter nach kleinsten Quadraten ergibt einen durch Ausreißer verfälschten Verlauf des Profils. Die richtige Nachbildung des Profils bekommt man, wenn die Schätzung der Parameter robustifiziert wird, indem die Ausreißer kaum die Schätzung beeinflussen. In diesem Fall wurden die Parameter iterativ mit der Verlustfunktion nach Huber,

$$\rho(v) = \begin{cases} v^2/2, & |v| \leq c \\ c(|v| - c/2), & |v| > c, \end{cases} \quad (3.2-8)$$

geschätzt. Mit dem Parameter c werden die Grenzen der Zufallsfehler festgelegt. Die entsprechende Gewichtsfunktion

$$p(v) = \begin{cases} 1, & |v| \leq c \\ c/|v|, & |v| > c \end{cases} \quad (3.2-9)$$

ist in der Abbildung 3.2-2 veranschaulicht. Nach dieser Funktion wurden die grobfehlerverdächtigen Messwerte in einem iterativen Vorgang herabgewichtet.

Die M-Schätzer sind hauptsächlich die auf einer Mischverteilung basierenden Maximum-Likelihood-Schätzer, deren stochastische Eigenschaften gut untersucht und beschrieben sind. Für die praktische Anwendung steht jedoch die Effizienz der Schätzer, die Fähigkeit, Ausreißer zu identifizieren und unschädlich zu machen, im Vordergrund. In der Praxis spielt auch die heuristische Vorgehensweise eine bedeutende Rolle. Die Verlust- oder auch sofort die Gewichtsfunktionen werden datenabhängig konstruiert. Es gibt auch robuste Verfahren, wie die sogenannte Dänische Methode, die von Anfang an als heuristische Verfahren entwickelt wurden. Zu den heuristischen Verfahren können auch Verfahren gezählt werden, die zur Filterung der Originaldaten für die Oberflächenmodellierung konstruiert wurden. Nachfolgend wird ein Überblick über existierende Verfahren für die Bearbeitung der Daten für die digitalen Höhenmodelle gegeben.

3.2.2 Verfahren für Höhendaten

Das Problem der Grobfehler in Höhenrohdaten tritt für die Beschreibung der Oberfläche eigentlich nur bei den Verfahren, die auf automatische Weise die Oberfläche abtasten, auf. Gegenwärtig begrenzt sich dieses Problem hauptsächlich auf Laserscanning. Als Grobfehler werden bei dieser Technik die Reflexionen an Gebäuden, Bäumen und anderen Punkten, die nicht zur Oberfläche gehören, betrachtet. Soll die Oberfläche approximiert

werden, so sind solche Punkte zu eliminieren bzw. so zu behandeln, dass sie keinen Einfluss auf die zu modellierende Oberfläche haben. Die sogenannte Durchdringungsrate im Wald beträgt zwischen 15% und 70% (KRAUS, 2000). Dies bedeutet, dass in bewaldeten Gebieten bis zu 85% aller Punkte nicht als Bodenpunkte klassifiziert werden können. Damit ist ersichtlich, dass ein großes Bedürfnis nach Verfahren besteht, die solche Ausreißer auf automatische Weise erkennen und eliminieren können. Die Ausreißer in solchen Datensätzen zeichnen sich dadurch aus, dass sie eine schiefe Verteilung erzeugen: Es treten ausschließlich nur Reflexionen an Punkten auf, die über der Geländeoberfläche liegen. Diese Tatsache erleichtert die Aufgabe ein wenig. In den letzten Jahren sind einige effiziente Verfahren entwickelt worden, die wir nachfolgend in einer knappen Darstellung präsentieren.

Ein inzwischen ausgereiftes Verfahren wurde im Institut für Photogrammetrie und Fernerkundung TU Wien erarbeitet (KRAUS und PFEIFER, 1997, 1998; KRAUS, 2000; KRAUS und PFEIFER, 2001; BRIESE et al., 2002b). In diesem, auf der linearen Prädiktion basierendem Algorithmus, können die folgenden Arbeitsschritte unterschieden werden:

a) Nachdem der Trend abgespalten wurde, wird zunächst die Interpolation nach kleinsten Quadraten (vgl. Abschnitt 3.1.6) durchgeführt, ohne individuelle Bewichtung der Stützwerte. Die Interpolation liefert Residuen v_i , die auch als Filterbeträge bezeichnet werden. Bei ausreichend vielen Bodenpunkten bekommen Ausreißer besonders große Residuen.

b) In einem zweiten Schritt bekommen die Daten die individuellen Gewichte in Abhängigkeit von den Residuen,

$$p_i = \frac{1}{1 + (a |v_i - g|)^b}, \quad (3.2-10)$$

wobei die Exzentrizität g aufgrund des Residuenhistogramms bestimmt wird und die Parameter a und b so zu wählen sind, dass die Gewichte für die Stützwerte mit großen Residuen gegen Null tendieren. Bei der Bestimmung dieser Parameter wird die Schiefe der Gewichtsfunktion und ihr Halbwert berücksichtigt.

c) Dann wird die Interpolation mit den neuen individuellen Gewichten durchgeführt und die Residuen als orientierter Abstand von der Oberfläche zu den Stützwerten neu berechnet. Dann wird zum zweiten Schritt übergegangen, um neue Gewichte zu ermitteln.

Dieser Iterationsprozess wird solange wiederholt, bis alle Grobfehler eliminiert sind.

Der Algorithmus arbeitet zuverlässig, wenn die Bodenpunkte und Vegetationsreflexionen eine einigermaßen homogene Mischung darstellen. Er versagt, wenn die Ausreißer clusterförmig über größere Flächen auftreten, z.B. Reflexionen an Dächern im bebauten Gelände. Für solche Daten wurde eine hierarchische Anwendung des präsentierten Algorithmus vorgeschlagen. Das hierarchische Verfahren ähnelt der Konstruktion der Bildpyramiden, um das Problem zunächst in einer groben Darstellung zu lösen und dann ins Detailniveau zu übertragen. Die Idee des robusten hierarchischen Verfahrens ist in der Abbildung 3.2-3 dargestellt. Dieses Verfahren besteht aus den nachstehenden Arbeitsschritten:

a) Zunächst werden die Datenpyramiden gebildet: Die Originaldaten (kleine Punkte in der Abbildung) werden durch die niedrigsten, in einem regulären Raster gewählten Datenwerte (große Punkte) repräsentiert.

b) Aus den im vorhergehenden Schritt erhaltenen Daten wird ein grobes DGM mit Hilfe der robusten Interpolation nach kleinsten Quadraten interpoliert. Die clusterförmigen Ausreißer werden in diesem Schritt nur durch einzelne Punkte vertreten und können somit auf iterativem Wege ohne weiteres eliminiert werden. Die Abbildung 3.2-3 zeigt die erste und die letzte Iteration.

c) Das grobe DGM wird um alle Originaldaten, die innerhalb eines Toleranzbandes liegen, ergänzt. Alle andere Daten werden, als nicht zur Oberfläche gehörige, entfernt.

d) Abschließend wird die robuste Interpolation mit allen Daten, die im vorhergehenden Schritt als Oberflächen-
daten klassifiziert wurden, erneut durchgeführt.

Eine weitere Vorgehensweise zur Elimination von Punkten aus Laserscanning-Messdaten, die keine Bodenpunkte sind, wurde von AXELSSON (1999, 2000) vorgeschlagen. Dieses, inzwischen in kommerzielle Software von TerraScan integrierte Verfahren, beruht auf einer iterativen Annäherung einer geeignet gewählten Startoberfläche an die Bodenpunkte. Zunächst wird ein rechteckiges Gitter über die ursprüngliche Punktwolke gelegt, wobei die Größe des Gitters interaktiv durch den Benutzer bestimmt wird. In jeder Masche des Gitters wird ein Punkt mit der niedrigsten Höhe gewählt und als Bodenpunkt klassifiziert. Die Triangulation (TIN) dieser Bodenpunkte bildet eine erste grobe Approximation der Oberfläche. Jeder weitere in die Triangulation eingeführte Bodenpunkt

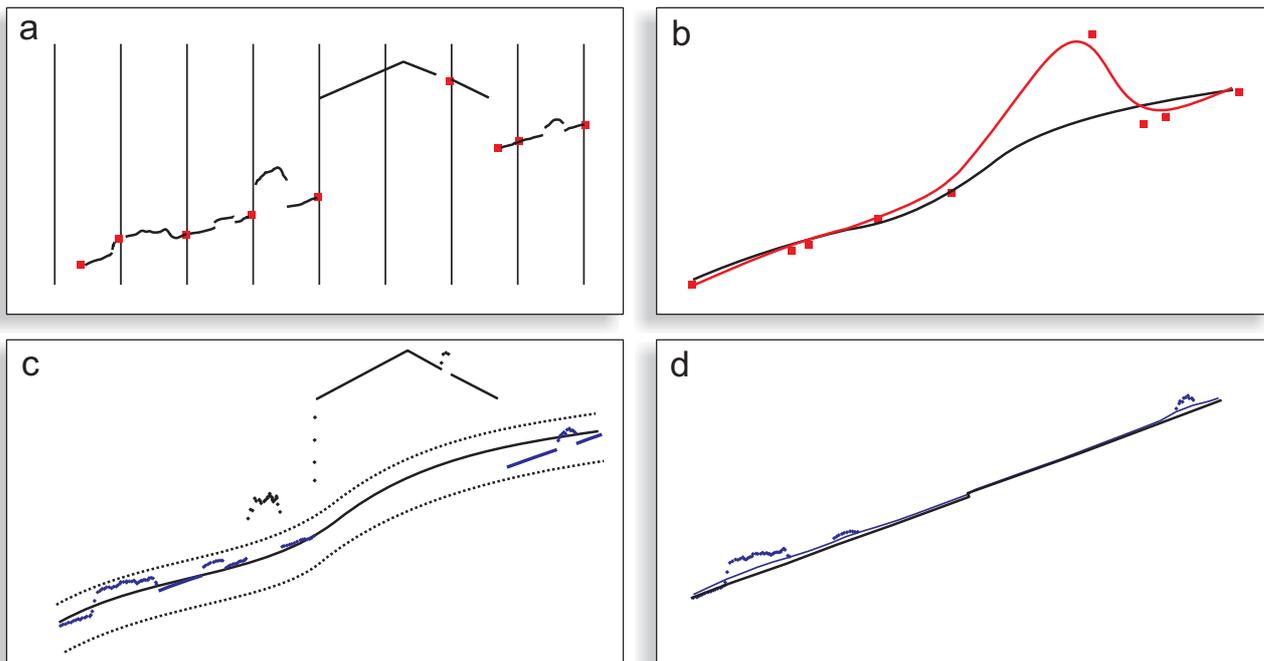


Abbildung 3.2-3: Hierarchische robuste Interpolation (nach BRIESE et al. (2002b)).

nähert diese Triangulation an die Oberfläche an. Die endgültige Approximation der Oberfläche bekommt man nach der sukzessiven Einfügung aller Bodenpunkte in die bestehende Triangulation. Die Einfügung der neuen Punkte kann nach unterschiedlichen Kriterien erfolgen, wobei Parameter der Kriterien durch den Benutzer interaktiv bestimmt werden. Ein Kriterium kann der Abstand zwischen der aktuellen TIN-Oberfläche und dem neuen Punkt sein. Ein weiteres Kriterium kann auf dem Winkel zwischen der Oberfläche mit einem neuen Punkt und der Oberflächen ohne diesen Punkt basieren.

Das letztere Kriterium ist auch, in einer anderen Form, im Verfahren von VOSSELMAN (2000), vgl. auch VOSSELMAN und MAAS (2001); SITHOLE (2001), benutzt. Das Verfahren basiert auf der Voraussetzung, dass im Gelände nur kleine bzw. mäßige Neigungen auftreten können. Dies bedeutet die Festlegung eines maximalen Höhenunterschiedes zwischen zwei Punkten als Funktion der Entfernung zwischen diesen Punkten, $\Delta h_{max}(d)$. Mit Hilfe der Operatoren der mathematischen Morphologie werden aus dem Datensatz Messpunkte entfernt, für die Δh_{max} überschritten wird. Eine Variante des Filters von VOSSELMAN und MAAS (2001) stellt das Verfahren von ROGGERO (2001, 2002) dar.

Die mathematische Morphologie wurde auch in den früheren Arbeiten von LINDENBERGER (1993) und KILLIAN et al. (1996) benutzt. In FRITSCH und KILLIAN (1994) wird die Filterung mit Hilfe von Spline-Funktionen durchgeführt. Eine Übersicht und Bewertung von existierenden Verfahren ist bei LOHMANN et al. (2000) zu finden.

Ein auf einem aktiven Oberflächenmodell basierendes Verfahren wurde von ELMQVIST et al. (2001); ELMQVIST (2002) entwickelt. Das benutzte aktive Oberflächenmodell zeigt gewisse Ähnlichkeiten mit dem Modell der deformierbaren Oberflächen, das wir im Kapitel 5 einführen. In diesem Modell wird die gesamte Energie der Oberfläche,

$$E(\nu) = \sum [E_{int}(\nu) + E_{im}(\nu) + E_{ext}(\nu)] \quad (3.2-11)$$

numerisch in einem iterativen Prozess minimiert. Man arbeitet mit regulären Daten, wobei die Regularisierung auf ähnlicher Weise wie im Verfahren von AXELSSON erfolgt. In jeder Masche eines regulären Gitter wird ein Punkt mit niedrigster Höhe aufgesucht und diese Höhe wird dem entsprechenden Gitterpunkt zugeordnet. Auf dieser Weise entsteht ein „Höhenbild“ $I(x, y)$. Das aktive Oberflächenmodell ist repräsentiert durch die Höhenmatrix $\nu(x, y)$, wobei die Dimension von ν gleich der von I ist. Die innere Energie des Modells ist eine Funktion der ersten, numerisch ermittelten Ableitung,

$$E_{int}(x, y) = C \sum_{m=x-1}^{x+1} \sum_{n=y-1}^{y+1} |\arctan[\nu(x, y) - \nu(m, n)]|, \quad (3.2-12)$$

die mit der Konstanten C skaliert wird. Diese Energie gibt der Oberfläche die Elastizität vor und sie verschwindet, wenn die Oberfläche gegen eine Ebene $\nu(x, y) \equiv \nu$ tendiert. E_{im} ist eine Funktion des Abstandes zwischen I und ν :

$$E_{im}(x, y) = -Ae^{-a[I(x,y)-\nu(x,y)]^2}. \quad (3.2-13)$$

Dieser Energieanteil leistet Widerstand gegen E_{int} und versucht das Modell an die Daten anzupassen. Über die Konstanten A und a wird die Anziehungskraft des Modells ν an die Daten I gesteuert. Die Anziehungskraft ist groß bei kleinen Abständen zwischen I und ν . Dies bedeutet, dass im Zusammenhang mit E_{int} , die Vegetationspunkte, die weit von der Oberfläche liegen, kaum einen Einfluss auf das Modell ν haben.

Ausgehend von einer unter dem niedrigsten Punkt von I liegenden Ebene wird die Energie in (3.2-12) minimiert, indem die Werte von ν in kleinen Schritten erhöht werden. Um den Iterationsprozess zu beschleunigen, wird in ersten Iterationsschritten die externe Energie eingesetzt:

$$E_{ext}(x, y) = -G\nu(x, y). \quad (3.2-14)$$

G ist wahrscheinlich ein zusätzlicher Skalierungsfaktor - die Benutzung dieses Energieanteiles ist anhand der Literatur nicht eindeutig ersichtlich. Abschließend wollen wir noch bemerken, dass die Anwendung der aktiven Modelle wie Snakes zur Modellierung von grobfehlerbehafteten Daten bereits von BORKOWSKI et al. (1997) vorgeschlagen und an Höhenprofilen getestet wurde.

In der letzten Zeit wurde von SITHOLE und VOSSELMAN (2003) ein Rapport über Filtermethoden der Laserscanner-Daten veröffentlicht. In diesem Rapport wurden acht existierende Filterverfahren an geeignet gewählten Datensätzen getestet. Die Filter wurden unter unterschiedlichen Gesichtspunkten analysiert, Filterungsergebnisse qualitativ verglichen. Es hat sich gezeigt, dass im komplizierten Gelände immer noch mehr oder weniger Probleme auftreten.

4 Snakes

Snakes als mathematisches Modell für deformierbare Kurven wurde zunächst zur Anwendung in der digitalen Bildverarbeitung konzipiert. Das universelle Prinzip dieses Modells, die Energieminimierung, stellt ein verallgemeinertes Optimalitätskriterium dar. Dadurch ist dieses Modell auch für andere Bereiche der Geodatenmodellierung attraktiv. Zur Oberflächenmodellierung, insbesondere im Rahmen von digitalen Geländemodellen oder topographischen Informationssystemen, sehen wir die folgenden Anwendungsmöglichkeiten der Snakestechnik:

- Modellierung von Höhenprofilen, insbesondere solcher mit Diskontinuitäten. Hierzu vereinfacht sich das klassische Snake-Modell auf eine Gleichung $z = z(x)$ (BORKOWSKI et al., 1997; BORKOWSKI und MEIER, 1999).
- Modellierung von Raumkurven, insbesondere in 3D-Modellen, inklusive solcher mit Diskontinuitäten. Dazu ist die Dimension des klassischen Modells zu erhöhen, $[x(s), y(s), z(s)]$, was zu einem erweiterten linearen Gleichungssystem führt (TRINDER und LI, 1996; GRÜN und LI, 1997a).
- Modellierung der topographischen Oberfläche mit einem verknüpften, genügend dichten Netz von 1D-, 2D- oder 3D-Snakes (GÜLCH, 1996; FUA, 1996; FUA et al., 2000).

4.1 Snakes - das Modell für deformierbare Kurven

Das Modell der deformierbaren Kurven wurde von KASS et al. (1987) für die Extraktion von unscharfen Linienobjekten aus digitalen Bildern eingeführt. Der zu modellierenden Kurve wird eine Energie zugeteilt, welche für die Gestaltung der Kurve sorgt. Andererseits verfügt das Medium, in dem sich die Kurve befindet, über eine eigene Energie (bzw. ein Potential), welche die innere Struktur dieses Mediums widerspiegelt. In einem Wechselspiel zwischen den Energien versucht die deformierbare Kurve, eine optimale Position, in der sich die Energieanteile ausgleichen, zu finden. Dieses *aktive* Kurvenmodell wird häufig als Snake bezeichnet.

Wir beginnen sogleich mit dem Modell für die Raumkurven, in Parameterdarstellung $[x = x(s), y = y(s), z = z(s)]$ mit dem Parameter (in der Regel die Bogenlänge) s . Es umfasst auch ebene Kurven $[x = x(s), y = y(s)]$ und Höhenprofile $z = z(s)$, wobei hier $s = x$, $z = z(x)$; vgl. Abschnitt 6.1. Für jede, zweckmäßig auf $s \in [0, 1]$ normierte Kurve, kann die optimale Snakes-Lage gefunden werden, indem man die (integrierte) Gesamtenergie der Kurve minimiert:

$$I[x(s), y(s), z(s)] = \int_0^1 (E_{int} + E_{ext}) ds \rightarrow \min. \quad (4.1-1)$$

Die innere Energie E_{int} , auch Gestalts- oder Formenergie genannt, beschreibt die geometrischen Eigenschaften der zu modellierenden Kurve,

$$E_{int} := \frac{1}{2} [\alpha(x_s^2 + y_s^2 + z_s^2) + \beta(x_{ss}^2 + y_{ss}^2 + z_{ss}^2)], \quad (4.1-2)$$

mit den Abkürzungen: $x_s := dx/ds$, $y_s := dy/ds$, $z_s := dz/ds$, $x_{ss} := d^2x/ds^2$, $y_{ss} := d^2y/ds^2$, $z_{ss} := d^2z/ds^2$. Der erste Term der inneren Energie, der sogenannte Elastizitätsterm, bewirkt die Längen-, der zweite, der auch Zähigkeitsterm oder Glattheitsterm genannt wird, die Krümmungsänderung. Mit den Parametern $\alpha = \alpha(s)$ und $\beta = \beta(s)$ werden die beiden Terme gegenseitig bewichtet. Durch das Variieren dieser Parameter lassen sich Kurven mit gewünschten geometrischen Eigenschaften erzeugen, insbesondere durch das Setzen $\beta = 0$ oder $\alpha = \beta = 0$ können Unstetigkeitsstellen bzw. Knickpunkte der Kurve nachgebildet werden.

Der zweite Term in (4.1-1), die externe Energie E_{ext} , macht die Schlange *aktiv* auf der Suche nach der Gleichgewichtslage. Diese Energie bewirkt die Verschiebung der Schlange solange, bis sich die Energierme ausgleichen. Die äußere Energie ist kontext-abhängig. Sie wird durch Daten generiert und ist immer der konkreten Anwendung angemessen zu definieren.

4.1.1 Lösungsmöglichkeiten der Energieminimierung

Die Energieminimierung kann zu mindestens auf den folgenden Wegen realisiert werden:

- Die einfachste Weise ist, die gesamte Energie an jeder einzelnen Stützstelle durch infinitesimale Verschiebung dieser Stützstelle zu minimieren. Die Ableitungen werden dabei explizit an jeder Stützstelle aus den

Daten berechnet. Auf diese Weise arbeitet der sogenannte Greedy-Algorithmus, vorgeschlagen von WILLIAMS und SHAH (1990). BURGHARDT und MEIER (1997a); BURGHARDT (2001) haben Vor- und Nachteile dieses Algorithmus benannt und Anwendungsbereiche aufgezeigt.

- Eine andere Möglichkeit, die Energierme zu minimieren, bietet die dynamische Programmierung. Die dynamische Programmierung ist ein stufenweiser Entscheidungsprozeß. Die optimale Entscheidung wird in jedem Schritt aus der begrenzten Menge der zulässigen Entscheidungen gewählt (AMINI et al., 1988, 1990; GRÜN und LI, 1995; LI, 1997).
- Durch Lösung des Variationsproblems (4.1-1) mit Hilfe der Variationsrechnung. Das ist das klassische Snakes-Konzept von KASS et al. (1987), das auf partielle Differentialgleichungen, die sogenannten Euler-Gleichungen führt, welche geeignet diskretisiert werden müssen. Im Abschnitt 4.1.2 wird auf diese Lösung näher eingegangen.
- Durch die sogenannte direkte Lösung des Variationsproblems (4.1-1). Dazu ist eine geeignete funktionale Darstellung der Kurve zu wählen, diese in die Energierme einzusetzen und das Minimum der Funktion zu finden. Auf diese Weise bekommt man die B-Snakes, wobei als Basisfunktionen die B-Splines eingesetzt werden (MENET et al., 1990, 1991; TRINDER und LI, 1995; GRÜN und LI, 1997a; LIANG et al., 1999; BORKOWSKI und KELLER, 2002).

4.1.2 Variationsproblem und Euler-Gleichungen

Mit Hilfe der Variationsrechnung sind Funktionen $x(s), y(s), z(s)$ zu finden, für welche die gesamte Energie E_{ges} , das Funktional (4.1-1) mit der Energiedefinition (4.1-2) minimal wird:

$$I[x(s), y(s), z(s)] = \int_0^1 E_{ges} ds = \int_0^1 E_{ges}(x, x_s, x_{ss}, y, y_s, y_{ss}, z, z_s, z_{ss}, s) ds \rightarrow \min. \quad (4.1-3)$$

Mit festen Randbedingungen,

$$x(0) = x_a, \quad y(0) = y_a, \quad z(0) = z_a, \quad x(1) = x_e, \quad y(1) = y_e, \quad z(1) = z_e$$

ist die Stationarität von I bei der Variation der gesuchten Funktionen die notwendige Bedingung:

$$\delta I[x + \delta x, y, z] = 0, \quad \delta I[x, y + \delta y, z] = 0, \quad \delta I[x, y, z + \delta z] = 0.$$

Die Lösung der Variationen führt zu Differentialgleichungen 4. Ordnung, den Euler-Gleichungen (die Herleitung kann z.B. bei BURGHARDT (2001) gefunden werden), die nach dem Einsetzen der inneren und der äußeren Energie die Form

$$\frac{\partial E_{ext}}{\partial x} - \alpha x_{ss} + \beta x_{ssss} = 0 \quad (4.1-4)$$

$$\frac{\partial E_{ext}}{\partial y} - \alpha y_{ss} + \beta y_{ssss} = 0 \quad (4.1-5)$$

$$\frac{\partial E_{ext}}{\partial z} - \alpha z_{ss} + \beta z_{ssss} = 0 \quad (4.1-6)$$

haben. Dabei wurde vorausgesetzt, dass die Steuerparameter entlang der Kurve konstant sind ($\alpha = const$ und $\beta = const$).

4.1.3 Diskretisierung mit finiten Differenzen

Die Euler-Gleichungen werden nun mit finiten Differenzen diskretisiert. Mit Bezeichnungen wie in der Abbildung 4.1-1 und der Diskretisierung mit Vorwärtsdifferenzen bekommt man die nötigen Ableitungen an der Stelle i , z.B. für die x-Koordinate,

$$x_{ss,i} = \frac{\Delta_1 x_{i-1} - (\Delta_{-1} + \Delta_1) x_i + \Delta_{-1} x_{i+1}}{\Delta_{-1} \Delta_1}, \quad (4.1-7)$$

$$x_{ssss,i} = \frac{x_{i-2}}{\Delta_{-2}} - \frac{(\Delta_{-1} + 3\Delta_{-2}) x_{i-1}}{\Delta_{-1} \Delta_{-2}} + \frac{3(\Delta_{-1} + \Delta_1) x_i}{\Delta_{-1} \Delta_1} - \frac{(\Delta_1 + 3\Delta_2) x_{i+1}}{\Delta_1 \Delta_2} + \frac{x_{i+2}}{\Delta_2}. \quad (4.1-8)$$

Dabei sind $\Delta_i := s_{i+1} - s_i$ die Stützstellenabstände. Diese Ableitungen sind nun in die Eulerschen Gleichungen einzusetzen und nach Umformen entsteht das lineare Gleichungssystem

$$cx_{i-2} + bx_{i-1} + ax_i + dx_{i+1} + ex_{i+2} + E_x^{ext} = 0. \quad (4.1-9)$$

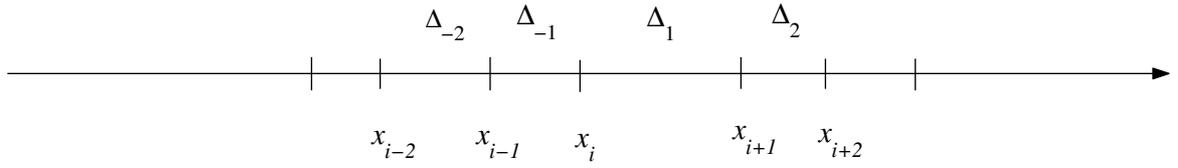


Abbildung 4.1-1: Irreguläre Verteilung der Stützstellen.

Von gleicher Struktur sind auch die Gleichungen für $y(s)$ und $z(s)$, in Matrixschreibweise

$$\mathbf{A}\mathbf{x} + \mathbf{E}_x^{ext} = 0, \quad (4.1-10)$$

$$\mathbf{A}\mathbf{y} + \mathbf{E}_y^{ext} = 0, \quad (4.1-11)$$

$$\mathbf{A}\mathbf{z} + \mathbf{E}_z^{ext} = 0, \quad (4.1-12)$$

mit der pentadiagonalen Matrix

$$\mathbf{A} = \begin{bmatrix} a & d & e & 0 & 0 & 0 & \dots \\ b & a & d & e & 0 & 0 & \dots \\ c & b & a & d & e & 0 & \dots \\ 0 & c & b & a & d & e & \dots \\ 0 & 0 & c & b & a & d & \dots \\ 0 & 0 & 0 & c & b & a & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (4.1-13)$$

und den Abkürzungen

$$c := \frac{1}{\Delta_{-2}}\beta, \quad (4.1-14)$$

$$b := -\frac{1}{\Delta_{-1}}\alpha - \frac{\Delta_{-1} + 3\Delta_{-2}}{\Delta_{-1}\Delta_{-2}}\beta, \quad (4.1-15)$$

$$a := \frac{\Delta_{-1} + \Delta_1}{\Delta_{-1}\Delta_1}(\alpha + 3\beta), \quad (4.1-16)$$

$$d := -\frac{1}{\Delta_1}\alpha - \frac{\Delta_1 + 3\Delta_2}{\Delta_1\Delta_2}\beta, \quad (4.1-17)$$

$$e := \frac{1}{\Delta_2}\beta. \quad (4.1-18)$$

Diese strenge Vorgehensweise wurde bisher nicht praktiziert. Stattdessen nimmt man an, dass die Stützstellen äquidistant entlang der Kurve verteilt sind ($\Delta_i = const$). Ist das nicht der Fall, unterscheiden sich insbesondere die Längen der benachbarten Kurvenabschnitte stark voneinander, so kann man immer die ursprünglichen Stützstellen mittels Interpolation quasi äquidistant verdichten. In BORKOWSKI und MEIER (2001) wurde festgestellt, dass diese Vorgehensweise in manchen Fällen notwendig sein kann, damit man eine akzeptable Lösung bekommt. Solche Probleme treten natürlich nicht auf, wenn die Daten im Rasterformat vorliegen. Das letztere betrifft offensichtlich Kurven im \mathbb{R}^2 .

Eine andere Strategie wurde in BURGHARDT und MEIER (1997a), vgl. auch BURGHARDT (2001), entwickelt: Man arbeitet zwar mit den ursprünglichen, nicht äquidistanten Vektordaten, die direkt in die lineare Gleichungen (4.1-10) und (4.1-11) eingehen, die externe Energie rechnet man aber aus den dicht und äquidistant interpolierten Daten. Diese Vorgehensweise stellt eine Kompromisslösung dar.

Setzt man die äquidistante Abtastung der Kurve voraus, so kann man $\Delta_i \equiv 1$ setzen, und damit vereinfachen sich die entsprechenden Ausdrücke von (4.1-14) bis (4.1-18) zu

$$a = 2\alpha + 6\beta, \quad b = d = -\alpha - 4\beta, \quad c = e = \beta. \quad (4.1-19)$$

Die Bandmatrix \mathbf{A} ist nun symmetrisch, allerdings nicht positiv definit. Deshalb muss geeignet regularisiert und das lineare Gleichungssystem (4.1-10) bis (4.1-12) iterativ gelöst werden. Die einzelnen Gleichungen dieses Systems bezüglich \mathbf{x} , \mathbf{y} , \mathbf{z} sind voneinander unabhängig und können daher getrennt gelöst werden. Die Matrix \mathbf{A} ist in jedem Fall die gleiche.

4.2 Snakes mit lokaler Steuerung

Will man die lokalen Eigenschaften, insbesondere die Knickstellen der Kurve an diskreten Punkten P_i modellieren, so muss auch die innere Energie lokal an diesen Punkten abgeschwächt werden, was durch Variieren der Steuerparameter $\alpha(s)$ und $\beta(s)$ geschieht. In KASS et al. (1987) wird die gesamte Energie bereits vor der Variation diskretisiert,

$$E_{ges} = \sum_{i=1}^n E_{int}(i) + E_{ext}(i), \quad (4.2-1)$$

womit auch die Steuerparameter diskretisiert sind. Dann wird zu den diskreten Euler-Gleichungen übergegangen:

$$\begin{aligned} & \alpha_i(\mathbf{v}_i - \mathbf{v}_{i-1}) - \alpha_{i+1}(\mathbf{v}_{i+1} - \mathbf{v}_i) + \beta_{i-1}(\mathbf{v}_{i-2} - 2\mathbf{v}_{i-1} + \mathbf{v}_i) + \\ & - 2\beta_i(\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}) + \beta_{i+1}(\mathbf{v}_i - 2\mathbf{v}_{i+1} + \mathbf{v}_{i+2}) + \frac{\partial E_{ext}}{\partial \mathbf{v}}(i) = 0. \end{aligned} \quad (4.2-2)$$

Dabei haben wir $x(s)$, $y(s)$ und $z(s)$ in dem Vektor $\mathbf{v} = [x(s) \ y(s) \ z(s)]^\top$ zusammengefasst. Nach Umformen bekommt man das lineare Gleichungssystem (4.1-10) bis (4.1-12) mit der Matrix \mathbf{A} , strukturiert wie in der Gleichung (4.1-13), mit

$$c := \beta_{i-1}, \quad (4.2-3)$$

$$b := -\alpha_i - 2(\beta_{i-1} + \beta_i), \quad (4.2-4)$$

$$a := \alpha_i + \alpha_{i+1} + \beta_{i-1} + 4\beta_i + \beta_{i+1}, \quad (4.2-5)$$

$$d := -\alpha_{i+1} - 2(\beta_i + \beta_{i+1}), \quad (4.2-6)$$

$$e := \beta_{i+1}. \quad (4.2-7)$$

Die kontinuierlichen Euler-Gleichungen lassen sich auch herleiten, wenn man die Variabilität der Steuerparameter zulässt (BORKOWSKI und MEIER, 1999). Mit $\alpha = \alpha(s)$ und $\beta = \beta(s)$ erweitern sich die Euler-Gleichungen (4.1-4) um drei zusätzliche Terme zu

$$\frac{\partial E_{ext}}{\partial \mathbf{v}} - \alpha \mathbf{v}_{ss} + \beta \mathbf{v}_{ssss} - \alpha_s \mathbf{v}_s + \beta_{ss} \mathbf{v}_{ss} + 2\beta_s \mathbf{v}_{sss} = 0. \quad (4.2-8)$$

Nach der Diskretisierung dieser Gleichung mit Rückwärtsdifferenzen ergeben sich die Koeffizienten der Matrix \mathbf{A} zu

$$c := 2\beta_{i-1} - \beta_i, \quad (4.2-9)$$

$$b := -\alpha_{i-1} - 5\beta_{i-1} + \beta_{i+1}, \quad (4.2-10)$$

$$a := \alpha_{i-1} + \alpha_i + 4(\beta_{i-1} + \beta_i) - 2\beta_{i+1}, \quad (4.2-11)$$

$$d := -\alpha_i - \beta_{i-1} - 4\beta_i + \beta_{i+1}, \quad (4.2-12)$$

$$e := \beta_i. \quad (4.2-13)$$

Die Koeffizienten-Matrix \mathbf{A} ist in den beiden Fällen eine nicht symmetrische pentadiagonale Matrix. Sind α und β konstant, so geht jeweils \mathbf{A} in die symmetrische Matrix mit den Koeffizienten von (4.1-19) über.

4.3 Snakes mit Nebenbedingungen

Für manche Aufgaben ist es sinnvoll, der Forderung, die gesamte Energie zu minimieren, eine zusätzliche Nebenbedingung $G(\mathbf{v}) = 0$ hinzuzufügen, der die gesuchte Funktion genügen soll. Man hat beispielsweise eine dreidimensionale Linie mit Snakes zu modellieren, gleichzeitig soll aber die Schlange auf einer vorgegebenen Oberfläche liegen. In diesem Falle ist das Variationsproblem mit Nebenbedingungen zu lösen. Die wichtigste Methode zur Behandlung solcher Probleme ist die Lagrangesche Multiplikatorenmethode. Man bildet zunächst die Grundfunktion,

$$F = E_{ges} + \lambda G, \quad (4.3-1)$$

dann ergeben sich die gesuchten Funktionen und die Parameterfunktion $\lambda = \lambda(t)$ aus den Eulerschen Differentialgleichungen (BRONSTEIN und SEMENDJAJEW, 1962):

$$\frac{\partial E_{ges}}{\partial \mathbf{v}} + \lambda(s) \frac{\partial G}{\partial \mathbf{v}} - \frac{d}{ds} \left(\frac{\partial E_{ges}}{\partial \mathbf{v}_s} \right) = 0. \quad (4.3-2)$$

Die Lösung der Eulerschen Gleichungen mit der zusätzlichen zu bestimmenden Parameterfunktion λ ist mit weiteren Schwierigkeiten verbunden. Verzichtet man jedoch auf die strenge Erfüllung der Nebenbedingung, so kann man die Lösung der Aufgabe mit weniger Aufwand erzielen. Da die Messdaten fehlerbehaftet sind, erwarten wir, dass die Abstände zwischen der Oberfläche z_o und der Schlange z minimal werden,

$$\frac{1}{2}\|z - z_o\|^2 \rightarrow \min, \quad (4.3-3)$$

dann ist $(z - z_o)z_s \equiv G(z) = 0$. Die Differentialgleichung für z erweitert sich um den zusätzlichen Term $\lambda(\partial G/\partial z) = \lambda z_s$ zu

$$\lambda z_s - \alpha z_{ss} + \beta z_{ssss} + \frac{\partial E_{ext}}{\partial z} = 0, \quad (4.3-4)$$

und die Gleichungen für die x - und y -Koordinaten bleiben unverändert. λ ist nun ein zusätzlicher Steuerparameter.

Eine Alternative zur Lagrange-Methode ist, die Nebenbedingung (4.3-3) in die externe Energie zu integrieren. Der neue Energieterm entsteht als lineare Kombination aus der herkömmlichen Energie und einer solchen aus der Nebenbedingung, wobei wiederum ein zusätzlicher Steuerparameter auftritt. Dabei ist von Vorteil, dass das bisherige Gleichungssystem und der Algorithmus erhalten bleiben. Allerdings ist hier nur die Erfüllung einer zusätzlichen Restriktion im Sinne von (4.3-3) möglich.

4.4 Iterationsmöglichkeiten

Nach der Diskretisierung der Euler-Gleichungen, z.B. mit finiten Differenzen, entsteht das lineare Gleichungssystem

$$\mathbf{A}\mathbf{v} + \mathbf{E}_v = \mathbf{0} \quad (4.4-1)$$

mit der nicht positiv definiten Koeffizientenmatrix \mathbf{A} . Um diese diskrete Gleichgewichtsbedingung zu lösen, platzieren wir die Schlange in einem viskosen Medium und lassen sie sich bewegen. Es ist somit die Bewegungsgleichung

$$\gamma \dot{\mathbf{v}} + \mathbf{A}\mathbf{v}(t) + \mathbf{E}_v = \mathbf{0} \quad (4.4-2)$$

mit dem Viskositätsparameter γ zu lösen. Diskretisiert man diese Gleichung mit Rückwärtsdifferenzen, so bekommt man:

$$\gamma(\mathbf{v}_t - \mathbf{v}_{t-1}) + \mathbf{A}\mathbf{v}_t = -\mathbf{E}_v|_{t-1}. \quad (4.4-3)$$

Nach Umformen bekommt man das semilineare Gleichungssystem

$$\mathbf{v}_t = (\mathbf{A} + \gamma\mathbf{I})^{-1}(\gamma\mathbf{v}_{t-1} - \mathbf{E}_v|_{t-1}), \quad (4.4-4)$$

welches iterativ aufgelöst wird. γ ist ein zusätzlicher Auflösungsparameter (meistens gleich Eins gewählt). Gleichung (4.4-4) stellt den klassischen Ansatz von KASS et al. (1987) vor, orientiert auf die Anwendungen in der digitalen Bildverarbeitung.

BURGHARDT und MEIER (1997a) haben die folgende Modifikation eingeführt:

$$\mathbf{v}_t = \mathbf{v}_0 + (\mathbf{A} + \gamma\mathbf{I})^{-1} \left[\gamma(\mathbf{v}_{t-1} - \mathbf{v}_0) - \mathbf{E}_v|_{t-1} \right], \quad (4.4-5)$$

vgl. auch (BURGHARDT und MEIER, 1997b; BURGHARDT, 2001). Dieser Ansatz wurde speziell für die Lösung der bei der kartographischen Generalisierung im Kartenbild entstehenden Konflikte entwickelt. Um solche Konflikte zu beseitigen, gleichzeitig aber die ursprüngliche Gestalt des Linienobjekts möglichst wenig zu ändern, wird immer die aktuelle Snakes-Lage \mathbf{v}_t auf die ursprüngliche \mathbf{v}_0 bezogen. Mit Snakes werden also nicht die aktuellen Koordinaten, sondern die Korrekturen zu den Anfangskoordinaten modelliert.

Dieser Gedankengang kann fortgesetzt werden, indem man fordert, dass die Koordinatenkorrekturen sich nicht auf die ursprünglichen, sondern auf die Lage der Schlange im vorhergehenden Schritt beziehen. Man bekommt dann die Iterationsvorschrift (BORKOWSKI et al., 1997):

$$\mathbf{v}_t = \mathbf{v}_{t-1} - (\mathbf{A} + \gamma\mathbf{I})^{-1}\mathbf{E}_v|_{t-1}. \quad (4.4-6)$$

Die Wahl des Iterationsverfahrens ist immer mit der konkreten Anwendung verbunden, denn die Definition der

externen Energie ist datenabhängig. Obwohl die Formeln (4.4-4), (4.4-5) und (4.4-6) unterschiedliche Vorgehensweisen zur iterativen Lösung des Variationsproblems darstellen, liefern sie praktisch vergleichbare Ergebnisse. Anders jedoch ist der Umgang mit den Formeln. Um eine unscharfe Kontur aus einem digitalen Bild zu extrahieren wird, im einfachsten Fall, die äußere Energie proportional zu den Grauwertstufen gewählt. Die Existenz eines Grauwertgradienten entlang der Kontur erlaubt, die Gleichung (4.4-4), d.h die innere zur äußeren Energie, im Gleichgewicht zu halten. Ist das nicht der Fall oder der Gradient ist klein genug, so läuft die Schlange an den Koordinatenursprung heran (Kollaps).

Die externe Energie, das Verdrängungspotential in der Gleichung (4.4-5), resultiert aus den Lagekonflikten im Kartenbild. Das Ziel ist, diese Konflikte zu minimieren. Das bedeutet, dass die externe Energie verschwinden soll: $E_v|_{t-1} \rightarrow 0$. Ist die externe Energie nicht mehr vorhanden, so kehrt die Schlange zur Ursprungslage zurück und es entstehen wiederum Konflikte. Ohne ein geeignetes Abbruchkriterium, das dem minimalen Wert der externen Energie entspricht, würde der Iterationsprozess nicht enden.

Die Iterationssvorschrift (4.4-6) wurde bei der Modellierung von Höhenprofilen aus Laserscanner-Daten angewendet (BORKOWSKI et al., 1997; BORKOWSKI und MEIER, 1999). Die externe Energie wurde dabei proportional zu den Differenzen zwischen der momentanen Schlangenlage und den Messdaten gewählt. Hier braucht man auch ein Abbruchkriterium, denn sonst würde die approximierende Schlange nach unendlich vielen Iterationsschritten an die Messpunkte laufen. Der Iterationsprozess kann beispielsweise abgebrochen werden, wenn die verbliebenen Residuen die Messfehler nicht überschreiten.

4.5 Snakes-Varianten

Durch das zugrunde liegende universelle Prinzip und ihre Effizienz hat die Snakes-Technik gute Dienste in vielen Bereichen der Geodatenbearbeitung geleistet, insbesondere bei der Objektextraktion aus digitalen Bildern. In der umfangreichen Snakes-Literatur können viele Beispiele gefunden werden, welche diese Aussage bestätigen. Gleichzeitig sind auch gewisse Probleme, die mit der Anwendung der Snakes-Technik verbunden sind, identifiziert worden:

1. Langsame Konvergenz des Iterationsprozesses,
2. Schwierigkeiten mit der Abstimmung der Steuerparameter, insbesondere in der lokalen Umgebung,
3. Schwierigkeiten mit der Identifizierung der optimalen Lage der Schlange in einer verrauschten Umgebung.

Zur Lösung der genannten Probleme wurden mehrere Ansätze und Snakes-Varianten vorgeschlagen. In der Weiterentwicklung der Modelle können drei Richtungen skizziert werden:

- *Konvergenzbeschleunigung.* Als Abhilfe wurde hier der Greedy-Algorithmus (WILLIAMS und SHAH, 1990) und die Dynamische Programmierung (AMINI et al., 1990) vorgeschlagen. Eine bessere Konvergenz kann man auch durch geeignete Wahl der Startschlage erreichen. Ein entsprechender Algorithmus wird in NEUENSCHWANDER et al. (1994) und YUEN et al. (1998) angegeben. Die Konvergenz des Iterationsprozesses wird durch die riesigen Datenmengen beeinträchtigt. Um das zu lösende lineare Gleichungssystem zu reduzieren, wurden unterschiedlichen Parametrisierungen benutzt. Zu dieser Gruppe gehören: FEM-Snakes, B-Snakes, LSB-Snakes und TAFUS-Algorithmus. Die Konvergenzeigenschaften des Iterationsprozesses wurden durch LEYMARIE und LEVINE (1993) und BORKOWSKI et al. (1999) untersucht.
- *Entwicklung der objektorientierten Modelle.* Dabei wird eine verbesserte Identifikation von Objekten einer gewissen Klasse erreicht. Ferner werden dadurch die Probleme vermindert, die unter 2. und 3. genannt sind. Dazu gehören: Ribbon Snakes, Twin Snakes, T-Snakes.
- *Lokale Kontrolle der Objekteigenschaften.* Darunter versteht man die optimale Anpassung der Schlange an jedem Teilstück, insbesondere Nachbildung der Knickstellen. Das letztere geschieht entweder durch das Variieren der Steuerparameter α und β (KASS et al., 1987; BORKOWSKI und MEIER, 1999) oder durch das Modifizieren der externen Energie (XU et al., 1994; XU und PRINCE, 1998a,b; CHAN und VESE, 2001; PARK et al., 2001).

Die Snakes-Modelle können nach unterschiedlichen Gesichtspunkten betrachtet werden. Eine umfangreiche Darstellung der Probleme bei der Geodaten-Verarbeitung mit Hilfe der Snakes-Approximation ist bei MEIER (2000a) angegeben. In dieser Arbeit kann man auch die Erklärung der auf Snakes bezogenen Begriffe finden.

Im Mittelpunkt dieser Arbeit steht die topographische Anwendung der Snakes-Approximation. Aus der Sicht

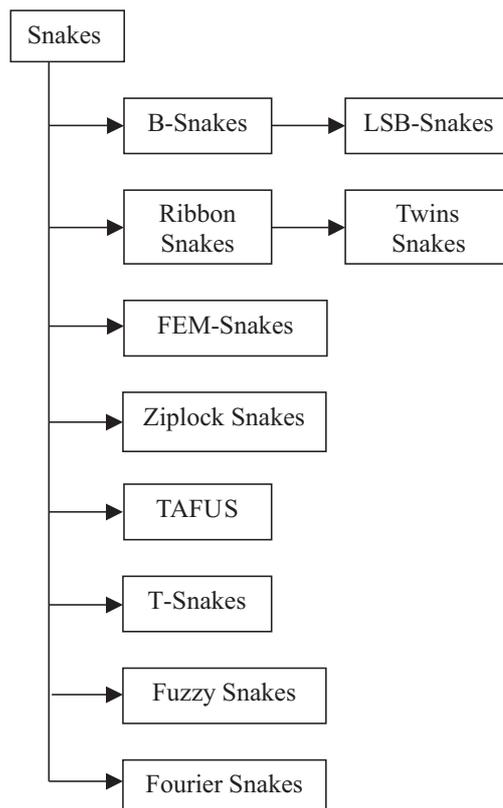


Abbildung 4.5-1: Zusammenstellung der Snakes-Varianten

der Geländemodellierung stehen Modelle und Ansätze im Vordergrund, welche die lokale Kontrolle der Objekteigenschaften ermöglichen. Ferner ist die Möglichkeit, Diskontinuitäten nachzubilden, von großer Bedeutung. Da die Daten nicht immer dicht vorliegen und ferner Lücken beinhalten können, sollen möglicherweise Modelle vorgezogen werden, die eine Zwischeninterpolation zulassen und damit die Datenverdichtung ermöglichen. Die Interpolationseigenschaft besitzen Modelle, welche mit einer parametrisierten Darstellung der Kurve arbeiten (B-Snakes, LSB-Snakes, FEM-Snakes).

Nachfolgend geben wir eine kurze Übersicht der existierenden Snakes-Varianten. In der Abbildung 4.5-1 sind diverse Snakes-Modelle zusammengestellt. Detaillierte Informationen kann der interessierte Leser in der jeweils angegebenen Literaturquelle finden. In der Arbeit von LIANG et al. (1999) ist eine, in einem einheitlichen Finite-Elemente-Konzept verallgemeinerte Darstellung der Snakes-Modelle gegeben. Zu bemerken ist, dass sich bis jetzt kein Modell eindeutig durchgesetzt hat. Alle Varianten haben ihre Vor- und Nachteile.

B-Snakes (*B-Spline Snakes*) entstehen durch die Parametrisierung der verformbaren Kurve mit B-Splines und direkter Lösung des Variationsproblems (vgl. Abschnitt 4.1.1). Die optimale Schlangenlage wird in diesem Modell durch Parameter des Splines beschrieben. Vorteile sind dabei die Interpolationsmöglichkeit, was Verdünnung der ursprünglichen Daten ermöglicht, und damit bessere Konvergenz. Ferner können über Knotengrade Knicke modelliert werden (MENET et al., 1990, 1991; TRINDER und LI, 1995, 1996; LI, 1997; WANG et al., 1999; BRIGGER et al., 2000).

LSB-Snakes (*Least Squares B-Spline Snakes*) stellen ein fortgeschrittene Modell der B-Snakes dar. Sie zeichnen sich vor allem dadurch aus, dass die Parameter der B-Splines nach der Methode der kleinsten Quadrate geschätzt werden. Darüber hinaus lässt dieses Modell die Genauigkeitsanalyse und die Berücksichtigung der geometrischen und photogrammetrischen Vorinformationen zu (GRÜN und LI, 1996, 1997a,b; LI, 1997; FUA et al., 2000).

Ribbon Snakes (*Bandschlangen*) wurden für die Modellierung von Linienobjekten mit konstanter Breite w , wie z.B. Straßen entwickelt. Bandschlangen werden durch drei Komponenten $\{x(s), y(s), w(s)\}$ beschrieben. Aus dem Breitenterm $w(s)$ resultiert der zusätzliche Term der inneren Energie, welcher dafür sorgt, dass die Objektbreite sich im Zuge der Optimierung möglichst minimal ändert (FUA, 1996; MAYER et al., 1997, 1998).

Twin Snakes (*twins; Schlangenpaare*) stellen das Alternativmodell zu den Bandschlangen dar. Linienobjekte mit nicht notwendig konstanter Breite werden (beidseitig) durch Paare von Schlangen begrenzt. Im Gegensatz zu Ribbon Snakes wird hier die Entfernung zwischen den Schlangenpaaren in die externe Energie integriert (GUNN und NIXON, 1997; KERSCHNER, 1998).

FEM-Snakes (*Finite Element Method Snakes*) entstehen durch Lösung des Variationsproblems nach der Methode der finiten Elemente. Die Motivation für die Entwicklung dieses Modells war dieselbe wie im Fall der B-Snakes: Die Schlange interpolationsfähig zu machen, Datenmengen zu reduzieren und folglich den Iterationsprozess zu beschleunigen (COHEN und COHEN, 1990, 1993; COHEN, 1991; BORKOWSKI und KELLER, 2002).

Ziplock Snakes (*reißverschlussartige Snakes*). Für die Extraktion der Objekte aus digitalen Bildern mit der Snake-Technik wird eine gute objektnahe Startposition der Schlange benötigt. Ist das nicht der Fall, so kann es dazu kommen, dass die Schlange das zu identifizierende Objekt überläuft und an einem anderen lokalen Minimum haltmacht. Um diesem Problem abzuweichen, wurden die Ziplock Snakes konzipiert. Man braucht in diesem Konzept nur wenige Punkte am Anfang und Ende des Objektes vorzugeben. Dann wird die Linie, von diesen Punkten an fortschreitend, an das Objekt optimal angepasst. Diese Vorgehensweise ähnelt dem Verhalten des Reißverschlusses, daher der Name (NEUENSCHWANDER et al., 1995, 1997).

TAFUS (*Tangent Angle Function Snakes*) entstehen, wenn man die Tangentenwinkeldarstellung der Kurve benutzt. Dann geht die Kurvenrichtung $\varphi(s)$ und die Richtungsänderung $\partial\varphi/\partial s$ in die innere Energie ein. Aus dem Variationsproblem resultiert nur eine Euler-Gleichung von 2. Ordnung. Man hat somit ein reduziertes Gleichungssystem mit einer tridiagonalen, deutlich besser konditionierten Matrix als die pentadiagonale Systemmatrix zu lösen (BORKOWSKI et al., 1999; BORKOWSKI und MEIER, 2001).

T-Snakes (*Topologically adaptable Snakes*). Durch Verwendung eines speziellen Gitters wird die Snakesreparametrisierung während des Iterationsprozesses und folglich eine bessere topologische Snakes-Anpassung unabhängig von der Startschlange erreicht (MCINERNEY und TERZOPOULOS, 1995).

Fuzzy Snakes Mit Hilfe der Fuzzy-Logik wird unsichere Vorinformation in das Snakes-Konzept integriert. Die Autoren (HÖWING et al., 1997, 2000) versprechen eine Reduktion des Rechenaufwandes und bessere Segmentierung.

Fourier Snakes benutzen die Snakesparametrisierung mit Fourier-Reihen. Durch die spektrale Zerlegung wird eine bessere Konvergenz erreicht (STAIB und DUNCAN, 1992).

4.6 Filtereigenschaften von Snakes

Die Eigenschaften von Snakes können auch aus der Sicht der Filtertheorie betrachtet werden. Diese Betrachtung kann Hinweise für die Wahl der Steuerparametern α und β geben. Die Analyse der Filtereigenschaften von Snakes, insbesondere in Hinsicht auf Modellierung von Diskontinuitäten in Höhenprofilen ist mit Hilfe der Deltafunktionen von MEIER (2000b) gegeben. In (BORKOWSKI und KELLER, 2002) ist die Filtervorschrift von Snakes mit Pseudodifferentialoperatoren dargestellt.

Die Filterung kann formal als Bedingungsgleichung, in der das gefilterte Signal z , kombiniert mit einem linearen Operator L , mit dem Ausgangssignal h übereinstimmen soll, angeschrieben werden:

$$Lz = h. \quad (4.6-1)$$

Der Operator L beschreibt die Filtereigenschaften und ist unabhängig von der Zeit. Aus der breiten Palette von L schränken wir uns ferner auf die Klasse der invarianten Pseudodifferentialoperatoren ein, welche durch

$$Lu = \mathcal{F}^{-1}\{a(\omega)\mathcal{F}\{u\}\} \quad (4.6-2)$$

definiert sind. \mathcal{F} und \mathcal{F}^{-1} stehen für die Fouriertransformation bzw. die inverse Fouriertransformation; ω bezeichnet die Kreisfrequenz. Damit ergibt sich das gefilterte Signal zu

$$z = L^{-1}h = \mathcal{F}^{-1}\left\{\frac{1}{a(\omega)}\mathcal{F}\{h\}\right\}. \quad (4.6-3)$$

Daraus ist ersichtlich, dass die Filtereigenschaften durch das sogenannte Symbol $a(\omega)$ ausgedrückt sind. Der Filtervorgang kann auch in folgender äquivalenter Form aufgefasst werden:

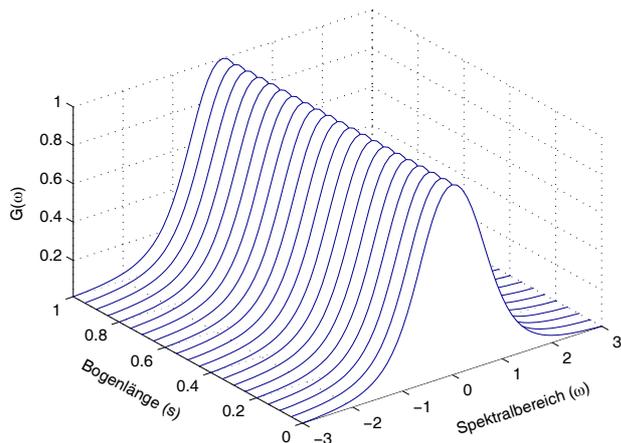


Abbildung 4.6-1: Stationäre Durchlasscharakteristik von Snakes mit $\alpha = const, \beta = const$.

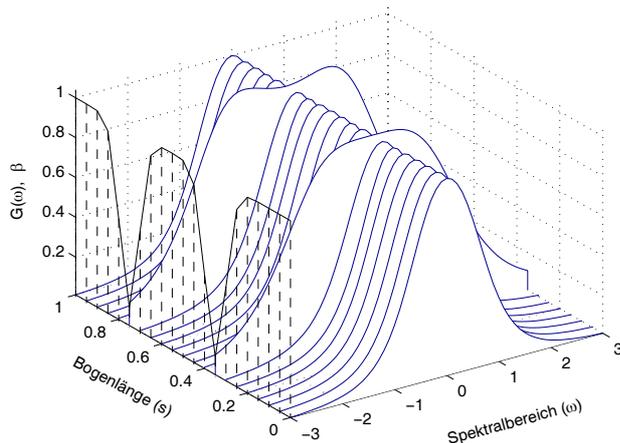


Abbildung 4.6-2: Instationäre Durchlasscharakteristik von Snakes mit dem eingetragenen Verlauf von β , $\alpha = const$

$$\mathcal{F}\{h\} = a(\omega)\mathcal{F}\{z\}. \quad (4.6-4)$$

Daran ist erkennbar, dass das mit $a(\omega)$ modifizierte Spektrum des gefilterten Signals mit dem Spektrum des ursprünglichen Signals übereinstimmen soll. Der Kehrwert von $a(\omega)$ entspricht der aus der Filtertheorie bekannten *Durchlasscharakteristik* $G(j\omega)$, welche die Fouriertransformierte der Gewichtsfunktion der linearen Filtervorschrift ist.

Die Korrespondenz zwischen der linearen Filterung und der Snakes-Technik ist offenbar, wenn man die zum Variationsproblem äquivalente Euler-Gleichungen betrachtet. Um das Problem anschaulicher zu machen, betrachten wir es am Beispiel eines Höhenprofils. Die externe Energie definieren wir als proportional zu den Quadraten der Abstände zwischen den Messwerten h und den Snakeshöhen z :

$$E_{ext} := \frac{1}{2}(h - z)^2, \quad \frac{\partial E_{ext}}{\partial z} = z - h. \quad (4.6-5)$$

Mit dieser Definition hat die Euler-Gleichung, z.B. (4.1-4) die Form

$$z - \alpha z_{ss} + \beta z_{ssss} = h. \quad (4.6-6)$$

Nun ersetzt man den Differentialoperator d/ds durch $j\omega$, $j^2 = -1$, und bekommt die gleichwertige Pseudodifferentialgleichung

$$(1 - \alpha(j\omega)^2 + \beta(j\omega)^4) \cdot \mathcal{F}\{z\} = \mathcal{F}\{h\}, \quad (4.6-7)$$

oder die korrespondierende Snakes-Filtervorschrift

$$z = \mathcal{F}^{-1}\{G(\omega) \cdot \mathcal{F}\{h\}\}. \quad (4.6-8)$$

Der Ausdruck

$$G(\omega) = \frac{1}{1 + \alpha\omega^2 + \beta\omega^4} \quad (4.6-9)$$

entspricht der Durchlasscharakteristik der linearen Übertragung von Snakes. Dies ist ein phasentreues Tiefpassfilter (sog. *Butterworth-Filter*) mit dem dominierenden Einfluss des Parameters β . Das Ergebnis der Snakesapproximation ist also immer ein geglättetes Signal.

In der praktischen Ausführung werden die Euler-Gleichungen diskretisiert und das gefilterte Signal bekommt man als Lösung des linearen Gleichungssystems, vgl. z.B. (4.1-10). Jede Zeile der inversen regularisierten Systemmatrix kann als Satz von Filterkoeffizienten $\{g_i\} =: g$ betrachtet werden, welcher mittels Faltung die Messwerte h in die Snakes-Koordinaten überträgt. Die Fouriertransformierte der Filterkoeffizienten ist gerade die Durchlasscharakteristik (4.6-9). Jede Zeile der Systemmatrix ist gleich, somit sind auch die Filtereigenschaften in jedem Datenpunkt gleich. Sie ändern sich entlang des Höhenprofils (der Kurve) nicht. Dieses ist in der Abbildung 4.6-1 veranschaulicht.

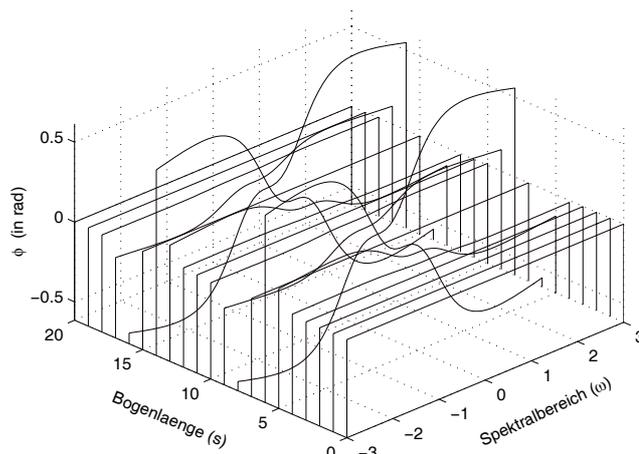


Abbildung 4.6-3: Phasenverschiebung.

Auch für Snakes mit lokaler Steuerung, d.h. mit variablen Koeffizienten $\alpha(s)$ und $\beta(s)$ lässt sich die Filtervorschrift in der Form von (4.6-8) auffassen. Die entsprechende Durchlasscharakteristik hängt nicht nur von den Steuerparametern, sondern auch von deren Ableitungen ab:

$$G(\omega) = \frac{1}{1 + \alpha(s)\omega^2 + \beta(s)\omega^4 - \alpha'(s)j\omega - \beta''(s)\omega^2 - 2\beta'(s)j\omega^3}. \quad (4.6-10)$$

Es ist ersichtlich, dass für konstante α und β die Gleichung (4.6-10) in die Gleichung (4.6-9) übergeht und der Snakes-Filter wird zu einem Tiefpassfilter. Sobald man diese Parameter variieren lässt, wird das Tiefpassfilter mit einem Hochpass kombiniert und es entsteht ein Tiefpassfilter mit Restauration. Dann sind auch die Anteile mit $j\omega$ in (4.6-10) verschieden von Null und die Durchlasscharakteristik ist nicht mehr reellwertig, sondern komplexwertig.

Sollen nun gewisse Stellen im Signal besonders behandelt, z.B. Knickstellen erhalten oder möglichst wenig geglättet werden, so sollen an diesen Stellen auch höhere Frequenzen durchgelassen werden. Dem Tiefpassfilter muss ein Hochpassfilter nachgeschaltet werden; die Filterung wird ortsabhängig. In der Abbildung 4.6-2 ist die Abfolge von Durchlasscharakteristiken für ein Signal mit zwei Knickstellen gezeigt. Gezeichnet sind die Betragswerte $|G(\omega)| = \sqrt{\Im^2(G) + \Re^2(G)}$. Dadurch wird es ersichtlich, wie einzelne (spektrale) Signalanteile geglättet bzw. verstärkt werden. Da aber der Imaginärteil \Im verschieden von Null ist, kommt es zu einer Phasenverschiebung

$$\phi(\omega) = \arctan \frac{\Im(G)}{\Re(G)} \quad (4.6-11)$$

zwischen dem gefilterten und dem ungefilterten Signal. Die Phasenverschiebung für die im Bild 4.6-2 veranschaulichte Filterabfolge ist in der Abbildung 4.6-3 dargestellt. Man sieht, dass für den stationär gefilterten Signalabschnitt keine Phasenverschiebung ($\phi = 0$) auftritt, während in einer Umgebung der Stelle, wo sich die Durchlasscharakteristik ändert, auch ϕ variiert. Das letztere betrifft die benachbarten Stützstellen; an der Knickstelle selbst tritt keine Phasenverschiebung auf. Das ist auch aus der Durchlasscharakteristik ersichtlich: Wenn $\beta(s)$ ein Minimum erreicht, verschwindet der Imaginäranteil und die Durchlasscharakteristik wird reellwertig sowie gerade. Ferner ist zu sehen, dass besonders hohe Frequenzanteile des Signalspektrums von der Phasenverschiebung betroffen sind und die Verschiebung antisymmetrisch zur Knickstelle ist.

Als Folge der Phasenverschiebung können unerwünschte Defekte im gefilterten Signal wie sekundäre Knickstellen auftreten. Diese Defekte können bei der numerischen Realisierung vermindert werden. Während im stationären Fall g immer gerade ist, sind einzelne Zeilen der Systemmatrix in der Knickstellenumgebung ungerade, je mehr desto größer ist die Phasenverschiebung. Man sollte diesen Effekt möglichst klein halten, ferner die Approximation iterativ in kleinen Schritten durchführen.

5 Flakes

5.1 Variationsproblem und Lösungsmöglichkeiten

Snakes sind hinreichend glatte Kurvenstücke, denen Energie zugeordnet wird. Durch Analogie kann man sich hinreichend glatte, energiegeladene Flächenstücke vorstellen. Solche Gebilde wurden in der Arbeit von BORKOWSKI et al. (1997) als *Flakes* bezeichnet. Um die Oberfläche aus (diskreten) Höhenwerten auf dem ebenen Bereich $\mathbb{B} \subset \mathbb{R}^2$ zu modellieren, ist in diesem Konzept die gesamte Energie aller Flächenstücke, das Funktional I zu minimieren:

$$I(z(x, y)) := \iint_{\mathbb{B}} E_{ges}(z; z_x, z_y; z_{xx}, z_{xy}, z_{yy}) dx dy \rightarrow \min. \quad (5.1-1)$$

Die gesamte Energie setzt sich wiederum aus der inneren Energie E_{int} , welche die geometrischen Eigenschaften der Approximationsoberfläche beinhaltet und der externen Energie E_{ext} , welche die Deviation zwischen den zu approximierenden Daten und der Oberfläche widerspiegelt, zusammen:

$$E_{ges} = E_{int} + E_{ext}. \quad (5.1-2)$$

Die externe Energie ist datenabhängig zu definieren, die innere Energie soll Neigungs- und Krümmungsterme beinhalten. Letztere kann als gewichtete Summe der Neigung und der Steifigkeit der Oberfläche aufgefasst werden, was sich als Quadratsumme der Norm des Gradienten und der Norm der Hesse-Matrix realisieren lässt:

$$E_{int} = \frac{\alpha}{2} (z_x^2 + z_y^2) + \frac{\beta}{2} (z_{xx}^2 + 2z_{xy}^2 + z_{yy}^2). \quad (5.1-3)$$

Der erste Term dieses Ausdruckes wird als *membrane* bezeichnet, der zweite als *thin plate* (TERZOPOULOS, 1986). Während der *membrane*-Term den Flächeninhalt der Oberfläche beeinflusst, gestaltet der *thin-plate*-Term die Krümmung dieser Oberfläche.

Eine Funktion $z(x, y)$, welche die Minimalaufgabe (5.1-1) erfüllt, kann auf zwei Wegen gesucht werden:

- Mit Hilfe der Variationsrechnung kann die für die Erfüllung des Variationsproblems notwendige Bedingung, die Eulergleichung, formuliert werden. Diese Gleichung muss dann geeignet diskretisiert werden. Dazu steht die Methode der finiten Differenzen oder die Kollokationsmethode zu Verfügung (GROSSMANN und ROOS, 1992). Das Differenzenverfahren basiert in der Regel auf einem regelmäßigen Gitter. Die gleiche Regularität müssen auch die Daten aufweisen. Der Vorteil liegt bei diesem Verfahren darin, dass die Koeffizientenmatrix des dabei entstehenden Gleichungssystems schwach besetzt ist. Beim Kollokationsverfahren können die Daten absolut irreguläre Struktur aufweisen; dafür ist die Koeffizientenmatrix des Gleichungssystems voll besetzt.
- Statt über die Euler-Gleichungen kann auch das ursprüngliche Variationsproblem direkt mit dem sogenannten *Ritz-Verfahren* gelöst werden. Das Ritz-Verfahren ist eine Näherungsmethode zur Lösung von Variationsproblemen. In diesem Verfahren versucht man, die unbekannte Lösung als eine lineare Kombination von bekannten, beliebig skalierten Ansatzfunktionen darzustellen. Als Unbekannte treten dabei die Skalierungskoeffizienten auf.

5.2 Euler-Gleichungen

Der ebene Bereich \mathbb{B} , auf dem das Funktional (5.1-1) definiert ist, sei begrenzt mit einer zumindest stückweise glatten Kurve L . Das Verschwinden der ersten Variation des Funktionals $I(z(x, y))$,

$$\delta I[z(x, y) + \delta z(x, y)] = 0,$$

mit einer geeigneten Randbedingung entlang L führt zu der Euler-Gleichung 4. Ordnung (BRONSTEIN und SEMENDJAJEW, 1962)

$$E_z - \frac{\partial}{\partial x} E_{z_x} - \frac{\partial}{\partial y} E_{z_y} + \frac{\partial^2}{\partial x^2} E_{z_{xx}} + \frac{\partial^2}{\partial x \partial y} E_{z_{xy}} + \frac{\partial^2}{\partial y^2} E_{z_{yy}} = 0. \quad (5.2-1)$$

Diese Gleichung stellt die notwendige Bedingung für die Existenz einer Extremallösung des ursprünglichen Variationsproblems dar.

Nun sind die entsprechenden Ausdrücke für die Ableitungen zu finden. Diese sind, für $z = z(x, y)$ und die innere Energie nach (5.1-3), im Anhang A angegeben. Nach dem Einsetzen dieser Ableitungen in (5.2-1) und Umformen bekommt man bei konstanten Steuerparametern α und β die Differentialgleichung 4. Ordnung

$$\frac{\partial E_{ext}}{\partial z} - \alpha(z_{xx} + z_{yy}) + \beta(z_{xxxx} + 2z_{xyxy} + z_{yyyy}) = 0. \quad (5.2-2)$$

Will man die Krümmung und Neigung der zu modellierenden Oberfläche lokal gestalten, so muss die lokale Steuerung zugelassen werden. Für die variablen $\alpha = \alpha(x, y)$ und $\beta = \beta(x, y)$ hat die Eulersche Gleichung die Struktur

$$\begin{aligned} \frac{\partial E_{ext}}{\partial z} &- \alpha(z_{xx} + z_{yy}) + \beta(z_{xxxx} + 2z_{xyxy} + z_{yyyy}) - \\ &- (\alpha_x z_x + \alpha_y z_y) + (\beta_{xx} z_{xx} + 2\beta_{xy} z_{xy} + \beta_{yy} z_{yy}) + \\ &+ 2[\beta_x(z_{xxx} + z_{xyy}) + \beta_y(z_{yyy} + z_{xyx})] = 0. \end{aligned} \quad (5.2-3)$$

Natürlich geht (5.2-3) in (5.2-2) über, wenn die Steuerparameter konstant sind. Die Eulerschen Differentialgleichungen müssen nun diskretisiert werden.

5.2.1 Diskretisierung mit finiten Differenzen

Die Ableitungen in den Euler-Gleichungen werden nun durch finite Ausdrücke ersetzt. Während im eindimensionalen Fall (Snakes) sich solche Ausdrücke auch für unregelmäßige Daten problemlos ermitteln lassen, ist es im zweidimensionalen Fall (Flakes) ohne eine Regularisierung der Daten praktisch unmöglich. Es gibt zwar Ansätze für irreguläre Daten (HEINRICH, 1987), wie kurvenlineare Vermaschung, sie sind aber mit einem zusätzlichen, oft erheblichen Aufwand verbunden. In der Praxis dominieren regelmäßige Strukturen der sogenannten Sterne, wie Drei-, Vier- oder Fünfecke. Nachfolgend beschränken wir uns auf den Fall der Daten auf einem regulären Rechteck- oder Quadratgitter (vgl. Abbildung 5.2-1).

Mit der Indizierung der Gitterpunkte wie in der Abbildung (5.2-1) bekommen wir die Approximationsausdrücke für die nötigen Ableitungen:

$$z_{xx}|_{i,j} = \frac{1}{\Delta_x^2}(z_{i+1,j} - 2z_{i,j} + z_{i-1,j}), \quad (5.2-4)$$

$$z_{yy}|_{i,j} = \frac{1}{\Delta_y^2}(z_{i,j+1} - 2z_{i,j} + z_{i,j-1}), \quad (5.2-5)$$

$$z_{xxxx}|_{i,j} = \frac{1}{\Delta_x^4}(z_{i+2,j} - 4z_{i+1,j} + 6z_{i,j} - 4z_{i-1,j} + z_{i-2,j}), \quad (5.2-6)$$

$$z_{yyyy}|_{i,j} = \frac{1}{\Delta_y^4}(z_{i,j+2} - 4z_{i,j+1} + 6z_{i,j} - 4z_{i,j-1} + z_{i,j-2}), \quad (5.2-7)$$

$$\begin{aligned} z_{xyxy}|_{i,j} &= \frac{1}{16\Delta_x^2\Delta_y^2}(z_{i-2,j+2} - 2z_{i,j+2} + z_{i+2,j+2} - 2z_{i-2,j} + \\ &4z_{i,j} - 2z_{i+2,j} + z_{i-2,j-2} - 2z_{i,j} - 2 + z_{i+2,j-1}). \end{aligned} \quad (5.2-8)$$

Im Weiteren setzen wir voraus, dass $\Delta_x = \Delta_y = 1$ gilt. Nun können die finiten Ausdrücke für die Ableitungen in (5.2-2) eingesetzt werden und man bekommt nach Umformen die Euler-Gleichung in diskreter Form

$$\begin{aligned} \frac{\partial E_{ext}}{\partial z} &- (\alpha + 4\beta)z_{i,j+1} + (4\alpha + 12\frac{1}{2}\beta)z_{i,j} - (\alpha + 4\beta)z_{i,j-1} - (\alpha + 4\beta)z_{i+1,j} - \\ &- (\alpha + 4\beta)z_{i-1,j} + \frac{3}{4}\beta z_{i,j+2} + \frac{3}{4}\beta z_{i,j-2} + \frac{3}{4}\beta z_{i+2,j} + \frac{3}{4}\beta z_{i-2,j} + \\ &\frac{1}{8}\beta z_{i-2,j+2} + \frac{1}{8}\beta z_{i+2,j+2} + \frac{1}{8}\beta z_{i-2,j-2} + \frac{1}{8}\beta z_{i+2,j-2} = 0 \end{aligned} \quad (5.2-9)$$

oder

$$\begin{aligned} \frac{\partial E_{ext}}{\partial z} &- dz_{i-2,j+2} + cz_{i,j+2} + dz_{i+2,j+2} + bz_{i,j+1} + cz_{i-2,j} + bz_{i-1,j} + az_{i,j} + \\ &+ bz_{i+1,j} + cz_{i+2,j} + bz_{i,j-1} + dz_{i-2,j-2} + cz_{i,j-2} + dz_{i+2,j-2} = 0, \end{aligned} \quad (5.2-10)$$

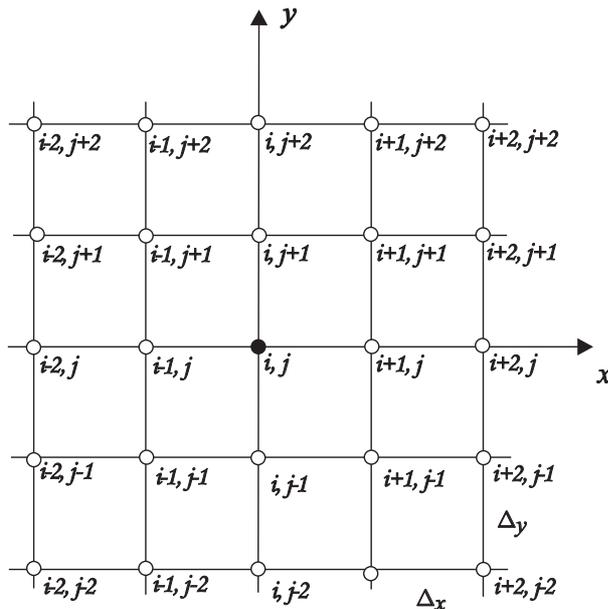


Abbildung 5.2-1: Rechteckiges Gitter für die Diskretisierung mit finiten Differenzen.

wobei die Abkürzungen

$$a := 4\alpha + \frac{25}{2}\beta, \quad b := -(\alpha + 4\beta), \quad (5.2-11)$$

$$c := \frac{3}{4}\beta, \quad d := \frac{1}{8}\beta \quad (5.2-12)$$

eingeführt wurden. Für einen Punkt $P(i, j)$ können die Koeffizienten in Matrixschreibweise zusammengefasst werden:

$$\begin{bmatrix} d & 0 & c & 0 & d \\ 0 & 0 & b & 0 & 0 \\ c & b & a & b & c \\ 0 & 0 & b & 0 & 0 \\ d & 0 & c & 0 & d \end{bmatrix}. \quad (5.2-13)$$

Damit ist ersichtlich, dass in Koordinatenrichtungen alle Differenzen bis zur vierten Ordnung auftreten. Indessen kommen in den Diagonalrichtungen nur die vierten Differenzen vor. Rechnet man ferner die Determinante von (5.2-13) aus, so wird auch ersichtlich, dass die Koeffizientenmatrix nicht positiv definit ist.

Die Gleichung (5.2-10) ist für jeden Datenpunkt aufzustellen. Es entsteht somit ein lineares Gleichungssystem

$$\mathbf{Az} = \mathbf{b}. \quad (5.2-14)$$

Liegen die zu modellierenden Daten auf einem Quadratgitter

$$\begin{matrix} z_{1,1} & z_{1,2} & z_{1,3} & \dots & z_{1,m} \\ z_{2,1} & z_{2,2} & z_{2,3} & \dots & z_{2,m} \\ z_{3,1} & z_{3,2} & z_{3,3} & \dots & z_{3,m} \\ \vdots & & & & \vdots \\ z_{n,1} & z_{n,2} & z_{n,3} & \dots & z_{n,m} \end{matrix}$$

vor, so haben die Vektoren in der Gleichung (5.2-14) die Struktur:

$$\mathbf{z} = [z_{1,1} \ \dots \ z_{1,m} \ z_{2,1} \ \dots \ z_{2,m} \ \dots \ z_{n,1} \ \dots \ z_{n,m}]^T, \quad (5.2-15)$$

$$\mathbf{b} = - [E_z|_{1,1} \ \dots \ E_z|_{1,m} \ E_z|_{2,1} \ \dots \ E_z|_{2,m} \ \dots \ E_z|_{n,1} \ \dots \ E_z|_{n,m}]^T, \quad (5.2-16)$$

einen hohen Grad von Regularität aufweisen. Aus diesem Grunde werden wir im Weiteren auf die Einzelheiten dieser Methode nicht eingehen, stattdessen wenden wir uns dem Ritz-Verfahren, das auch ein Ansatzverfahren ist und gewisse Ähnlichkeiten mit der Kollokationsmethode aufweist, zu. Es hat aber den Vorteil, dass schon Ansatzfunktionen mit niedrigerem Regularitätsgrad ausreichend sind.

5.3 Direkte Lösung des Variationsproblems

Anstatt die Ableitungen bis 4. Ordnung in den Euler-Gleichungen zu approximieren, versuchen wir nun, das Variationsproblem (5.1-1), wo Ableitungen bis 2. Ordnung auftreten, direkt zu lösen, indem wir für die gesuchte Lösung des Problems den Ansatz (5.2-17) machen. Da nicht alle Funktionen als Ansatzfunktionen zugelassen werden, sondern solche der Bauart (5.2-17), wird das Funktional nicht auf dem gesamten Definitionsbereich D , sondern in einem Unterraum $D_n \subset D$, definiert durch

$$D_n := \text{span}\{\varphi_1, \varphi_2, \dots, \varphi_n\}, \quad (5.3-1)$$

minimiert. Deshalb kann nach diesem Verfahren nicht die strenge, sondern eine approximative Lösung des Variationsproblems gefunden werden.

Mit $z(x, y) \approx \phi(x, y)$ definieren wir nun die äußere Energie $E_{ext} := E_{ext}(\phi)$ und die innere Energie

$$E_{int} := \frac{1}{2} \iint_{\mathbb{R}^2} [\alpha (\phi_x^2 + \phi_y^2) + \beta (\phi_{xx}^2 + 2\phi_{xy}^2 + \phi_{yy}^2)] dx dy. \quad (5.3-2)$$

Die gesamte Energie

$$E_{ges} := E_{int} + E_{ext} \rightarrow \min_{D_n} \quad (5.3-3)$$

soll zum Minimum gebracht, oder das Variationsproblem

$$\begin{aligned} & \frac{1}{2} \iint_{\mathbb{R}^2} \{ \alpha [\phi_x^2(x, y, \mathbf{c}) + \phi_y^2(x, y, \mathbf{c})] + \\ & + \beta [\phi_{xx}^2(x, y, \mathbf{c}) + 2\phi_{xy}^2(x, y, \mathbf{c}) + \phi_{yy}^2(x, y, \mathbf{c})] \} dx dy + E_{ext}(x, y, \mathbf{c}) \rightarrow \min_{\mathbf{c}} \end{aligned} \quad (5.3-4)$$

gelöst werden, wobei in dem Vektor $\mathbf{c} = [c_1, c_2, \dots, c_n]^T$ die unbekanntenen Koeffizienten zusammengefasst sind. Nun setzen wir die Funktion (5.2-17) in (5.3-4) ein und erhalten

$$\begin{aligned} & \frac{1}{2} \iint_{\mathbb{R}^2} \left\{ \alpha \left[\left(\sum_{i=1}^n c_i \varphi_{i,x} \right)^2 + \left(\sum_{i=1}^n c_i \varphi_{i,y} \right)^2 \right] + \beta \left[\left(\sum_{i=1}^n c_i \varphi_{i,xx} \right)^2 + 2 \left(\sum_{i=1}^n c_i \varphi_{i,xy} \right)^2 + \right. \right. \\ & \left. \left. + \left(\sum_{i=1}^n c_i \varphi_{i,yy} \right)^2 \right] \right\} dx dy + E_{ext}(x, y, \mathbf{c}) \rightarrow \min, \end{aligned} \quad (5.3-5)$$

$$\begin{aligned} & \frac{1}{2} \iint_{\mathbb{R}^2} \left\{ \alpha \left[\sum_{i,j=1}^n c_i c_j \varphi_{i,x} \varphi_{j,x} + \sum_{i,j=1}^n c_i c_j \varphi_{i,y} \varphi_{j,y} \right] + \beta \left[\sum_{i,j=1}^n c_i c_j \varphi_{i,xx} \varphi_{j,xx} + 2 \sum_{i,j=1}^n c_i c_j \varphi_{i,xy} \varphi_{j,xy} + \right. \right. \\ & \left. \left. + \sum_{i,j=1}^n c_i c_j \varphi_{i,yy} \varphi_{j,yy} \right] \right\} dx dy + E_{ext}(x, y, \mathbf{c}) \rightarrow \min, \end{aligned} \quad (5.3-6)$$

und nach Umformen bekommen wir das zu (5.3-3) äquivalente Variationsproblem

$$\begin{aligned} & \frac{1}{2} \sum_{i,j=1}^n c_i c_j \iint_{\mathbb{R}^2} \{ \alpha [\varphi_{i,x} \varphi_{j,x} + \varphi_{i,y} \varphi_{j,y}] + \beta [\varphi_{i,xx} \varphi_{j,xx} + 2\varphi_{i,xy} \varphi_{j,xy} + \varphi_{i,yy} \varphi_{j,yy}] \} dx dy + \\ & + E_{ext}(x, y, \mathbf{c}) \rightarrow \min. \end{aligned} \quad (5.3-7)$$

Aus der notwendigen Bedingung für die Existenz eines Extremums

$$\frac{\partial E_{ges}}{\partial c_j} = 0 \quad (5.3-8)$$

ergibt sich

$$\sum_{i=1}^n c_i \iint_{\mathbb{R}^2} \{ \alpha [\varphi_{i,x} \varphi_{j,x} + \varphi_{i,y} \varphi_{j,y}] + \beta [\varphi_{i,xx} \varphi_{j,xx} + 2\varphi_{i,xy} \varphi_{j,xy} + \varphi_{i,yy} \varphi_{j,yy}] \} dx dy + \frac{\partial E_{ext}(x, y, \mathbf{c})}{\partial c_j} = 0. \quad (5.3-9)$$

Nun führen wir die folgenden Abkürzungen für die Integrale

$$a_{ij} := \iint_{\mathbb{R}^2} \varphi_{i,x} \varphi_{j,x} dx dy, \quad (5.3-10)$$

$$b_{ij} := \iint_{\mathbb{R}^2} \varphi_{i,y} \varphi_{j,y} dx dy, \quad (5.3-11)$$

$$c_{ij} := \iint_{\mathbb{R}^2} \varphi_{i,xx} \varphi_{j,xx} dx dy, \quad (5.3-12)$$

$$d_{ij} := \iint_{\mathbb{R}^2} \varphi_{i,yy} \varphi_{j,yy} dx dy, \quad (5.3-13)$$

$$e_{ij} := \iint_{\mathbb{R}^2} \varphi_{i,xy} \varphi_{j,xy} dx dy \quad (5.3-14)$$

ein und bekommen die notwendige Bedingung in der Form des linearen Gleichungssystems

$$\sum_{i=1}^n c_i [\alpha (a_{ij} + b_{ij}) + \beta (c_{ij} + 2d_{ij} + e_{ij})] + \frac{\partial E_{ext}(x, y, \mathbf{c})}{\partial c_j} = 0. \quad (5.3-15)$$

Der wichtigste Schritt ist nun die Wahl der geeigneten Ansatzfunktion φ . Durch diese Funktion werden die numerischen und die Approximationseigenschaften der Methode bestimmt. Wünschenswert ist eine Funktion, welche die folgenden Eigenschaften verbürgern würde:

- Sie sei zumindest zweimal differenzierbar,
- sie habe einen beschränkten Träger, was auf ein schwach besetztes Gleichungssystem führt,
- sie lasse möglichst beliebige Datenstrukturen zu,
- sie habe eine solche Gestalt, dass sich die Integrale (5.3-10) bis (5.3-14) möglichst analytisch auswerten lassen.

Ferner muss die externe Energie zweckmäßig definiert werden.

5.3.1 Linearer Finite-Elemente-Ansatz für reguläre Daten

Zunächst versuchen wir die Koeffizienten $a_{ij}, b_{ij}, \dots, e_{ij}$ für die regelmäßig im \mathbb{R}^2 verteilten Daten auszurechnen, die auf einem regulären Gitter vorliegen. Für solche Daten liegt es nahe, eine Funktion $\varphi_i(x, y)$ in der Form des Tensorprodukts

$$\varphi_i(x, y) = \varphi_i(x) \varphi_i(y), \quad (5.3-16)$$

als Ansatzfunktion für die Gleichung (5.2-17) zu wählen. Als Basisfunktion für $\varphi_i(x)$ und $\varphi_i(y)$ wählen wir die lineare Funktion mit beschränktem Träger

$$\varphi_i(s) = \begin{cases} \frac{1}{\Delta}(s - (i-1)\Delta) & , \quad (i-1)\Delta \leq s \leq i\Delta \\ 1 - \frac{1}{\Delta}(s - i\Delta) & , \quad i\Delta \leq s \leq (i+1)\Delta \\ 0 & , \quad \text{sonst} \end{cases} \quad (5.3-17)$$

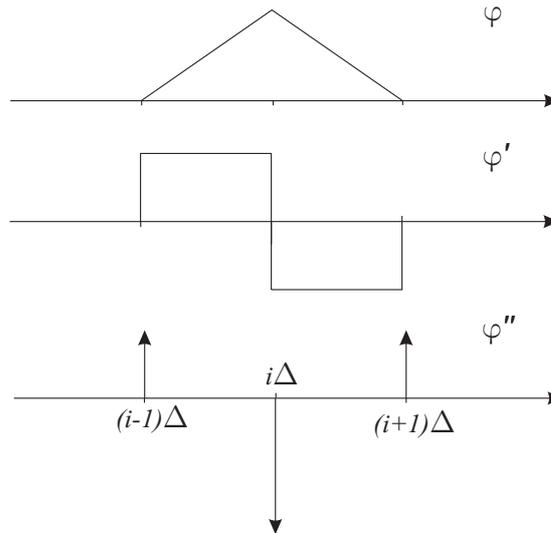


Abbildung 5.3-1: Lineare Basisfunktionen und ihre Ableitungen.

womit dieser Ansatz ein Finite-Elemente-Ansatz ist. Diese Basisfunktion und ihre Ableitungen

$$\varphi'_i(s) = \begin{cases} \frac{1}{\Delta} & , \quad (i-1)\Delta \leq s \leq i\Delta \\ -\frac{1}{\Delta} & , \quad i\Delta \leq s \leq (i+1)\Delta \\ 0 & , \quad \text{sonst} \end{cases} \quad (5.3-18)$$

und

$$\varphi''_i(s) = \frac{1}{\Delta} [\delta(s - (i-1)\Delta) - 2\delta(s - i\Delta) + \delta(s - (i+1)\Delta)], \quad (5.3-19)$$

wobei $\delta(\cdot)$ die Deltadistribution ist, sind in der Abbildung 5.3-1 veranschaulicht. Die zweidimensionale Ansatzfunktion (5.3-16) ist in der Abbildung 5.3-2 gezeigt. Die benötigten Ableitungen von $\varphi_i(x, y)$ ergeben sich zu

$$\begin{aligned} \varphi_x &:= \frac{\partial \varphi(x, y)}{\partial x} = \varphi'(x)\varphi(y), & \varphi_y &:= \frac{\partial \varphi(x, y)}{\partial y} = \varphi(x)\varphi'(y), \\ \varphi_{xx} &:= \frac{\partial^2 \varphi(x, y)}{\partial x^2} = \varphi''(x)\varphi(y), & \varphi_{yy} &:= \frac{\partial^2 \varphi(x, y)}{\partial y^2} = \varphi(x)\varphi''(y), \\ \varphi_{xy} &:= \frac{\partial^2 \varphi(x, y)}{\partial x \partial y} = \varphi'(x)\varphi'(y). \end{aligned} \quad (5.3-20)$$

Nun können die Integrale (5.3-10) bis (5.3-14) ausgewertet werden. Einzelheiten zur Integration können im Anhang C gefunden werden; hier geben wir die Endresultate an:

$$a_{ij} = \frac{\Delta y}{\Delta x} \begin{cases} \frac{4}{3} & , \quad i = k; j = l \\ \frac{1}{3} & , \quad i = k; |j - l| = 1 \\ -\frac{2}{3} & , \quad |i - k| = 1; j = l \\ -\frac{1}{6} & , \quad |i - k| = 1; |j - l| = 1 \\ 0 & , \quad \text{sonst,} \end{cases} \quad (5.3-21)$$

$$b_{ij} = \frac{\Delta x}{\Delta y} \begin{cases} \frac{4}{3} & , \quad i = k; j = l \\ -\frac{2}{3} & , \quad i = k; |j - l| = 1 \\ \frac{1}{3} & , \quad |i - k| = 1; j = l \\ -\frac{1}{6} & , \quad |i - k| = 1; |j - l| = 1 \\ 0 & , \quad \text{sonst,} \end{cases} \quad (5.3-22)$$

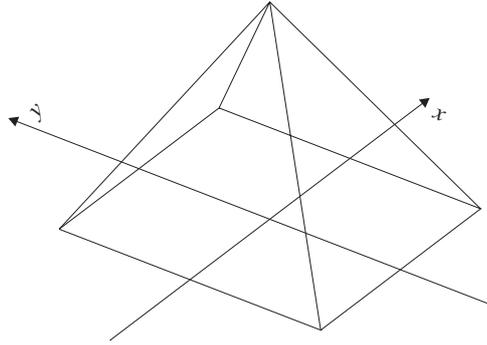


Abbildung 5.3-2: Zweidimensionale Basisfunktion für Finite-Elemente-Flakes.

$$c_{ij} = \frac{\Delta_y}{\Delta_x^2} \begin{cases} 4 & , \quad i = k; j = l \\ -\frac{8}{3} & , \quad i = k; |j - l| = 1 \\ \frac{2}{3} & , \quad i = k; |j - l| = 2 \\ 1 & , \quad |i - k| = 1; j = l \\ -\frac{2}{3} & , \quad |i - k| = 1; |j - l| = 1 \\ \frac{1}{6} & , \quad |i - k| = 1; |j - l| = 2 \\ 0 & , \quad \text{sonst,} \end{cases} \quad (5.3-23)$$

$$d_{ij} = \frac{\Delta_x}{\Delta_y^2} \begin{cases} 4 & , \quad i = k; j = l \\ 1 & , \quad i = k; |j - l| = 1 \\ -\frac{8}{3} & , \quad |i - k| = 1; j = l \\ -\frac{2}{3} & , \quad |i - k| = 1; |j - l| = 1 \\ \frac{2}{3} & , \quad |i - k| = 2; j = l \\ \frac{1}{6} & , \quad |i - k| = 2; |j - l| = 1 \\ 0 & , \quad \text{sonst,} \end{cases} \quad (5.3-24)$$

$$e_{ij} = \frac{1}{\Delta_x \Delta_y} \begin{cases} 4 & , \quad i = k; j = l \\ -2 & , \quad i = k; |j - l| = 1 \\ -2 & , \quad |i - k| = 1; j = l \\ 1 & , \quad |i - k| = 1; |j - l| = 1 \\ 0 & , \quad \text{sonst.} \end{cases} \quad (5.3-25)$$

Um die Aufgabe ein wenig zu vereinfachen, setzen wir wieder voraus, dass die Daten auf einem Quadratgitter vorliegen: $\Delta_x = \Delta_y =: 1$. Da die Ansatzfunktion $\varphi(x, z)$ die lineare Funktion mit dem beschränkten Träger ist, können wir ferner die diskreten Werte z_i der gesuchten Lösung $z(x, y)$ als die Koeffizienten der Ansatzfunktion benutzen. Dann sind die entsprechenden Koeffizienten zu summieren:

$$a_{ij} + b_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{6} & -\frac{2}{3} & -\frac{1}{6} & 0 \\ 0 & \frac{1}{3} & \frac{4}{3} & \frac{1}{3} & 0 \\ 0 & -\frac{1}{6} & -\frac{2}{3} & -\frac{1}{6} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{6} & \frac{1}{3} & -\frac{1}{6} & 0 \\ 0 & -\frac{2}{3} & \frac{4}{3} & -\frac{2}{3} & 0 \\ 0 & -\frac{1}{6} & -\frac{1}{3} & -\frac{1}{6} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & 0 \\ 0 & -\frac{1}{3} & \frac{8}{3} & -\frac{1}{3} & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.3-26)$$

und auf gleiche Weise

$$c_{ij} + 2d_{ij} + e_{ij} = \begin{bmatrix} 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \\ \frac{1}{6} & \frac{2}{3} & -\frac{17}{3} & \frac{2}{3} & \frac{1}{6} \\ \frac{2}{3} & -\frac{17}{3} & 16 & -\frac{17}{3} & \frac{2}{3} \\ \frac{1}{6} & \frac{2}{3} & -\frac{17}{3} & \frac{2}{3} & \frac{1}{6} \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \end{bmatrix}. \quad (5.3-27)$$

Die Resultate von (5.3-26) und (5.3-27) sind nun mit den Parametern α und β entsprechend zu multiplizieren

und dann zu summieren. Das Ergebnis der Summation bezeichnen wir mit \mathbf{B} und führen noch die Matrix

$$\mathbf{Z} = \begin{bmatrix} z_{i-2,j+2} & z_{i-1,j+2} & z_{i,j+2} & z_{i+1,j+2} & z_{i+2,j+2} \\ z_{i-2,j+1} & z_{i-1,j+1} & z_{i,j+1} & z_{i+1,j+1} & z_{i+2,j+1} \\ z_{i-2,j} & z_{i-1,j} & z_{i,j} & z_{i+1,j} & z_{i+2,j} \\ z_{i-2,j-1} & z_{i-1,j-1} & z_{i,j-1} & z_{i+1,j-1} & z_{i+2,j-1} \\ z_{i-2,j-2} & z_{i-1,j-2} & z_{i,j-2} & z_{i+1,j-2} & z_{i+2,j-2} \end{bmatrix} \quad (5.3-28)$$

ein. Dann bilden wir das *Hadamardsche* Produkt

$$\mathbf{B} * \mathbf{Z} := (a_{ij} \cdot b_{ij}). \quad (5.3-29)$$

Die Summe der Elemente der gewonnenen Matrix ergibt die zu der Eulergleichung (5.2-10) äquivalente lineare Bedingungsgleichung

$$\begin{aligned} & ez_{i-1,j+2} + cz_{i,j+2} + ez_{i+1,j+2} + ez_{i-2,j+1} + dz_{i-1,j+1} + bz_{i,j+1} + dz_{i+1,j+1} + ez_{i+2,j+1} + \\ & + cz_{i-2,j} + bz_{i-1,j} + az_{i,j} + bz_{i+1,j} + cz_{i+2,j} + ez_{i-2,j-1} + dz_{i-1,j-1} + bz_{i,j-1} + \\ & + dz_{i+1,j-1} + ez_{i+2,j-1} + ez_{i-1,j-2} + cz_{i,j-2} + ez_{i+1,j-2} + \frac{\partial E_{ext}}{\partial z_i} = 0 \end{aligned} \quad (5.3-30)$$

mit den Koeffizienten

$$a := \frac{8}{3}\alpha + 16\beta, \quad b := -\frac{1}{3}(\alpha + 17\beta), \quad (5.3-31)$$

$$c := \frac{2}{3}\beta, \quad d := -\frac{1}{3}(\alpha - 2\beta), \quad (5.3-32)$$

$$e := \frac{1}{6}\beta. \quad (5.3-33)$$

Um die beiden Verfahren zu vergleichen, fassen wir wieder die Koeffizienten im Punkt $P(i, j)$ in die Matrix

$$\begin{bmatrix} 0 & e & c & e & 0 \\ e & d & b & d & e \\ c & b & a & b & c \\ e & d & b & d & e \\ 0 & e & c & e & 0 \end{bmatrix} \quad (5.3-34)$$

zusammen. Aus dem Vergleich der Matrizen (5.2-13) und (5.3-34) ist ersichtlich, dass man mit linearen finiten Elementen eine bessere Approximation der Ableitungen bekommt als durch die Diskretisierung der Eulergleichungen mit finiten Differenzen. Im letzteren Fall treten nur die ersten und die zweiten Differenzen in den Koordinatenrichtungen und die zweiten Differenzen in den Hauptdiagonalenrichtungen auf. Überwiegend wird die Information entlang der Koordinatenrichtungen genutzt. Bei der Approximation der Ableitungen im Finite-Elemente-Ansatz werden alle Differenzen außer der zweiten in den Hauptdiagonalenrichtungen berücksichtigt. Ein weiterer Vorteil dieses Ansatzes stellen seine Interpolationseigenschaften dar. Im Gegensatz zu den Eulergleichungen können mit dem Finite-Elemente-Ansatz bei Bedarf die ursprünglichen Daten zwischeninterpoliert werden.

Für die praktische Anwendung muss noch die äußere Energie zweckmäßig definiert und ggf. ihre Ableitung numerisch berechnet werden. Fordert man indessen, dass diese Energie proportional zu dem Datendefekt wird,

$$E_{ext} := \frac{1}{2} \iint_{\mathbb{R}^2} \left[\sum_{i=1}^n z_i \varphi_i(x, y) - w \right]^2 dx dy, \quad (5.3-35)$$

so ist

$$\frac{\partial E_{ext}}{\partial z_i} = \sum_{i=1}^n z_i \iint_{\mathbb{R}^2} \varphi_i(x, y) \varphi_j(x, y) dx dy - \iint_{\mathbb{R}^2} w \varphi_i(x, y) dx dy. \quad (5.3-36)$$

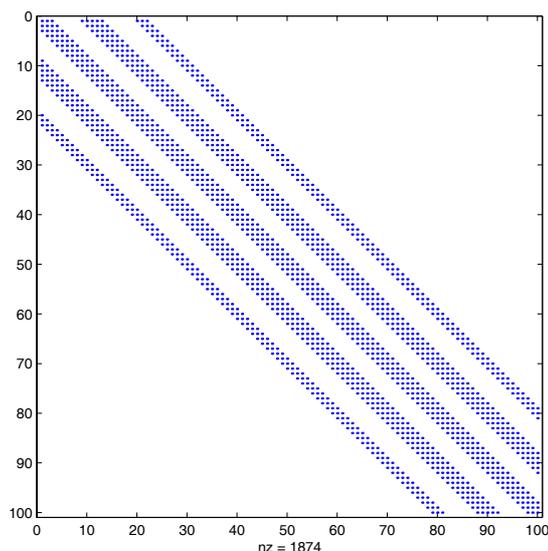


Abbildung 5.3-3: Struktur der schwach besetzten Systemmatrix.

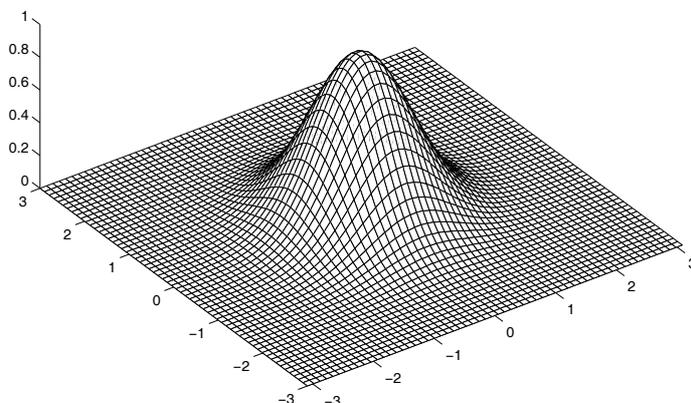


Abbildung 5.3-4: Zweidimensionale Gaußsche Basisfunktion

Das zweite Integral in diesem Ausdruck ist gleich den Daten- bzw. Messwerten w_k , das erste Integral ergibt sich (vgl. Anhang C) zu

$$\iint_{\mathbb{R}^2} \varphi_i(x, y) \varphi_j(x, y) dx dy = \Delta_x \Delta_y \begin{cases} \frac{4}{9} & , \quad i = k; j = l \\ \frac{1}{9} & , \quad i = k; |j - l| = 1 \\ \frac{1}{9} & , \quad |i - k| = 1; j = l \\ \frac{1}{36} & , \quad |i - k| = 1; |j - l| = 1 \\ 0 & , \quad \text{sonst.} \end{cases} \quad (5.3-37)$$

Mit diesen Ergebnissen nimmt die lineare Bedingungsgleichung die Form

$$\begin{aligned} & e z_{i-1, j+2} + c z_{i, j+2} + e z_{i+1, j+2} + e z_{i-2, j+1} + \tilde{d} z_{i-1, j+1} + \tilde{b} z_{i, j+1} + \tilde{d} z_{i+1, j+1} + e z_{i+2, j+1} + \\ & + c z_{i-2, j} + \tilde{b} z_{i-1, j} + \tilde{a} z_{i, j} + \tilde{b} z_{i+1, j} + c z_{i+2, j} + e z_{i-2, j-1} + \tilde{d} z_{i-1, j-1} + \tilde{b} z_{i, j-1} + \\ & + \tilde{d} z_{i+1, j-1} + e z_{i+2, j-1} + e z_{i-1, j-2} + c z_{i, j-2} + e z_{i+1, j-2} = w_k \end{aligned} \quad (5.3-38)$$

an und gilt für alle Datenwerte w_k ; $k = 1, 2, \dots, n$. Die Koeffizienten a, \dots, e sind durch die Gleichungen (5.3-31) bis (5.3-33) definiert, wobei die folgenden Koeffizienten zu korrigieren sind:

$$\tilde{a} := a + 4/9 \qquad \tilde{b} := b + 1/9 \qquad \tilde{d} := d + 1/36. \quad (5.3-39)$$

Das lineare Gleichungssystem (5.3-30) bzw. (5.3-38) lässt sich natürlich auch in Matrixschreibweise darstellen. Die Systemmatrix ist dabei eine schwach besetzte Bandmatrix mit ähnlicher Struktur wie \mathbf{A} in Abschnitt 5.2.1, was ein großer Vorteil aus numerischer Sicht ist. Leider ist die Anwendung dieser Lösung auf reguläre bzw. regularisierte Daten begrenzt. Die schwach besetzte (Band-)Struktur von \mathbf{A} ist in Abbildung 5.3-3 visualisiert.

5.3.2 Quasi-Finite-Elemente Ansatz für irreguläre Daten

In diesem Abschnitt wollen wir das Ansatzverfahren auf irregulär in \mathbb{R}^2 verteilte Datenstützstellen erweitern. Während für reguläre Daten unterschiedliche Ansatzfunktionen, insbesondere Splines in Tensorproduktform, eingesetzt werden können, ist die Wahl der Ansatzfunktionen für irreguläre Daten eher begrenzt. Eine naheliegende Ansatzfunktion, welche die im Abschnitt 5.3 formulierten Anforderungen recht gut erfüllt, ist die Funktion vom Gauß-Typ:

$$\varphi_i(x, y) = e^{-\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma_i^2}}. \quad (5.3-40)$$

Diese Funktion hat im Punkt P_i den Wert eins und nimmt dann mit wachsender Entfernung von diesem Punkt ab (vgl. Abbildung 5.3-4). Die Schnelligkeit des Abklingens wird durch den frei wählbaren Parameter σ_i gesteuert. Die signifikanten Werte der Funktion konzentrieren sich um den Punkt P_i herum. Für Punkte, die weit

genug vom Punkt P_i entfernt sind, liegen die Werte der Ansatzfunktion nahe Null und können als unbedeutend betrachtet werden. Daher bezeichnen wir diesen Ansatz als Quasi-Finite-Elemente-Ansatz.

Mit der Ansatzfunktion (5.3-40) sind die Integrale (5.3-10) bis (5.3-14) auszuwerten. Dazu benötigen wir die ersten Ableitungen

$$\nabla \varphi_i(x, y) = -\frac{1}{\sigma_i^2} \left| \begin{array}{c} (x - x_i) \\ (y - y_i) \end{array} \right| \varphi_i(x, y) \quad (5.3-41)$$

und die zweiten Ableitungen

$$\begin{aligned} \nabla^2 \varphi_i(x, y) &= -\frac{1}{\sigma_i^2} \nabla \left| \begin{array}{c} (x - x_i) \\ (y - y_i) \end{array} \right| \varphi_i(x, y) - \frac{1}{\sigma_i^2} \left| \begin{array}{c} (x - x_i) \\ (y - y_i) \end{array} \right| \nabla \varphi_i(x, y) \\ &= -\frac{1}{\sigma_i^2} \left\{ \mathbf{I} - \frac{1}{\sigma_i^2} \left| \begin{array}{cc} (x - x_i)^2 & (x - x_i)(y - y_i) \\ (x - x_i)(y - y_i) & (y - y_i)^2 \end{array} \right| \right\} \varphi_i(x, y). \end{aligned} \quad (5.3-42)$$

∇ steht hier für den Nablaoperator und \mathbf{I} für die Einheitsmatrix. Die Ausführung der Integration, die im Anhang D skizziert ist, ergibt die Koeffizienten:

$$a_{ij} = \frac{2\pi}{a} \exp \left[\frac{b_x^2 + b_y^2 - a(c_x + c_y)}{2a\sigma_i^2\sigma_j^2} \right] \left[\frac{\sigma_i^2\sigma_j^2}{a} + \frac{b_x}{a} \left(\frac{b_x}{a} - x_i - x_j \right) + x_i x_j \right], \quad (5.3-43)$$

$$b_{ij} = \frac{2\pi}{a} \exp \left[\frac{b_x^2 + b_y^2 - a(c_x + c_y)}{2a\sigma_i^2\sigma_j^2} \right] \left[\frac{\sigma_i^2\sigma_j^2}{a} + \frac{b_y}{a} \left(\frac{b_y}{a} - y_i - y_j \right) + y_i y_j \right], \quad (5.3-44)$$

$$\begin{aligned} c_{ij} &= \frac{2\pi}{a\sigma_i^2\sigma_j^2} \exp \left[\frac{b_x^2 + b_y^2 - a(c_x + c_y)}{2a\sigma_i^2\sigma_j^2} \right] [m_{4x} - 2(x_i + x_j)m_{3x} + \\ &\quad + (x_i^2 + x_j^2 + 4x_i x_j - a)m_{2x} - 2(x_i x_j^2 + x_j x_i^2 - b_x)m_{1x} + x_i^2 x_j^2 + \sigma_i^2 \sigma_j^2 - c_x], \end{aligned} \quad (5.3-45)$$

$$\begin{aligned} d_{ij} &= \frac{2\pi}{a\sigma_i^2\sigma_j^2} \exp \left[\frac{b_x^2 + b_y^2 - a(c_x + c_y)}{2a\sigma_i^2\sigma_j^2} \right] [m_{4y} - 2(y_i + y_j)m_{3y} + \\ &\quad + (y_i^2 + y_j^2 + 4y_i y_j - a)m_{2y} - 2(y_i y_j^2 + y_j y_i^2 - b_y)m_{1y} + y_i^2 y_j^2 + \sigma_i^2 \sigma_j^2 - c_y] \end{aligned} \quad (5.3-46)$$

und

$$\begin{aligned} e_{ij} &= \frac{2\pi}{a\sigma_i^2\sigma_j^2} \exp \left[\frac{b_x^2 + b_y^2 - a(c_x + c_y)}{2a\sigma_i^2\sigma_j^2} \right] [m_{2x}m_{2y} - m_{2x}m_{1y}(y_i + y_j) - m_{1x}m_{2y}(x_i + x_j) + \\ &\quad + y_i y_j [m_{2x} - m_{1x}(x_i + x_j)] + x_i x_j [m_{2y} - m_{1y}(y_i + y_j)] + \\ &\quad + m_{1x}m_{1y}(x_i + x_j)(y_i + y_j) + x_i x_j y_i y_j]. \end{aligned} \quad (5.3-47)$$

In diesen Ausdrücken wurden die folgenden Abkürzungen verwendet:

$$\begin{aligned} a &= \sigma_i^2 + \sigma_j^2 & b_x &= \sigma_i^2 x_j + \sigma_j^2 x_i \\ b_y &= \sigma_i^2 y_j + \sigma_j^2 y_i & c_x &= \sigma_i^2 x_j^2 + \sigma_j^2 x_i^2 \\ c_y &= \sigma_i^2 y_j^2 + \sigma_j^2 y_i^2 & m_{1y} &= \frac{b_y}{a} \\ m_{1x} &= \frac{b_x}{a} & m_{3(x,y)} &= m_{1(x,y)}^3 + 3 \frac{\sigma_i^2 \sigma_j^2}{a} m_{1(x,y)} \\ m_{2(x,y)} &= m_{1(x,y)}^2 + \frac{\sigma_i^2 \sigma_j^2}{a} & m_{4(x,y)} &= m_{1(x,y)}^4 + 6 \frac{\sigma_i^2 \sigma_j^2}{a} m_{1(x,y)}^2 + 3 \frac{\sigma_i^4 \sigma_j^4}{a^2}. \end{aligned} \quad (5.3-48)$$

Die Momente m_2 , m_3 und m_4 sind sowohl für die x - als auch für die y -Koordinate zu bestimmen.

Die externe Energie definieren wir erneut proportional zum Datendefekt,

$$E_{ext} = \frac{1}{2} \sum_{k=1}^n [\phi_k(x, y) - w_k]^2 \quad (5.3-49)$$

über alle Datenpunkte P_k ; $k = 1, 2, \dots, n$. Mit dem Ansatz (5.2-17) ist dann

$$E_{ext} = \frac{1}{2} \sum_{k=1}^n \left[\sum_{i=1}^n \sum_{j=1}^n c_i c_j \varphi_i(x_i, y_i) \varphi_j(x_j, y_j) - 2 \sum_{i=1}^n c_i \varphi_i(x_i, y_i) w_k + w_k^2 \right]$$

und

$$\frac{\partial E_{ext}}{\partial c_j} = \sum_{i=1}^n \sum_{k=1}^n c_i \varphi_i(x_k, y_k) \varphi_j(x_k, y_k) - \sum_{k=1}^n \varphi_i(x_k, y_k) w_k. \quad (5.3-50)$$

Die zu (5.3-15) äquivalente Bedingungsgleichung wird dann

$$\sum_{i=1}^n c_i \left[\alpha(a_{ij} + b_{ij}) + \beta(c_{ij} + 2d_{ij} + e_{ij}) + \sum_{k=1}^n \varphi_i(x_k, y_k) \varphi_j(x_k, y_k) \right] = \sum_{k=1}^n \varphi_i(x_k, y_k) w_k. \quad (5.3-51)$$

Nun führen wir noch die Matrizen

$$\mathbf{A}_{\alpha, \beta} := (\alpha(a_{ij} + b_{ij}) + \beta(c_{ij} + 2e_{ij} + d_{ij})) \quad (5.3-52)$$

und

$$\mathbf{L} := (\varphi_j(x_i, y_i)) \quad (5.3-53)$$

ein und schreiben die Bedingungsgleichungen in der Matrixnotation

$$(\mathbf{A}_{\alpha, \beta} + \mathbf{L}^\top \mathbf{L}) \mathbf{c} = \mathbf{L}^\top \mathbf{w}. \quad (5.3-54)$$

Die Auflösung dieses linearen Gleichungssystem ergibt die Skalierungsfaktoren $\mathbf{c} = [c_1, c_2, \dots, c_n]^\top$ an den Datenpunkten. Weitere Einzelheiten, insbesondere zur numerischen Seite dieses Ansatzes, können bei BORKOWSKI und KELLER (2003) gefunden werden.

5.4 Filtereigenschaften von Flakes

Die filtertheoretische Betrachtung von Snakes lässt sich leicht auch auf den zweidimensionalen Fall erweitern. Die Veranschaulichung ist aber nicht mehr so gut überschaubar. Analog wie bei Snakes kann man fordern, dass das zweidimensionale deformierbare Modell $z(x, y)$ möglichst gut die Messdaten h approximiert und man muss die Euler-Gleichung (5.2-2) entsprechend umformen. Zu der Euler-Gleichung kann man die gleichwertige Pseudodifferentialoperator-Gleichung oder die korrespondierende Flakes-Filtervorschrift

$$z = \mathcal{F}^{-1}\{G(\omega_x, \omega_y) \cdot \mathcal{F}\{h\}\} \quad (5.4-1)$$

formulieren. \mathcal{F} ist nun die zweidimensionale Fouriertransformation und die Durchlasscharakteristik G ist eine Funktion zweier Variablen ω_x und ω_y . Im stationären Fall ($\alpha = const$, $\beta = const$) ist das Flakes-Filter ein zweidimensionales Tiefpassfilter mit

$$G(\omega_x, \omega_y) = \frac{1}{1 + \alpha(\omega_x^2 + \omega_y^2) + \beta(\omega_x^4 + 2\omega_x^2\omega_y^2 + \omega_y^4)}. \quad (5.4-2)$$

Man erkennt wieder den dominierenden Einfluss des vor dem Glattheitsterm im Flakesansatz stehenden Parameters β . Die phasentreue und rotationssymmetrische Durchlasscharakteristik (5.4-2) ist in der Abbildung 5.4-1 gezeigt. Bei Flakes mit konstanten Steuerparametern sind die Filtereigenschaften, somit die Durchlasscharakteristik in jedem Datenpunkt gleich, also stationär auf dem gesamten Definitionsbereich der Daten.

Wird die Variabilität von $\alpha = \alpha(x, y)$ und $\beta = \beta(x, y)$ zugelassen, so wird die Durchlasscharakteristik

$$G(\omega_x, \omega_y) = 1 / \left\{ 1 + \alpha \|\omega\|^2 + \beta(\omega^4 + 2\omega_x^2\omega_y^2 + \omega_y^4) - j(\alpha_x\omega_x + \alpha_y\omega_y) - (\beta_{xx}\omega_x^2 + 2\beta_{xy}\omega_x\omega_y + \beta_{yy}\omega_y^2) - 2j[\beta_x(\omega_x^3 + \omega_x\omega_y^2) + \beta_y(\omega_y^3 + \omega_y\omega_x^2)] \right\} \quad (5.4-3)$$

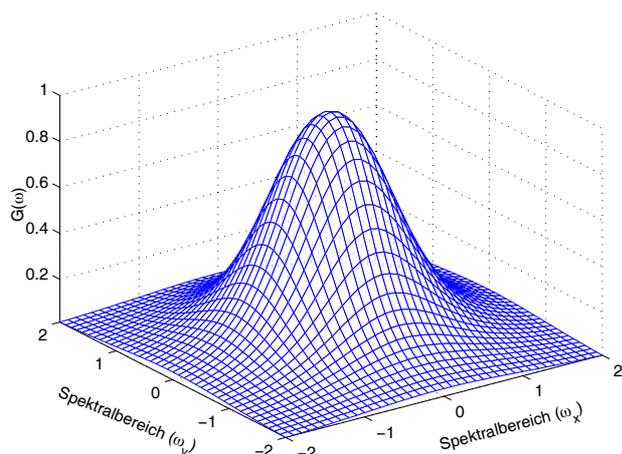


Abbildung 5.4-1: Stationäre Durchlasscharakteristik von Flakes mit $\alpha = \text{const}$, $\beta = \text{const}$

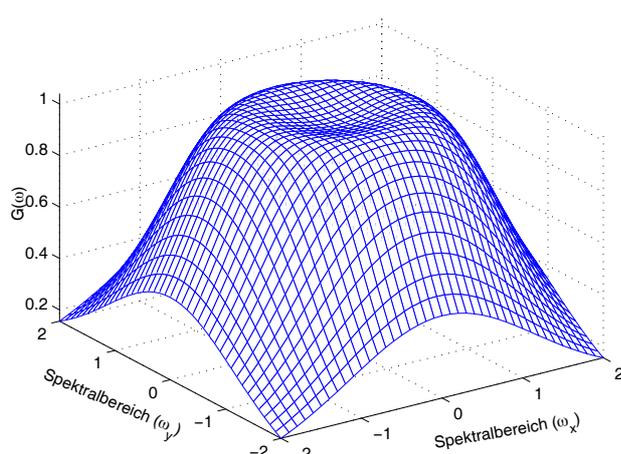


Abbildung 5.4-2: Durchlasscharakteristik von Flakes mit variablem β , $\alpha = \text{const}$

komplexwertig und ist daher allgemein nicht mehr phasentreu. Ähnlich wie im eindimensionalen Fall ist dieses Flakes-Filter ein Tiefpassfilter mit Restauration: Ein Tiefpass-, das mit einem Hochpassfilter kombiniert wurde. Ein Beispiel ist in der Abbildung 5.4-2 gezeigt. Dabei wurde lediglich die Variabilität von β zugelassen. Das Bild zeigt die Durchlasscharakteristik in einem zur Kante gehörenden Datenpunkt, wo $\beta = \beta(x, y)$ ein Minimum erreicht; daher ist $G(\omega_x, \omega_y)$ reellwertig und rotationssymmetrisch. In den benachbarten Datenpunkten werden Ableitungen β_x bzw. β_y verschieden von Null und die Durchlasscharakteristik wird komplexwertig mit einer Phasenverschiebung ungleich Null. Es können ähnliche Defekte wie im eindimensionalen Fall auftreten. In sonstigen, außerhalb des Kantenbereiches liegenden Datenpunkten wird die Durchlasscharakteristik von dem im Bild 5.4-1 dargestellten Typ. Die Filterung wird instationär bzw. ortsabhängig.

6 Approximation von Profilen mit Snakes

6.1 Glatte Profile

Den experimentellen Teil dieser Abhandlung beginnen wir mit der eindimensionalen Approximation der Daten durch die deformierbaren Modelle. Die Approximationseigenschaften dieser Modelle wollen wir an grobfehlerbehafteten Höhenprofilen demonstrieren. Die Schlange soll natürlich die Daten optimal approximieren und gleichzeitig die Grobfehler beseitigen. Dazu ist die externe Energie geeignet zu definieren. In BORKOWSKI et al. (1997) haben wir bereits einen einfachen Ansatz für die externe Energie,

$$E_{ext}(r) \propto \begin{cases} \frac{1}{2}r^2 & \text{wenn } r < \varepsilon, \\ 0 & \text{sonst} \end{cases} \quad (6.1-1)$$

gemacht. Danach ist E_{ext} (quadratisch) proportional zu den Residuen $r := z_d - z_t$, wobei durch z_d die gegebenen Messhöhen und durch z_t die momentanen "Snakeshöhen" bezeichnet werden. Die Anziehungskraft wirkt nur in einem begrenzten Bereich von z_t . Der Schwellwert ε wird proportional zur Standardabweichung $\sigma_{a-priori}$ des Abtastfehlers im grobfehlerlosen Gelände, etwa $\varepsilon := \gamma\sigma_{a-priori}$, $\gamma = 2$ bis 3 gewählt. Der diskutierte Energieansatz braucht eine gute Näherung für die Startschlange, was sich mittels Tiefpassfilterung realisieren lässt, allerdings mit einem zusätzlichen Aufwand. Darüber hinaus können Schwierigkeiten bei der Approximation von Daten mit clusterförmigen Ausreißern auftreten.

Die genannten Probleme lassen sich umgehen, wenn man die Struktur der Ausreißer berücksichtigt. Geht man davon aus, dass die grobfehlerbehafteten Daten hauptsächlich *über* der zu modellierenden Oberfläche liegen, so liegt es nahe, die äußere Energie folgendermaßen zu bestimmen:

$$E_{ext}(r) \propto \begin{cases} -\frac{1}{2}r^2 & \text{wenn } r < 0, \\ \frac{\sigma^2}{2}e^{-r^2/\sigma^2} & \text{wenn } r \geq 0. \end{cases} \quad (6.1-2)$$

Ihre Ableitung ergibt sich unmittelbar zu

$$\frac{\partial E_{ext}}{\partial z_t} = \begin{cases} r & \text{wenn } r < 0, \\ re^{-r^2/\sigma^2} & \text{wenn } r \geq 0. \end{cases} \quad (6.1-3)$$

Dieser Ansatz bedeutet, dass die Ableitung der externen Energie eine zum Abstand zwischen den Daten und dem momentanen Snakesverlauf proportionale Anziehungskraft ist, welche dafür sorgt, dass die Snakesfunktion an die Daten gezogen wird. Für die positiven Residuen r wird diese Abstandsfunktion durch die Gaußfunktion gedämpft. Dadurch wirkt diese Kraft nur auf einem relativ kleinen Abstand und sperrt somit die Grobfehler aus. Die Größe der Sperre wird über den Parameter σ^2 unter Kontrolle gehalten. Die negativen Residuen werden nicht gedämpft. Das (negative) Potential wird umso größer, je höher die momentane Snakesfunktion über den Daten liegt. Dadurch wird die Approximationsfunktion im Bereich von Punkten mit den niedrigsten Höhen verankert. Die Wirkung $E_z := \frac{\partial E_{ext}}{\partial z_i}$ der externen Energie ist in Abhängigkeit von den Residuen r in der Abbildung 6.1-1 dargestellt. Das Maximum dieser Funktion liegt bei $r_m = \sigma/\sqrt{2}$.

Das deformierbare Modell wird an die Höhendaten iterativ nach der Vorschrift

$$\mathbf{z}_t = (\mathbf{A} + \mathbf{I})^{-1}(\mathbf{z}_{t-1} + \mathbf{E}_{\mathbf{z},t-1}) \quad (6.1-4)$$

angepasst, wobei \mathbf{A} die Matrix (4.1-13) ist und E_z in jedem Datenpunkt aufgrund der Ordinaten aus dem $(t-1)$ ten Iterationsschritt ermittelt wird. Der Iterationsprozess endet, wenn das Gleichgewicht zwischen der inneren und der äußeren Energie erreicht ist. Dann ist $z_t \approx z_{t-1}$.

Die Snakesapproximation demonstrieren wir an zwei Beispielen. Der in der Abbildung 6.1-2 veranschaulichte Test 1 zeigt ein mit einem Laserscanner abgetastetes Höhenprofil mit einzelnen Reflektionen an der Vegetation, wobei die Messdaten durch Punkte repräsentiert sind. Der Test 2 in der Abbildung 6.1-3 zeigt eine Stadtsilhouette. Es ist ein Profilausschnitt aus dem Datensatz *Stockholm* mit großen Gebäudekomplexen (vgl. Abbildung 7.3-1). Als Startschlange wurde in beiden Fällen eine über die Daten gelegte, etwa auf der Höhe des höchsten Punktes liegende Gerade gewählt. Die beiden Tests zeigen eine korrekte Approximation der Profile mit der vollständigen Grobfehlerbeseitigung. Die unterschiedliche Steuerung in den Tests musste der Datenstruktur angepasst werden. Während im Test 1 die Schlange relativ "weich" war, musste ihre Steifigkeit im Test 2 deutlich

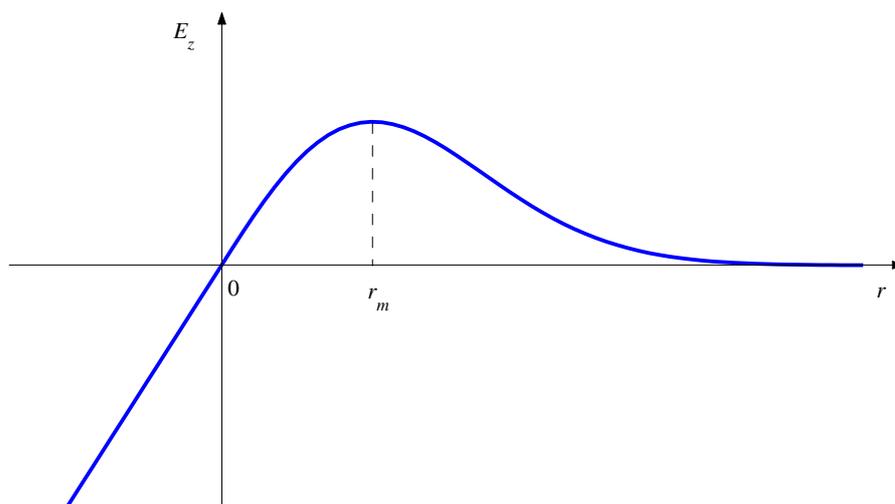
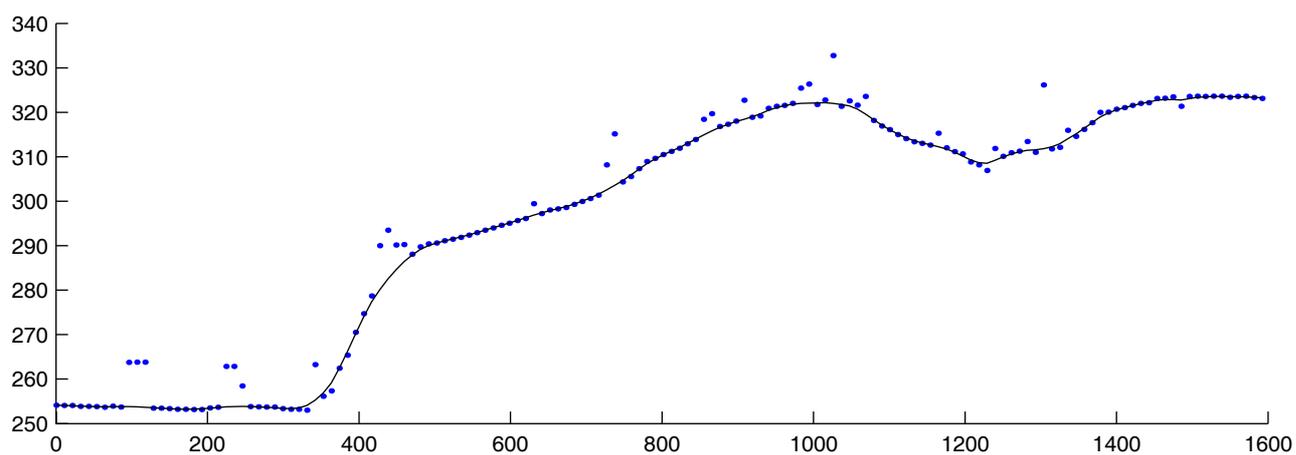
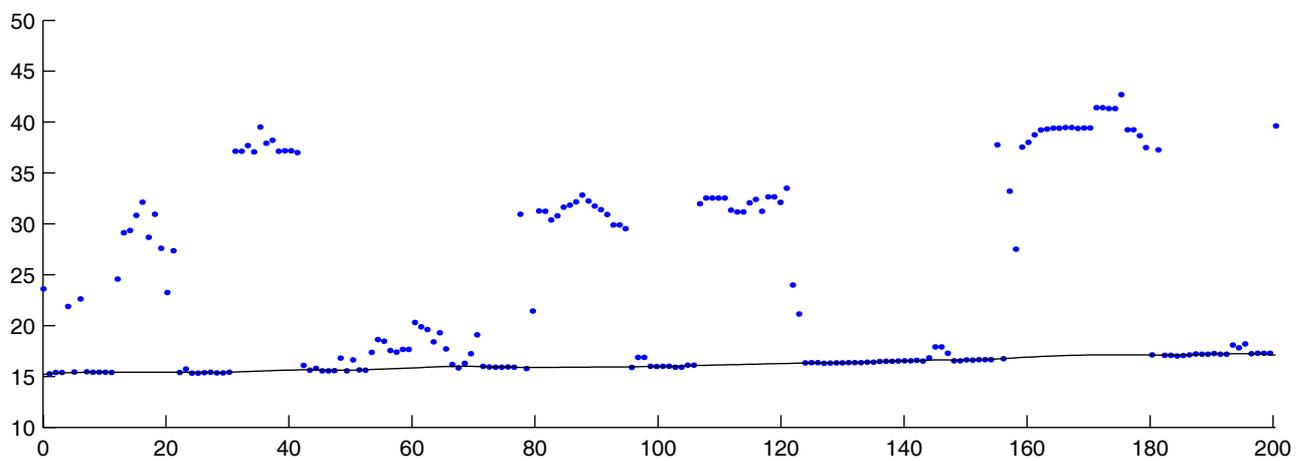


Abbildung 6.1-1: Die Ableitung der externen Energie

Abbildung 6.1-2: Robuste Profilapproximation mit Snakes - Test 1. (Steuerparameter: $\alpha = 1$, $\beta = 1$, $\sigma^2 = 3.5$)Abbildung 6.1-3: Robuste Profilapproximation mit Snakes - Test 2. (Steuerparameter: $\alpha = 5$, $\beta = 5$, $\sigma^2 = 0.2$)

erhöht werden. Eine zu weiche Schlange würde das durch die Dächer generierte Potential nicht überwinden können und die Schlange würde an den Gebäuden "hängen bleiben". Auch die Wirkungsbreite der externen Energie wurde deutlich kleiner ausgelegt, damit die kleinen Ausreißer (wahrscheinlich Reflexionen an Fahrzeugen) vollständig beseitigt werden. Bei der Laserabtastung treten gelegentlich und vereinzelt auch Punkte, die unter der Oberfläche liegen, auf. Solche Abtastfehler entstehen durch Mehrwegreflexionen. Die innere Steifigkeit der Schlange ermöglicht es, auch solche Fehler zu minimieren. Ein Beispiel dazu ist im Test 1 gegen das rechte Ende des Profils zu finden.

Generell kann festgestellt werden, dass die Modellierung von glatten Profilen mit Snakes keine Schwierigkeiten bereitet. Die Approximationseigenschaften können beliebig mit Hilfe der Parameter α und β gestaltet werden. In demselben Approximationsvorgang können auch Grobfehler beseitigt werden. Das letztere betrifft auch Fehlerkonstellationen, die mit anderen Verfahren schwierig bzw. mit zusätzlichem Aufwand beseitigt werden können. Das effiziente Mittel dazu ist die vorgeschlagene Formulierung (6.1-2) der externen Energie.

6.2 Profile mit Diskontinuitäten

Die Modellierung von Diskontinuitäten kann in den deformierbaren Modellen zumindest auf drei Wegen realisiert werden:

1. Durch das Entziehen der inneren Energie an den zu modellierenden Knickstellen. Dies bedeutet, dass in den diskreten Punkten P_i $\alpha(x_i) = \beta(x_i) = 0$ zu setzen ist (*point break*).
2. Durch das Entziehen bzw. Herabsetzen des Krümmungsterms (*tangent break*). Dies bedeutet, dass lediglich die Variabilität des Parameters β zugelassen wird. $\alpha > 0$ wird dabei durchweg konstant gehalten.
3. Durch das Kompensieren der inneren Energie mittels entsprechender Korrekturterme zur äußeren Energie.

Während die ersten zwei Ansätze eine ortsabhängige Steuerung und damit die ortsabhängige Gestaltung der inneren Energie bedeuten, werden in dem letzten Ansatz die Steuerparameter konstant belassen. Ortsabhängig ist dabei die externe Energie. Denkbar sind auch Kombinationen der Ansätze, welche eine ortsabhängige Gestaltung sowohl der inneren als auch der äußeren Energie zulassen. Die Voraussetzung für eine erfolgreiche Modellierung von Diskontinuitäten ist ein hinreichendes Vorwissen über das Vorhandensein und die etwaige Lage derselben.

Der 3. Ansatz wurde von XU et al. (1994) vorgeschlagen und ist mit Erfahrung in der digitalen Bildverarbeitung begründet. Anstatt die innere Energie an den Knickstellen herabzusetzen, was immer eine Modifikation der Systemmatrix bedeutet, wird die Systemmatrix unverändert gelassen ($\alpha = const$, $\beta = const$) und die externe Energie wird durch entsprechende Korrekturterme verstärkt. Die Korrekturterme sind von der lokalen Krümmung κ_i in den Punkten P_i abhängig und nehmen bei konstantem α die Form

$$k_i := \beta |\kappa_i|^3 \quad (6.2-1)$$

an. Damit soll der Wölbungsterm der inneren Energie in dem Punkt P_i kompensiert werden. Die lokale Krümmung kann numerisch verschiedenartig geschätzt werden. Diesen Ansatz haben wir bereits in der Arbeit (BORKOWSKI und MEIER, 1999) untersucht und konnten die positive Erfahrung aus der digitalen Bildverarbeitung bei der Profilmmodellierung nicht bestätigen.

In der genannten Arbeit haben wir auch unterschiedliche Vorgehensweisen zur Modellierung von Knickstellen durch das Herabsetzen des Wölbungsterms in der inneren Energie untersucht. Hierzu gibt es Kombinationen aus zwei Lösungsvarianten der Euler-Gleichungen mit variablen Steuerparametern: Die Systemmatrix \mathbf{A} kann entweder mit den Koeffizienten (4.2-3) bis (4.2-7) oder (4.2-9) bis (4.2-13) aufgebaut werden (vgl. Abschnitt 4.2). Das lineare Gleichungssystem kann nach der Iterationsvorschrift (4.4-4) oder (4.4-6) aufgelöst werden. Um den Wölbungsterm herabzusetzen, muss der Parameter β an den diskreten Stellen x_i herabgesetzt werden:

$$\beta(x) = \beta - \beta_0 \sum_i \delta(x - x_i), \quad (6.2-2)$$

wobei $\delta(\cdot)$ die Deltafunktion bezeichnet und $0 < \beta_0 \leq \beta$. Da die Lage der Knicke in der Regel nur unscharf bekannt ist, kann es sinnvoll sein, die Schlange in einer gewissen Umgebung der vermutlichen Lage x_i zu schwächen, z.B. mit dem Ansatz

$$\beta(x) = \beta - \beta_0 \sum_i e^{-(x-x_i)^2/2\epsilon^2}, \quad (6.2-3)$$

wobei 2ε die Breite des Schwankungsbereiches bedeutet. Damit kann auch eine Phasenverschiebung, die zu den im Abschnitt 4.6 diskutierten Defekten führen kann, minimiert werden.

Aus der gesamten Palette der durchgeführten Tests haben wir sechs gewählt und in der Abbildung 6.2-1 veranschaulicht. Sie repräsentieren unterschiedliche Signale in Form einer Böschungsfolge, einer Sprung- und einer Dreieckfunktion. Die detaillierten Informationen zu den einzelnen Testbeispielen sind in der Tabelle 6.2-1 zusammengestellt. Unter der Bezeichnung "konventionell" sind die Euler-Gleichungen (4.2-2) zu verstehen, wenn die gesamte Energie bereits vor der Variation diskretisiert wird. Die exakte Lösung mit variablen Koeffizienten (4.2-8) ist als "exakt" bezeichnet. Bei der Iteration "klassisch" bzw. "modifiziert" handelt sich um die Iterationsvorschriften (4.4-4) bzw. (4.4-6) entsprechend. Im Gegensatz zur robusten Approximation, die wir an glatten Profilen demonstriert haben, war hier die äußere Energie immer eine symmetrische, zu den Residuenquadraten proportionale Funktion:

$$E_{ext}(r) \propto \frac{1}{2}r^2. \quad (6.2-4)$$

Mit Kreisen sind in der Abbildung 6.2-1 fehlerbehaftete Messwerte gezeichnet, wobei vorausgesetzt wurde, dass mögliche Grobfehler vorher, in einem getrennten Vorgang eliminiert worden sind. Die ausgezogenen Linien repräsentieren die Snakesapproximation der Messdaten. Die punktierten Linien zeigen, in einer unterschiedlichen Skalierung, den ortsabhängigen Verlauf des Parameters β , wenn der Ansatz (6.2-3) benutzt wurde. Als Startschlange wurden immer die stark tiefpassgefilterten ursprünglichen Daten verwendet. Test A und B repräsentieren den klassischen Ansatz von KASS et al. (1987), allerdings unterschiedlich realisiert. Die Ergebnisse sind vergleichbar. Test C und D vertreten die exakte Lösung des Variationsproblems mit variabler Steuerung und einer stark asymmetrischen Systemmatrix. Anders war die Iterationsvorschrift und demzufolge auch die Steuerung. Bei diesem Ansatz treten mehr oder weniger sichtbare Defekte in Form von kleinen Sekundärknicken neben den Knicken auf. Diese Defekte sind umso größer, je "stärker" die Schlange ist. Sie lassen sich teilweise im modifizierten Iterationsprozess reduzieren. Test E und F belegen, dass auch für extreme Signalformen die Knicke modelliert werden können.

In der Abbildung 6.2-2 zeigen wir ein Beispiel, an dem wir versucht haben, in *einem* Iterationsvorgang die Ausreißer zu eliminieren, die Messfehler zu glätten und die Knicke zu modellieren bzw. zu erhalten. Es wurde der konventionelle Snakes-Algorithmus mit der Iterationsvorschrift (4.4-4) benutzt. Für die Beschreibung der externen Energie wurde die bewertete Funktion (6.1-2) mit $\sigma^2 = 5$ verwendet. Als Startschlange wurde eine über die Daten gelegte Gerade gewählt. Die Knicke wurden als *point break* modelliert, wobei sowohl für β als auch für α der Ansatz (6.2-2) mit $\alpha = \beta = \alpha_0 = \beta_0 = 1.5$ benutzt wurde. Dies bedeutet, dass an den Knickstellen die innere Energie vollständig beseitigt wurde. Die Messdaten wurden an diesen Stellen nicht mehr geglättet. Die Qualität dieser Approximation ist zufriedenstellend und entspricht unserer Erwartung.

Nun wollen wir unsere Erkenntnisse und Erfahrung zur Modellierung von Profilen mit Diskontinuitäten zusammenfassen:

- Mit dem deformierbaren Snakes-Modell ist eine Kantenmodellierung fast immer möglich, aber nicht immer einfach. Die Vorgehensweise und die Qualität der Modellierung hängt stark von den Daten bzw. von der Signalform ab: Art und Abstand benachbarter Kanten, Hangneigung, Signal-Rausch-Verhältnis.
- Die Anforderungen, Messfehler zu glätten und die Kanten zu erhalten, stehen im Widerspruch. Für die Glättung muss die Schlange genügend angespannt werden (großes β), für die Kantenmodellierung hingegen soll die Schlange (lokal) genügend "weich" sein. Generell: Je stärker man glätten will, umso schwieriger sind Kanten zu modellieren.

Tabelle 6.2-1: Ausführungsvarianten und Steuerparameter zu den Tests der Abbildung 6.2-1

Test	Euler-Gl.	Iteration	α	β
A	konventionell	klassisch	0.1	(6.2-2), $\beta = \beta_0 = 1$
B	konventionell	modifiziert	10	(6.2-3), $\beta_0 = 100$, $\varepsilon = 1$
C	exakt	klassisch	0.1	(6.2-3), $\beta_0 = 2$, $\varepsilon = 2.5$
D	exakt	modifiziert	10	(6.2-3), $\beta_0 = 10$, $\varepsilon = 1$
E	konventionell	modifiziert	1	(6.2-2), $\beta = \beta_0 = 20$
F	konventionell	modifiziert	10	(6.2-2), $\beta = \beta_0 = 100$

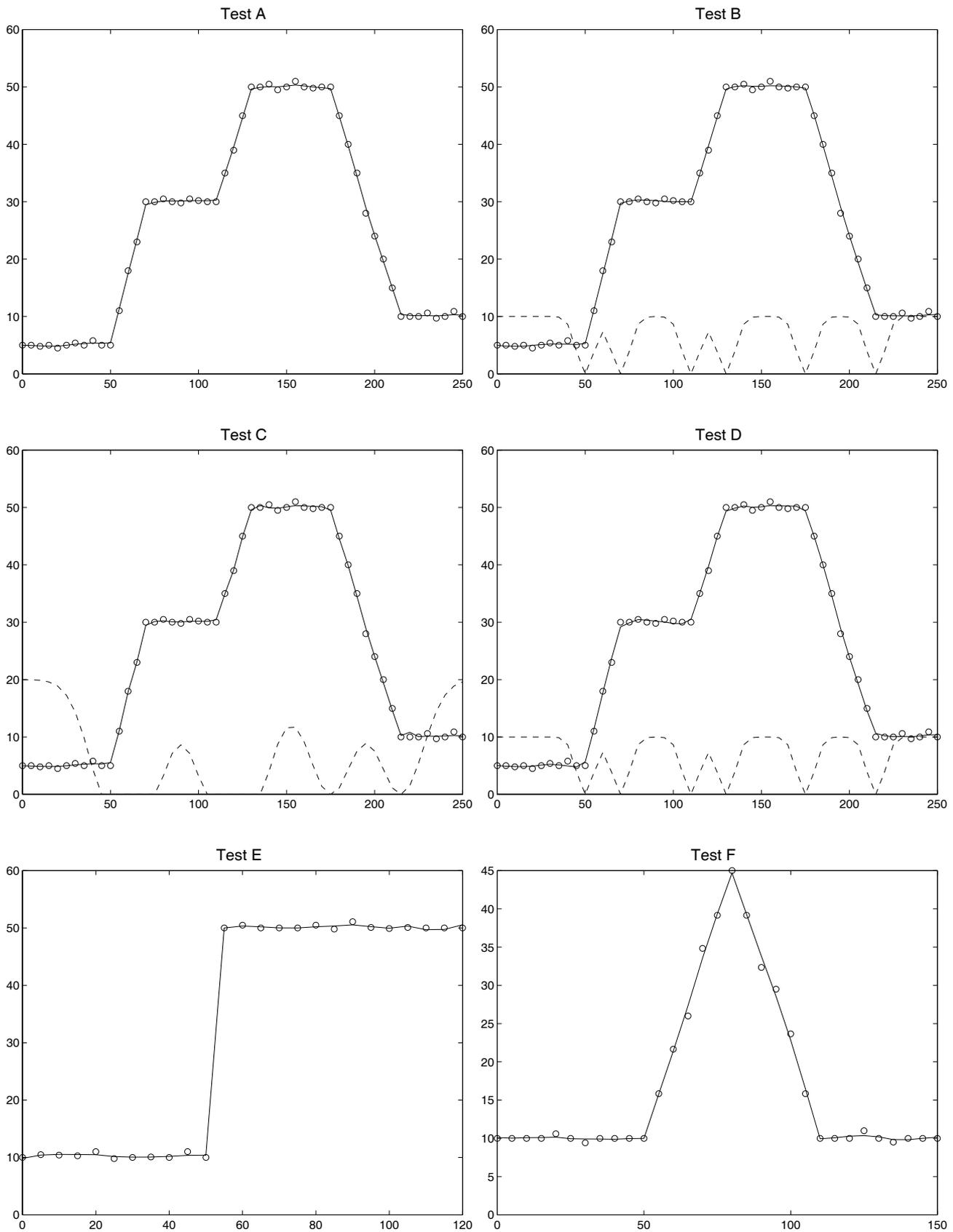


Abbildung 6.2-1: Snakes-Approximation einer Böschungsfolge, einer Sprung- und einer Dreieckfunktion mit Nachbildung der Diskontinuitäten. Einzelheiten in der Tabelle 6.2-1 sowie im Text

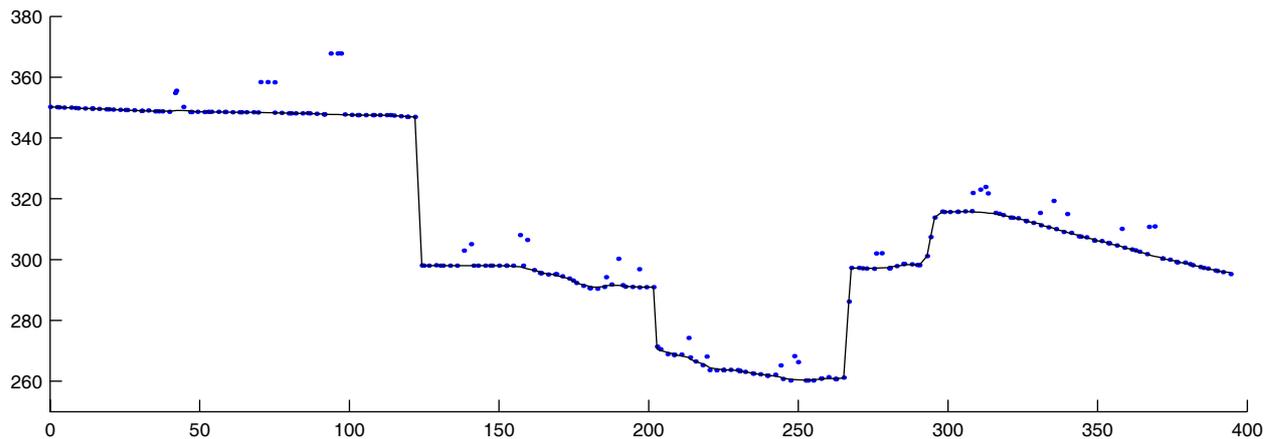


Abbildung 6.2-2: Robuste Snakes-Approximation mit Erhaltung von Diskontinuitäten

- Am einfachsten lassen sich Knicke als *point break* modellieren. Will man indessen auch im Kantenbereich Messfehler glätten, so muss $\alpha > 0$ sein. Dann ist eine geeignete Kombination der Parameter α und β , insbesondere ihr gegenseitiges Verhältnis zu finden, was nur interaktiv geschehen kann und einer gewissen Erfahrung bedarf. Mögliche Defekte lassen sich weitgehend reduzieren, wenn man die modifizierte Iterationsvorschrift verwendet. Dieses Verfahren braucht ein geeignetes Abbruchkriterium, sonst läuft die Schlange nach unendlich vielen Iterationen an die Messdaten heran.
- Jeder Datensatz soll individuell betrachtet (modelliert) werden. In den meisten Fällen wird jedoch der klassische Snakes-Algorithmus, kombiniert mit der externen Energie (6.1-2) und dem Ansatz (6.2-2) bzw. (6.2-3) für β , ggf. auch für α , ein effizientes Mittel für die robuste Profilapproximation mit Knicken sein. Die exakte Lösung der Euler-Gleichungen mit variabler Steuerung, obwohl theoretisch korrekt, ist in der praktischen Anwendung schwieriger handhabbar. Die Ursachen sind im Abschnitt 4.6 diskutiert. Die konventionelle Iterationsvorschrift hat im Vergleich zur modifizierten den Vorteil, dass man hier kein Abbruchkriterium und in der Regel weniger Iterationen braucht. Die modifizierte Iterationsvorschrift lässt hingegen eine größere Steifigkeit der Schlange zu.
- Die erwähnte Kombination der Ansätze ermöglicht auch die Kantenmodellierung, die Messfehlerglättung und die Grobfehlerbeseitigung in *einem* Iterationsprozess, wenn nur genügend "Bodeninformation" zur Verfügung steht und wenn die Ausreißer nicht mit Kanteninformation überlagert sind. Die Snakes-Approximation kann in diesem Fall als instationäre robuste Filterung bezeichnet werden.

7 Experimente zur Filterung von Laserscanner-Daten

Im Abschnitt 3.2 haben wir einen Überblick über robuste Verfahren, insbesondere solche zur Elimination von Grobfehlern in Laserscannerdaten gegeben. Es gibt derzeit zumindest einige ausgereifte und effiziente Verfahren. Anhand der einschlägigen Literatur kann jedoch festgestellt werden, dass es in Gebieten, wo Ausreißer clusterförmig auf größeren Bereichen, wie Reflexionen an Dächern auftreten, immer noch Schwierigkeiten gibt. Ein Ausweg ist in solchen Gebieten, das Datenpyramidenkonzept zu verwenden. Wir stellen dazu eine Alternative, ein auf TPS (vgl. Abschnitt 3.1.5) basierendes Verfahren vor. Ferner versuchen wir, dieses Problem mit Flakes, dem aktiven Modell der deformierbaren Oberflächen, zu lösen. Der Ausgangspunkt kann hierzu die im Abschnitt 6.1 vorgestellte Approximation von (Höhen-)Profilen mit Snakes sein.

7.1 Hilfsalgorithmen für lineare Gleichungssysteme

7.1.1 Iterative Erweiterung des Gleichungssystems

Da einerseits die Modellierungsverfahren in dieser Arbeit oft iterativ eingesetzt werden und andererseits die Datensätze sehr groß sind, ist es besonders sinnvoll, zur Lösung von linearen Gleichungssystemen solche Algorithmen zu verwenden, die den Rechenaufwand reduzieren. Vorausgesetzt sei bereits die Lösung eines Gleichungssystems,

$$\mathbf{x}_0 = \mathbf{A}^{-1}\mathbf{e}. \quad (7.1-1)$$

Nun soll dieses Gleichungssystem um zusätzliche Gleichungen und Unbekannte erweitert werden, z.B. sollen zu einer bereits bestimmten Approximationsfunktion neue Stützstellen hinzukommen. Statt das neue Gleichungssystem zu lösen, können die unbekannt Parameter aufgrund der bestehenden Lösung (7.1-1) ermittelt werden. Dazu schreiben wir das erweiterte Gleichungssystem in Blockmatrixform auf:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{f} \end{bmatrix}. \quad (7.1-2)$$

Daraus ergeben sich die neuen Parameter zu

$$\mathbf{y} = (\mathbf{D} - \mathbf{CA}^{-1}\mathbf{B})^{-1}(\mathbf{f} - \mathbf{Cx}_0). \quad (7.1-3)$$

Die bestehenden Parameter sind auf folgender Weise zu modifizieren:

$$\mathbf{x} = \mathbf{x}_0 - \mathbf{A}^{-1}\mathbf{B}\mathbf{y}. \quad (7.1-4)$$

Soll das Gleichungssystem in einem weiteren Iterationsschritt noch einmal erweitert werden, so ist auch die Inverse der Systemmatrix zu modifizieren. Bezeichnen wir die Blockmatrix in (7.1-2) mit \mathbf{E} , so ist die Inverse

$$\mathbf{E}^{-1} = \begin{bmatrix} \mathbf{A}^{-1}(\mathbf{I} + \mathbf{BF}^{-1}\mathbf{CA}^{-1}) & -\mathbf{A}^{-1}\mathbf{BF}^{-1} \\ -\mathbf{F}^{-1}\mathbf{CA}^{-1} & \mathbf{F}^{-1} \end{bmatrix} \quad (7.1-5)$$

mit

$$\mathbf{F} = \mathbf{D} - \mathbf{CA}^{-1}\mathbf{B}.$$

Die Beziehungen (7.1-3) und (7.1-4) nehmen besonders einfache Form an, wenn die bestehende Lösung nur um eine Gleichung erweitert werden soll. Es ergibt sich

$$y = \frac{f - \mathbf{c} \cdot \mathbf{x}_0}{d - \mathbf{c} \cdot \mathbf{A}^{-1}\mathbf{b}} \quad (7.1-6)$$

und

$$\mathbf{x} = \mathbf{x}_0 - y\mathbf{A}^{-1}\mathbf{b}, \quad (7.1-7)$$

wobei (\cdot) das Skalarprodukt bezeichnet. Entsprechend einfacher wird auch die Beziehung (7.1-5).

7.1.2 Formel von Sherman-Morrison

Soll in der Matrix \mathbf{A} , zu der wir \mathbf{A}^{-1} bereits haben, eine kleine Änderung vorgenommen werden, soll z.B. ein Element a_{ij} geändert werden, so kann dieses mit der Formel von *Sherman-Morrison* realisiert werden, ohne die Inverse neu zu berechnen. Lässt sich die Korrektur als *Kroneckerprodukt* (\otimes) zweier Vektoren auffassen, so gilt (PRESS et al., 2001)

$$(\mathbf{A} + \mathbf{u} \otimes \mathbf{v})^{-1} = \mathbf{A}^{-1} - \frac{(\mathbf{A}^{-1}\mathbf{u}) \otimes (\mathbf{v} \cdot \mathbf{A}^{-1})}{1 + \lambda} \quad (7.1-8)$$

wobei

$$\lambda := \mathbf{v} \cdot \mathbf{A}^{-1}\mathbf{u}. \quad (7.1-9)$$

Zur Lösung von linearen Gleichungssystemen kann die *Sherman-Morrison-Formel* indirekt in folgender Weise angewendet werden: Vorausgesetzt lässt sich das zu lösende Gleichungssystem auffassen zu

$$(\mathbf{A} + \mathbf{u} \otimes \mathbf{v})\mathbf{x} = \mathbf{b}. \quad (7.1-10)$$

Dann löst man zwei Ersatzgleichungssysteme,

$$\mathbf{A}\mathbf{y} = \mathbf{b}, \quad \mathbf{A}\mathbf{z} = \mathbf{u}, \quad (7.1-11)$$

bezüglich \mathbf{y} und \mathbf{z} auf. Mit diesen Vektoren lassen sich nun die unbekannt Parameter ermitteln:

$$\mathbf{x} = \mathbf{y} - \left[\frac{\mathbf{v} \cdot \mathbf{u}}{1 + (\mathbf{v} \cdot \mathbf{z})} \right] \mathbf{z}. \quad (7.1-12)$$

7.1.3 Formel von Woodbury

Wenn man mehr als eine Korrektur zu der Matrix \mathbf{A} addieren will, so kann die verallgemeinerte Version der *Sherman-Morrison-Formel*, die *Woodbury-Formel* (PRESS et al., 2001),

$$(\mathbf{A} + \mathbf{U}\mathbf{V}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}^T\mathbf{A}^{-1} \quad (7.1-13)$$

verwendet werden. Auf einen ersten Blick bringt diese Formel nichts, denn in den Klammern auf der rechten Seite von (7.1-13) hat man wieder eine Matrix zu invertieren. Geht man jedoch davon aus, dass die Matrizen \mathbf{U} und \mathbf{V} $n \times m$ groß sind, während \mathbf{A} $n \times n$ groß ist und dabei gilt $m < n$, meistens $m \ll n$, so ist die zu invertierende Matrix dem Umfang nach klein gegenüber \mathbf{A} .

7.2 Unterteilungsalgorithmus für große Datensätze

In der einschlägigen, insbesondere mathematisch orientierten Literatur, werden immer wieder sehr günstige Approximationseigenschaften von TPS hervorgehoben. Diesem folgt aber nicht eine breitere Anwendung dieser Approximationsmethode in den Geowissenschaften, vor allem aus numerischen Gründen. Bei großen Datensätzen, insbesondere bei stark irregulär in der Ebene verteilten Daten, kann das lineare Gleichungssystem instabil sein; darüber hinaus ist die Größe des Gleichungssystems gleich der Anzahl der Stützstellen. Bei großen Datensätzen ist die Aufgabe einfach nicht mehr lösbar. Diese Probleme können umgangen werden, indem man das gesamte Approximationsgebiet in Untergebiete aufteilt, um die Modellierung lokal anzuwenden.

Die Idee der Unterteilung nutzt die Tatsache aus, dass für große Datensätze das Verhalten der Approximationsfunktion einen lokalen Charakter hat, obwohl die Funktion als solche eine globale ist. Die Approximationsfunktion wird auf einem kleinen begrenzten Bereich kaum oder unbedeutend, im numerischen Sinne, beeinflusst durch Daten, die *genügend weit* von diesem Bereich entfernt sind. Die genügend weite Entfernung ist sowohl im Sinne der Distanz als auch im Sinne der Anzahl der Daten, die dazwischen liegen, zu verstehen, denn je mehr es sind, desto kleiner kann die Distanz sein. Das lokale Verhalten von TPS kann besonders bedenkenlos bei Laserscanner-Daten vorausgesetzt werden. Diese Daten sind regellos, aber gleichmäßig und lückenlos in der Ebene verteilt.

Unser Unterteilungsalgorithmus ist durch eine von MITASOVA und MITAS (1993) vorgeschlagene Segmentierungsmethode inspiriert. Wir gehen davon aus, dass die Daten ein reguläres, meistens rechteckiges Gebiet B , ohne große Konkavitäten bedecken. Diese Voraussetzung ist beim flugzeuggetragenen Laserscanner völlig legitim. Das gesamte Gebiet B teilen wir nun in ein reguläres Gitter mit $n \times m$ quadratischen Untergebieten D_{ij} ;

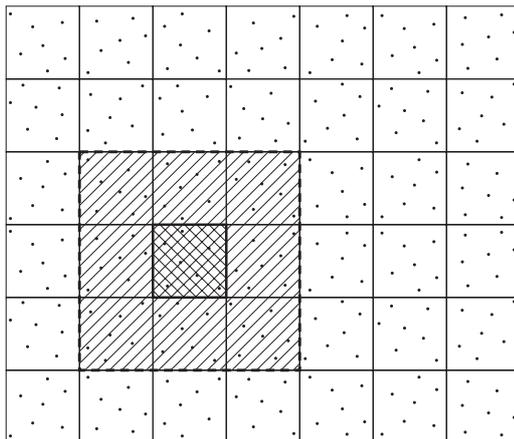


Abbildung 7.2-1: Unterteilung des Approximationsgebietes. Kernbereich und Überlappungsbereiche.

$i = 1, 2, \dots, n; j = 1, 2, \dots, m$ auf, und zwar so, dass die Anzahl der Daten in 3×3 benachbarten Boxen ungefähr einer Maximalzahl k_{max} entspricht:

$$\sum_{i=i-1}^{i=i+1} \sum_{j=j-1}^{j=j+1} N(D_{ij}) \approx k_{max}. \quad (7.2-1)$$

k_{max} ist die Anzahl der linearen Gleichungen, die am verfügbaren Computer noch effizient gelöst werden können. Die Gitterweite a kann iterativ abgestimmt werden, indem die Daten in einzelnen Boxen gezählt und mit (7.2-1) verglichen werden. Bei gleichmäßig regellos in $B \subset \mathbb{R}^2$ verteilten Laserscanner-Daten ist das nicht nötig: Man kann die Eigenschaften von *Poisson-Punktfeldern* ausnutzen. Für ein homogen-isotropes Poisson-Feld gilt (STOYAN und STOYAN, 1992)

$$EN(B) = \lambda A(B). \quad (7.2-2)$$

Die gesamte Anzahl $EN(B)$ der Daten *einer* Realisierung des Poisson-Feldes, und die Fläche $A(B)$ von B sind in der Regel aus der Projektdokumentation bekannt, können auch leicht abgeschätzt werden. λ bezeichnet die Intensität des Feldes oder die mittlere Punktdichte und kann zur Bestimmung der Gitterweite benutzt werden:

$$a^2 = \frac{k_{max}}{9\lambda}. \quad (7.2-3)$$

Weichen die Daten stark von den genannten Voraussetzungen ab, weisen insbesondere eine starke Inhomogenität auf, so ist a iterativ zu bestimmen. Bei solchen Daten ist es auch sinnvoll, einen weiteren Parameter, eine minimale Anzahl der Daten (k_{min}), die zur Modellierung einbezogen werden, einzufügen.

Die Ausführung der Approximation in diesem Unterteilungskonzept ist in der Abbildung 7.2-1 schematisch dargestellt. Die Modellierungsfunktion wird für jede Box D_{ij} getrennt aufgestellt. Um einen glatten Übergang zwischen den einzelnen Segmenten zu sichern, werden zur Bestimmung der Parameter der Modellierungsfunktion in D_{ij} die Daten aus dieser Box *und* aus den 3×3 benachbarten Boxen benutzt.

Das Problem der großen Datensätze tritt bei allen globalen Modellierungsmethoden, wie z. B. Multiquadratische Methode, Kriging, Flakes u.a. auf. Die vorgeschlagene Unterteilungsmethode kann für alle globalen Verfahren ohne Einschränkung angewendet werden.

7.3 TPS basiertes Verfahren

Zur Identifikation von Bodenpunkten in Laserscanner-Datensätzen benutzen wir TPS in einem iterativen Vorgang. Zunächst wird das gesamte Gebiet in Untergebiete aufgeteilt. Dann werden in jedem Untergebiet zwei bis drei Stützstellen mit niedrigsten Höhen aufgesucht. Diese Punkte werden als Bodenpunkte angenommen. Anhand der gewählten Stützstellen aus allen Segmenten des gesamten Gebietes werden die Parameter von TPS bestimmt. Die durch diese Daten bestimmte Oberfläche kann als Trendoberfläche betrachtet werden. Weiter arbeitet das Verfahren iterativ:

1. Es wird ein erstes Untergebiet als Arbeitsgebiet genommen.

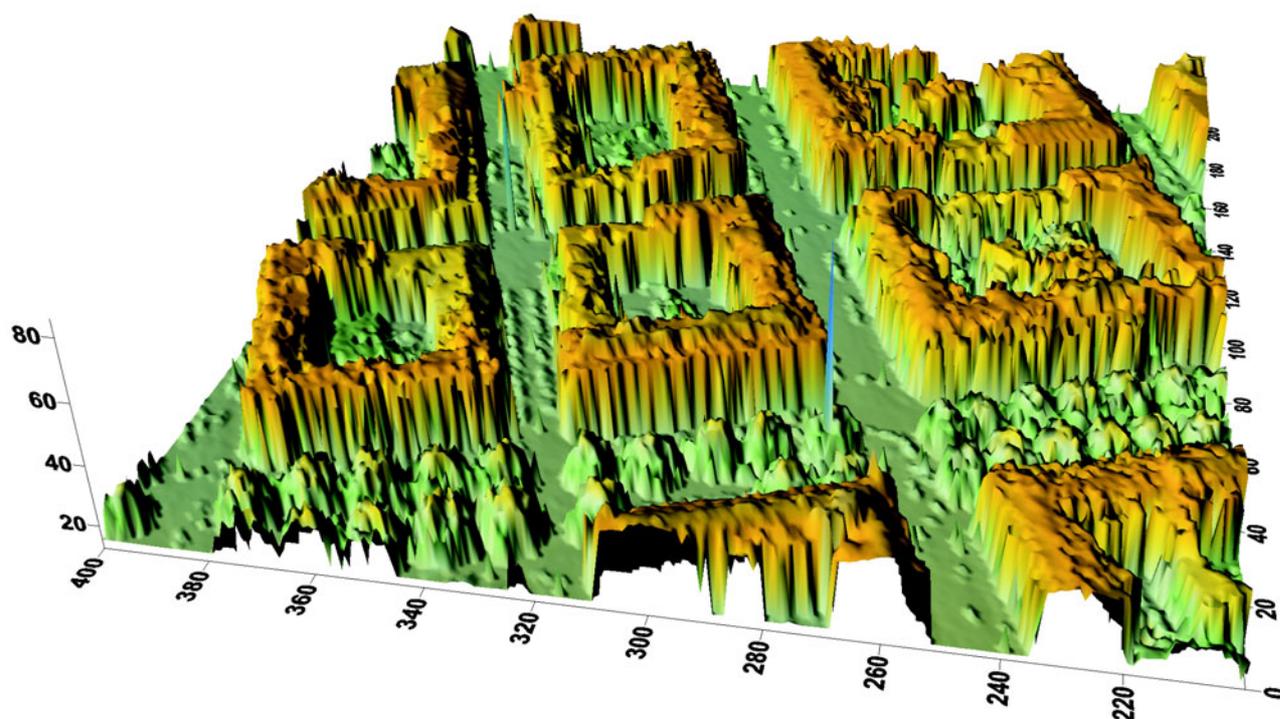


Abbildung 7.3-1: Testgebiet Stockholm (Quelle: OEEPE Working Group on Laser Data Acquisition; http://www.geomatics.kth.se/~fotogram/OEEPE/oeepe_laser_main.htm)

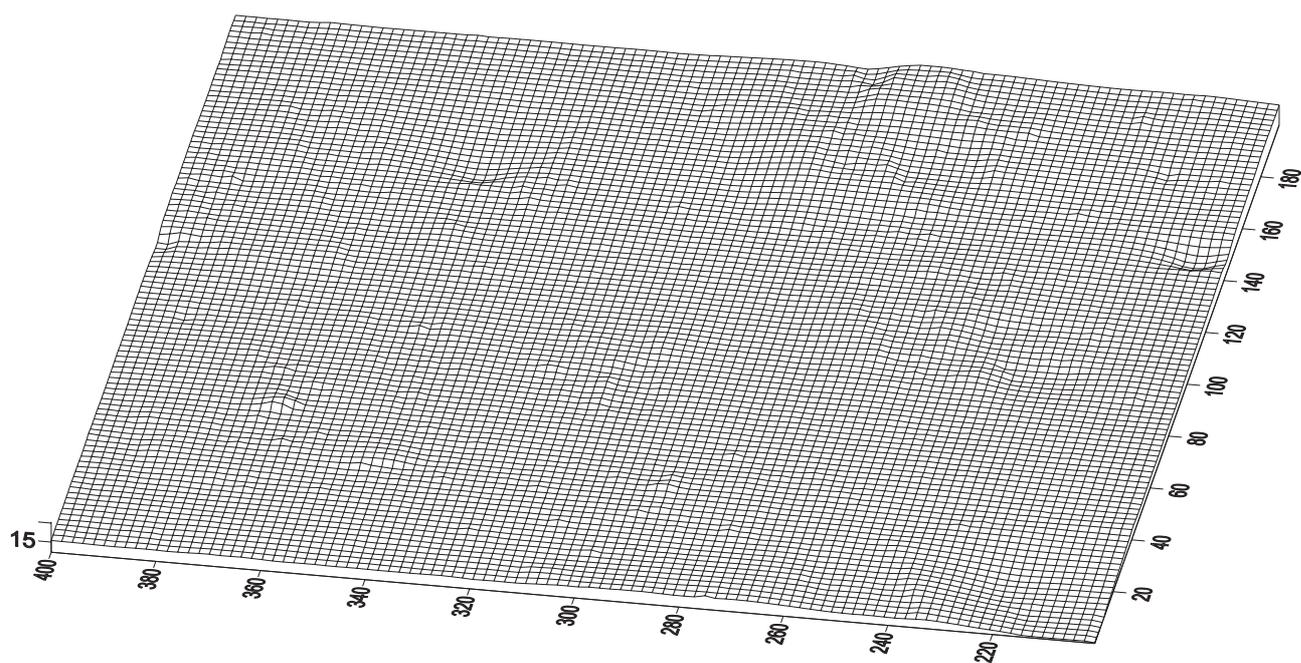


Abbildung 7.3-2: Testgebiet Stockholm: Aus Bodenpunkten interpolierte Oberfläche

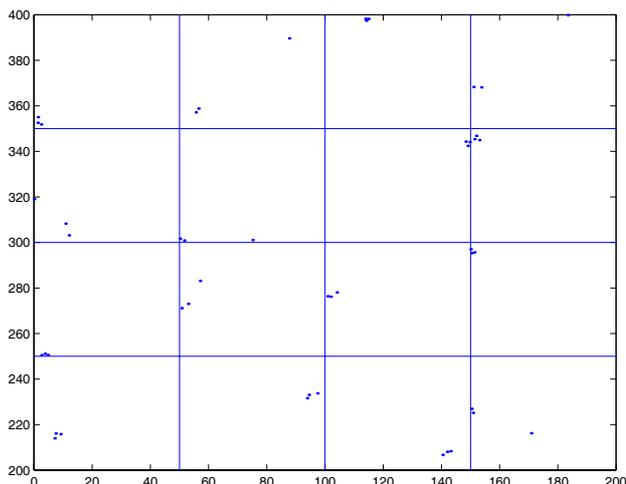


Abbildung 7.3-3: Untergebiete und Startpunkte

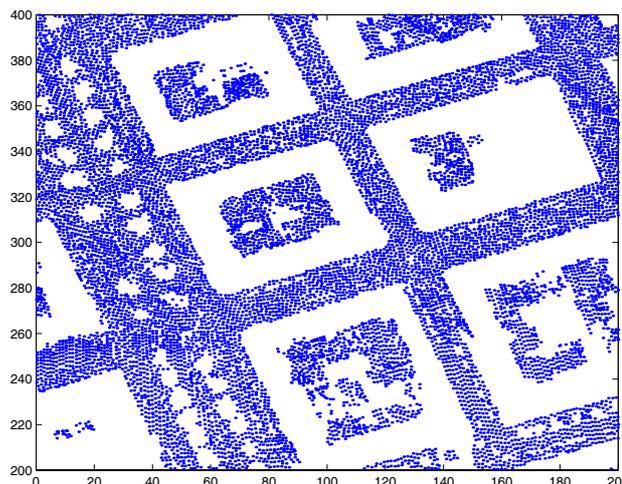


Abbildung 7.3-4: Bodenpunkte

2. Für *alle* Datenpunkte des Arbeitsgebiets, außer jenen, die als Bodenpunkte bereits klassifiziert wurden, werden die Abstände dieser Daten von der aktuellen, mit TPS bestimmten Oberfläche berechnet.
3. Die Punkte, die genügend nahe der Oberfläche liegen, d.h. mit $|\Delta z_i| \leq \Delta z_{max}$, werden als Bodenpunkte angenommen.
4. Anhand *aller* bis jetzt als Bodenpunkte klassifizierten Daten werden die TPS-Parameter neu berechnet.
5. Es wird zum Schritt 2 zurückgegangen. Der Iterationsprozess endet, wenn keine neuen Punkte als Bodenpunkte hinzukommen. Dann wird zur ursprünglichen Trendoberfläche zurückgekehrt, ein weiteres Untergebiet als Arbeitsgebiet genommen und der Iterationsprozess erneut gestartet.

An dieser Stelle wollen wir noch eine Bemerkung zur numerischen Ausführung von TPS machen. In manchen Fällen, insbesondere bei großen Datensätzen, kann die Lösung des Gleichungssystems zur Bestimmung von TPS-Parametern instabil sein, wenn man mit Absolutwerten von Koordinaten, ggf. mit Landeskoordinaten arbeitet. Aus numerischen Gründen ist es also zweckmäßig, die Skalierung

$$\begin{aligned}\tilde{x} &= (x - x_{min})/s \\ \tilde{y} &= (y - y_{min})/s\end{aligned}$$

der Originalkoordinaten x, y zu verwenden, wobei

$$s := \max \begin{cases} (x_{max} - x_{min}) \\ (y_{max} - y_{min}) \end{cases}.$$

Das vorgeschlagene Verfahren wollen wir nun an einem Beispiel demonstrieren. Wir benutzen einen Ausschnitt aus dem Datensatz *Stockholm*, entnommen der Homepage des Projekts "OEEPE Project on Laser Data Acquisition" (European Organisation for Experimental Photogrammetric Research). Das Testbeispiel ist im Bild 7.3-1 gezeigt. Das gesamte Gebiet wurde in 16 Untergebiete mit der maximalen Anzahl der Daten in einer Box von etwa 4800 aufgeteilt. In der Abbildung 7.3-3 sind die Untergebiete und die Stützstellen der Daten, die zur Bestimmung der Trendoberfläche gewählt wurden, veranschaulicht. Dann wurden die Daten iterativ klassifiziert. In diesem Prozess wurden die Daten, die von der momentanen Oberfläche bis zu $\Delta z_{max} = 0,40m$ abweichen, als Bodenpunkte angenommen. Die Verteilung der Bodenpunkte in der Ebene ist im Bild 7.3-4 gezeigt. Es ist zu sehen, dass nicht nur Reflexionen an Gebäuden und Bäumen, sondern auch an Fahrzeugen entfernt wurden.

Das vorgestellte Verfahren klassifiziert die Daten, so wie sie sind; es wird sowohl keine Regularisierung als auch keine Bildung von Datenpyramiden benötigt. Ferner scheint es besonders geeignet für Gebiete zu sein, wo Ausreißer clusterförmig auf großen Bereichen, wie etwa sehr große Gebäudekomplexe oder Bereiche mit einer sehr geringen Durchdringungsrate im bewaldeten Gelände auftreten. Die Interpolationseigenschaften von TPS erlauben, solche Lücken gut auszufüllen. Darüber hinaus kann mit dem *gleichen* Verfahren ein DGM, inklusive Glättung von Messfehlern, interpoliert werden.

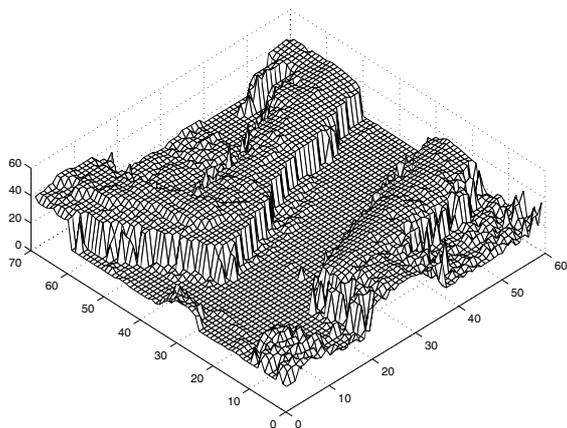


Abbildung 7.4-1: Ausschnitt aus dem Testgebiet Stockholm

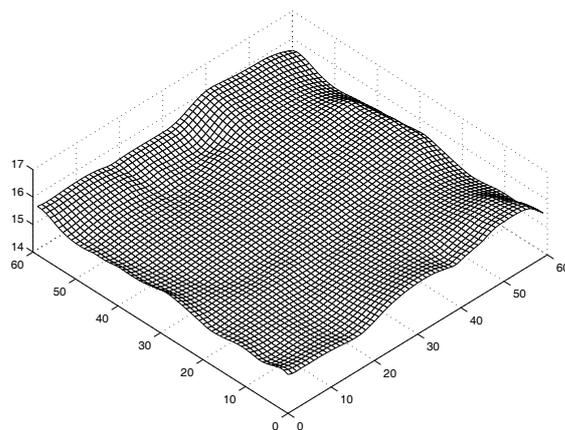


Abbildung 7.4-2: Approximation mit Flakes

7.4 Filterung mit Flakes

Die Filterung von Grobfehlern in Laserscanner-Daten kann auch mit Flakes durchgeführt werden. Dazu beutzen wir das deformierbare Oberflächenmodell für reguläre Daten (vgl. Abschnitt 5.3.1). Die ursprünglichen, irregulären Daten werden zunächst nach dem in Abschnitt 3.1.1 vorgestellten Verfahren regularisiert, wobei die Gitter- bzw. Rasterweite etwa dem mittleren Abstand im Originaldatensatz im \mathbb{R}^2 entsprechend gewählt wird. Ausgehend von einer Ebene, die über den höchsten Punkt im Datensatz gelegt wird, wird das deformierbare Modell an Höhendaten iterativ nach der Vorschrift (6.1-4) angepasst, wobei die Matrix \mathbf{A} mit den Koeffizienten (5.3-31) bis (5.3-33) aufgebaut ist. Die externe Energie wird als Funktion des Abstandes r zwischen den gegebenen Messhöhen z_d und den Höhen z_t , welche die momentane Gestalt von Flakes in dem $(t-1)$ -ten Iterationsschritt beschreiben, nach (6.1-2) definiert und in jedem Datenpunkt z_{ij} ; $i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$ ermittelt. Genauso wie im eindimensionalen Fall sorgt die äußere Energie dafür, dass sich die Approximationsfunktion in Richtung der Datenpunkte mit den niedrigsten Höhen bewegt und im Bereich dieser Punkte verankert wird. Man geht davon aus, dass die Ausreißer nur nach oben auftreten. Ist das nicht der Fall, so ist die symmetrisch gedämpfte Energiefunktion zu verwenden. Diese Energiefunktion kann auch für andere Zwecke angewendet werden. Um z.B. die Waldhöhe zu approximieren, reicht es aus, den linken Ast der Funktion mit dem rechten zu vertauschen. Die innere Energie von Flakes wird durch die Parameter α und β gestaltet. Der Iterationsprozess endet, wie bei der Profilapproximation, wenn das Gleichgewicht zwischen der inneren und der äußeren Energie gefunden ist, d.h. wenn $z_t \approx z_{t-1}$.

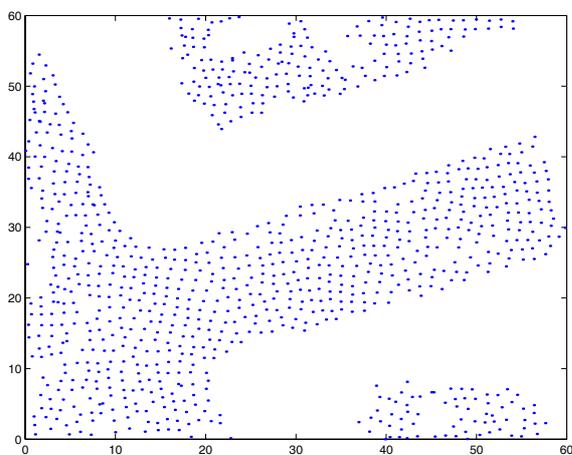


Abbildung 7.4-3: Bodenpunkte zum Testgebiet von Abb. 7.4-1

Den vorgestellten Flakesalgorithmus demonstrieren wir an einem Ausschnitt aus dem Testgebiet Stockholm (Abbildung 7.4-1). Die regularisierten Höhendaten wurden durch Flakes mit den Parametern $\alpha = 1$, $\beta = 1$ und $\sigma^2 = 0.1$ approximiert. Die Approximationsoberfläche ist in der Abbildung 7.4-2 gezeigt. Man bekommt eine glatte, von Grobfehlern freie Oberfläche. Kleine Defekte sind an Rändern des Gebietes zu sehen. Diese lassen sich durch die Randbedingungen für Differentialgleichungen erklären und durch Einfügen zusätzlicher Daten am Rande reduzieren. Ferner können auf Bereichen, wo große Gebäude stehen, mehr oder weniger Dellen auftreten, wenn das deformierbare Modell nicht genügend stark angespannt ist, weil in solchen Bereichen die externe Energie nicht vorhanden ist. Dieses ist kein Nachteil, wenn man das Verfahren zur Grobfehlerfilterung benutzt. Nachdem der Iterationsprozess beendet ist, werden die Flakeshöhen auf dem regulären Gitter an die ursprünglichen Punkte im \mathbb{R}^2 nach dem gleichen Verfahren rückübertra-

ragen. Das Vergleichen der neuen Höhen mit den Originalhöhen erlaubt, die Bodenpunkte aus dem Datensatz zu isolieren. In unserem Beispiel wurden als Bodenpunkte (vgl. Abbildung 7.4-3) jene Daten klassifiziert, deren Höhen sich um nicht mehr als $0,1m$ von den Flakeshöhen unterscheiden.

8 Modellierung von Diskontinuitäten in Laserscanner-Daten

8.1 Bisherige Ansätze

Die Modellierung von Geländekanten mit Hilfe von Laserscanner-Daten ist ein relativ neues und aktuelles Problem. Über längere Zeit wurde die automatische Kantenextraktion aus solchen Daten wegen zu geringer Dichte für nicht erfolgversprechend gehalten. Geländekanten und Formlinien wurden mit photogrammetrischen Verfahren erfasst und in die Interpolation von DGM mit Laserscanner-Daten integriert. Die steigende Auflösung der Laserabtastung in den letzten Jahren ergab die echte Möglichkeit, die Kanten aufgrund solcher Daten zu identifizieren und zu modellieren.

Erste Versuche zur Modellierung von Diskontinuitäten in Höhenprofilen wurden von BORKOWSKI und MEIER (1999) unternommen. Dazu wurde die Snakes-Technik angewendet. In der Arbeit von BRÜGELMANN (2000) wurden zur Kantendetektion die Bildverarbeitungsmethoden angewendet. Aus der ursprünglichen irregulären Punktwolke wird ein reguläres Gitter interpoliert. Nun reicht es, aus den Höhenwerten die Graustufen zu bilden, und dann kann man die wohl bekannten Verfahren zur Kantenextraktion aus digitalen Bildern anwenden. BRÜGELMANN hat Laplacesche und LoG (Laplacian of Gaussian) Operatoren an die Daten mit einer Dichte von ca. 7 Punkte pro m^2 angewendet. Die erzielte Lagegenauigkeit der identifizierten Kanten entlang von Staudämmen war vergleichbar mit der Genauigkeit, welche bei der Messung dieser Linien in photogrammetrischen Bildern im Maßstab 1:4000 erzielt wurde. Die beschriebene Vorgehensweise ist auch von SUI (2002, 2003) angewendet worden. Es wurden lediglich andere Extraktionsalgorithmen aus der breiten Palette der digitalen Bildverarbeitungsmethoden benutzt.

Die mit den Methoden der digitalen Bildverarbeitung abgeleiteten Geländekanten werden in einem am Institut für Photogrammetrie und Fernerkundung der TU Wien (KRAUS und PFEIFER, 2001; BRIESE et al., 2002a; BRIESE und KRAUS, 2003) entwickelten Verfahren als Näherung für die *dreidimensionale* Kantenmodellierung benutzt. In diesem Ansatz werden Originaldaten, welche vorher von den Laserpunkten auf Bäumen, Gebäuden etc. (vorläufig) befreit wurden, in die Modellierung einbezogen. Die vorläufig im \mathbb{R}^2 definierte Kante K teilt die Originaldaten in zwei Klassen. Eine Klasse bilden die Punkte, die links von K , die andere Klasse bilden die Punkte, die rechts von K liegen. Zusätzlich werden die Daten aus den beiden Klassen in überlappende Patches entlang der Kante einsortiert (vgl. Abbildung 8.1-1). In jedem Patch werden die Daten links und rechts von K mit Ebenen E , $E \subset \mathbb{R}^3$ approximiert. Die Schnittgerade der beiden Ebenen ist eine Tangente an die gesuchte dreidimensionale Geländekante. Die sechs Parameter des Ebenenpaares werden nach der Methode der kleinsten Quadrate bestimmt. Die Approximation wird fortlaufend in jedem Patch durchgeführt. Die gewonnenen Schnittgeraden dienen in einem weiteren Rechengang zur erneuten Klassifizierung der Punkte in die Klassen "links" und "rechts" der Schnittgeraden. Bei Bedarf werden weitere Iterationen durchgeführt. Die Schnittgerade ist zwar eine Tangente an die gesuchte Kante, der Berührungspunkt ist aber unbekannt. Es wird daher ein repräsentativer 3D-Punkt in der Mitte des Patches berechnet. Die repräsentativen Punkte werden dann mit einer Spline-Interpolation zur dreidimensionalen Geländekante verdichtet. Durch Einführung von individuellen Gewichten erfolgt die Einpassung der Ebenenpaare durchaus robust. Ferner existieren in vielen Fällen photogrammetrisch ausgewertete Geländekanten, die im Grundriss genauer als die Laserscanner-Informationen aber in der Höhe wesentlich ungenauer als das Laserscanning sind. Diese Tatsache wird auch durch entsprechende Bewichtung berücksichtigt.

8.2 Approximation mit Flakes

8.2.1 Testbeispiele

Eine stationäre Approximation lässt sich mit Flakes einfach nach dem gleichen Algorithmus, der zur Grobfehlerfilterung im Abschnitt 7.4 benutzt wurde, durchführen. Man braucht lediglich den asymmetrischen Ansatz für die externe Energie (6.1-2) durch den symmetrischen (6.2-4) zu ersetzen. Eine solche Approximation würde auch (Gelände-)Kanten gleichmäßig abrunden. Um sie zu erhalten, muss eine variable Steuerung herangezogen werden.

Die Flakes-Approximation mit Erhaltung der Kanten demonstrieren wir zunächst an einem synthetischen Beispiel, das in der Abbildung 8.2-1-A gezeigt ist. Eine Geländekante ist hier mit Rauschen überlagert. Die stationäre Approximation wurde mit den Parametern $\alpha = \beta = 0.2$ durchgeführt und ist im Bild B gezeigt. Die Messfehler wurden geglättet. Dabei wurde auch die Kante verschliffen. Ihr scharfer Verlauf ist nicht mehr erkennbar. Im Bild C ist das Ergebnis der Approximation mit den gleichen Parametern gezeigt; entlang der Kante

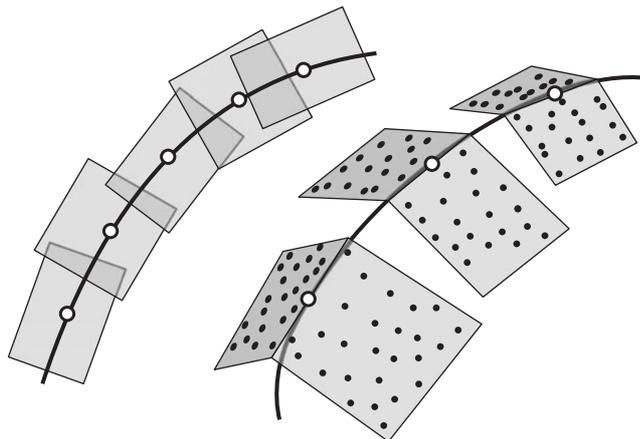


Abbildung 8.1-1: Zu modellierende Geländekante mit Patches (links) und approximierende Ebenenpaare mit Schnittgeraden und repräsentativen Punkten im \mathbb{R}^3 (rechts) (nach KRAUS und PFEIFER (2001))

wurden jedoch sowohl α als auch β zu Null gesetzt. Außerhalb des Kantenbereiches ist die Filterung gleich der im Bild B (qualitativ kann man dies durch Vergleichen von Residuen feststellen), entlang der Kante ist der Filterungsprozess ausgeschaltet. Deshalb hat die Kante ihre ursprüngliche scharfe Definition erhalten. Sowohl im Test A als auch im Test B wurde das FEM-Flakes-Modell (Abschnitt 5.3.1) mit den Koeffizienten der Systemmatrix nach (5.3-31) bis (5.3-33) angewendet. Vergleichbare Resultate erzielt man, wenn die Euler-Gleichungen mit finiten Differenzen (Abschnitt 5.2.1) statt mit finiten Elementen diskretisiert sind. Das Bild D zeigt unser nach diesem Ansatz bearbeitetes Testbeispiel, wobei die Steuerparameter unverändert blieben (entlang der Kante $\alpha = \beta = 0$; außerhalb der Kante $\alpha = \beta = 0.2$). Optisch sind keine Unterschiede erkennbar. Das Vergleichen der Residuen zeigt, dass die Unterschiede zwischen den beiden Approximationsvarianten C und D im Bereich unter 10^{-2} liegen. Der Test E wurde mit dem gleichen Verfahren wie D bearbeitet. Die Steuerparameter wurden aber dabei wie folgt modifiziert: außerhalb der Kante $\alpha = \beta = 0.2$, entlang der Kante $\alpha = 0.2$; $\beta = 0$ (tangent break). Hier ist die Kante geringfügig geglättet, jedoch sind kleine Defekte in Form von Aufhebung (sekundäre Knicke) hinter der Kante sichtbar als in anderen Tests. Zu bemerken ist noch, dass Defekte auch am Rande des bearbeiteten Gebietes auftreten. Diese lassen sich durch Randbedingungen in den Differentialgleichungen erklären und sind leicht zu beheben, indem man den zu bearbeitenden Datensatz an Rändern um zusätzliche Spalten (Zeilen) erweitert bzw. fortsetzt.

Das gleiche Beispiel haben wir auch mit dem für irreguläre Daten konzipierten direkten Ansatz bearbeitet (Bild F). Hier hat man drei Steuerparameter, welche wie folgt ausgelegt wurden: $\alpha = 0.01$, $\beta = 0.2$ und σ etwa gleich dem Abstand zwischen benachbarten Datenpunkten. Entlang der Kante wurde jedoch $\alpha = \beta = 0$ und $\sigma^2 \rightarrow 0$ gesetzt. Das Vergleichen der Bilder C, D und F zeigt, dass bis auf feine Einzelheiten alle drei Ansätze vergleichbare Approximationsresultate liefern.

Bei dem letztgenannten Verfahren kann das Gleichungssystem (5.3-54) direkt aufgelöst werden, ohne iterieren zu müssen. Die Systemmatrix $\mathbf{A}_{\alpha,\beta}$ kann in manchen Fällen schwach konditioniert und die Lösung kann instabil sein. Außerdem kann der iterative Vorgang wegen einer interaktiven Kontrolle der Approximationsqualität bevorzugt werden. Die Bedingungsgleichung (5.3-54) kann auch iterativ aufgelöst werden. Durch Analogie zu Snakes platzieren wir wieder das deformierbare Flakes-Modell in einem viskosen Medium und lassen es sich bewegen. Die entsprechende Bewegungsgleichung lautet

$$\gamma \dot{\mathbf{c}} + (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^\top \mathbf{L}) \mathbf{c}(t) - \mathbf{L}^\top \mathbf{w} = 0. \quad (8.2-1)$$

Dann diskretisieren wir diese Differentialgleichung mit Rückwärtsdifferenzen,

$$\gamma (\mathbf{c}_t - \mathbf{c}_{t-1}) + (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^\top \mathbf{L}) \mathbf{c}_t = \mathbf{L}^\top \mathbf{w}, \quad (8.2-2)$$

und erhalten nach Umformen die Iterationsvorschrift

$$\mathbf{c}_t = (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^\top \mathbf{L} + \gamma \mathbf{I})^{-1} (\gamma \mathbf{c}_{t-1} + \mathbf{L}^\top \mathbf{w}) \quad (8.2-3)$$

für die Bestimmung der Parameter \mathbf{c} . Man kann auch gleich die "Flakeshöhen" bestimmen. Setzt man $\alpha = \beta = 0$ in die Gleichung (5.3-54) ein (reines Interpolationsproblem), so ist ersichtlich, dass der Operator \mathbf{L} die Parameter

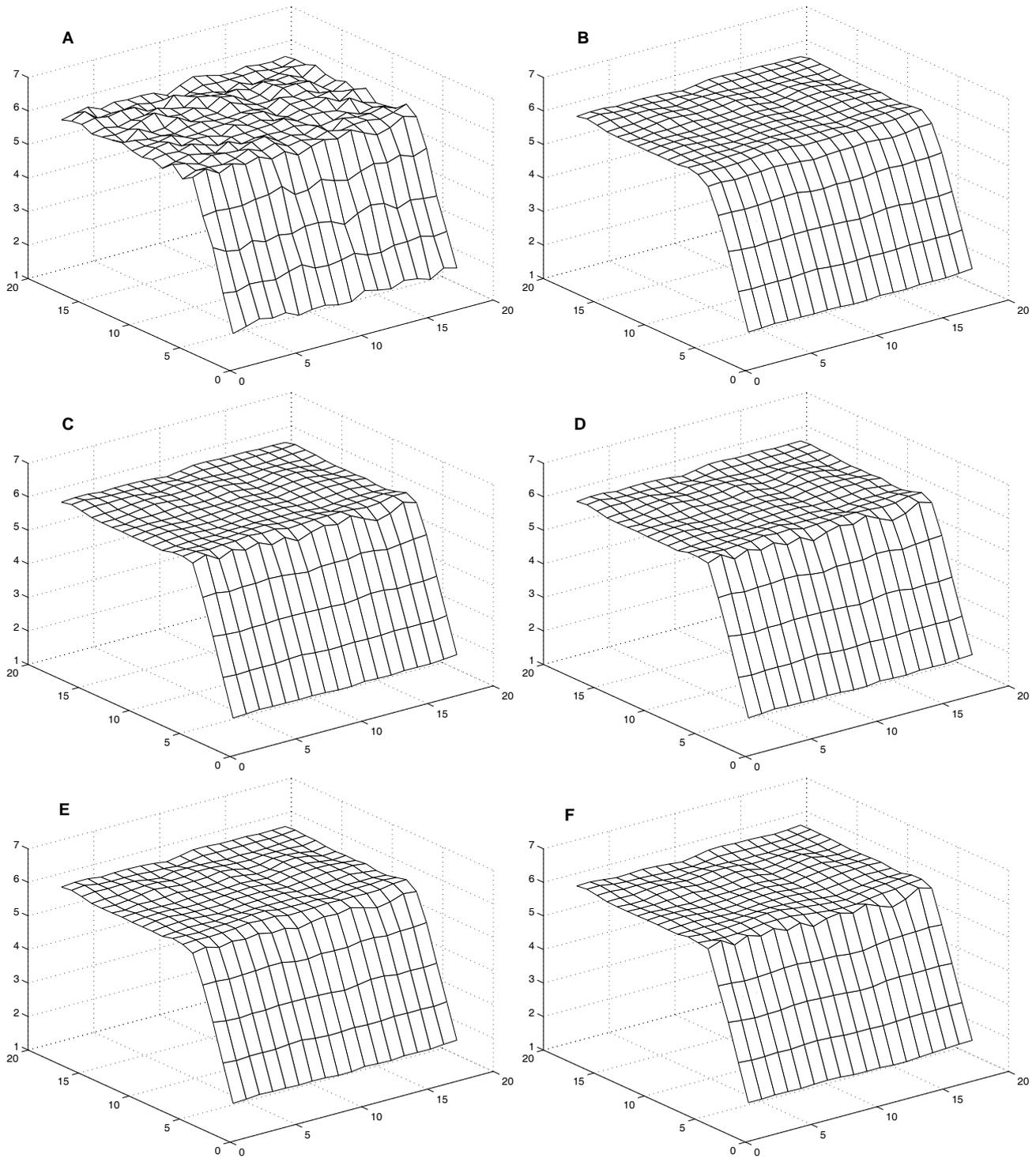


Abbildung 8.2-1: Synthetisches Testbeispiel: Daten mit überlagertem Rauschen (A), stationäre Approximation mit Flakes (B) und unterschiedliche Varianten der instationären Flakes-Approximation mit Erhaltung der Kante (C), (D), (E) und (F); Einzelheiten im Text

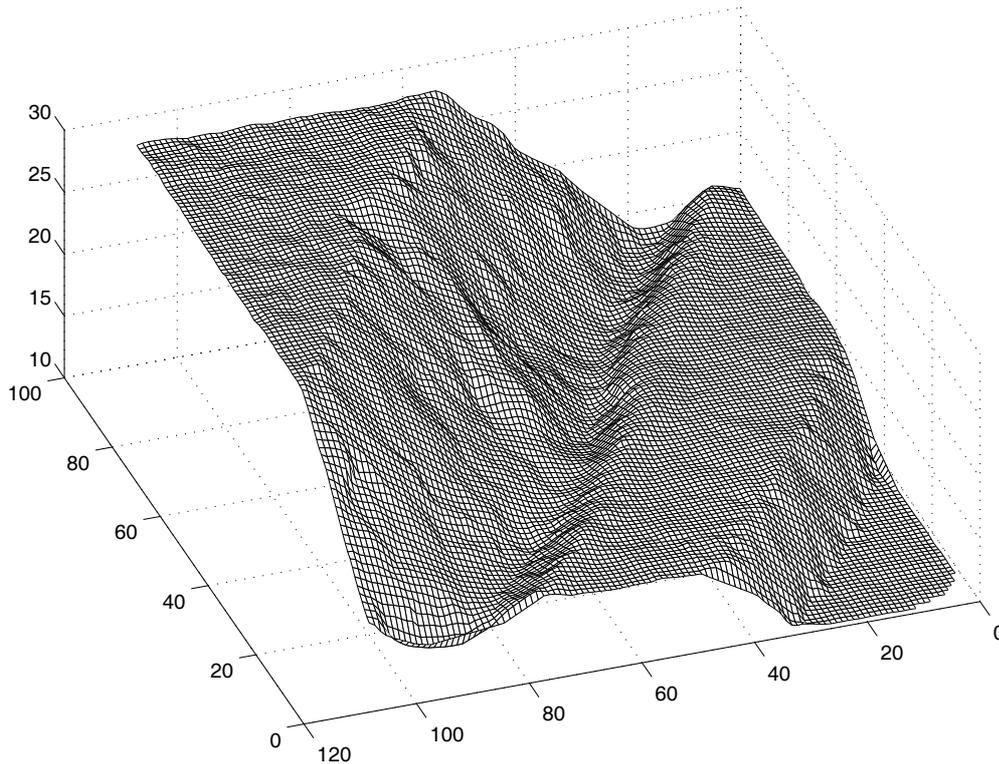


Abbildung 8.2-2: Testbeispiel Tagebau

\mathbf{c} in die Datenkoordinaten überträgt: $\mathbf{Lc} = \mathbf{w}$. Multipliziert man nun die Gleichung (8.2-3) mit \mathbf{L} , so ergeben sich die momentanen "Flakeshöhen" \mathbf{z}_t zu

$$\mathbf{z}_t = (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^\top \mathbf{L} + \gamma \mathbf{I})^{-1} (\gamma \mathbf{z}_{t-1} + \mathbf{L} \mathbf{L}^\top \mathbf{w}). \quad (8.2-4)$$

Für kleine σ , etwa $\sigma \ll r$; $r^2 = (x - x_i)^2 + (y - y_i)^2$, insbesondere für $\sigma^2 \rightarrow 0$, kann man $\mathbf{L} \approx \mathbf{I}$ setzen. Dann bekommt man die Näherungsvorschrift

$$\mathbf{z}_t = (\mathbf{A}_{\alpha,\beta} + 2\gamma \mathbf{I})^{-1} (\gamma \mathbf{z}_{t-1} + \mathbf{w}). \quad (8.2-5)$$

Das Verfahren für irreguläre Daten demonstrieren wir noch an einem der Arbeit von BORKOWSKI und KELLER (2003) entnommenem Testbeispiel (Abb. 8.2-2), welches einen Ausschnitt aus einem Laserscanner-Datensatz eines Tagebaues darstellt. Die Oberflächenpunkte $\{x, y, z\}$ sind irregulär in der Ebene verteilt (vgl. Abb. 8.2-3). Um die Geländekanten zwecks Modellierung zu markieren, wurde mit Hilfe der Bildverarbeitungsmethoden die vorläufige xy -Lage der Kanten gefunden. Aus den Daten wurde die Delaunay-Triangulation gebildet und die Triangulationsseiten, welche die vorläufigen Kanten schneiden, wurden als zu Kanten gehörende markiert (Abb. 8.2-3). Die Daten wurde nun sowohl mit Kantenerhaltung als auch ohne modelliert. Die Effekte der Modellierung wollen wir anhand der in den Abbildungen 8.2-4 und 8.2-5 veranschaulichten Residuen betrachten. Die stationäre Modellierung wurde in beiden Fällen mit gleichen Steuerparametern $\alpha = \beta = 1/50$ und $\sigma_i^2 = 10$ über alle Datenpunkte durchgeführt. Um die Glättung entlang der Diskontinuität minimal zu halten, wurde wieder α und β in den Punkten des Kantenbereiches Null gesetzt. Zusätzlich bekam auch der Parameter σ_i^2 in diesen Punkten sehr kleine Werte: $\sigma_i^2 = 0.1$ (Abb. 8.2-5). In der Abbildung 8.2-5 sind deutliche Streifen erkennbar, deren Lage der Kantenlage entspricht und wo die Glättung nur minimal aufgetreten ist und die Residuen sich nicht signifikant von Null unterscheiden, während an sonstigen Datenpunkten starke Glättung vorgenommen wurde. Wurden keine Maßnahmen zur Kantenerhaltung vorgenommen, so ist die Glättung etwa gleichmäßig (Abb. 8.2-4). Als Maß für die Qualität der Approximation kann die Standardabweichung σ_r der Residuen benutzt werden. Im letztgenannten Fall ist σ_r sowohl im Kantenbereich als auch im sonstigen Bereich etwa gleich und beträgt $\sigma_r = 0,11m$. Im Falle der Kantenerhaltung ist σ_r um Null im Kantenbereich, worauf ca. 400 Punkte entfallen, und $\sigma_r = 0,12m$ im sonstigen Bereich, worauf ca. 1500 Punkte entfallen.

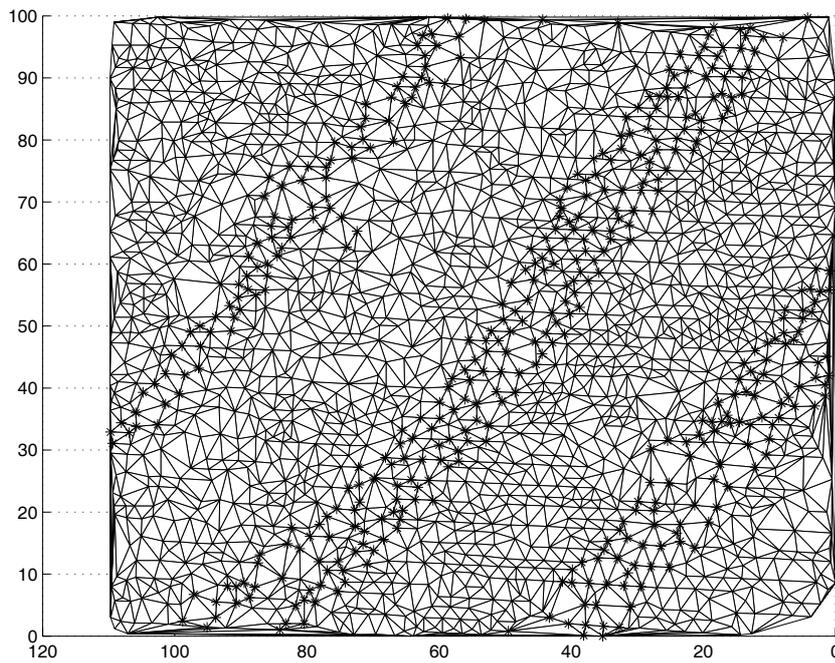


Abbildung 8.2-3: Delaunay-Triangulation der Daten mit markierten Kantenpunkten (*)

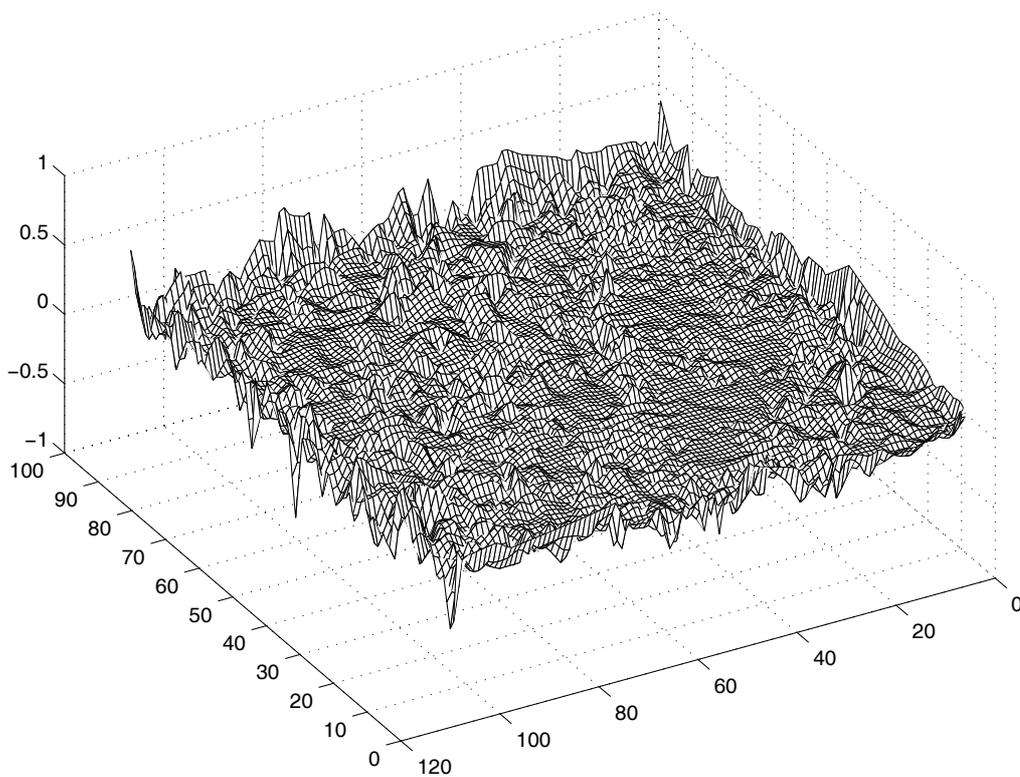


Abbildung 8.2-4: Residuen nach der Filterung ohne Kantenerhaltung

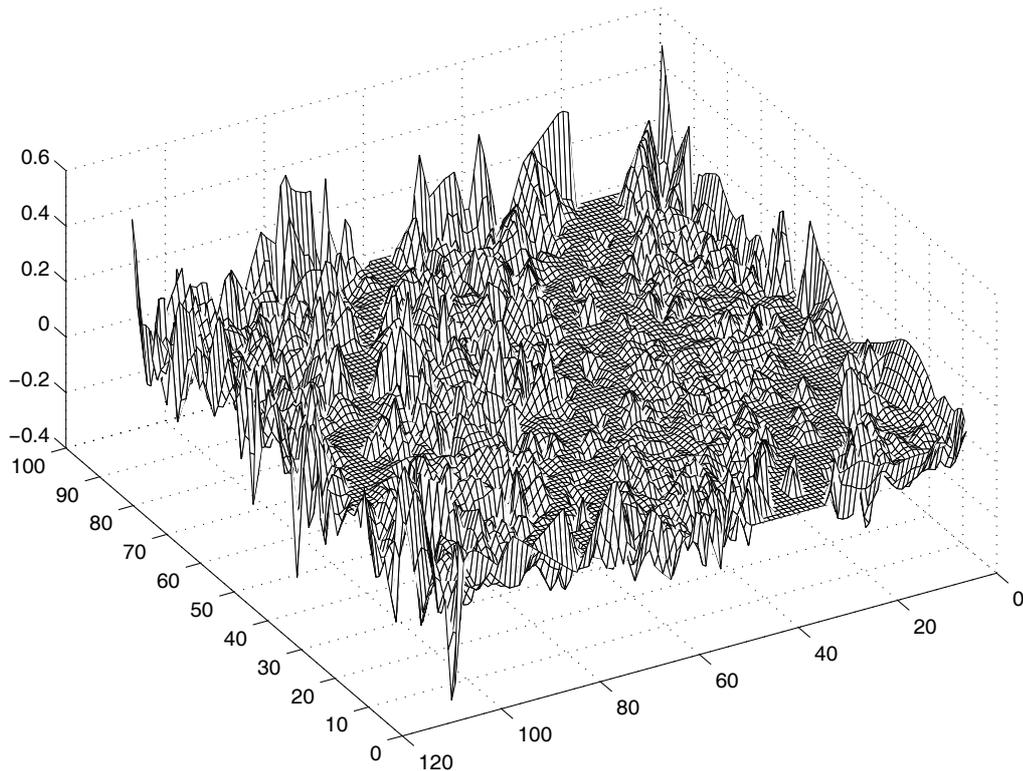


Abbildung 8.2-5: Residuen nach der Filterung mit Kantenerhaltung

Nun wollen wir unsere Ergebnisse und Erfahrung zur Modellierung von Oberflächen mit Flakes zusammenfassen. Prinzipiell können alle Aussagen zur Profilmodellierung sinngemäß mit entsprechender Schwierigkeitssteigerung auf Oberflächen erweitert werden. Einige Bemerkungen wollen wir jedoch zusätzlich formulieren:

- In allen Flakes-Modellen ist die Anzahl der zu lösenden linearen Gleichungen gleich der Anzahl der Datenpunkte. Bei Laserscanner-Daten erreicht man schnell die Grenzen des verfügbaren Rechners und die Datensätze müssen geteilt werden, um sie überhaupt bearbeiten zu können. Hierzu kann die im Abschnitt 7.2 vorgeschlagene Segmentierungsmethode behilflich sein.
- Der Flakes-Ansatz für reguläre Daten ist wegen der schwach besetzten Bandstruktur der Systemmatrix numerisch günstiger und mit nur zwei Parametern α und β leichter steuerbar, bedarf aber zusätzlicher Vorarbeit, um die ursprünglichen irregulären Daten zu regularisieren. Im Gegensatz zum Ansatz für irreguläre Daten ist mit diesem Verfahren auch eine robuste Approximation möglich.
- Der Flakes-Ansatz für irreguläre Daten lässt eine beliebige Datenstruktur zu. Es wird keine Regularisierung benötigt; man arbeitet mit Originaldaten. Der Preis dafür ist eine vollbesetzte Systemmatrix, welche auch mit größerem numerischem Aufwand aufzustellen ist. Darüberhinaus wird dieses Verfahren über drei Parameter α , β und σ^2 gesteuert.
- Mit beiden Verfahren ist eine qualitätsmäßig vergleichbare Kantenerhaltung während der Approximation möglich. Beide Verfahren zeichnen sich durch eine große Flexibilität aus, was von Vorteil ist. Die Flexibilität der Verfahren ist aber gleichzeitig auch deren Nachteil, denn das Finden der richtigen Kombination der Steuerparameter beim konkreten Datensatz verlangt eine gewisse Erfahrung und ist mit mehreren numerischen Experimenten verbunden.
- Gewisse allgemeine Richtlinien lassen sich jedoch formulieren: Am einfachsten ist es, im Bereich der Kanten $\alpha = \beta = 0$ zu setzen und ggf. auch zusätzlich σ^2 gegen Null tendieren zu lassen, während im sonstigen Bereich die Parameter der gewünschten Approximationsqualität entsprechend zu variieren sind.

8.2.2 Genauigkeit der Approximation

Der Flakes-Ansatz für reguläre Daten kann als lineares Ersatzsystem aufgefasst werden:

$$\mathbf{Az} = -\mathbf{E}_z. \quad (8.2-6)$$

Wird dann die externe Energie proportional zum Datendefekt definiert, etwa $\mathbf{E} = \frac{1}{2}(\mathbf{w} - \mathbf{z})^2$, so ist

$$\mathbf{E}_{\mathbf{z}} = \mathbf{z} - \mathbf{w}. \quad (8.2-7)$$

Aus (8.2-6) und (8.2-7) ergibt sich die "naive Lösung" für die "Flakeshöhen" zu

$$\mathbf{z} = (\mathbf{A} + \mathbf{I})^{-1}\mathbf{w}. \quad (8.2-8)$$

Nach der allgemeinen Fehlerfortpflanzung bekommt man die Kovarianzmatrix der "Flakeshöhen"

$$\mathbf{C}_{\mathbf{zz}} = (\mathbf{A} + \mathbf{I})^{-1}\mathbf{C}_{\mathbf{ww}}(\mathbf{A} + \mathbf{I})^{-1}, \quad (8.2-9)$$

weil $(\mathbf{A} + \mathbf{I})$ symmetrisch ist. Für gleichgenaue Messwerte \mathbf{w} ist

$$\mathbf{C}_{\mathbf{ww}} = \text{diag}(\sigma_w^2 \ \sigma_w^2 \ \sigma_w^2 \ \dots \ \sigma_w^2) = \sigma_w^2 \mathbf{I} \quad (8.2-10)$$

und

$$\mathbf{C}_{\mathbf{zz}} = \sigma_w^2 [(\mathbf{A} + \mathbf{I})(\mathbf{A} + \mathbf{I})]^{-1}. \quad (8.2-11)$$

Für $\alpha = \beta = 0$ ist $\mathbf{A} = \mathbf{0}$ und die Genauigkeit der "Flakeshöhen" ist gleich der Genauigkeit der Eingangsdaten: $\mathbf{C}_{\mathbf{zz}} = \sigma_w^2 \mathbf{I}$. Die Beziehungen (8.2-9) bzw. (8.2-11) gelten auch für snakes-approximierte Profile; sie sind auch bereits in (MEIER, 2000b) angegeben. Man braucht lediglich die Matrix \mathbf{A} entsprechend zu modifizieren.

Bei dem Flakes-Ansatz für irreguläre Daten ergeben sich die Parameter \mathbf{c} aus (5.3-54) sofort zu

$$\mathbf{c} = (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{w} \quad (8.2-12)$$

und nach der Fehlerfortpflanzung deren Kovarianzmatrix

$$\mathbf{C}_{\mathbf{cc}} = (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{C}_{\mathbf{ww}} \mathbf{L} (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \quad (8.2-13)$$

wobei $(\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})$ wieder symmetrisch ist. Mit $\mathbf{z} = \mathbf{Lc}$ ergibt sich die Kovarianzmatrix der "Flakeshöhen" zu

$$\mathbf{C}_{\mathbf{zz}} = \mathbf{L} (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{C}_{\mathbf{ww}} \mathbf{L} (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T. \quad (8.2-14)$$

Für gleichgenaue Messwerte mit der Kovarianzmatrix nach (8.2-10) gilt

$$\mathbf{C}_{\mathbf{zz}} = \sigma_w^2 \mathbf{L} (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{L} (\mathbf{A}_{\alpha,\beta} + \mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T. \quad (8.2-15)$$

Man kann sich auch leicht überzeugen, dass für $\alpha = \beta = 0$, d.h. auch $\mathbf{A}_{\alpha,\beta} = \mathbf{0}$, $\mathbf{C}_{\mathbf{zz}} = \sigma_w^2 \mathbf{I}$ gilt.

Mit den angegebenen Beziehungen können auch die Steuerparameter numerisch und iterativ vorabgeschätzt werden. Bei einer vorgegebenen Approximationsgenauigkeit σ_z^2 und gewählten Parametern α und β ist die entsprechende Matrix \mathbf{A} zusammenzustellen und $\mathbf{C}_{\mathbf{zz}}$ auszurechnen. Diese ist mit der vorgegebenen Genauigkeit zu vergleichen, die Parameter sind zu modifizieren und $\mathbf{C}_{\mathbf{zz}}$ ist neu zu bestimmen, ggf. das Ganze zu wiederholen.

MEIER (2000b) hat gezeigt, dass eine Qualitätsbeurteilung von snakes-approximierten Höhenprofilen unter gewissen Einschränkungen über Spektralanalyse möglich ist. Prinzipiell könnte dieser Ansatz auch auf zweidimensionale Signale erweitert werden. Dies wäre jedoch mit großem Aufwand verbunden: Man müsste mit zweidimensionalen verallgemeinerten δ -Funktionen arbeiten. Bereits im eindimensionalen Fall sind jedoch geschlossene Formeln nur für bestimmte Signaltypen möglich.

8.3 Kante als Schnitt zweier Oberflächen

Mit Flakes sind wir in der Lage, Kanten während der Approximation weitgehend zu erhalten oder zu modellieren. Diese deformierbaren Modelle der Oberflächen liefern aber keine direkte Information zur Lage der Kanten im \mathbb{R}^2 bzw. im \mathbb{R}^3 . Räumliche Information über Diskontinuitäten, insbesondere in Form von Vektordaten, ist aber von großer Bedeutung für die praktische Nutzung. Nachfolgend widmen wir uns der Gewinnung solcher Information.

In dem an der TU Wien entwickelten Verfahren wird die Geländekante als Schnittlinie von einem entlang der gesuchten Geländekante gleitenden Ebenenpaar ermittelt. Diese Idee wollen wir aufnehmen und verallgemeinern. Der Ausgangspunkt für unser Verfahren ist der gleiche: Die Originaldaten müssen in die Klassen "links" und "rechts" der gesuchten Kante eingeteilt werden. Dazu werden die Methoden der digitalen Bildverarbeitung

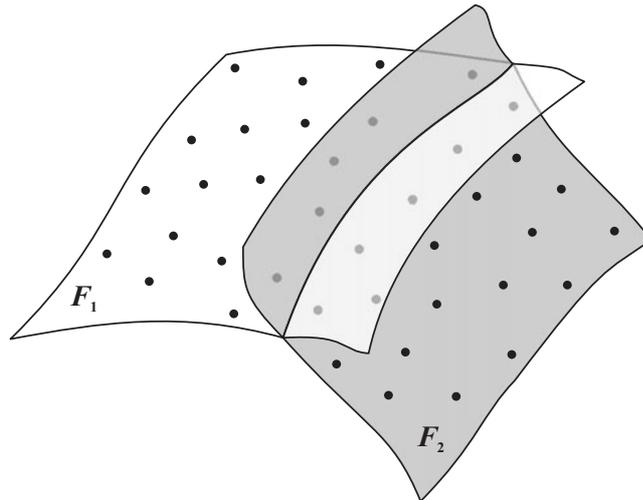


Abbildung 8.3-1: Geländekante als Schnitt zweier Oberflächen

eingesetzt. Mögliche Vorinformation zur Kantenlage soll auch berücksichtigt werden. Die Datenpunkte in den beiden Klassen werden mit Oberflächen in Funktionsform $z = f_i(x, y)$; $i = 1, 2$ approximiert (vgl. Abb. 8.3-1). Die Geländekante ergibt sich als Schnittkurve $[x(s), y(s), z(s)]^T$ der beiden Oberflächen. Die Aufgabe kann auf die Bestimmung der in die xy -Ebene projizierten Schnittkurve zurückgeführt werden. Die "Höhen" der Schnittkurve können durch das Einsetzen der projizierten Kurve in eine von den oberflächenapproximierenden Funktionen,

$$[x(s), y(s), z(s)]^T = [x(s), y(s), f_i(x(s), y(s))]^T, \quad i = 1, 2 \quad (8.3-1)$$

ermittelt werden. Entlang der Verschneidungskurve sind die Funktionswerte z für die beiden Oberflächen bekanntlich gleich. Somit ist die Projektion der Schnittkurve in die xy -Ebene durch

$$F(x(s), y(s)) := f_2(x(s), y(s)) - f_1(x(s), y(s)) = 0. \quad (8.3-2)$$

gegeben. Für $f_i(x, y)$ kann eigentlich jede Approximationsmethode, von den einfachen Polynomialflächen bis zur linearen Prädiktion genommen werden. Man muss nur beachten, dass die Approximationsfunktion auch eine sinnvolle Extrapolation ermöglicht. Wir benutzen die *thin plate splines*.

Die Verschneidung von Oberflächen (ggf. auch Kurven) gehört zu den Standardaufgaben der Computergraphik. Die Fülle der bereits vorhandenen Lösungen und Algorithmen ist kaum noch überschaubar. Gleichzeitig aber sind Programme bzw. Algorithmen, welche alle Anforderungen wie geeignete Genauigkeit, Zuverlässigkeit, Schnelligkeit und Selbständigkeit verbürgen, eher selten. Ein guter Überblick über vorhandene Algorithmen ist bei HOSCHEK und LASSER (1992) zu finden. Eine große Rolle spielen in der Computergraphik algebraische Methoden, welche für einfache Oberflächen explizite Lösungen bereitstellen. Diese sind aber für unsere Aufgabenstellung nicht geeignet. Im einfachsten Fall kann die projizierte Schnittlinie nach der Gittermethode, wie bei der Isolineninterpolation, gefunden werden. Zu diesem Zweck sind die Funktionswerte $F(x(s), y(s))$ auf ein Gitter zu interpolieren und dann kann mit vorhandenen Programmen die "Höhenlinie" Null interpoliert werden.

8.3.1 Einordnung der Daten in die Teiloberflächen

Ein wichtiger Schritt bei der dreidimensionalen Kantenmodellierung als Schnitt zweier Oberflächen ist die Einordnung der Daten in die einzelnen sich schneidenden Teiloberflächen. Dies wird mit Hilfe der digitalen Bildverarbeitungsmethoden realisiert, indem man die ursprünglichen Daten auf ein Quadratgitter interpoliert, dieses in ein Grauwertbild umwandelt und ferner daraus die vorläufige Lage der Kanten ableitet. Für kleinere Datensätze kann die Einordnung der Datenpunkte direkt im folgenden interaktiven und iterativen Vorgang erfolgen:

1. Zunächst muss die Anzahl n der Teiloberflächen T_i vorgegeben werden. In jedem Teilbereich T_i ; $i = 1, 2, \dots, n$ sind drei Datenpunkte zu wählen.
2. Aus den zu T_i gehörenden Punkten wird die konvexe Hülle H_i gebildet und alle Punkte innerhalb von H_i werden T_i zugeordnet.
3. Aufgrund der T_i zugeordneten Daten wird mittels TPS die Approximationsoberfläche O_i berechnet.

4. Zu jedem T_i wird ein neuer Punkt P zugefügt, welcher die Bedingungen

$$\min\{\text{dist}(H_i, P_j)\}, \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, m$$

und

$$\min\{\text{dist}(O_i, P_j)\}, \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, m$$

erfüllt.

5. Es wird zum Punkt 2. zurückgegangen.

Dieses Verfahren kann auch zusammen mit den Bildverarbeitungsmethoden eingesetzt werden. Innerhalb eines schmalen Bereiches entlang der Kante können Punkte nach den letztgenannten Methoden, wegen Regularisierung der Daten, falsch klassifiziert werden. Es ist sinnvoll, die Daten aus dem direkten Kantebereich mit dem vorgestellten Verfahren nachzuarbeiten.

8.3.2 Ein Verfolgungsalgorithmus

Die Bedingung (8.3-2) kann in Form der Differentialgleichung

$$\frac{d}{ds}F(x(s), y(s)) = F_x(x(s), y(s))\frac{dx}{ds} + F_y(x(s), y(s))\frac{dy}{ds} = 0 \quad (8.3-3)$$

aufgefasst werden. Daraus folgt unmittelbar:

$$\frac{dy}{dx} = -\frac{F_x(x, y)}{F_y(x, y)}, \quad F_y \neq 0 \quad (8.3-4)$$

$$\frac{dx}{dy} = -\frac{F_y(x, y)}{F_x(x, y)}, \quad F_x \neq 0. \quad (8.3-5)$$

Ausgehend von einem gegebenen Anfangspunkt (x_0, y_0) kann die Projektion der Schnittlinie durch Lösung der gewöhnlichen Differentialgleichung (8.3-4) für y als Funktion von x oder für x als Funktion von y (Gleichung (8.3-5)) fortgesetzt werden.

8.3.3 Numerische Ausführung

Ausgehend von einem Anfangspunkt können die gewöhnlichen Differentialgleichungen (8.3-4) bzw. (8.3-5) numerisch als Anfangswertaufgaben mit einem Einzelschrittverfahren näherungsweise gelöst werden:

$$y_{i+1} = y_i + \Phi_x(x_i, y_i), \quad x_{i+1} = x_i + h, \quad i = 0, 1, 2, \dots, \quad (8.3-6)$$

$$x_{i+1} = x_i + \Phi_y(y_i, x_i), \quad y_{i+1} = y_i + h, \quad i = 0, 1, 2, \dots. \quad (8.3-7)$$

In der Regel werden äquidistante Stützstellen mit der Schrittweite h verwendet. Benutzt man das klassische *Runge-Kutta*-Verfahren vierter Ordnung, so lautet das Rechenschema z.B. für den Schritt von x_i nach $x_{i+1} = x_i + h$ (SCHWETLICK und KRETZSCHMAR, 1991):

$$y_{i+1} = y_i + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \quad (8.3-8)$$

mit

$$K_1 = -\frac{F_x(x_i, y_i)}{F_y(x_i, y_i)}, \quad (8.3-9)$$

$$K_2 = -\frac{F_x(x_i + h/2, y_i + hK_1/2)}{F_y(x_i + h/2, y_i + hK_1/2)}, \quad (8.3-10)$$

$$K_3 = -\frac{F_x(x_i + h/2, y_i + hK_2/2)}{F_y(x_i + h/2, y_i + hK_2/2)}, \quad (8.3-11)$$

$$K_4 = -\frac{F_x(x_i + h, y_i + hK_3)}{F_y(x_i + h, y_i + hK_3)}. \quad (8.3-12)$$

Das gleiche Rechenschema wird angewendet, wenn die Differentialgleichung (8.3-5) gelöst werden soll, d.h. wenn $F_y \approx 0$. Man braucht lediglich die Variablen x und y zu vertauschen.

8.3.4 Ein Verfolgungsalgorithmus in Parameterdarstellung

Bei dem im vorhergehenden Abschnitt vorgestellten Algorithmus kann die Schnittlinie nach der Differentialgleichung (8.3-4) oder (8.3-5) verfolgt werden. Im allgemeinen Fall lässt sich nicht mit *einer* Gleichung auskommen. Man wird dazu gezwungen sein, zwischen den Differentialgleichungen zu schalten. Dies bedarf einer gesonderten Steuerung, insbesondere wenn die gesuchte Kurve nahezu parallel einer Koordinatenrichtung verläuft, denn dabei können an den Schaltstellen Unstetigkeiten der Kurve generiert werden. Diese Schwierigkeiten lassen sich umgehen, wenn man eine Parameterdarstellung der gesuchten Kurve benutzt.

Die Differentialgleichung (8.3-3) kann in folgender Kurzform aufgefasst werden:

$$F_x \frac{dx}{ds} + F_y \frac{dy}{ds} = 0. \quad (8.3-13)$$

Da diese Differentialgleichung die Bedingung

$$\frac{\partial F_x}{\partial y} = \frac{\partial F_y}{\partial x} \quad (8.3-14)$$

erfüllt, handelt es sich dabei um eine sogenannte exakte Differentialgleichung (BRONSTEIN et al., 1993), die eindeutig lösbar ist. Verwendet man nun die Bogenlänge s als Kurvenparameter, so entstehen zwei gekoppelte Differentialgleichungen

$$F_x \frac{dx}{ds} + F_y \frac{dy}{ds} = 0, \quad (8.3-15)$$

$$dx^2 + dy^2 = ds^2 \quad (8.3-16)$$

für die Beschreibung der gesuchten Schnittlinie. Werden diese Differentialgleichungen mit der Schrittweite $h = s_{n+1} - s_n$ diskretisiert, so entstehen die Differenzgleichungen

$$F_x(x_{n+1} - x_n) + F_y(y_{n+1} - y_n) = 0, \quad (8.3-17)$$

$$(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2 = h^2, \quad (8.3-18)$$

welche einen neuen Punkt (x_{n+1}, y_{n+1}) der gesuchten Kurve als Schnittpunkt einer Geraden mit einem Kreis vom Radius h im Punkt (x_n, y_n) darstellen. Die Richtung der Geraden ist gleich der Richtung des Tangentenvektors an die Kurve und ergibt sich sofort aus (8.3-17) zu

$$\alpha = \arctan \frac{-F_x}{F_y}. \quad (8.3-19)$$

Die beiden möglichen Schnittpunkte lassen sich aus (8.3-18) mit der positiven und der negativen Schrittweite berechnen:

$$(x_{n+1} - x_n) = \pm h \frac{(x_{n+1} - x_n)}{h} = \pm h \cos \alpha, \quad (8.3-20)$$

$$(y_{n+1} - y_n) = \pm h \frac{(y_{n+1} - y_n)}{h} = \pm h \sin \alpha. \quad (8.3-21)$$

Je nach Wahl des Vorzeichens wird die Schnittlinie in der einen oder in der anderen Richtung durchlaufen. Zusammenfassend bekommt man für die Verfolgung der Schnittlinie die folgende Iterationsvorschrift:

$$\alpha_n = \arctan \frac{-F_x(x_n, y_n)}{F_y(x_n, y_n)}, \quad (8.3-22)$$

$$x_{n+1} = x_n \pm h \cos \alpha_n, \quad (8.3-23)$$

$$y_{n+1} = y_n \pm h \sin \alpha_n. \quad (8.3-24)$$

Im Vergleich zu dem vorhergehenden Algorithmus kommt man hier mit den drei einfachen Beziehungen (8.3-22) bis (8.3-24) aus, ohne zwischen den Differentialgleichungen schalten zu müssen. Dagegen ist die Konsistenzordnung dieses Algorithmus deutlich niedriger als bei dem *Runge-Kutta*-Verfahren, was eine schlechtere numerische Genauigkeit dieses Integrationsverfahrens bedeutet. Die Konsistenzordnung, somit auch die Integrationsgenauigkeit kann bei Bedarf auf unterschiedliche Weise erhöht werden, z.B. indem man das Prädiktor-Korrektor-Verfahren (ROSS und SCHWETLICK, 1999) anwendet.

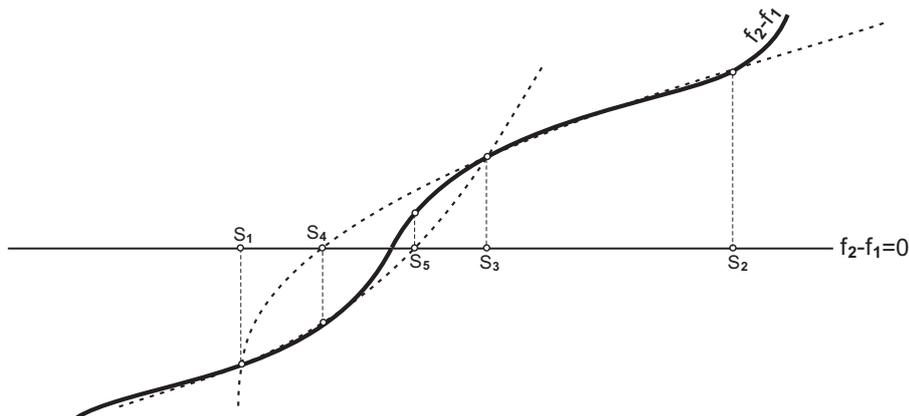


Abbildung 8.3-2: Schematische Darstellung der Startpunktsuche

8.3.5 Startpunktsuche

Um einen Verfolgungsalgorithmus laufen zu lassen, muss noch ein Startpunkt angegeben werden. Diesen kann man mit dem folgenden Suchverfahren iterativ finden. Zunächst setzen wir voraus, dass zwei Punkte, welche die Bedingung

$$F(\mathbf{v}_1) \cdot F(\mathbf{v}_2) < 0 \quad (8.3-25)$$

erfüllen, gegeben sind, wobei $\mathbf{v}_1 = [x_1 \ y_1]$ und $\mathbf{v}_2 = [x_2 \ y_2]$. Dann wird in der Mitte der dritte Punkt definiert:

$$\mathbf{v}_3 := \frac{\mathbf{v}_1 + \mathbf{v}_2}{2}. \quad (8.3-26)$$

Weiter arbeitet das Suchverfahren iterativ. Zunächst wird aus diesen Punkten die Folge $\{s_k\}$ gebildet zu

$$s_1 := 0, \quad (8.3-27)$$

$$s_2 := 1, \quad (8.3-28)$$

$$s_3 := \frac{\|\mathbf{v}_3 - \mathbf{v}_1\|}{\|\mathbf{v}_2 - \mathbf{v}_1\|}. \quad (8.3-29)$$

Dann wird die Folge $\{(s_k, F_k)\}$, $F_k := F(\mathbf{v}_k)$; $k = 1, 2, 3$, mit dem Polynom

$$p_k = As^2 + Bs + C = 0 \quad (8.3-30)$$

approximiert. Die Koeffizienten A , B und C ergeben sich aus dem entsprechenden linearen Gleichungssystem zu

$$A = \frac{(1 - s_3)F_1 + s_3F_2 - F_3}{s_3(1 - s_3)}, \quad (8.3-31)$$

$$B = \frac{-(1 + s_3^2)F_1 - s_3^2F_2 + F_3}{s_3(1 - s_3)}, \quad (8.3-32)$$

$$C = F_1. \quad (8.3-33)$$

Die Wurzeln von p_k bezeichnen wir mit λ_1 und λ_2 und nehmen die, die innerhalb des Intervalls

$$\lambda := \{\lambda_1, \lambda_2\} \cap [\min\{s_k\}, \max\{s_k\}] \quad (8.3-34)$$

liegt. Der neue Punkt \mathbf{v}_{k+1} berechnet sich zu

$$\mathbf{v}_{k+1} = \mathbf{v}_1 + \lambda(\mathbf{v}_2 - \mathbf{v}_1). \quad (8.3-35)$$

Wenn $F(\mathbf{v}_3) \neq 0$, ist zur Gleichung (8.3-27) zurückzukehren und die Approximation erneut durchzuführen. Dazu werden, außer dem Punkt \mathbf{v}_{k+1} , noch die folgenden zwei Punkte genommen:

$$\mathbf{v}_k \leftarrow \mathbf{v}_i \quad \text{wenn} \quad \min\{\|\lambda - s_i\|\}; \quad F(\mathbf{v}_{k+1}) \cdot F(\mathbf{v}_i); \quad i = 1, 2, 3 \quad (8.3-36)$$

$$\mathbf{v}_{k-1} \leftarrow \mathbf{v}_i \quad \text{wenn} \quad \min\{\|\lambda - s_i\|\}; \quad i = 1, 2. \quad (8.3-37)$$

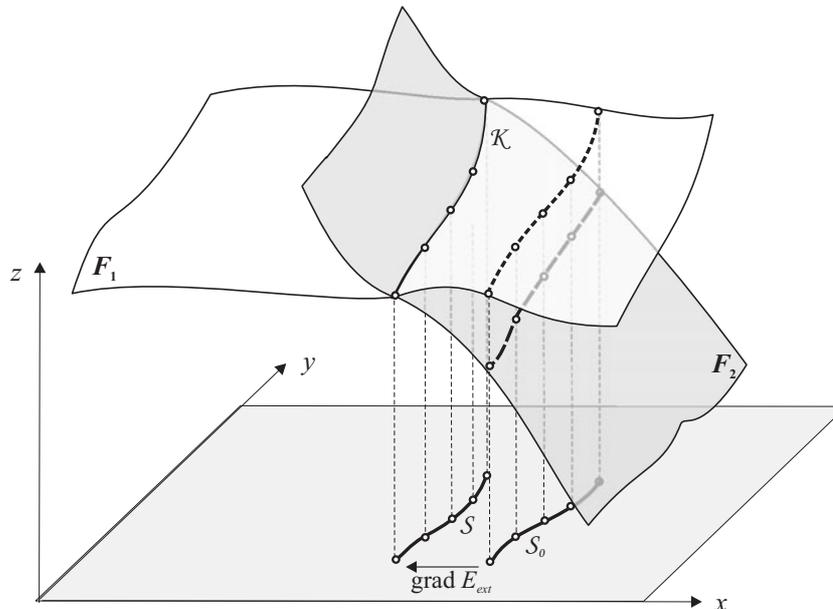


Abbildung 8.3-3: Identifikation der Schnittkurve K zweier Oberflächen mit Snake S

Die Idee des Suchalgorithmus ist in der Abbildung 8.3-2 veranschaulicht. Es werden die ersten zwei Iterationsschritte gezeigt. In jedem Iterationsschritt werden aus vier Punkten die drei genommen, die am dichtesten beieinander liegen und zwei davon die Bedingung (8.3-25) erfüllen.

Die durchgeführten Tests haben gezeigt, dass dieses Suchverfahren bereits nach wenigen Schritten konvergiert. Von dem gefundenen Startpunkt an wird die Schnittlinie in die eine und in die entgegengesetzte Richtung verfolgt.

8.3.6 Kantensuche mit Snakes

Die Lage der Kante kann auch mit Snakes gefunden werden. Die Idee ist in der Abbildung 8.3-3 veranschaulicht: In die xy -Ebene legen wir (fast beliebig) eine zweidimensionale Schlange S_0 . Von dieser Anfangslage an soll die Schlange sich im \mathbb{R}^2 bewegen und verformen und zwar solange, bis die optimale Approximation der Projektionslinie der Schnittkurve S durch diese Schlange erreicht wird. Wir benutzen den klassischen Algorithmus; man braucht lediglich die äußere Energie geeignet zu definieren. Logischerweise soll diese Energie proportional zu Abständen zwischen den beiden Oberflächen in der momentanen Schlangenlage sein. Es seien $z_1 = f_1(x, y)$ und $z_2 = f_2(x, y)$ die "Snakeshöhen" auf der ersten und auf der zweiten Oberfläche entsprechend. Die externe Energie

$$E_{ext} \propto \frac{\mu}{2}(z_2 - z_1)^2 \quad (8.3-38)$$

skalieren wir zusätzlich mit dem Gewichtskoeffizienten μ . Daraus ergeben sich die benötigten Ableitungen

$$\frac{\partial E_{ext}}{\partial x} = \mu(z_2 - z_1) \left\{ \frac{\partial f_2(x, y)}{\partial x} - \frac{\partial f_1(x, y)}{\partial x} \right\}, \quad (8.3-39)$$

$$\frac{\partial E_{ext}}{\partial y} = \mu(z_2 - z_1) \left\{ \frac{\partial f_2(x, y)}{\partial y} - \frac{\partial f_1(x, y)}{\partial y} \right\}. \quad (8.3-40)$$

Für TPS ist

$$\frac{\partial f}{\partial x} = \sum_{i=1}^n \lambda_i (x - x_i) (\ln r_i^2 + 1) + \nu_{10} \quad (8.3-41)$$

und

$$\frac{\partial f}{\partial y} = \sum_{i=1}^n \lambda_i (y - y_i) (\ln r_i^2 + 1) + \nu_{01}. \quad (8.3-42)$$

Im Vergleich zum Verfolgungsalgorithmus braucht man bei diesem Verfahren weder einen Startpunkt zu suchen, noch eine gesonderte Steuerung, um zwischen den Differentialgleichungen zu schalten. Darüber hinaus hat die Schlange die Fähigkeit zu "wachsen" und zu "schrumpfen": Es werden zusätzliche Snakespunkte eingeführt,

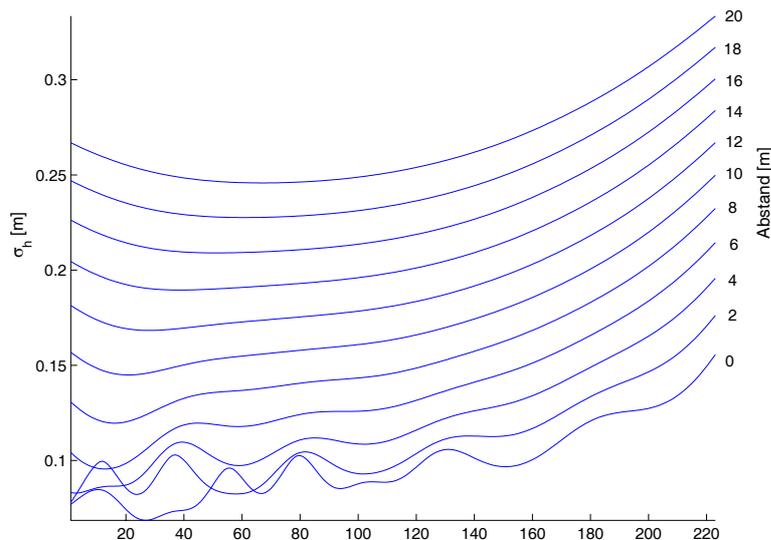


Abbildung 8.3-4: Modellrechnung: Höhengengenauigkeit einer Schnittlinie für unterschiedliche Abstände dieser Linie von den Datenpunkten

wenn die Schlangenelemente zu groß werden oder Snakespunkte werden beseitigt, wenn die Schlangenelemente zu klein werden. Dadurch erreicht man eine hohe gleichmäßige Genauigkeit der identifizierten Schnittlinie. Im Verfolgungsverfahren nimmt diese Genauigkeit mit wachsendem Abstand vom Startpunkt ab.

8.3.7 Genauigkeitsabschätzung

Um die Genauigkeit der identifizierten Schnittlinie zu bestimmen, müsste man ihre wahre Lage kennen. Diese ist jedoch unbekannt. Man hat lediglich die Funktionswerte einer unbekanntes Funktion in diskreten Punkten. Geht man davon aus, dass die Genauigkeit der Lagekoordinaten deutlich, in der Regel um eine Größenordnung höher ist als die Genauigkeit der Höhen, so kann man die Lagekoordinaten als fehlerfrei anhalten. Dann lässt sich auch nur die Höhengengenauigkeit der Schnittlinie abschätzen.

Das Gleichungssystem (3.1-12) kann in der einfachen Form

$$\mathbf{B} \cdot \mathbf{p} = \mathbf{z} \quad (8.3-43)$$

aufgefasst werden. Nach der allgemeinen Fehlerfortpflanzung bekommt man die Kovarianzmatrix der Parameter \mathbf{p} ,

$$\mathbf{C}_{\mathbf{pp}} = \mathbf{B}_z^{-1} \mathbf{C}_{\mathbf{zz}} \mathbf{B}_z^{-1}, \quad (8.3-44)$$

wobei $\mathbf{C}_{\mathbf{zz}}$ die Kovarianzmatrix der zur Bestimmung der Parameter dienenden Eingangsdaten ist, meistens

$$\mathbf{C}_{\mathbf{zz}} = \text{diag}\{\sigma_{w_1}^2 \quad \sigma_{w_2}^2 \quad \sigma_{w_3}^2 \quad \dots \quad \sigma_{w_n}^2 \quad 0 \quad 0 \quad 0\}. \quad (8.3-45)$$

Mit \mathbf{B}_z ist die Systemmatrix bezeichnet, die aufgrund der Ausgangsdaten für die Bestimmung der Parameter \mathbf{p} aufgestellt wird. Mit \mathbf{B}_h bezeichnen wir die Systemmatrix, welche aufgrund der Schnittlinienkoordinaten zusammengestellt wird. Dann lässt sich die Kovarianzmatrix der Schnittlinienhöhen berechnen zu

$$\mathbf{C}_{\mathbf{hh}} = \mathbf{B}_h \mathbf{B}_z^{-1} \mathbf{C}_{\mathbf{zz}} \mathbf{B}_z^{-1} \mathbf{B}_h. \quad (8.3-46)$$

Daraus ist ersichtlich, dass die Höhengengenauigkeit der Schnittlinie gleich der Genauigkeit der Daten ist, wenn die Schnittlinie durch die Datenpunkte geht.

Im Zusammenhang mit der Genauigkeit stellt sich auch die Frage: Wie weit dürfen Datenpunkte von der zu modellierenden Schnittlinie liegen, damit die Modellierung noch sinnvoll wird? Im strengen Sinne kann diese Fragen wohl nicht beantwortet werden. Mit Hilfe von (8.3-46) können lediglich Modellrechnungen gemacht werden.

In der Abbildung 8.3-4 sind die Ergebnisse einer solchen Modellrechnung für eine Kante aus dem Testbeispiel 1 (vgl. Abb. 8.3-7) veranschaulicht. Die Höhengengenauigkeit der Kante wurde für unterschiedliche Abstände der

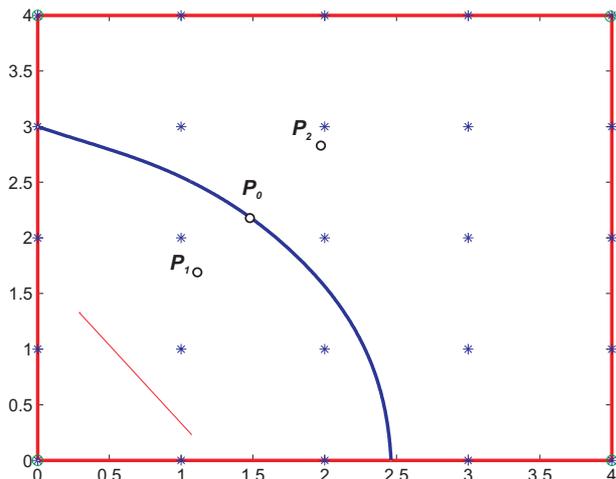


Abbildung 8.3-5: Datenpunkte, Startschlange (dünne Linie) und die identifizierte Kante

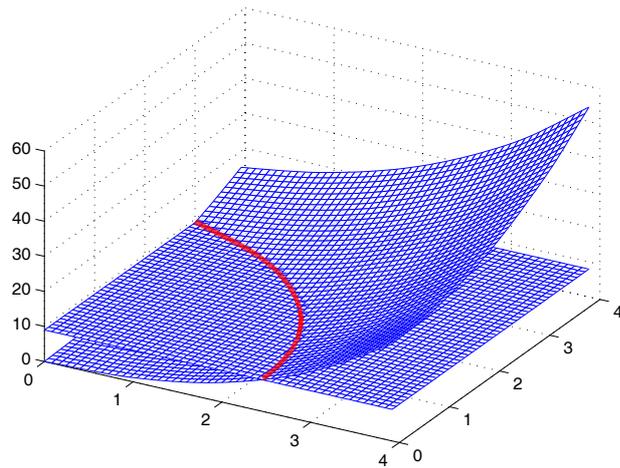


Abbildung 8.3-6: Zwei Oberflächen mit snakes-identifizierter Schnittlinie

Kante von den die Oberfläche definierenden Datenpunkten abgeschätzt. Dabei wurde die gleiche Genauigkeit der Eingangsdaten $\sigma_w = 0.10m$ vorausgesetzt. Daraus ist ersichtlich, dass für Laserscanner-Daten mit einem Punktabstand von wenigen Metern die diskutierte Methode eine Höhengenaugkeit der Kante etwa auf dem Niveau der Genauigkeit der Eingangsdaten sichert.

8.3.8 Testbeispiele

Mit den vorgestellten Verfahren zur Schnittlinienidentifikation zweier Oberflächen haben wir unterschiedliche Tests durchgeführt, welche gezeigt haben, dass sowohl die numerische Integration als auch der Snakes-Algorithmus vergleichbare Resultate liefern können. Für die in diesem Abschnitt angegebenen Testbeispiele verwenden wir den Snakes-Algorithmus und den im Abschnitt 8.3.4 präsentierten Verfolgungsalgorithmus.

Das prinzipielle Funktionieren der Algorithmen zeigen wir zunächst an einem synthetischen Beispiel, welches das Verschneiden einer Ebene mit einem Paraboloid darstellt. In der Abbildung 8.3-5 sind die Datenpunkte gezeigt, welche die Oberflächen beschreiben: Sterne gehören zum Paraboloid, vier Kreise zur Ebene. Mit der dünnen Linie ist die Startschlange und mit der dicken Linie die Lage der identifizierten Schnittlinie gezeichnet. Die Daten wurden mit TPS approximiert, die Schnittlinie mit Snakes identifiziert. Ausgehend von der Startlage bewegt sich eine kleine Schlange in Richtung der sinkenden externen Energie und "wächst" mit jedem Iterationsschritt solange, bis sie die Grenze des Arbeitsgebietes erreicht hat. Diese Grenze bildet in unserem Algorithmus die konvexe Hülle der Datenpunkte, welche aufgrund der Daten gebildet oder vorgegeben werden kann. Dabei wurden die Steuerparameter $\alpha = \beta = 1$ und $\mu = 0.01$ verwendet. Während für die Steuerparameter α und β eine relativ große Variationsbreite zulässig ist (man bekommt lediglich mehr oder weniger glatte Kurven), konvergiert das Verfahren nicht mehr, wenn man μ sogar nur geringfügig erhöht. Um sicher zu gehen, ist es sinnvoll, mit einem sehr kleinen Wert von μ zu beginnen oder sogar zu arbeiten. Das hat zur Konsequenz, dass man einige zusätzliche Iterationen durchführen muss. Wenn das Verfahren konvergiert, so konvergiert es in wenigen Iterationsschritten, fast unabhängig von der Lage der Startschlange. Die Abbildung 8.3-6 zeigt die beiden Oberflächen mit der identifizierten Schnittlinie.

Als Startwerte für den Verfolgungsalgorithmus werden zwei Punkte P_1 und P_2 gewählt, die durch die zu findende Schnittlinie getrennt sind (vgl. Abb. 8.3-5). Weiter läuft der Algorithmus automatisch: Auf der Verbindungslinie $P_1 - P_2$ wird mit Hilfe des im Abschnitt 8.3.5 beschriebenen Verfahrens der Startpunkt P_0 gefunden. Von diesem Punkt an läuft der Verfolgungsalgorithmus in die eine Richtung bis zur Grenze des Arbeitsgebietes, dann wird zum Punkt P_0 zurückgekehrt, das Vorzeichen in den Formel (8.3-23) und (8.3-24) wird geändert, und die Linie wird in der entgegengesetzten Richtung verfolgt.

Nun wollen wir unsere Verfahren mit reellen Daten konfrontieren. Zu diesem Zweck wurden zwei Testobjekte gemessen, welche sich innerhalb der Stadt Wroclaw (Breslau) befinden und künstlich entstandene Geländeformen repräsentieren. Die Testobjekte wurden im Gelände mit der tachymetrischen Aufnahme erfasst, wobei die Massenpunkte gleichmäßig, aber irregulär wie beim Laserscanning verteilt wurden. Die Einordnung der Punkte zu einzelnen Teiloberflächen erfolgte im Gelände. Um die Qualität der aufgrund von Massenpunkten modellier-

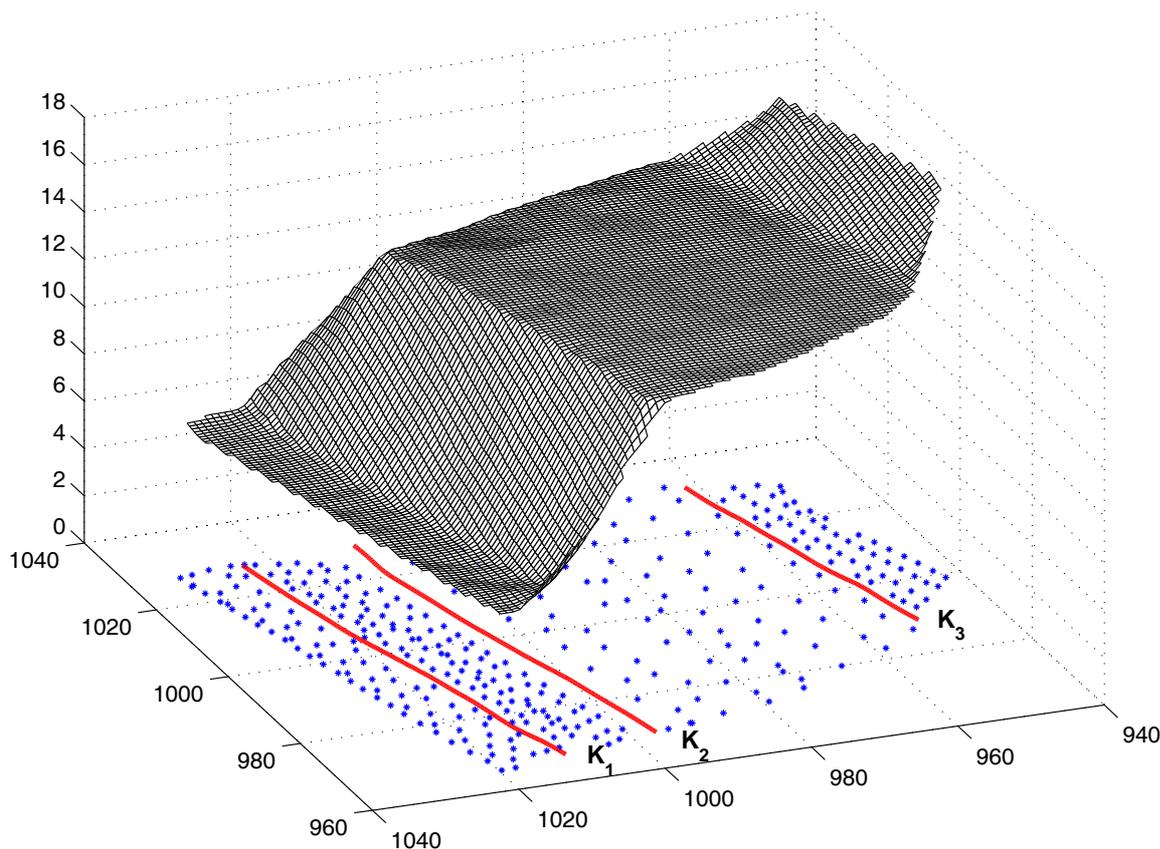


Abbildung 8.3-7: Testbeispiel 1: Datenpunkte und die identifizierten Kanten

ten Kanten zu beurteilen, wurden die Kanten ebenso im Gelände sorgfältig identifiziert und getrennt erfasst.

In der Abbildung 8.3-7 ist das Testbeispiel 1 gezeigt. Im Grundriss sind die Datenpunkte visualisiert, welche zur Approximation einzelner Teiloberflächen und ferner zur Modellierung der Kanten benutzt wurden. Die mit Snakes identifizierte drei Kantenlinien K_1 , K_2 und K_3 , wobei $\alpha = \beta = \mu = 1$ eingesetzt wurden, sind ebenso in der xz -Ebene gezeichnet. In der Abbildung 8.3-8 sind die Lage- und Höhenfehler der identifizierten Kanten entlang dieser Kanten graphisch dargestellt. Im Fall der Lagefehler (Bilder in der oberen Zeile) sind mit den durchgezogenen Linien die modellierten Kanten und mit den punktierten Linien die im Gelände gemessenen Kanten gezeichnet. Zahlenmäßig ist die Modellierungsgenauigkeit der Kanten in der Tabelle 8.3-1 dargestellt, und zwar für den Snakes-Algorithmus und für den Verfolgungsalgorithmus. Damit die Ergebnisse der beiden Verfahren vergleichbar sind, wurde für die Integrationsschrittweite $h = 0.2m$ die Länge des Snakeselements gewählt. Um die Lagegenauigkeit abzuschätzen, wurden Abstände d_i ; $i = 1, 2, 3, \dots$, zwischen der gemessenen und der modellierten Kante, wie im Bild 8.3-9 dargestellt, berechnet. Für die einzelnen Kanten sind der maximale Wert dieses Abstandes d_{max} , der Mittelwert \bar{d} und die Standardabweichung σ_d angegeben. Für die Abschätzung der Höhengenaugkeit wurden einfach die Höhendifferenzen in entsprechenden Punkten der Kante gebildet und daraus wurden die maximale Abweichung Δz_{max} , der Mittelwert $\overline{\Delta z}$ und die Standardabweichung $\sigma_{\Delta z}$ abgeleitet. Aus den angegebenen Genauigkeitsparametern ist ersichtlich, dass die als Schnitt zweier Oberflächen abgeleiteten Kanten die Erwartungen zur praktischen Nutzung gut erfüllen können, auch wenn man die maximale Abweichung der modellierten von der wahren Kante als Genauigkeitskriterium nimmt. Die Genauigkeitsparameter für die beiden Verfahren sind gleich.

Das in der Abbildung 8.3-10 visualisierte Testbeispiel 2 ist komplizierter strukturiert; die Kanten treten hier als gekrümmte Raumkurven auf. Die Kanten K_i ; $i = 1, 2, 3, 4$, wurden hier nach dem gleichen Snakes-Verfahren mit den Steuerparametern $\alpha = \beta = \mu = 1$ identifiziert. Die Visualisierung der Daten und der Ergebnisse der Modellierung erfolgte hier nach dem gleichen Muster wie im vorhergehenden Testbeispiel. Die Abweichungen der identifizierten Kanten von den gemessenen in der Lage und in der Höhe sind im Bild 8.3-11 gezeichnet. Die Tabelle 8.3-2 beinhaltet die maximalen Abweichungen in der Lage und in der Höhe sowie deren Mittelwerte und Standardabweichungen für jede Kante und für die zwei Identifikationsverfahren. Dieses Beispiel bestätigt die hohe Genauigkeit der Kantenmodellierung sowohl mit dem Snakes-Algorithmus als auch mit dem Verfol-

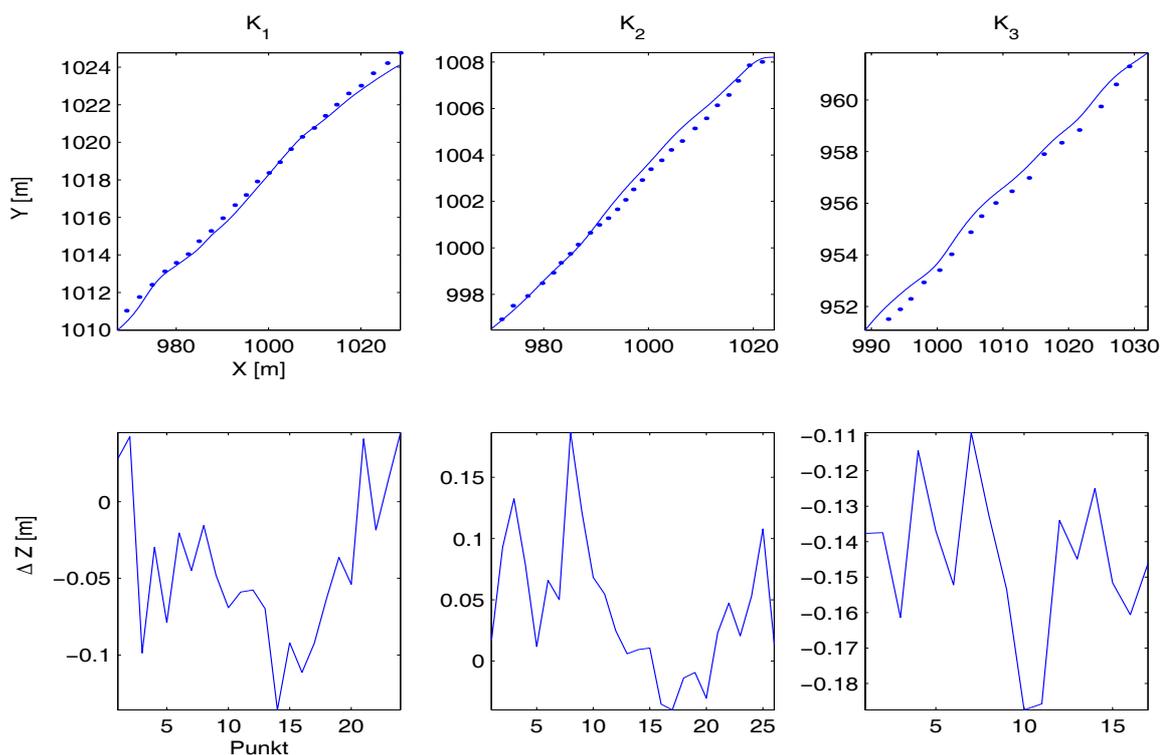


Abbildung 8.3-8: Testbeispiel 1: Lage- und Höhenfehler der identifizierten Kanten; Einzelheiten im Text

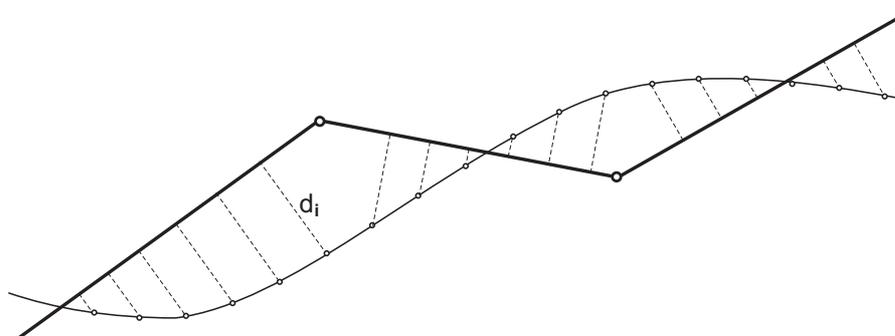


Abbildung 8.3-9: Zur Abschätzung der Lagegenauigkeit; die gemessene (dicke Linie) und die modellierte Kante (dünne Linie)

Tabelle 8.3-1: Testbeispiel 1: Genauigkeitsparameter der identifizierten Kanten [in m]

Kante	d_{max}	\bar{d}	σ_d	Δz_{max}	$\bar{\Delta z}$	$\sigma_{\Delta z}$
Snakes-Algorithmus						
K_1	0,65	0,29	0,17	-0,14	-0,04	0,05
K_2	0,56	0,27	0,18	0,18	0,04	0,06
K_3	0,53	0,36	0,13	-0,18	-0,15	0,02
Verfolgungsalgorithmus						
K_1	0,65	0,28	0,17	-0,14	-0,04	0,05
K_2	0,57	0,27	0,18	0,19	0,04	0,06
K_3	0,58	0,40	0,14	-0,18	-0,15	0,02

Tabelle 8.3-2: Testbeispiel 2: Genauigkeitsparameter der identifizierten Kanten [in m]

Kante	d_{max}	\bar{d}	σ_d	Δz_{max}	$\overline{\Delta z}$	$\sigma_{\Delta z}$
Snakes-Algorithmus						
K_1	0,88	0,42	0,27	-0,15	-0,02	0,07
K_2	0,32	0,11	0,07	-0,10	0,03	0,04
K_3	1,06	0,55	0,31	0,23	0,04	0,08
K_4	0,89	0,46	0,25	0,16	0,04	0,05
Verfolgungsalgorithmus						
K_1	0,85	0,41	0,26	-0,15	-0,02	0,07
K_2	0,37	0,12	0,07	-0,12	0,01	0,05
K_3	1,06	0,53	0,30	0,22	0,03	0,08
K_4	0,87	0,42	0,24	0,14	0,03	0,04

gungsalgorithmus. Allerdings muss zugegeben werden, dass diese Testbeispiele besonders zu den angewendeten Verfahren prädestiniert sind: Die exakte Klassifizierung der Punkte erfolgte bereits im Gelände, die Kanten waren scharf und eindeutig im Gelände identifizierbar, daher auch die hohe Genauigkeit. Im natürlichen Gelände würde wohl die Modellierungsgenauigkeit zurückgehen. Die natürlichen Geländekanten bzw. Formlinien sind aber auch mit anderen Verfahren mit einer niedrigeren Genauigkeit erfassbar, denn sie sind schwieriger zu identifizieren. Andererseits treten solche Geländekanten wie in den Testbeispielen oft auf, z.B. auf Dämmen.

Im Bezug auf die gemessenen Geländekanten liefern die untersuchten Verfahren die modellierten Geländekanten mit einer vergleichbaren Genauigkeit. Die untersuchten Verfahren können auch im Bezug auf die innere Genauigkeit miteinander verglichen werden. Wird einmal die Lage der Kante als Schnittlinie zweier Oberflächen identifiziert, so können die Höhenkoordinaten dieser Linie sowohl für die eine Oberfläche Z_1 als auch für die andere Z_2 ermittelt werden. Die Differenz ΔZ der Höhen kann als Maß für die innere Genauigkeit der Verfahren benutzt werden. Die Abbildungen 8.3-12 und 8.3-13 zeigen die Differenzen ΔZ für die einzelnen Geländekanten der beiden Testbeispiele, wenn die Kanten mit dem Snakes-Verfahren bzw. mit dem Verfolgungsalgorithmus identifiziert wurden. Im Testbeispiel 1 ist die Höhengenaugigkeit der beiden Algorithmen vergleichbar. Im Testbeispiel 2, wo die Geländekanten eine größere Krümmung als im Testbeispiel 1 besitzen, ist die Genauigkeit des Snakes-Algorithmus um eine Größenordnung höher als beim Verfolgungsalgorithmus. Darüber hinaus ist die Eigenschaft der numerischen Integration ersichtlich: Ausgehend vom Startpunkt sinkt die Genauigkeit mit der Bogenlänge. Die Genauigkeit der numerischen Integration kann erhöht werden, indem man die Schrittweite h verkleinert oder die Konsistenzordnung erhöht, z.B. dadurch, dass man für den Richtungswinkel α im Punkt n den Mittelwert aus den Punkten n und $n + 1$ nimmt. Die Genauigkeit des Snakes-Algorithmus kann erhöht werden, indem man die Länge des Snakeelements verkürzt oder auch für die Steuerparameter α und β , ggf. auch für μ kleinere Werte einsetzt.

Mit den beschriebenen Verfahren haben wir auch das Testbeispiel Tagebau bearbeitet: Die ursprünglichen Daten wurden automatisch mit dem im Abschnitt 8.3.1 vorgeschlagenen Verfahren in sechs Teiloberflächen eingeordnet. Ferner wurden die Kanten automatisch mit dem Snakes-Verfahren modelliert. Das Ergebnis dieser Modellierung ist in der Abbildung 8.3-14 gezeigt. Es entspricht unserer Erwartung.

Abschließend wollen wir zu diesem Abschnitt die folgenden Bemerkungen formulieren:

- Mit den vorgeschlagenen Verfahren können (Gelände-)Kanten zuverlässig und mit einer hohen Genauigkeit identifiziert werden. Die Voraussetzung dazu ist eine vollständige Grobfehlerbeseitigung und Einordnung der Daten in die einzelnen Teiloberflächen.
- Die Genauigkeit der Kantenmodellierung hängt von der Genauigkeit der Daten und deren Dichte sowie vom Verschneidungswinkel der sich zu schneidenden Oberflächen ab. Die numerische Genauigkeit der Kantenidentifikation lässt sich in der Regel bis zu einem akzeptablen Niveau absenken, so dass diese Fehlerquelle als unbedeutend für die praktische Anwendung gehandelt werden kann.
- Eine falsche Einordnung der Datenpunkten kann zu einer fehlerbehafteten Kantenmodellierung führen. Im Zweifelsfall können für die Punkte aus einem unscharfen Kantenbereich kleinere Gewichte eingeführt werden.

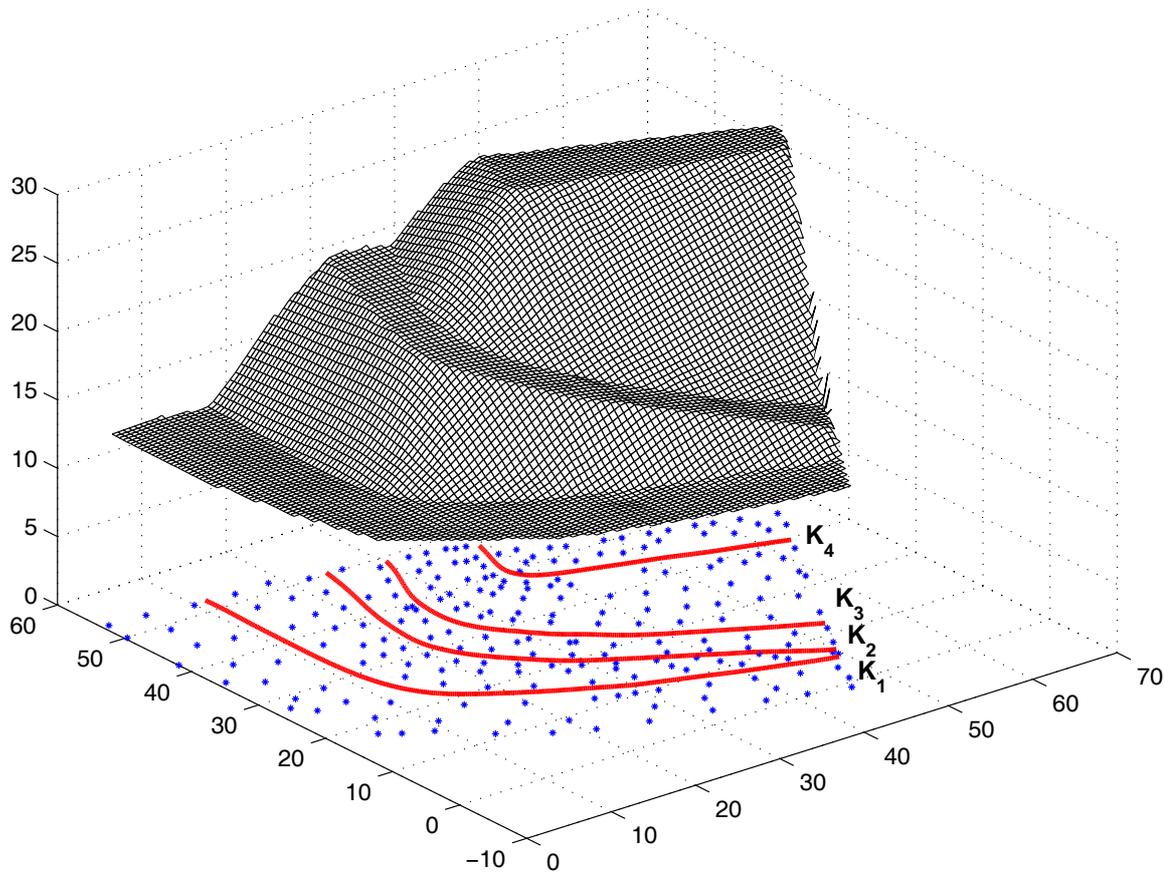


Abbildung 8.3-10: Testbeispiel 2: Datenpunkte und die identifizierten Kanten

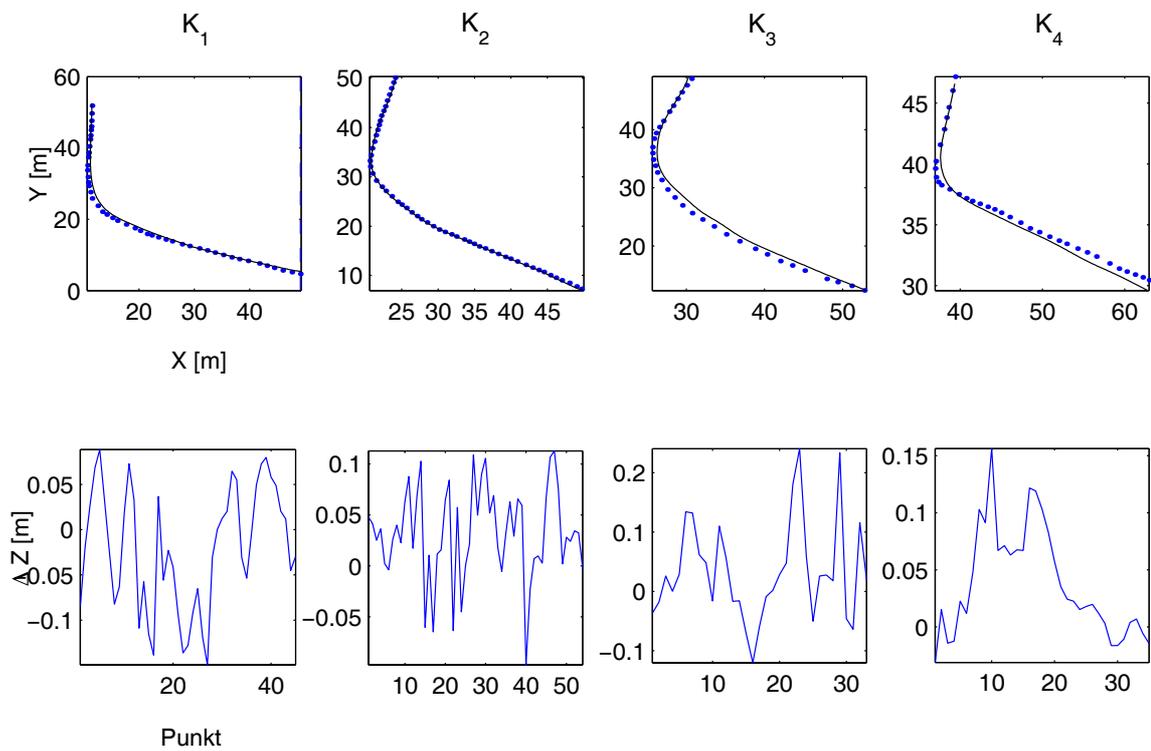


Abbildung 8.3-11: Testbeispiel 2: Lage- und Höhenfehler der identifizierten Kanten; Einzelheiten im Text

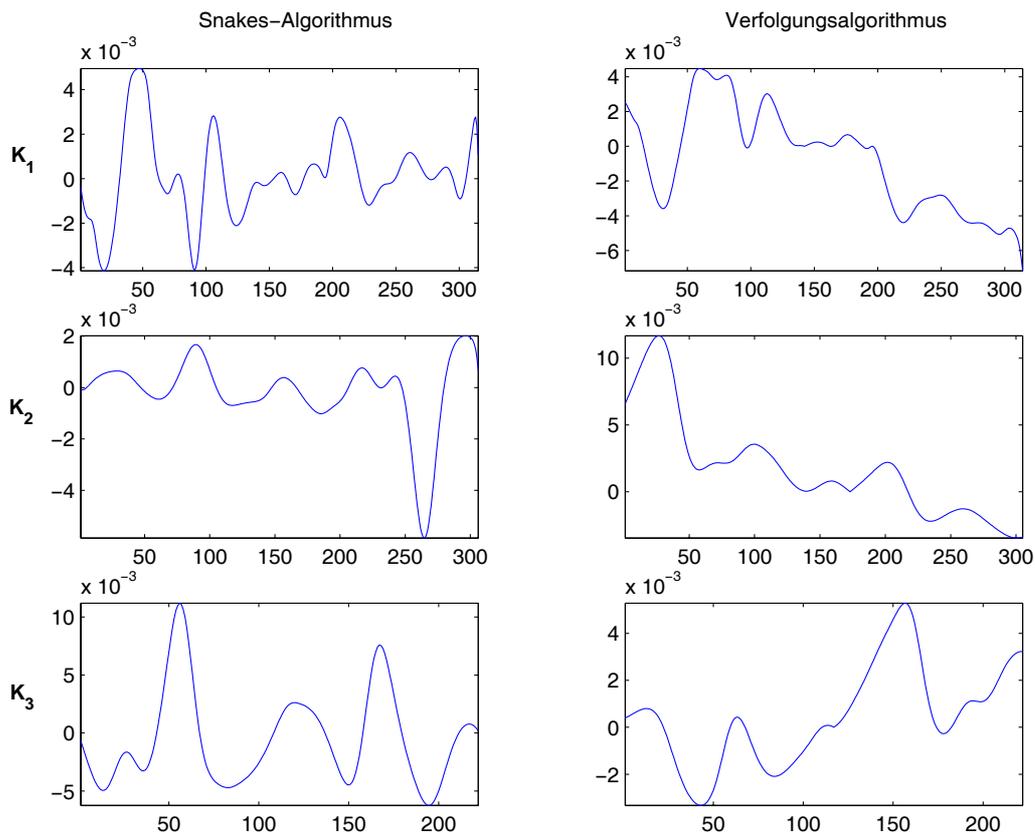


Abbildung 8.3-12: Testbeispiel 1: Die innere Höhengenaugkeit der identifizierten Kanten [in m]

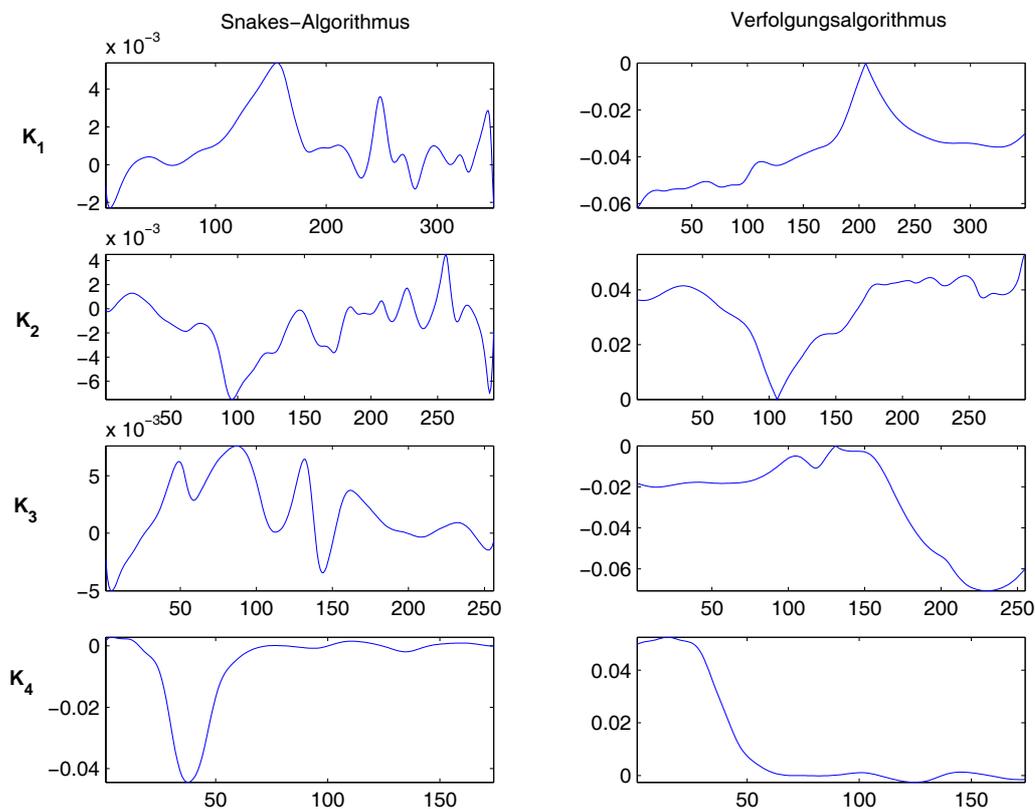


Abbildung 8.3-13: Testbeispiel 2: Die innere Höhengenaugkeit der identifizierten Kanten [in m]

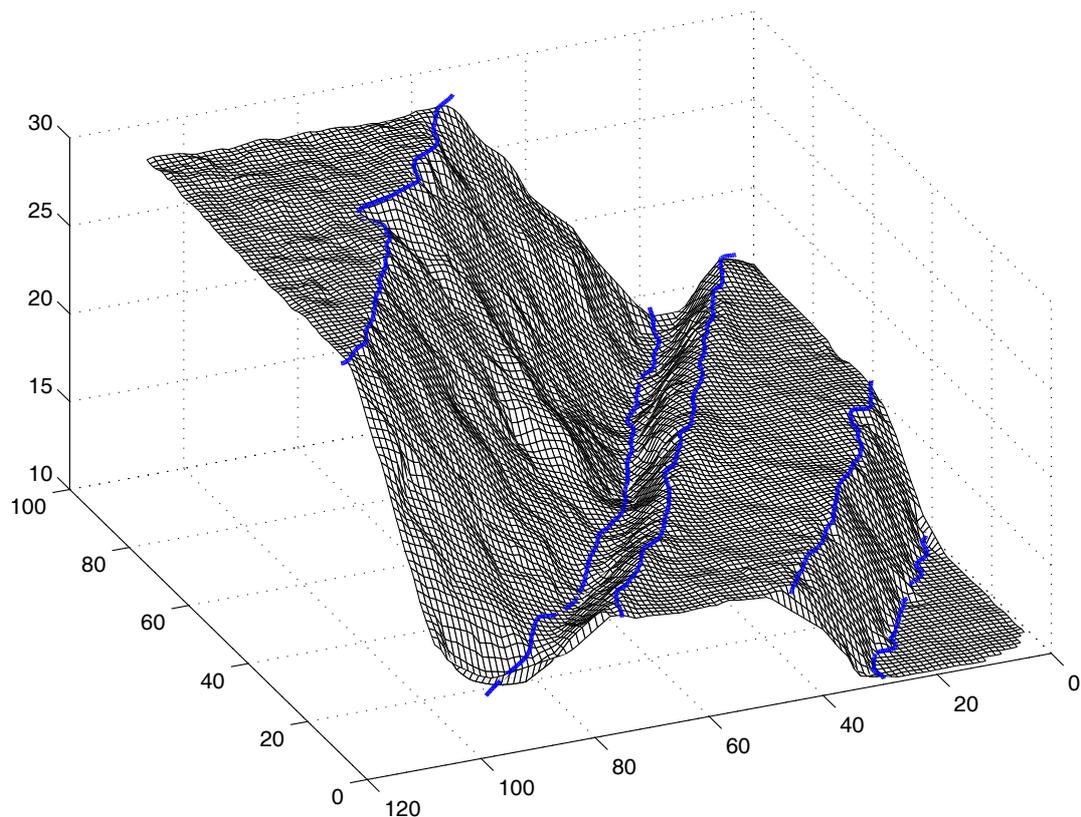


Abbildung 8.3-14: Testbeispiel Tagebau mit identifizierten Kanten

- Das am Institut für Photogrammetrie und Fernerkundung der TU Wien entwickelte und im Abschnitt 8.1 erwähnte Verfahren kann als Sonderfall des vorgeschlagenen Verfahrens angesehen werden. Für genügend großes α im Gleichungssystem (3.1-12) geht die Schnittkurve in eine dreidimensionale Schnittgerade über.
- Im Gegensatz zur numerischen Integration ermöglicht das Snakes-Verfahren auch Messfehlerglättung in Bezug auf die zu modellierende Kante. Dies kann durch Berücksichtigung von Gewichten im Gleichungssystem (3.1-12) oder durch "Verstärkung" der Suchschlange (größere Werte für α , β) oder beides erfolgen.
- Das Snakes-Verfahren zeichnet sich durch eine große Flexibilität aus. Außerdem braucht man bei diesem Verfahren keinen Startpunkt zu suchen.
- Beim Verfolgungsalgorithmus wird die Kante lokal erfasst. Zur numerischen Integration wird nur ein vorübergehender Punkt benötigt (Einschrittverfahren). Das Verfahren läuft sehr schnell.
- Beim Snakes-Algorithmus wird die Kante global erfasst. Je länger die zu identifizierte Kante, desto aufwendiger das Verfahren. Je höher die erwartete Genauigkeit der Identifikation, umso mehr Iterationen sind in der Regel erforderlich.

9 Zusammenfassung

Die Modellierung von erdoberflächenbezogenen Daten, der Geodaten, ist ein vielseitiges und interdisziplinäres Problem. Die Fülle der existierenden Ansätze und Modelle ist kaum noch überschaubar. In vielen Bereichen, wie zum Beispiel digitale Geländemodelle, existieren effiziente und bewertete Lösungen. Neue Methoden der Datengewinnung, die aus dem technischen Fortschritt resultieren, stellen auch neue Anforderungen an die Datenbearbeitungsmethoden und eröffnen neue Möglichkeiten der Oberflächenmodellierung. In den letzten Jahren hat das Laserscanning die Datengewinnung für digitale Geländemodelle revolutioniert. Dieses Verfahren liefert eine sehr dichte diskrete Information über die zu modellierende Oberfläche in Form einer gleichmäßig-regellos in der Ebene verteilten Punktwolke $\{x, y, z\} \subset \mathbb{R}^3$. Diese dichte, teilweise auch redundante Information erlaubt eine effiziente Modellierung von besonderen Strukturen der Oberfläche wie Geländekanten, welche für eine hochqualitative Beschreibung der Oberfläche unentbehrlich sind. Gleichzeitig stellen Nicht-Geländepunkte (Reflexionen an Gebäuden, Bäumen etc.) einen bedeutenden Anteil an den Laserscanner-Datensätzen dar. Solche Daten sind während der Modellierung effizient und zuverlässig aus den Datensätzen zu beseitigen.

Die Arbeit beginnt mit einer sehr knappen Einführung in die Problematik der digitalen Geländemodelle und einer Übersicht der herkömmlichen Modellierungsmethoden von Oberflächen in der expliziten Darstellung $z = z(x, y)$. Damit ordnen wir diese Arbeit in die geowissenschaftliche Aufgabenstellung ein. Hierzu sind auch robuste Verfahren, die zur Filterung von Laserscanner-Daten konzipiert wurden, diskutiert worden.

Den Kern der Arbeit bilden die deformierbaren Modelle von Kurven und Oberflächen. Sie basieren auf dem physikalischen Prinzip der Energieminimierung und entstehen als Lösung von Variationsproblemen. Die gesamte Energie setzt sich aus der inneren und der äußeren Energie zusammen. Die letztere Energie wird kontextabhängig durch Daten generiert; meistens beschreibt sie die Deviation zwischen den Daten und dem Modell. Die innere Energie beschreibt die geometrischen Eigenschaften der Kurve und setzt sich aus einem Elastizitäts- und einem Zähigkeitsterm zusammen. Die beiden Terme werden mit den ortsabhängigen Steuerparametern α und β gegenseitig bewichtet. Durch das Variieren dieser Parameter lassen sich Kurven mit gewünschten geometrischen Eigenschaften erzeugen. Das Prinzip der Energieminimierung kann auf unterschiedlichen Wegen realisiert werden und führt zu unterschiedlichen *Snakes*-Varianten, welche im Kapitel 4 ausführlich dargestellt und diskutiert sind. Die Snakes-Approximation haben wir zur profilweisen Modellierung von Oberflächen angewendet. Hierzu haben wir eine Formulierung der externen Energie gefunden, welche die robuste Modellierung von Profilen ermöglicht: In *einem* Iterationsprozess können die Grobfehler beseitigt, die Messfehler geglättet und die Diskontinuitäten (Knicke) modelliert bzw. erhalten werden. Die Anpassung des Snakes-Modells an Daten erfolgt iterativ, wobei die Steuerparameter abhängig von den Daten auszulegen sind. Um die drei genannten Anforderungen in einem Iterationsprozess zu verbürgen, ist die eingeführte nicht-symmetrische Definition der externen Energie mit den Steuerparametern $\alpha > 0$ und $\beta > 0$, welche um die Knickstelle geeignet (bis zu $\alpha = \beta = 0$) herabgesetzt werden, zu verwenden.

Als Verallgemeinerung von Snakes haben wir das Modell von hinreichend glatten, energiegeladenen Flächenstücken eingeführt und mit *Flakes* bezeichnet. Die innere Energie beim Flakes-Modell setzt sich aus dem *membrane*- und dem *thin-plate*-Kern zusammen, welche die Neigung und die Krümmungseigenschaften der zu modellierende Oberfläche beschreiben. Die Energieanteile werden wiederum mit den ortsabhängigen Steuerparametern α und β gegenseitig bewichtet. Die Minimierung der gesamten Energie führt wieder zu einem Variationsproblem, welches mit unterschiedlichen Mitteln und Methoden gelöst wurde. Durch Formulierung der zu dem ursprünglichen Variationsproblem äquivalenten Euler-Gleichungen und deren Diskretisierung mit finiten Differenzen entstand das Flakes-Modell für reguläre Daten. Alternativ zu den Euler-Gleichungen kann auch das ursprüngliche Variationsproblem direkt mit dem sogenannten Ritz-Verfahren gelöst werden. Hierzu sind geeignete Basisfunktionen zu wählen. Mit einer linearen Basisfunktion haben wir ein fortgeschrittenes Flakes-Modell für reguläre Daten entwickelt. Um diese Modelle anwenden zu können, müssen die Daten die gleiche reguläre Struktur aufweisen, d.h. ggf. regularisiert werden. Mit einer Basisfunktion vom Gauß-Typ konnte aber das Flakes-Modell für die Modellierung von irregulären Daten entwickelt werden. Hierzu kommt ein zusätzlicher Steuerparameter σ^2 , welcher das Einzugsgebiet für die einzelnen Basisfunktionen bestimmt.

Die Modellierung der Daten mit Flakes führt zu entsprechenden linearen Gleichungssystemen, die in der Regel iterativ gelöst werden. Der Flakes-Ansatz für reguläre Daten zeichnet sich durch die schwachbesetzte Systemmatrix mit einer Bandstruktur aus und ist daher numerisch günstiger. Mit nur zwei Steuerparametern α und β ist er auch leichter steuerbar, bedarf aber der Regularisierung der Daten. Mit diesem Verfahren ist auch eine robuste Approximation möglich. Dabei ist der gleiche, nicht symmetrische Ansatz für die externe Energie wie im eindimensionalen Fall zu verwenden. Der Flakes-Ansatz für irreguläre Daten lässt eine beliebige Datenstruktur

zu. Es wird keine Regularisierung benötigt und man arbeitet mit Originaldaten. Der Preis dafür ist die vollbesetzte Systemmatrix, welche auch mit einem größeren numerischen Aufwand aufgestellt werden muss.

Die Flakes-Verfahren zeichnen sich durch eine große Flexibilität aus und erlauben eine weitgehende Erhaltung von Oberflächendiskontinuitäten während der Approximation. Um dieses Ziel zu erreichen, ist es am einfachsten, im Bereich der Kanten $\alpha = \beta = 0$ zu setzen und ggf. σ^2 gegen Null tendieren zu lassen, während im sonstigen Bereich die Steuerparameter entsprechend der erwarteten Approximationsqualität zu variieren sind.

Mit den deformierbaren Modellen lassen sich (Gelände-)Kanten nachbilden bzw. weitgehend erhalten. Die Verfahren geben aber keine Auskunft über den räumlichen Verlauf der Kanten. Eine solche Information wird aber in vielen Anwendungen benötigt. Um aus einer unstrukturierten Punktwolke $\{x, y, z\}$, insbesondere aus Laserscanner-Daten, die räumliche Beschreibung von Kanten im Vektorformat abzuleiten, haben wir vorgeschlagen, die grobfehlerfreien Daten mit Oberflächenfunktionen zu beschreiben und die Kanten als Schnitt zweier Oberflächen $z_i = f_i(x, y)$; $i = 1, 2$ zu ermitteln. Zu diesem Zweck müssen zunächst die Daten in die einzelnen Teilerflächen eingeordnet werden. Hierzu können die Standardmethoden der digitalen Bildverarbeitung oder das in dieser Arbeit vorgeschlagene Verfahren eingesetzt werden. Es wird die Projektion $\{x(s), y(s)\}$ der Kante in die xy -Ebene gefunden und die z -Koordinaten ergeben sich aus $f_i(x(s), y(s))$. Für die Schnittkurve zweier Oberflächen gilt $f_1(x(s), y(s)) = f_2(x(s), y(s)) = 0$. Auf dieser Bedingung basierend haben wir zwei Verfahren zur Identifikation der Schnittkurve entwickelt. Ein Verfolgungsalgorithmus beruht auf der numerischen Integration einer dem Problem entsprechenden Differentialgleichung. Für die numerische Integration wird ein Startpunkt benötigt. Hierzu haben wir ein Suchverfahren vorgeschlagen. Dem lokalen Verfolgungsalgorithmus haben wir ein globales Verfahren, welches den Snakes-Ansatz mit einer geeigneten Definition der externen Energie benutzt, gegenübergestellt. Die beiden Algorithmen haben wir an natürlichen Datensätzen getestet, wobei die zu identifizierenden Kanten im Gelände erfasst wurden, um die Genauigkeit der Modellierung abzuschätzen. Für die Beschreibung von Oberflächen haben wir die *thin plate* Splines benutzt. Es können aber alle in den digitalen Geländemodellen verwendeten Approximationsmethoden, wie die lineare Prädiktion (Kriging), die radialen Basisfunktionen, die finiten Elemente etc., benutzt werden.

Die durchgeführten Tests haben gezeigt, dass mit den vorgeschlagenen Verfahren die Geländekanten zuverlässig und mit einer hohen Genauigkeit aus einer unstrukturierten Punktwolke identifiziert werden können. Besonders möchten wir das auf dem Snakes-Algorithmus basierende Verfahren hervorheben. Es zeichnet sich durch eine große Flexibilität aus, bedarf keines Startpunktes; die Startschlange kann beliebig platziert werden und konvergiert meistens problemlos, denn die externe Energie ist hier im Gegensatz zur traditionellen Anwendung in der digitalen Bildverarbeitung eindeutig definiert. Darüber hinaus passt dieses Verfahren zum Konzept dieser Abhandlung, in der die deformierbaren Modelle die dominierende Rolle spielen.

Literaturverzeichnis

- AMINI, A. A., TEHRANI, S. und WEYMOUTH, T. E. (1988). *Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints*. Proc. Second International Conference on Computer Vision, 95–99.
- AMINI, A. A., WEYMOUTH, T. E. und JAIN, E. C. (1990). *Using dynamic programming for solving variational problems in vision*. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 12, No. 9, 855–867.
- AXELSSON, P. (1999). *Processing of laser scanner data - algorithms and applications*. ISPRS Journal of Photogrammetry and Remote Sensing, 54(2), 138–147.
- AXELSSON, P. (2000). *DEM generation from laser scanner data using adaptive TIN models*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part B4/1, 110–117.
- BAMLER, R. (1989). *Mehrdimensionale lineare Systeme. Fourier-Transformation und δ -Funktionen*. Springer-Verlag, Berlin.
- BARTELME, N. (1995). *Geoinformatik - Modelle Strukturen Funktionen*. Springer-Verlag, Berlin, Heidelberg.
- BORKOWSKI, A., BURGHARDT, D. und MEIER, S. (1997). *Zur optimalen Approximation von Höhenprofilen*. Österr. Zeitschrift für Vermessung und Geoinformation, 281–285.
- BORKOWSKI, A., BURGHARDT, D. und MEIER, S. (1999). *A fast snakes algorithm using the tangent angle function*. Internat. Arch. Photogr. Rem. Sensing 32 (3-2W5), 61–65.
- BORKOWSKI, A. und KELLER, W. (2002). *Von stationären und instationären Filtern*. Festschrift zum 65. Geburtstag von Prof. Dr.-Ing. habil. Sigfried Meier. Technische Universität Dresden, 53–65.
- BORKOWSKI, A. und KELLER, W. (2003). *Modelling of irregularly sampled surfaces by two-dimensional snakes*. Journal of Geodesy, Vol. 77, No 9, 543–553.
- BORKOWSKI, A. und MEIER, S. (1999). *Versuche zur robusten Sakes-Approximation von Höhenprofilen mit Diskontinuitäten*. Z. Photogrammetrie, Fernerkundung, Geoinformation, 381–390.
- BORKOWSKI, A. und MEIER, S. (2001). *Robustification of tangent angle function snakes*. Z. Photogrammetrie, Fernerkundung, Geoinformation, 129–135.
- BRIESE, C. und KRAUS, K. (2003). *Datenreduktion dichter Laser-Geländemodelle*. ZfV - Zeitschrift für Geodäsie, Geoinformation und Landmanagement, 312–317.
- BRIESE, C., KRAUS, K. und PFEIFER, N. (2002a). *Modellierung von dreidimensionalen Geländekanten in Laser-Scanner-Daten*. Festschrift zum 65. Geburtstag von Prof. Dr.-Ing. habil. Sigfried Meier. Technische Universität Dresden, 47–52.
- BRIESE, C., PFEIFER, N. und DORNINGER, P. (2002b). *Applications of the robust interpolation for DTM determination*. Symposium ISPRS Commission III, Photogrammetric Computer Vision, Graz, 9 - 13 September 2002. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV / 3A, 55 – 61.
- BRIGGER, P., HOEG, J. und UNSER, M. (2000). *B-spline snakes: A flexible tool for parametric contour detection*. IEEE Transactions on image processing Vol. 9, No 9, 1484–1496.
- BRONSTEIN, I. und SEMENDJAJEW, K. (1962). *Taschenbuch der Mathematik*. B.G. Teubner Verlagsgesellschaft, Leipzig.
- BRONSTEIN, I., SEMENDJAJEW, K., MUSIOL, G. und MÜHLIG, H. (1993). *Taschenbuch der Mathematik*. Verlag Harri Deutsch.
- BRÜGELMANN, R. (2000). *Automatic breaklines detection from airborne laser scanner data*. Int. Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, Part B3, Amsterdam, 109–116.
- BURGHARDT, D. (2001). *Automatisierung der kartographischen Verdrängung mittels Energieminimierung*. Deutsche Geodätische Kommission, München, Reihe C, H. 536.
- BURGHARDT, D. und MEIER, S. (1997a). *Cartographic displacement using the snakes concept*. In: Förstner, W.; Plümer, L. (eds.): Semantic Modeling for the Acquisition of Topographic Information from Images and Maps, Birkhäuser Verlag, Basel, 59–71.

- BURGHARDT, D. und MEIER, S. (1997b). *Kartographische Verdrängung nach Extremalprinzipien*. Z. f. Vermessungswesen, Heft 8, 377–386.
- CARLSON, R. und FOLEY, T. (1991). *The parameter r^2 in multiquadric interpolation*. Computers and Mathematics with Applications, 21, 29–42.
- CASPARY, W. (2002). *Kinematische Messung von Fahrbahnoberflächen*. Festschrift zum 65. Geburtstag von Prof. Dr.-Ing. habil. Sigfried Meier. Technische Universität Dresden, 33–38.
- CHAN, T. F. und VESE, L. A. (2001). *Active contours without edges*. IEEE Transactions on Image Processing, Vol. 10, 266–277.
- CHEN, G. (1996). *Robuste Verfahren zur Analyse linearer stochastischer Prozesse im Zeitbereich*. Schriftenreihe, Studiengang Vermessungswesen Universität der Bundeswehr München, H. 53.
- COHEN, L. D. (1991). *On active contour models and balloons*. Computer Vision, Graphics, and Image Processing: Image Understanding, Vol. 53 No. 2, 211–218.
- COHEN, L. D. und COHEN, I. (1990). *A finite-element method applied to new active contour models and 3D reconstruction from cross sections*. Proceedings. Third International Conference on Computer Vision, Osaka, Japan; IEEE Comput. Soc. Press, 587–591.
- COHEN, L. D. und COHEN, I. (1993). *Finite-element methods for active contour models and balloons for 2-D and 3-D images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15 No. 11, 1131–1147.
- CRESSIE, N. (1993). *Statistics for Spatial Data*. John Wiley and Sons, New York.
- DE BOOR, C. und RON, A. (1992). *Computational aspects of polynomial interpolation in several variables*. Mathematics of Computation, 48, 705–727.
- DUCHON, J. (1976). *Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces*. R.A.I.R.O. Analyse Numérique, Vol. 10, 5–12.
- EBNER, H. (1990). *Digitale Geländemodelle aus heutiger Sicht*. Festschrift für Rüdiger Finsterwalder zum 60. Geburtstag, Technische Universität München, 51–60.
- EBNER, H., HOFMAN-WELLENHOF, B., REISS, P. und STEIDLER, F. (1980). *HIFI - Ein Minicomputer-Programmsystem für Höheninterpolation mit finiten Elementen*. Zeitschrift für Vermessungswesen, 105, 215–225.
- ELMQVIST, M. (2002). *Ground surface estimation from airborne laser scanner data using active shape models*. ISPRS, Commission III, Symposium Photogrammetric Computer Vision, September 9-13, Graz, 114–118.
- ELMQVIST, M., JUNGERT, E., PERSSON, A. und SODERMAN, U. (2001). *Terrain modelling and analysis using laser scanner data*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV-3/W4, Annapolis, Maryland, 22-24 October, 219–227.
- FARIN, G. (1993). *Curves and Surfaces for Computer Aided Geometric Design - A Practical Guide*. Academic Press.
- FRANKE, R. (1982). *Scattered data interpolation. Test of some methods*. Mathematics of Computation, 38, 181–200.
- FRANKE, R. und NIELSON, G. (1980). *Smooth interpolation of large sets of scattered data*. International Journal for Numerical Methods in Engineering, 15, 1691–1704.
- FRITSCH, D. (1991). *Raumbezogene Informationssysteme und digitale Geländemodelle*. Deutsche Geodätische Kommission, München, Reihe C, H. 369.
- FRITSCH, D. und KILLIAN, J. (1994). *Filtering and calibration of laser scanner measurements*. International Archives of Photogrammetry and Remote Sensing, Vol. XXX, part B3/1, 227–234.
- FUA, P. (1996). *Model-based optimization : Accurate and consistent site modeling*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXI, part B3, 222–223.

- FUA, P., GRÜN, A. und LI, H. (2000). *Optimization-based approaches to feature extraction from aerial images*. In: Dermanis, A.; Grün A.; Sansó, F. (eds.): *Geometric Methods for the Analysis of Data in Earth Sciences*, Springer-Verlag, Berlin, Heidelberg, 190–228.
- GOLDEN SOFTWARE (2002). *Surfer. User's Guide. Contouring and 3D Surface Mapping for Scientists and Engineers*. Golden Software, Inc.
- GOLUB, G. H. und LOAN, C. F. V. (1996). *Matrix Computations, Third edition*. The Johns Hopkins University Press.
- GRADSTEIN, I. und RISHIK, I. (1981). *Tables of Series, Products and Integrals*. Harri Deutsch, Frankfurt/M.
- GROSSMANN, C. und ROOS, H.-G. (1992). *Numerik partieller Differentialgleichungen*. Teubner Studienbücher Mathematik, B.G. Teubner, Stuttgart.
- GRÜN, A. und LI, H. (1995). *Road extraction from aerial and satellite images by dynamic programming*. ISPRS Journal of Photogrammetry and Remote Sensing 50(4), 11–20.
- GRÜN, A. und LI, H. (1996). *Linear feature extraction with LSB-Snakes from multiple images*. International Archives of Photogrammetry and Remote Sensing. Vol. XXXI, Part B3, 266–272.
- GRÜN, A. und LI, H. (1997a). *Linear feature extraction with 3-D LSB-snakes*. In: Automatic Extraction of Man-Made Objects from Aerial and Space Images (II) Monte Verita, Birkhäuser Verlag, Basel, 287–298.
- GRÜN, A. und LI, H. (1997b). *Semi-automatic linear feature extraction by dynamic programming and lsb-snakes*. Photogrammetric Engineering & Remote Sensing, 63(8), 985–995.
- GÜLCH, E. (1996). *Deformable models as a photogrammetric tool - potential and problems*. Internat. Arch. Photogr. Rem. Sensing, Vol. 31, part B3, 270–284.
- GUNN, S. R. und NIXON, M. S. (1997). *A robust snake implementation; a dual active contour*. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 19, No. 1, 63–68.
- HAMPEL, F., RONCHETTI, E., ROUSSEEUW, P. und STAHEL, W. (1986). *Robust Statistics*. John Wiley & Sons, New York.
- HARDY, R. L. (1971). *Multiquadric equations of topography and other irregular surfaces*. Journal of Geophysical Research, 76, 1905–1915.
- HARDY, R. L. (1972). *Geodetic applications of multiquadric analysis*. Allgemeine Vermessungsnachrichten, 79, 399–406.
- HARDY, R. L. (1990). *Theory and applications of the multiquadric-biharmonic method*. Computers and Mathematics with Applications, 19, 163–208.
- HEINRICH, B. (1987). *Finite difference method of irregular networks*. Akademie Verlag, Berlin.
- HOSCHEK, J. und LASSER, D. (1992). *Grundlagen der geometrischen Datenverarbeitung*. B.G. Teubner, Stuttgart.
- HÖWING, F., DOOLEY, L. S. und WERMSE, D. (2000). *Fuzzy active contour*. IEEE Proceedings on Vision, Image and Signal Processing, Vol. 147, August 2000, 322–330.
- HÖWING, F., WERMSE, D. und DOOLEY, L. S. (1997). *Fuzzy snakes*. International Conference on Image Processing and its Applications IPA97, Dublin, July 1997, 627–630.
- HUBER, J. (1981). *Robust Statistics*. John Wiley & Sons, New York.
- HUBER, P. (1964). *Robust estimation of a location parameter*. Annals of Mathematical Statistics, vol. 35, 73–101.
- KASS, M., WITKIN, A. und TERZOPOULOS, D. (1987). *Snakes: Active contour models*. Proceedings of the First International Conference on Computer Vision, IEEE Comput. Soc. Press, 259–268.
- KERSCHNER, M. (1998). *Homologous twin snakes integrated in a bundle block adjustment*. International Archives of Photogrammetry and Remote Sensing, Vol. 32 3/1, 244–249.

- KICIAK, P. (2000). *Podstawy modelowania krzywych i powierzchni - Zastosowania w grafice komputerowej*. Wydawnictwo Naukowo-Techniczne Warszawa.
- KILLIAN, J., HAALA, N. und ENGLISCH, M. (1996). *Capture and evaluation of airborne laser scanner data*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXI, Part B3, 383–388.
- KOCH, K.-R. (1973). *Höhenlinieninterpolation mittels gleitender Schrägebene und Prädiktion*. Vermessung, Photogrammetrie, Kulturtechnik, 71, 229–232.
- KOCH, K.-R. (1985). *Digitales Geländemodell mittels Dreiecksvermaschung*. Vermessungswesen und Raumordnung, 47, 129–135.
- KOCH, K.-R. (1987). *Parameterschätzung und Hypothesentests in linearen Modellen*. Dümmler, Bonn.
- KOCH, K.-R. (1996). *Robuste Parameterschätzung*. Allgemeine Vermessungsnachrichten, 103, 1–18.
- KRAUS, K. (1998). *Interpolation nach kleinsten Quadraten versus Krige-Schätzer*. Österr. Zeitschrift für Vermessung und Geoinformation, 86, 45–48.
- KRAUS, K. (2000). *Photogrammetrie. Band 3. Topographische Informatonssysteme*. Dümmler, Köln.
- KRAUS, K. und PFEIFER, N. (1997). *A new method for surface reconstruction from laser scanner data*. International Archives of Photogrammetry and Remote Sensing. Vol. XXXII, Part 3-2W3, Haifa, 80–86.
- KRAUS, K. und PFEIFER, N. (1998). *Determination of terrain models in wooded areas with aerial laser scanner data*. ISPRS Journal of Photogrammetry and Remote Sensing vol.53, 193–203.
- KRAUS, K. und PFEIFER, N. (2001). *Advanced DTM generatin from LIDAR data*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV-3/W4, Annapolis, Maryland, 22-24 October, 23–30.
- LEYMARIE, F. und LEVINE, M. D. (1993). *Tracking deformable objects in the plane using an active contour model*. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 15, No. 6, 617–634.
- LI, H. (1997). *Semi-Automatic Road Extraction from Satellite and Aerial Images*. PhD thesis, Institute of Geodesy and Photogrammetry, ETH Zürich.
- LIANG, J., MCINERNEY, T. und TEZOPOULOS, D. (1999). *United snakes*. IEEE Proceedings of the International Conference on Computer Vision, Vol. 2, 933–940.
- LINDENBERGER, J. (1993). *Laser-Profilmessungen zur topographischen Geländeaufnahme*. Deutsche Geodätische Kommission, München, Reihe C, H. 400.
- LODHA, S. K. und FRANKE, R. (1999). *Scattered data techniques for surfaces*. In: Hagen, H.; Nielsen, G.M.; Post F. (eds.): Dagstuhl'97 Scientific Visualization, IEEE Computer Society, Los Alamitos, 181–222.
- LOHMANN, P., KOCH, A. und SCHAEFFER, M. (2000). *Approaches to the filtering of laser scanner data*. Int. Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, Part B3, Amsterdam, 540–547.
- MAYER, H., LAPTEV, I. und BAUMGARTNER, A. (1998). *Mulit-scale and snakes for automatic road extraction*. Proceedings of the 5th European Conference on Computer Vision, Springer-Verlag, Berlin, 720–733.
- MAYER, H., LAPTEV, I., BAUMGARTNER, A. und STEGER, C. (1997). *Automatic road extraction based on mulit-scale modelling, context, and snakes*. Internatonal Archives of Photogrammetry and Remote Sensing, Vol. 32 3-2W3, 106–113.
- MCINERNEY, T. und TERZOPOULOS, D. (1995). *Topologically adaptable snakes*. IEEE Proceedings 5th International Conference on Computer Vision, 840–845.
- MEIER, S. (2000a). *Die Snakes-Approximation als Hilfsmittel der Geodaten-Verarbeitung*. AVN, Heft 2, 50–57.
- MEIER, S. (2000b). *Zur Qualität Snakes-approximierter Höhenprofile mit Diskontinuitäten*. Photogrammetrie, Fernerkundung, Geoinformation 6, 399–409.
- MEIER, S. und KELLER, W. (1990). *Geostatistik. Einführung in die Theorie der Zufallsprozesse*. Springer - Verlag Wien, New York.

- MENET, S., SAINT-MARC, P. und MEDIONI, G. (1990). *Active contour models: Overview, implementation and applications*. IEEE, International Conference on System, Man and Cybernetics, Conference Proceedings, 194–199.
- MENET, S., SAINT-MARC, P. und MEDIONI, G. (1991). *B-snakes: Implementation and application to stereo*. Artificial Intelligence and Computer Vision. Proceedings of the Seventh Israeli Conference; Ramat Gan, Israel; 26-27 Dec. 1990; Amsterdam, Netherlands, 223–236.
- MENZ, J. und PILZ, J. (1990). *Kollokation, universelles Kriging und Bayesscher Zugang*. Das Merkscheidewesen 101, 62–66.
- MITASOVA, H. und MITAS, L. (1993). *Interpolation by regularized spline with tension: I theory and implementation*. Mathematical Geology 25, 641–655.
- MORITZ, H. (1989). *Advanced Physical Geodesy (2nd ed.)*. Herbert Wichmann, Karlsruhe.
- NEUENSCHWANDER, W., FUA, P., IVERSON, L., SZEKELY, G. und KÜBLER, O. (1997). *Ziplock snakes*. International Journal of Computer Vision Vol. 25 No. 3, 191–201.
- NEUENSCHWANDER, W., FUA, P., SZEKELY, G. und KÜBLER, O. (1994). *Initializing snakes*. Conference on Computer Vision and Pattern Recognition. Proceedings CVPR '94. IEEE Computer Society, 658–663.
- NEUENSCHWANDER, W., FUA, P., SZEKELY, G. und KÜBLER, O. (1995). *From ziplock snake to velcroTM surfaces*. In: Automatic Extraction of Man-Made Objects from Aerial and Space Images, Monte Verita, Birkhäuser Verlag, Basel, 105–114.
- PARK, H. W., SCHOEPLIN, T. und KIM, Y. (2001). *Active contour model with gradient directional information: Directional snake*. IEEE Transaction on Circuits and Systems for Video Technology Vol. 11, No. 2, 252–256.
- PRESS, W. H., TEUKOLSKY, S. A., VETTERLING, W. T. und FLANNERY, B. P. (2001). *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press.
- REISS, P. (1985). *Aufbau digitaler Höhenmodelle auf der Grundlage einfacher finiter Elemente*. Deutsche Geodätische Kommission, München, Reihe C, H. 315.
- REINHARDT, W. (1991). *Aufbau hochqualitativer digitaler Geländemodelle an photogrammetrischen Stereosystemen*. Deutsche Geodätische Kommission, München, Reihe C, H. 381.
- ROGGERO, M. (2001). *Airborne laser scanning: Clustering in row data*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV-3/W4, Annapolis, Maryland, 22-24 October, 227–232.
- ROGGERO, M. (2002). *Object segmentation with region growing and principal component analysis*. Symposium ISPRS Commission III, Photogrammetric Computer Vision, Graz, 9 - 13 September 2002. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV / 3A, 289–294.
- ROSS, H.-G. und SCHWETLICK, H. (1999). *Numerische Mathematik*. B.G. Teubner Stuttgart, Leipzig.
- SCHWETLICK, H. und KRETZSCHMAR, H. (1991). *Numerische Verfahren für Naturwissenschaftler und Ingenieure*. Fachbuchverlag, Leipzig.
- SHEPARD, D. (1968). *A two-dimensional interpolation function for irregularly spaced data*. Proceedings 23rd. National Conference ACM, 517–523.
- SIBSON, R. (1981). *A brief description of natural neighbor interpolation*. Interpreting Multivariate Data, V. Barnett (ed.), John Wiley and Sons, New York, 21–36.
- SITHOLE, G. (2001). *Filtering of laser altimetry data using a slope adaptive filter*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV-3/W4, Annapolis, Maryland, 22-24 October, 203–210.
- SITHOLE, G. und VOSSELMAN, G. (2003). *Report ISPRS: Comparison of filters*. <http://www.commission3.isprs.org/wg3/>.
- STAIB, L. und DUNCAN, J. (1992). *Boundary finding with parametrically deformable models*. IEEE Transaction on Pattern Analysis and Machine Intelligence Vol. 14 No. 11, 1061–1075.
- STOYAN, D. und STOYAN, H. (1992). *Fraktale - Formen - Punktfelder: Methoden der Geometrie-Statistik*. Akademie-Verlag, Berlin.

- SUI, L. (2002). *Ableitung topographischer Strukturlinien aus Laserscannerdaten mit Methoden der Bildverarbeitung*. Z. Photogrammetrie, Fernerkundung, Geoinformation, 423–434.
- SUI, L. (2003). *Auswertung von Laserscannerdaten und automatische Ableitung von Strukturlinien mit digitalen Bildverarbeitungsmethoden*. ZfV - Zeitschrift für Geodäsie, Geoinformation und Landmanagement, 318–325.
- TERZOPOULOS, D. (1986). *Regularization of inverse visual problems involving discontinuities*. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol PAMI-8, No. 4, 413–423.
- TRINDER, J. C. und LI, H. (1995). *Semi-automatic feature extraction by snakes*. In: Automatic Extraction of Man-Made Objects from Aerial and Space Images, Monte Verita, Birkhäuser Verlag, Basel, 95–104.
- TRINDER, J. C. und LI, H. (1996). *Extraction of man-made features by 3-D active contour models*. Internat. Arch. Photogr. Rem. Sensing, Vol. 31, part B3, 874–879.
- VOSSELMAN, G. (2000). *Slope based filtering of laser altimetry data*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, Part B3, Amsterdam, 935–942.
- VOSSELMAN, G. und MAAS, H.-G. (2001). *Adjustment and filtering of raw laser altimetry data*. OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Elevation Models, Stockholm, 1.-3. march.
- WANG, Y., KHUANG, E. und SHEN, D. (1999). *Lane detection using B-snake*. IEEE International Conference in Information Intelligence and Systems, Rockville, Maryland, 31.03-3.04 1999, 438–443.
- WILD, E. (1983). *Die Prädiktion mit Gewichtsfunktionen und deren Anwendung zur Beschreibung von Gelände- flächen bei topographischer Geländeaufnahme*. Deutsche Geodätische Kommission, München, Reihe C, H. 277.
- WILLIAMS, D. J. und SHAH, M. (1990). *A fast algorithm for active contours and curvature estimation*. Proceedings of the Third International Conference on Computer Vision, IEEE Comput. Soc. Press, 592–595.
- XU, C. und PRINCE, J. L. (1998a). *Generalized gradient vector flow external forces for active contours*. Signal Processing Vol. 71, 131–139.
- XU, C. und PRINCE, J. L. (1998b). *Snakes, shapes and gradient vector flow*. IEEE Transactions on Image Processing Vol. 7, 359–369.
- XU, G., SEGAWA, E. und TSUJI, S. (1994). *Robust active contours with insensitive parameters*. Pattern Recognition Vol. 27, No.7, 879–884.
- YUEN, P., FENG, G. und ZHOU, J. (1998). *A contour detection method: Initialization and contour model*. Pattern Recognition Letters 20, 141–148.

Anhang A

Ableitungen für die Euler-Gleichungen

Die Ableitungen in den Euler-Gleichungen haben für konstante α und β die Form

$$\begin{aligned}
 \frac{\partial E_{z_x}}{\partial x} &= \alpha z_{xx}, & \frac{\partial E_{z_y}}{\partial y} &= \alpha z_{yy}, & E_{z_{xy}} &= 2\beta z_{xy}, \\
 E_{z_{xx}} &= \beta z_{xx}, & E_{z_{yy}} &= \beta z_{yy}, \\
 \frac{\partial E_{z_{xx}}}{\partial x} &= \beta z_{xxx}, & \frac{\partial^2 E_{z_{xx}}}{\partial x^2} &= \beta z_{xxx}, \\
 \frac{\partial E_{z_{yy}}}{\partial y} &= \beta z_{yyy}, & \frac{\partial^2 E_{z_{yy}}}{\partial y^2} &= \beta z_{yyy}, \\
 \frac{\partial E_{z_{xy}}}{\partial x} &= 2\beta z_{xyx}, & \frac{\partial^2 E_{z_{xy}}}{\partial x \partial y} &= 2\beta z_{xyxy}.
 \end{aligned}$$

Für variable $\alpha = \alpha(x, y)$ und $\beta = \beta(x, y)$ sind die entsprechenden Ausdrücke komplizierter:

$$\begin{aligned}
 E_{z_x} &:= \frac{\partial E}{\partial z_x} = \alpha z_x, & E_{z_y} &:= \frac{\partial E}{\partial z_y} = \alpha z_y, \\
 \frac{\partial E_{z_x}}{\partial x} &= \alpha_x z_x + \alpha z_{xx}, & \frac{\partial E_{z_y}}{\partial y} &= \alpha_y z_y + \alpha z_{yy}, \\
 E_{z_{xx}} &:= \frac{\partial E}{\partial z_{xx}} = \beta z_{xx}, & E_{z_{yy}} &:= \frac{\partial E}{\partial z_{yy}} = \beta z_{yy}, \\
 \frac{\partial E_{z_{xx}}}{\partial x} &= \beta_x z_{xx} + \beta z_{xxx}, & \frac{\partial E_{z_{yy}}}{\partial y} &= \beta_y z_{yy} + \beta z_{yyy}, \\
 \frac{\partial^2 E_{z_{xx}}}{\partial x^2} &= \beta_{xx} z_{xx} + \beta_x z_{xxx} + & \frac{\partial^2 E_{z_{yy}}}{\partial y^2} &= \beta_{yy} z_{yy} + \beta_y z_{yyy} + \\
 &+ \beta_x z_{xxx} + \beta z_{xxxx}, & &+ \beta_y z_{yyy} + \beta z_{yyyy}, \\
 &= \beta_{xx} z_{xx} + 2\beta_x z_{xxx} + \beta z_{xxxx} & &= \beta_{yy} z_{yy} + 2\beta_y z_{yyy} + \beta z_{yyyy}, \\
 E_{z_{xy}} &:= \frac{\partial E}{\partial z_{xy}} = 2\beta z_{xy}, \\
 \frac{\partial E_{z_{xy}}}{\partial x} &= 2(\beta_x z_{xy} + \beta z_{xyx}), \\
 \frac{\partial^2 E_{z_{xy}}}{\partial x \partial y} &= 2(\beta_{xy} z_{xy} + \beta_x z_{xyy} + \beta_y z_{xyx} + \beta z_{xyxy}).
 \end{aligned}$$

Anhang B

Approximation der Ableitungen mit finiten Differenzen

Mit der Indizierung der Stützstellenlage wie in Abbildung 5.2-1 und Diskretisierung mit Vorwärtsdifferenzen sind die ersten Ableitungen, z.B. in x-Richtung,

$$z_x|_{i,j} = \frac{1}{\Delta_x}(z_{i+1,j} - z_{i,j}), \quad z_x|_{i-1,j} = \frac{1}{\Delta_x}(z_{i,j} - z_{i-1,j})$$

und die zweite Ableitung

$$z_{xx}|_{i,j} = \frac{1}{\Delta_x}(z_x|_{i,j} - z_x|_{i-1,j}) = \frac{1}{\Delta_x^2}(z_{i+1,j} - 2z_{i,j} + z_{i-1,j}),$$

identisch mit (5.2-4). Um die vierte Ableitung zu approximieren, ist z_{xx} zusätzlich an den Stellen $(i+1, j)$ und $(i-1, j)$ zu ermitteln. Dann ist

$$z_{xxxx}|_{i,j} = \frac{1}{\Delta_x^2}(z_{xx}|_{i+1,j} - 2z_{xx}|_{i,j} + z_{xx}|_{i-1,j}) = \frac{1}{\Delta_x^4}(z_{i+2,j} - 4z_{i+1,j} + 6z_{i,j} - 4z_{i-1,j} + z_{i-2,j}),$$

identisch mit (5.2-6).

Die gemischte Ableitung z_{xyxy} wird aus den finiten Ausdrücken für

$$\begin{aligned} z_{xy}|_{i+1,j+1} &= \frac{1}{4\Delta_x\Delta_y}(z_{i+2,j+2} + z_{i,j} - z_{i,j+2} - z_{i+2,j}), \\ z_{xy}|_{i-1,j+1} &= \frac{1}{4\Delta_x\Delta_y}(z_{i,j+2} + z_{i-2,j} - z_{i-2,j+2} - z_{i,j}), \\ z_{xy}|_{i-1,j-1} &= \frac{1}{4\Delta_x\Delta_y}(z_{i,j} + z_{i-2,j-2} - z_{i-2,j} - z_{i,j-2}), \\ z_{xy}|_{i+1,j-1} &= \frac{1}{4\Delta_x\Delta_y}(z_{i+2,j} + z_{i,j-2} - z_{i,j} - z_{i+2,j-2}) \end{aligned}$$

approximiert zu

$$z_{xyxy}|_{i,j} = \frac{1}{4\Delta_x\Delta_y}(z_{xy}|_{i+1,j+1} + z_{xy}|_{i-1,j-1} - z_{xy}|_{i-1,j+1} - z_{xy}|_{i+1,j-1}).$$

Nach dem Auswerten dieses Ausdrucks bekommt man die Beziehung (5.2-8).

Anhang C

Auswertung der Integrale vom Abschnitt 5.3.1

Um die Doppelintegrale (5.3-10) bis (5.3-14) mit der Ansatzfunktion (5.3-16) und (5.3-17) auszuwerten, müsste man mit verallgemeinerten δ -Funktionen, und zwar nicht nur mit δ -Impulsen, sondern auch mit δ -Linien (BAMLER, 1989), arbeiten. Die Schwierigkeiten, welche damit verbunden sind, lassen sich umgehen, wenn man die Tensorprodukteigenschaften von $\varphi(x, y)$ ausnutzt. Mit (5.3-17) bis (5.3-19) lassen sich die eindimensionalen Integrale ohne weiteres auswerten (vgl. auch BORKOWSKI und KELLER (2002)):

$$\int_{\mathbb{R}} \varphi_i(s) \varphi_k(s) ds = \begin{cases} \frac{\Delta}{6} & , \quad |i - k| = 1 \\ \frac{2\Delta}{3} & , \quad i = k \\ 0 & , \quad \text{sonst,} \end{cases}$$

$$\int_{\mathbb{R}} \varphi'_i(s) \varphi'_k(s) ds = \begin{cases} -\frac{1}{\Delta} & , \quad |i - k| = 1 \\ \frac{2}{\Delta} & , \quad i = k \\ 0 & , \quad \text{sonst,} \end{cases}$$

$$\int_{\mathbb{R}} \varphi''_i(s) \varphi''_k(s) ds = \begin{cases} \frac{1}{\Delta^2} & , \quad |i - k| = 2 \\ -\frac{4}{\Delta^2} & , \quad |i - k| = 1 \\ \frac{6}{\Delta^2} & , \quad i = k \\ 0 & , \quad \text{sonst.} \end{cases}$$

Die Resultate der zweidimensionalen Integration ergeben sich entsprechend (5.3-20) durch Multiplizieren von Vektoren, die aus eindimensionalen Resultaten zusammengestellt werden. Als Beispiele geben wir das Resultat für (5.3-37),

$$\iint_{\mathbb{R}^2} \varphi_i(x, y) \varphi_j(x, y) dx dy = \begin{bmatrix} \frac{\Delta_x}{6} \\ \frac{2\Delta_x}{3} \end{bmatrix} \begin{bmatrix} \frac{\Delta_y}{6} & \frac{2\Delta_y}{3} \end{bmatrix} = \Delta_x \Delta_y \begin{bmatrix} \frac{1}{36} & \frac{1}{9} \\ \frac{1}{9} & \frac{4}{9} \end{bmatrix},$$

und für (5.3-23),

$$\iint_{\mathbb{R}^2} \varphi_{i,xx}(x, y) \varphi_{j,xx}(x, y) dx dy = \begin{bmatrix} \frac{1}{\Delta_x^2} \\ -\frac{4}{\Delta_x^2} \\ \frac{6}{\Delta_x^2} \end{bmatrix} \begin{bmatrix} 0 & \frac{\Delta_y}{6} & \frac{2\Delta_y}{3} \end{bmatrix} = \frac{\Delta_y}{\Delta_x^2} \begin{bmatrix} 0 & \frac{1}{6} & \frac{2}{3} \\ 0 & -\frac{2}{3} & -\frac{8}{3} \\ 0 & 1 & 4 \end{bmatrix},$$

an. Auf gleiche Weise bekommt man die sonstigen Integrale.

Anhang D

Auswertung der Integrale vom Abschnitt 5.3.2

Die Auswertung der Doppelintegrale (5.3-10) bis (5.3-14) mit der Gaußschen Ansatzfunktion demonstrieren wir am Beispiel des Koeffizienten a_{ij} ,

$$a_{ij} = \iint_{\mathbb{R}^2} \frac{x - x_i}{\sigma_i^2} \exp \left\{ -\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma_i^2} \right\} \frac{x - x_j}{\sigma_j^2} \exp \left\{ -\frac{(x - x_j)^2 + (y - y_j)^2}{2\sigma_j^2} \right\} dx dy.$$

Unter Verwendung der Bezeichnungen (5.3-48) bekommt man nach Umformen

$$\begin{aligned} a_{ij} &= \frac{1}{\sigma_i^2 \sigma_j^2} \int_{\mathbb{R}} \exp -\frac{ay^2 - 2b_y y + c_y}{2\sigma_i^2 \sigma_j^2} dy \int_{\mathbb{R}} [x^2 - (x_i + x_j)x + x_i x_j] \exp -\frac{ax^2 - 2b_x x + c_x}{2\sigma_i^2 \sigma_j^2} dx \\ &= \frac{1}{\sigma_i^2 \sigma_j^2} \exp \left\{ -\frac{c_x + c_y}{2\sigma_i^2 \sigma_j^2} \right\} I_1 (I_2 - I_3 + I_4) \end{aligned}$$

Die Integrale I_1 , I_2 , I_3 und I_4 lassen sich mit dem Tafelwerk von GRADSTEIN und RISHIK (1981) auswerten Integral Nr. 3.323.2:

$$I_1 := \int_{\mathbb{R}} \exp \left\{ -\frac{a}{2\sigma_i^2 \sigma_j^2} y^2 + \frac{b_y}{\sigma_i^2 \sigma_j^2} y \right\} dy = \sqrt{\frac{2\pi\sigma_i^2 \sigma_j^2}{a}} \exp \left\{ \frac{b_y^2}{2a\sigma_i^2 \sigma_j^2} \right\},$$

Integral Nr. 3.462.2:

$$I_2 := \int_{\mathbb{R}} x^2 \exp \left\{ -\frac{a}{2\sigma_i^2 \sigma_j^2} x^2 + \frac{b_x}{\sigma_i^2 \sigma_j^2} x \right\} dx = \left(\frac{\sigma_i^2 \sigma_j^2}{a} + \frac{b_x^2}{a^2} \right) \sqrt{\frac{2\pi\sigma_i^2 \sigma_j^2}{a}} \exp \left\{ \frac{b_x^2}{2a\sigma_i^2 \sigma_j^2} \right\},$$

$$I_3 := (x_i + x_j) \int_{\mathbb{R}} x \exp \left\{ -\frac{a}{2\sigma_i^2 \sigma_j^2} x^2 + \frac{b_x}{\sigma_i^2 \sigma_j^2} x \right\} dx = (x_i + x_j) \frac{b_x}{a} \sqrt{\frac{2\pi\sigma_i^2 \sigma_j^2}{a}} \exp \left\{ \frac{b_x^2}{2a\sigma_i^2 \sigma_j^2} \right\}$$

und

$$I_4 := x_i x_j \int_{\mathbb{R}} \exp \left\{ -\frac{a}{2\sigma_i^2 \sigma_j^2} x^2 + \frac{b_x}{\sigma_i^2 \sigma_j^2} x \right\} dx = x_i x_j \sqrt{\frac{2\pi\sigma_i^2 \sigma_j^2}{a}} \exp \left\{ \frac{b_x^2}{2a\sigma_i^2 \sigma_j^2} \right\}.$$

Nach dem Zusammenfassen dieser Ausdrücke und Umformen ergibt sich die Beziehung (5.3-43).

Modellierung von Oberflächen mit Diskontinuitäten

Von der Fakultät für Forst-, Geo- und Hydrowissenschaften
der Technischen Universität Dresden
zur Verleihung der Lehrfähigkeit
für das Fachgebiet "Geodäsie"
genehmigte Habilitationsschrift

vorgelegt von

Dr.-Ing. Andrzej Borkowski

München 2004

Verlag der Bayerischen Akademie der Wissenschaften
in Kommission beim Verlag C. H. Beck

Adresse der Deutschen Geodätischen Kommission:

Deutsche Geodätische Kommission

Marstallplatz 8 • D – 80 539 München

Telefon (089) 23 031 113 • Telefax (089) 23 031 – 283/– 100

E-mail hornik@dgfi.badw.de • <http://dgk.badw.de/>

Gutachter: Prof. Dr.-Ing.habil. S. Meier (TU Dresden)
o.Univ.Prof. Dipl.-Ing. Dr.techn. K. Kraus (TU Wien)
Prof. Dr.-Ing. Dr.-Ing.E.h. Dr.-Ing.E.h. K.-R. Koch (Univ. Bonn)

Tag der Einreichung: 26.01.2004

Tag des Habilitationskolloquiums: 30.06.2004

© 2004 Deutsche Geodätische Kommission, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,
die Veröffentlichung oder Teile daraus auf photomechanischem Wege (Photokopie, Mikrokopie) zu vervielfältigen