

Flächenhafte Plausibilisierung von Peildaten

Harry Wirth

Flächenhafte Plausibilisierung von Peildaten

Harry Wirth

Inhalt:

1	EINLEITUNG.....	3
2	MATHEMATISCHE MODELLBILDUNG (FUNKTIONALES MODELL)	3
3	PARAMETERSCHÄTZUNG UND STOCHASTISCHES MODELL	4
4	REALISIERUNG DES VERFAHRENS	5
4.1	DATENFLUß	5
4.2	ERWEITERUNG DURCH GEOMETRISCHE PARAMETER	7
5	AUSWERTESTRATEGIEN FÜR VERSCHIEDENE MEBWERTVERTEILUNGEN	8
5.1	LINIENPEILUNGEN MIT EINFACHSCHWINGERSYSTEMEN.....	8
5.2	FLÄCHENPEILUNGEN MIT MEHRFACHSCHWINGERSYSTEMEN UND FÄCHERLOTEN.....	11
5.3	ABSCHLUßKONTROLLE UND ZUSAMMENFASSUNG	13
6	SCHLUß.....	13
	ERGÄNZUNG ZUR FLÄCHENHAFTEN PLAUSIBILISIERUNG	15

Koblenz, im September 1999

Zusammenfassung

In der Gewässervermessung werden die Rohdatenmessungen (unbereinigte Koordinatentripel) oft mit erheblichem Zeitaufwand auf manuellem Wege plausibilisiert, wobei die manuelle Editierung meist durch graphische Verfahren unterstützt wird. Ein digitales Höhenmodell (DHM) des Gewässergrundes kann die Realität nur dann approximieren, wenn die Ausgangsdaten fehlerfrei sind. Die herkömmliche, sehr stark manuelle Vorgehensweise zur Fehlersuche und -bereinigung ist mit Massendaten sehr zeitaufwendig und anspruchsvoll. Vorgestellt wird ein Verfahren, das im wesentlichen vollautomatisch, sehr zuverlässig und ohne eine Vielzahl von Parametern die Meßwerte von groben Fehlern befreien kann.

1 Einleitung

Bei vielen bautechnischen Aufgabenstellungen im Gewässerausbau und der Unterhaltung, bzw. als Eingangsgröße für gewässerkundliche Modellrechnungen werden hochauflösende digitale Höhenmodelle (DHM) des Gewässergrundes benötigt. Meßtechnisch werden die Eingangsdaten für die DHM-Berechnung durch Abfahren von Meßprofilen mit Einzelschwingerecholoten oder auch mit Flächenvermessungssystemen digital erfaßt (Wirth 1995, 1998). Insbesondere die modernen Flächenvermessungssysteme erzeugen Meßwertdichten von leicht 100 Meßwerten pro Quadratmeter. Die Datenmengen werden innerhalb kurzer Meßdauer erzeugt. Bei der Datenbereinigung entsteht durch den oft relativ hohen Fehleranteil jedoch ein großer zeitlicher Aufwand, der durch Einsatz von automatischen, flächenhaften Plausibilisierungsverfahren reduziert werden kann.

Der Meßauftrag an Bord der Schiffe ist erst dann erfolgreich abgeschlossen, wenn die Qualität der Rohdaten geprüft ist. Der Prüfvorgang kann verschiedene Aspekte umfassen, z.B. ob die geforderte Meßwertdichte erreicht wurde oder ob die geforderte Meßgenauigkeit eingehalten wurde. Die eigentliche Datenauswertung im Büro hat dann mehr die Zielstellung, das Datenmaterial auszudünnen und geeignete Ergebnisdarstellungen zu erzeugen.

2 Mathematische Modellbildung (Funktionales Modell)

Der Gewässeruntergrund soll mathematisch durch eine stetige Fläche im dreidimensionalen kartesischen Raum beschrieben werden. Die Form des Gewässeruntergrundes wird dazu durch ganze rationale Polynome approximiert:

$$Z = F(X, Y) = \sum_{i=0}^n \sum_{j=0}^m a_{ij} \cdot X^i \cdot Y^j$$

$$n, m \in [0 \rightarrow \infty]$$

(2-1)

In Abhängigkeit von der Struktur des Gewässergrundes (Morphologie), von der gewünschten Anpassungsgenauigkeit des Modells und nicht zuletzt in Abhängigkeit von der gewünschten Größe eines gerade noch aufzudeckenden Fehlers muß über den Polynomgrad verfügt werden. Sind im Meßgebiet ausgeprägte Riffel, Bruchkan-

ten oder starke Gefällwechsel wie z. B. am Rande eines Kolkes vorhanden, so ist ein entsprechend hoher Polynomgrad zu wählen.

Bauwerke mit senkrechten Mauern (z.B. Schleusen, Kaimauern) oder Treppen unter Wasser bilden Sonderfälle, die ggf. gesondert behandelt werden müssen.

3 Parameterschätzung und stochastisches Modell

Die Aufgabe der Parameterschätzung besteht darin, in einem linearen Modell mit Hilfe der redundanten Meßwerte die Parameter des funktionalen Modells zu bestimmen. Im allgemeinen wird im Gauß-Markov-Modell unter der Annahme, daß die Beobachtungen normalverteilt und voneinander unabhängig sind die Methode der kleinsten Quadrate verwendet (Koch, 1987). Im Anschluß daran wird das Datenmaterial mit statistischen Hypothesentests auf eventuell vorkommende Ausreißer hin untersucht.

Normalerweise geht man davon aus, daß sich im Beobachtungsmaterial wenige grobe Fehler befinden. Die herkömmlichen Ausreißertests sind gut geeignet, wenn sich die Redundanzanteile gleichmäßig auf die Beobachtungen verteilen. Ausreißer verzerren jedoch die Parameterschätzung, so daß bei großen Ausreißern und vor allem bei einem hohen Anteil von groben Fehlern sowohl die Parameter falsch (verzerrt) geschätzt werden, als auch die Ausreißer nicht mehr richtig erkannt werden (Koch 1996).

Ausreißer bewirken eine Änderung der Verteilung der Beobachtungen. In der Hydrographie kann der Anteil von Ausreißern im Datenmaterial örtlich begrenzt (z.B. durch Schraubenwasser oder Bewuchs) sehr hoch sein. Durch diese ungünstige Fehlerverteilung versagen die herkömmlichen Parameterschätzungen und Ausreißertests. Um dennoch Fehler sicher aufdecken zu können, erscheint der Einsatz der robusten Parameterschätzung, sogenannten M-Schätzungen geboten. M-Schätzungen sind gegenüber groben Fehlern bzw. Änderungen der Verteilung der Stichprobe unempfindlich, so daß die Ausreißersuche auch noch bis nahezu 50 % Fehleranteil erfolgreich ist.

Die Parameterschätzung erfolgt im Gauß-Markov-Modell unter der Annahme, daß die Beobachtungen unkorreliert sind und gleiche Gewichte haben:

$$\underline{X} \cdot \underline{\beta} = E(\underline{y}) = \underline{y} + \underline{e} \quad \text{mit} \quad D(\underline{y}) = \sigma^2 \cdot \underline{I} \quad (3-1)$$

Hierin bedeuten \underline{X} die $n \times u$ Koeffizientenmatrix mit vollem Spaltenrang, $\underline{\beta}$ der $u \times 1$ Vektor unbekannter Parameter, \underline{y} der $n \times 1$ Vektor der Beobachtungen, \underline{e} der $n \times 1$ Vektor der Residuen. $E(\underline{y})$ bezeichnet den Erwartungswert und $D(\underline{y})$ die Kovarianzmatrix der Beobachtungen.

Die numerische Berechnung der Parameter erfolgt durch iterativ nach der üblichen Methode der kleinsten Quadrate. Die Beobachtungen werden dabei in Abhängigkeit der Größe der Residuen in jeder Iteration regewichtet. Die Gewichte w_i werden in der

Diagonalmatrix W zusammengefaßt. Die Iteration wird so lange durchgeführt, bis sich die Residuen, bzw. die Varianz der Gewichtseinheit nicht mehr ändern. Die Schätzwerte β der Iteration m erhält man aus:

$$\hat{\beta}^{(m)} = (X'W^{(m-1)}X)^{-1} X'W^{(m-1)}y \quad \text{und} \quad \hat{e}^{(m)} = X\hat{\beta}^{(m)} - y \quad (3-2)$$

Anschließend wird mit Hilfe der studentisierten Residuen für jede Beobachtung eine Testgröße berechnet und mit einem Signifikanzniveau von 95% die Hypothese getestet, ob in dieser Beobachtung ein Ausreißer vorliegt (Koch u. a., 1997). Die Ausreißersuche wird iterativ so lange durchgeführt, bis keine weiteren Ausreißer mehr erkannt werden.

4 Realisierung des Verfahrens

4.1 Datenfluß

Um ein großes Meßgebiet durch ein gut angepaßtes Polynom beschreiben zu können, müßte man bei bewegter Morphologie einen sehr hohen Polynomgrad wählen. Der Rechenaufwand steigt mit der Parameteranzahl jedoch exponentiell an. In der Praxis muß daher der wirtschaftlichste Kompromiß zwischen Anpassungsfähigkeit des Polynoms und kurzer Rechenzeit gesucht werden. Dazu wird das ursprüngliche Meßgebiet in ein regelmäßiges Raster unterteilt. Der die Fähigkeit des Schätzverfahrens, trotz hoher Fehleranzahl unverzerrte Schätzungen zu liefern, ist besonders wertvoll, weil in der Gewässervermessung grobe Fehler oft regional gehäuft und damit beispielsweise begrenzt in wenigen Maschen auftreten.

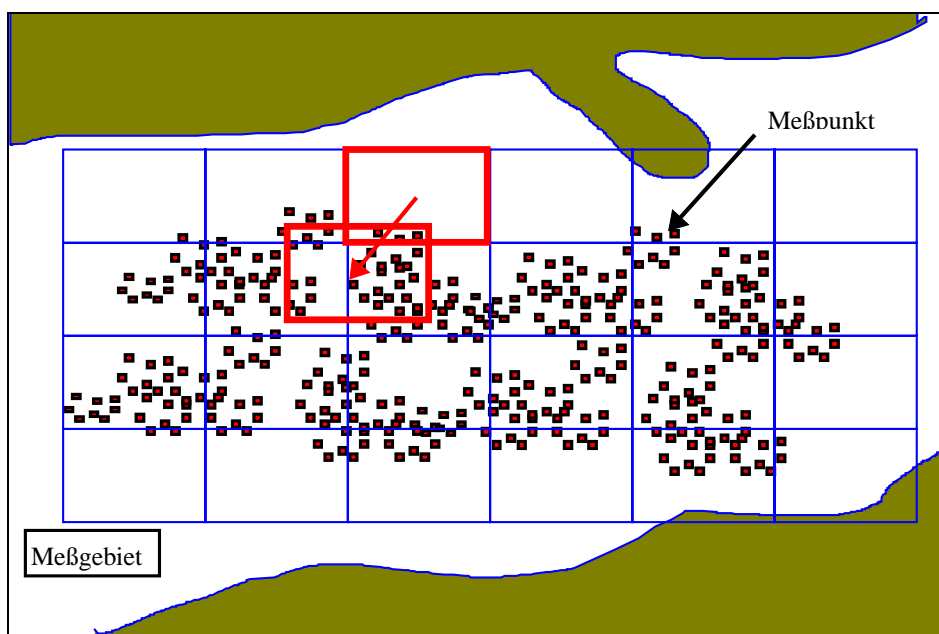


Abbildung 4.1-1: Translation der Maschen bei ungenügender Füllung

Das Verfahren wird numerisch instabil, wenn eine Rastermasche nicht regelmäßig, bzw. nur einseitig gefüllt ist. In diesen Fällen wird der Ort der Stichprobennahme ver-

ändert, indem benachbarte Rastermaschen in die robuste Parameterschätzung mit einbezogen werden (siehe Abbildung 4.1-1).

Die geschilderten Algorithmen wurden in einem Programmsystem (**Hydrographische Datenpräsentation, Auswertung und Plausibilisierung HyDAP**) realisiert und getestet. Die Rechenparameter sind über graphische Dialoge eingebbar.

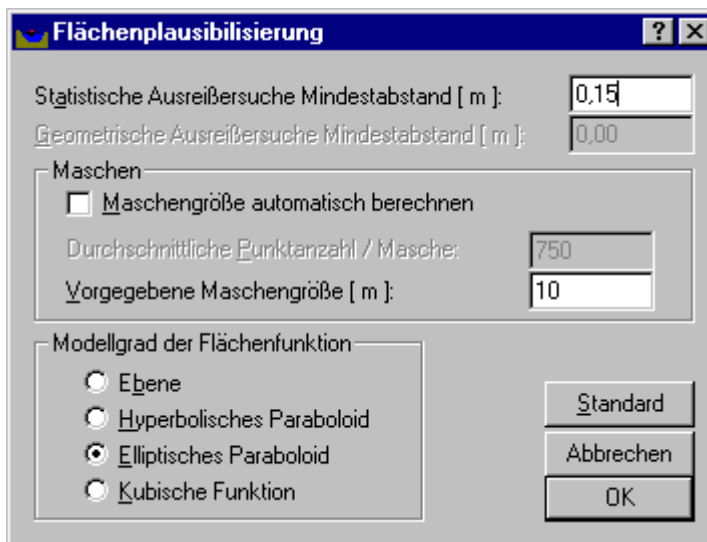
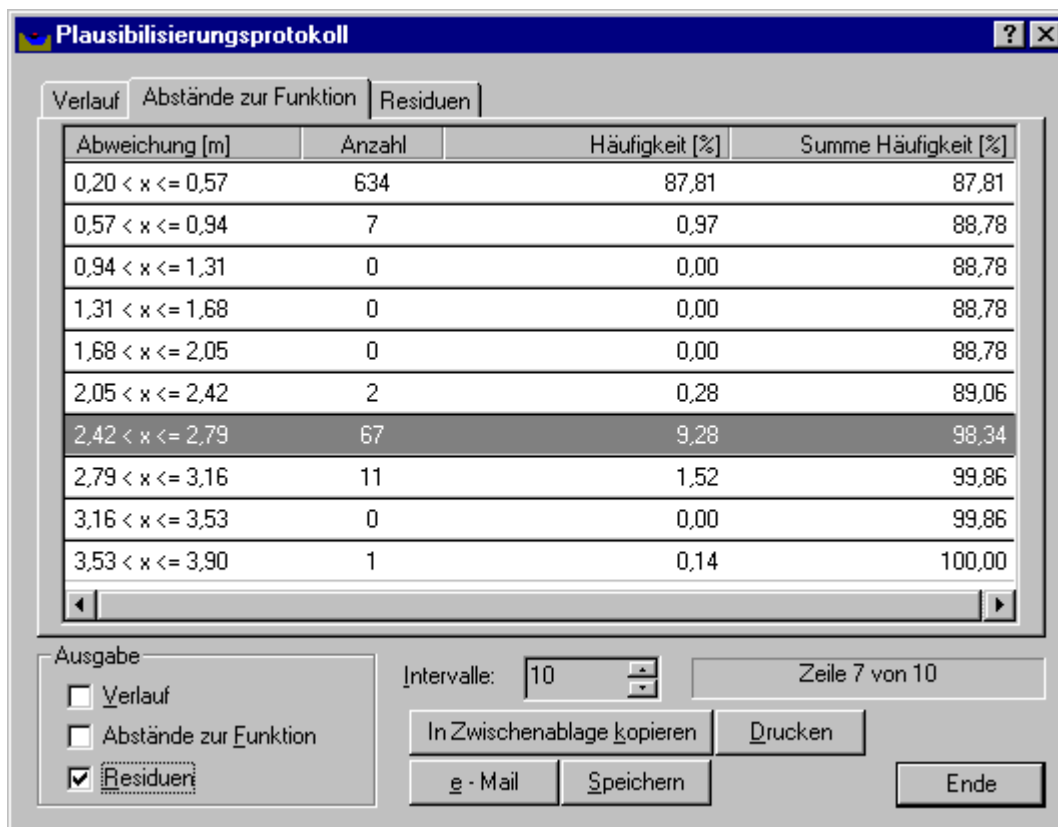
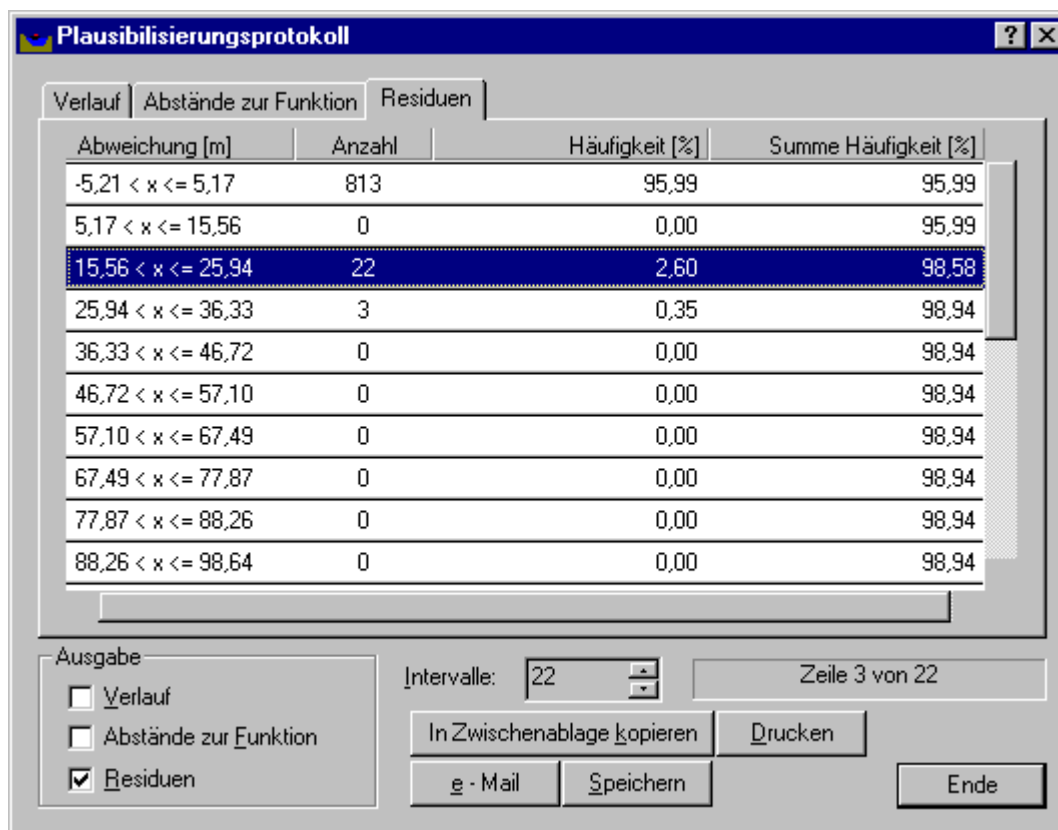


Abbildung 4.1-2: Dialog zur Plausibilisierung





The screenshot shows a software window titled 'Plausibilisierungsprotokoll' with three tabs: 'Verlauf', 'Abstände zur Funktion', and 'Residuen'. The 'Residuen' tab is active, displaying a table with the following data:

Abweichung [m]	Anzahl	Häufigkeit [%]	Summe Häufigkeit [%]
-5,21 < x <= 5,17	813	95,99	95,99
5,17 < x <= 15,56	0	0,00	95,99
15,56 < x <= 25,94	22	2,60	98,58
25,94 < x <= 36,33	3	0,35	98,94
36,33 < x <= 46,72	0	0,00	98,94
46,72 < x <= 57,10	0	0,00	98,94
57,10 < x <= 67,49	0	0,00	98,94
67,49 < x <= 77,87	0	0,00	98,94
77,87 < x <= 88,26	0	0,00	98,94
88,26 < x <= 98,64	0	0,00	98,94

Below the table, there is an 'Ausgabe' section with checkboxes for 'Verlauf', 'Abstände zur Funktion', and 'Residuen' (checked). To the right, there is an 'Intervalle:' field with a dropdown set to '22' and a 'Zeile 3 von 22' indicator. At the bottom, there are buttons for 'In Zwischenablage kopieren', 'Drucken', 'e-Mail', 'Speichern', and 'Ende'.

Abbildung 4.1-2: Statistik der Gesamtberechnung

Das Programm bearbeitet sequentiell alle Rastermaschen. Innerhalb jeder Rastermasche wird das im Kapitel 3 beschriebene Verfahren iterativ so lange durchgeführt, bis keine weiteren Ausreißer mehr erkannt werden. Gleichzeitig werden gefundene Ausreißer markiert. In jeder Iteration werden die verwendeten Parameter auf Signifikanz getestet. Beginnt die Berechnung beispielsweise mit 10 Parametern und der Test ergibt nur für 5 Parameter eine ausreichende Signifikanz, so wird die Berechnung in der nächsten Iteration nur mit diesen Parametern weitergeführt. Dieses Verfahren erspart in erster Linie unnötige Rechenzeit.

Das am Ende der Berechnung vorliegende Plausibilisierungsprotokoll gibt Informationen über den Verlauf der Berechnung, aber auch zur Verteilung der Ausreißer wieder. Auch die verbleibende Anzahl von Parametern kann dem Protokoll entnommen und bei zukünftigen Auswertungen weiter verwendet werden.

4.2 Erweiterung durch geometrische Parameter

Wenn der gewählte Polynomgrad die aktuelle Geländeoberfläche gut approximiert, ergibt sich eine kleine Varianz der Gewichtseinheit, so daß der Test schon bei kleinen Residuen sehr empfindlich reagiert und Meßwerte aufgrund der statistischen Signifikanz zu Ausreißern erklärt, obwohl sie noch innerhalb der Bandbreite der natürlichen Geländerauheit oder Meßgenauigkeit liegen. Vor allem in wenig strukturiertem Gelände würde wegen der kleinen Standardabweichung eine hohe Anzahl von Meßwerten -ggf. zu Unrecht- verworfen. Um die ungerechtfertigte Selektion von Meßwerten mit kleinen Residuen unterhalb der Größe der Bodenrauigkeit und der Polynom Anpassungsfähigkeit zu unterdrücken, wird in Ergänzung zu den statisti-

schen Tests ebenfalls überprüft, ob der potentielle statistisch gesicherte Ausreißer eine frei vorgebbare geometrische Schranke überschreitet (Mindestabstand).

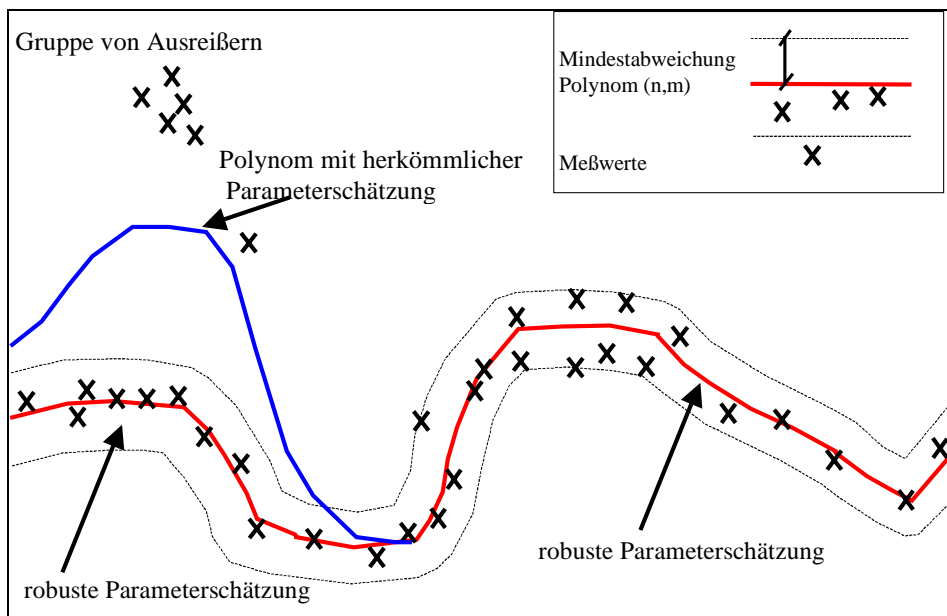


Abbildung 4.2-1: Robuste Parameterschätzung und Ausreißerdetektion

Die längs der Lotlinie gemessenen Abweichungen der Meßwerte von der ausgleichenden Fläche entsprechen den Residuen der Parameterschätzung. In extremen Hangneigungsbereichen entstehen so jedoch große Residuen, die statistisch als Ausreißer markiert werden. In diesen Punkten ist jedoch der lotrechte Abstand zur ausgleichenden Fläche relativ klein. Deshalb wird neben der statistischen Signifikanz als weiteres Kriterium geprüft, ob der kürzeste Abstand (das Lot) zur ausgleichenden Fläche ebenfalls die vorgegebene geometrische Schranke überschreitet.

Das Lot auf eine mehrparametrische Fläche kann nur in wenigen Fällen durch eine geschlossene Formel angegeben werden, deshalb wird es mit einem numerischen Verfahren iterativ berechnet. Der Abstand zwischen einem Punkt und der Flächenfunktion soll minimal werden (Lot). Mathematisch muß eine Extremwertaufgabe gelöst werden. Dazu muß die erste Ableitung der Funktion existieren und als notwendige Bedingung für ein Minimum Null werden. Die Nullstellen der Ableitung werden numerisch nach dem Newton - Verfahren, auch Tangentenverfahren berechnet (Gellert u.a. 1978).

5 Auswertestrategien für verschiedene Meßwertverteilungen

5.1 Linienpeilungen mit Einfachschwingersystemen

Bei einer hydrographischen Vermessung mit einem Einstrahlecholot werden i.d.R. in Abständen von mehreren Metern parallele Linien (Sollprofile) abgefahren. Aufgrund von Störgrößen wie Strömung und Schiffsverkehr werden diese Linien selten exakt eingehalten. Auf dem Profil ergibt sich eine relativ hohe Meßpunktdichte. Zwischen den Profilen ist aufgrund der Navigationsgenauigkeit oder auftrags- bzw. gebietsbe-

zogen ein Mindestabstand vorhanden, der weitaus größer ist als die profilinterne Datendichte.

Ein hoher Polynomgrad würde wegen der hier vorliegenden ungleichmäßigen Meßwertverteilung im Bereich zwischen den Profilen zur Schätzung von Flächenformen führen, die wenig mit der Realität gemein haben. Die Polynome sollten bei diesem Meßverfahren einen möglichst geringen Grad haben, der den Trend des Geländes aber dennoch richtig wiedergeben kann. Außerdem sollte sichergestellt sein, daß die Maschen zumindest Meßwerte aus zwei benachbarten Profilen enthalten. Dies ist in der Regel erfüllt, wenn die Größe der Rastermaschen ca. dem 2-fachen Profilstand entspricht. Infolgedessen ist die Anpassungsfähigkeit der Polynome in der Regel geringer als die Meßgenauigkeit. Kleinere Bodenerhebungen (1-2 dm) oder kurzweilige Undulationen in den Profilen werden oft nicht mehr modelliert.

Einzelne Steine werden durch die statistischen Tests als Ausreißer markiert. Um dies zu verhindern, muß die Mindestgröße der als Ausreißer anzusehenden Abweichungen relativ großzügig bemessen werden. Ein optimaler Wert wird nur durch Geländekenntnis auf Anhieb angegeben werden können. Die flächenhafte Plausibilisierung von Einzelprofilen ist dennoch vorteilhaft, weil z.B. größere Bereiche mit Schraubenwasser automatisch detektiert und beseitigt werden können.

Die folgenden Beispiele sollen die Kriterien für die Wahl von geeigneten Parametern und aufdeckbaren Fehlergrößen (Mindestabweichungen) für verschiedene Anwendungsbereiche verdeutlichen. Die obere Abbildung 5.1-1 zeigt ein einzelnes Querprofil, in welches künstlich 75 Fehlmessungen eingefügt wurden. Die linke Gruppe von Fehlmessungen steht beispielhaft für die Situation bei Schraubenwasser, wohingegen die am rechten Profilende eingebrachten Fehlmessungen die Situation mit ufernahem Bewuchs und/oder Echolotfehleinstellungen wiedergibt.

Das dargestellte Profil wurde gemeinsam mit anderen Profilen automatisch flächenhaft plausibilisiert, wobei der kleinste, statistisch gesicherte Ausreißer eine Mindestabweichung von 0,25 m haben mußte. Im Ergebnis (untere Profildarstellung) zeigt sich, daß auch gehäufte grobe Fehler sehr sicher detektiert werden. Bis auf drei Fehler in der Größenordnung kleiner 0,2 m wurden alle Fehlmessungen beseitigt. Das Ergebnis des automatischen Verfahrens entspricht dem graphisch unterstützten manuellen Verfahren.

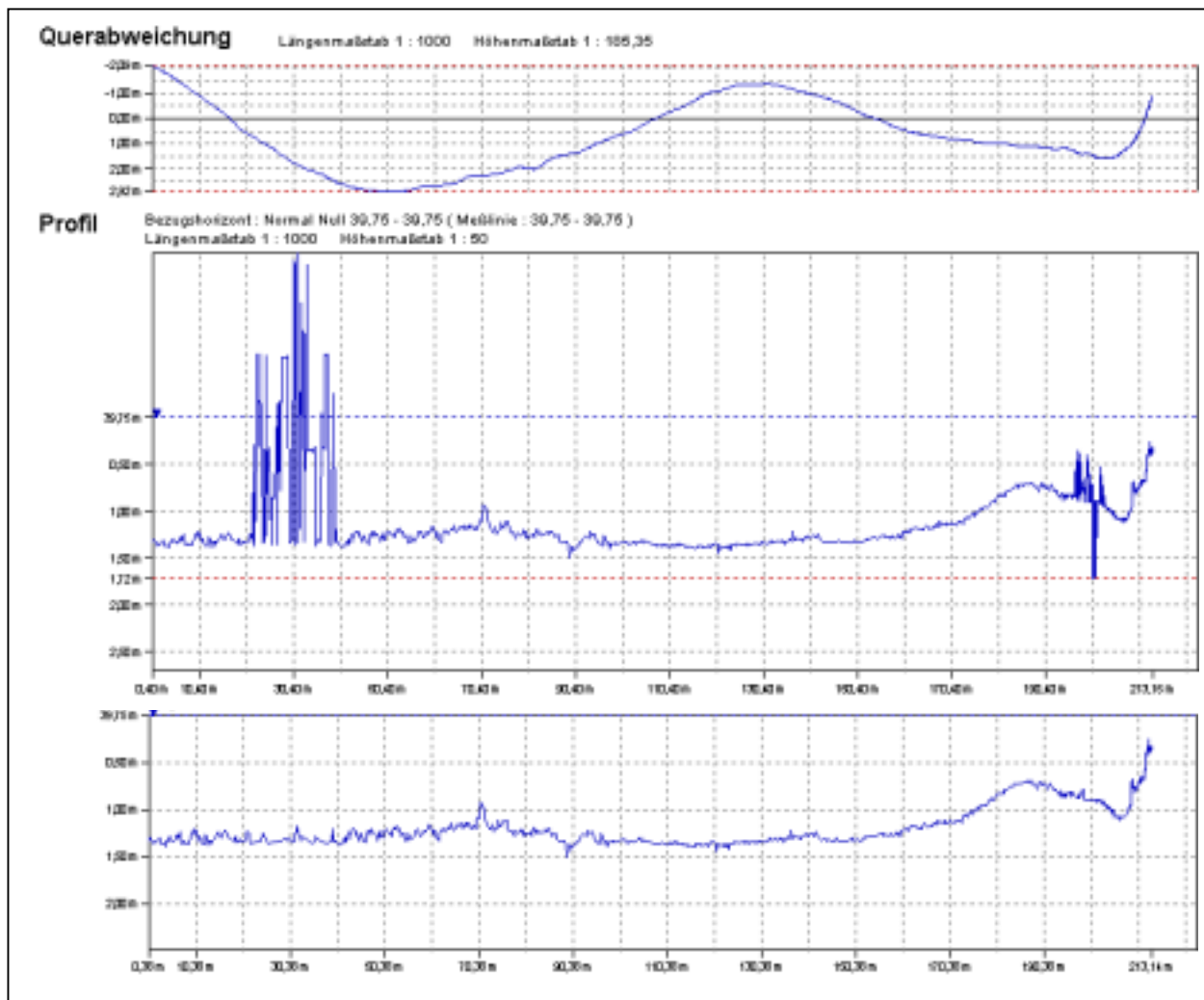


Abbildung 5.1-1: Vergleich eines Querprofils vor und nach der Plausibilisierung

Bei Linienpeilungen liegt in diesem Peilgebiet die Grenze der detektierbaren Fehler bei ca. 0,25 cm. Diese Größenordnung wird in der Natur auch schon von einzelnen Wasserbausteinen erreicht, so daß aufgrund der linienhaften Meßwertverteilung kleinere Fehlmessungen nicht von Objekten unterschieden werden können.

Die in der folgenden Tabelle angegebenen Parameter wurden bei der Auswertung von Peildaten (Rhein) empirisch bestimmt.

	1. Linienpeilung	2. Linienpeilung
Profilabstand	10 m	20 m
Meßwertdichte im Profil	Alle 12 cm	alle 12 cm
Morphologie	Frische Bruchkanten, z.B. Baggerei	keine Bruchkanten, Kiese und Sande
Maschenweite	25 m	50 m
Polynomgrad	Elliptisches Paraboloid (6 Parameter)	Hyperbolisches Paraboloid (4 Parameter)
Mindestabweichung	0,4 m	0,4 m

Tabelle 5.1-1: Parameterliste für die Plausibilisierung von flächenhaften Linienpeilungen

5.2 Flächenpeilungen mit Mehrfachschwingersystemen und Fächerloten

Bei sogenannten Flächenpeilungen sind die Meßwerte im wesentlichen ohne größere Lücken über den Gewässergrund verteilt. Durch die teilweise Überdeckung der Meßstreifen ergeben sich lokal unterschiedliche Meßwertdichten. Unter diesen Voraussetzungen kann nun versucht werden, mit Hilfe eines hochauflösenden Polynoms eine möglichst große Fläche zu plausibilisieren. Gegensätzlich wirkt die Forderung nach einer möglichst kleinen Mindestabweichung. Empirisch haben sich für ein Testgebiet an der Donau folgende Parameter ergeben:

	1. Flächenpeilung	2. Flächenpeilung
Profilabstand	ca. 6fache Wassertiefe	ca. 6-fache Wassertiefe
Meßwertdichte	100/m ²	100/m ²
Morphologie	Bruchkanten, Korngrößen bis 30 cm, Kolke, Buhnenköpfe	keine Bruchkanten, Korngrößen < 5-7 cm
Maschenweite	7 m	10 m
Polynomgrad	Fläche 3. Ordnung (10 Parameter)	Fläche 2. Ordnung (6 Parameter)
Mindestabweichung	0,3 m	0,15 m

Tabelle 5.2-1: Parameterliste für die Plausibilisierung von Fächerlotdaten

Ein digitales Höhenmodell (DHM) des Gewässergrundes kann die Realität nur richtig wiedergeben, wenn die Ausgangsdaten fehlerfrei sind. Dies verdeutlichen die nachfolgenden Abbildungen. Beide Abbildungen zeigen ein beleuchtetes Relief derselben Datenmenge. Als Grundinformation wurde ein regelmäßiges Höhenmodell mit einer Gitterweite von 0,8m berechnet. In der ersten Reliefdarstellung erscheinen sogenannte Artefakte, die auf Meßfehler zurückgeführt werden können.

Im Gegensatz zum ersten DHM wurden die Daten für das zweite DHM zuvor automatisch flächenhaft plausibilisiert. In der Darstellung sind keine Fehler mehr zu erkennen.

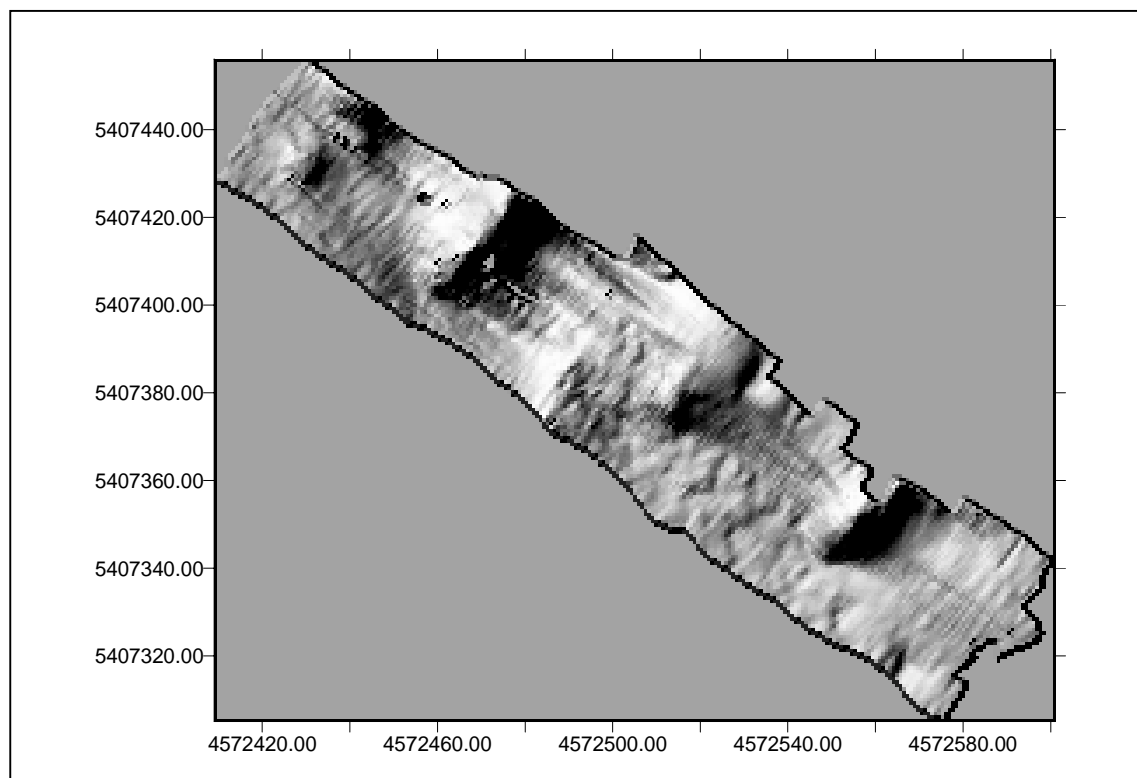
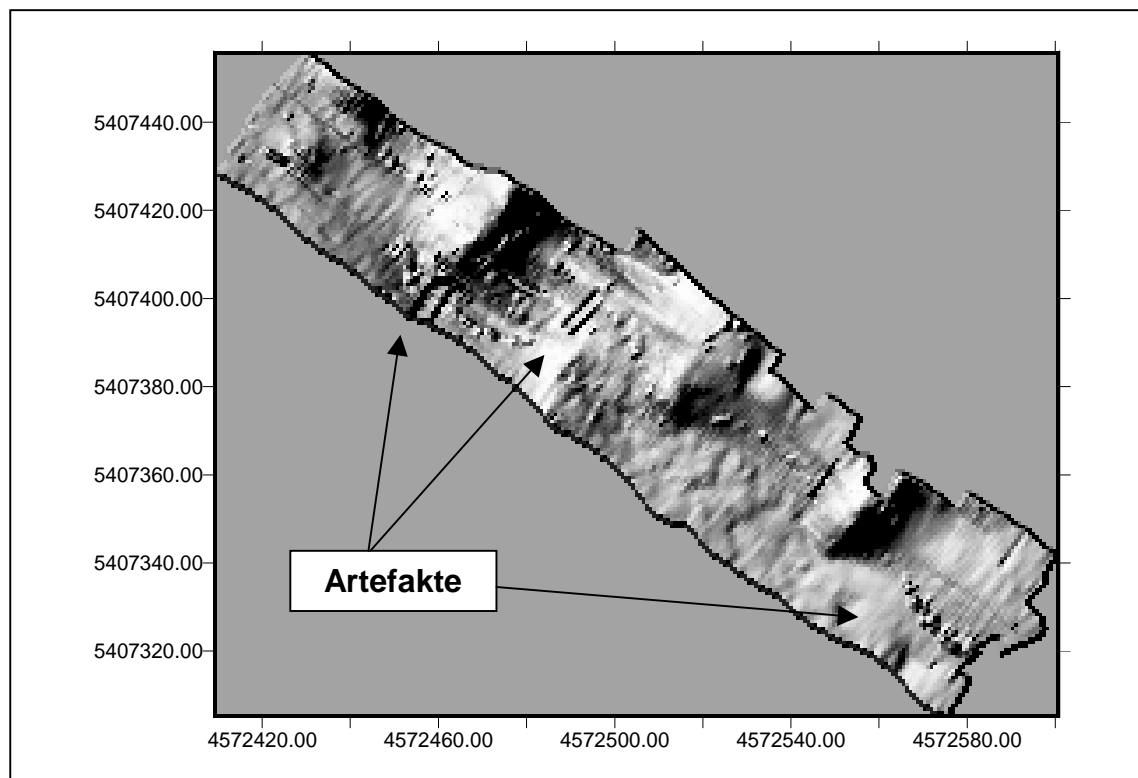


Abbildung 5.2-1: Vergleich eines Reliefs ohne und mit Plausibilisierung

5.3 Abschlußkontrolle und Zusammenfassung

Die oben genannten Beispiele zeigen mögliche Auswertestrategien auf, die in Abhängigkeit vom Meßgebiet und Meßverfahren erarbeitet wurden. In der Praxis läuft das vorgestellte Verfahren nach Programmstart vollautomatisch ab. Im Anschluß an den Rechengang muß der Nutzer jedoch mit einem Auswertesystem graphisch unterstützt den Erfolg seiner Einstellungen überprüfen und ggf. mit geänderten Einstellungen die Meßwerte erneut plausibilisieren. Einmal für ein Meßgebiet gefundene optimale Einstellungen können bei Folgemessungen wieder verwendet werden.

Obwohl zum Schluß der Bearbeitung zumindest beim ersten Mal noch ein Prüfvorgang nachgeschaltet werden muß, ergeben sich insgesamt gegenüber der manuellen Methodik bei großen Datenmengen wesentliche Zeitersparnisse. Außerdem werden die Qualitätsanforderungen und Auswahlkriterien in einem digitalen Protokoll dokumentiert. Durch den geringen Zeitbedarf ist dieses Verfahren für eine erste Sofortauswertung an Bord der Schiffe geeignet und kann auch von weniger qualifiziertem Personal ausgeführt werden.

6 Schluß

Vorgestellt wurde eine Entwicklung, die die Plausibilisierung von Tiefenmeßwerten direkt im Anschluß an die Meßwerterfassung bereits an Bord ermöglicht, so daß Auswertungen für Sofortmaßnahmen zur Verkehrssicherung auf verbesserten, bereinigten Daten aufbauen können und die Sicherheit der Aussagen gesteigert wird.

Die neuartigen automatisierten Methoden führen zur Zeitersparnis und bilden die Grundlage für die Dokumentation und geprüfte Einhaltung von Qualitätsstandards, so daß hier individuell für jedes Meßergebnis (Rohdaten) die erreichte Genauigkeit nachweisbar, bzw. angebbar ist. Im Vergleich zur bisherigen Erfassungsstrategie, wo den Meßwerten empirisch aufgrund des eingesetzten Meßverfahrens eine Genauigkeit zugeordnet wurde, ist hiermit ein weiterer Schritt zur Einhaltung von Qualitätsstandards erfolgt.

Literaturhinweise

- Gellert, W., Kästner, H. Neuber, S. (1978): *Fachlexikon ABC Mathematik*, Verlag Harri Deutsch, Frankfurt.
- Koch, K.R. (1987): *Parameterschätzung in linearen Modellen*, Dümmler, Bonn.
- Koch, K.R. (1996): *Robuste Parameterschätzung*, AVN Heft 1/1996, Wichmann Verlag, Heidelberg.
- Koch, K.R., Yuanxi Yang (1998): *Konfidenzbereiche und Hypothesentests für robuste Parameterschätzungen*, ZfV Heft 1/1998, Wittwer Verlag, Stuttgart.
- Wirth, H. (1995): *Gewässervermessung im Binnenbereich der BWaStr mit DGPS-Ortung*, DVW Schriftenreihe 14/1995, Wittwer Verlag, Stuttgart
- Wirth, H. (1998): *Hydrographisches Vermessungssystem HYMAS*, DVW Schriftenreihe 29/1998, Wittwer Verlag, Stuttgart

Anschrift des Verfassers

Dipl.-Ing. Harry Wirth
Bundesanstalt für Gewässerkunde
Postfach 200253
56002 Koblenz

Ergänzung zur Flächenhaften Plausibilisierung nach Harry Wirth

Im Folgenden wird der Datenfluß in der Software der BfG beschrieben. Da jede Realisierung in einer anderen Auswertesoftware auf einem anderen Datenmodell und anderer Datenorganisation basiert, wird hier lediglich beschrieben, wie der Algorithmus funktioniert, ein vollständiger Abdruck des Quellcodes macht wenig Sinn.

Das Modell wird in der Veröffentlichung „Flächenhafte Plausibilisierung von Peildaten“ Schriftenreihe 37/1999 des DVW, Verlag Wittwer, beschrieben. Vor allem in der dort angegebenen Literatur (Koch), bitte ich die umfangreiche Schilderung der zugrundeliegenden Theorie nachzulesen!

Die Form des Gewässeruntergrundes wird durch ganze rationale Polynome approximiert:

$$Z = F(X, Y) = \sum_{i=0}^n \sum_{j=0}^m a_{ij} \cdot X^i \cdot Y^j$$

$$n, m \in [0 \rightarrow \infty]$$

Die Parameterschätzung der Koeffizienten a_{ij} der Polynome erfolgt im Gauß-Markov-Modell unter der Annahme, daß die Beobachtungen unkorreliert sind und gleiche Gewichte haben:

$$\underline{X} \cdot \underline{\beta} = E(\underline{y}) = \underline{y} + \underline{e} \quad \text{mit} \quad D(\underline{y}) = \sigma^2 \cdot \underline{I} \quad (3-1)$$

Hierin bedeuten \underline{X} die $n \times u$ Koeffizientenmatrix mit vollem Spaltenrang, $\underline{\beta}$ der $u \times 1$ Vektor unbekannter Parameter, \underline{y} der $n \times 1$ Vektor der Beobachtungen, \underline{e} der $n \times 1$ Vektor der Residuen. $E(\underline{y})$ bezeichnet den Erwartungswert und $D(\underline{y})$ die Kovarianzmatrix der Beobachtungen.

Die numerische Berechnung der Parameter erfolgt iterativ nach der üblichen Methode der kleinsten Quadrate. Die Beobachtungen werden dabei in Abhängigkeit der Größe der Residuen in jeder Iteration regewichtet. Die Gewichte w_i werden in der Diagonalmatrix W zusammengefaßt.

$$W_{ii}^{(m)} = \frac{c \cdot \sigma}{|\hat{e}_i^{(m)}|} \quad \text{für} \quad |\hat{e}_i^{(m)}| > c \cdot \sigma$$

Und

$$W_{ii}^{(m)} = 1 \quad \text{für} \quad |\hat{e}_i^{(m)}| \leq c \cdot \sigma$$

Die Gewichtung wird um so empfindlicher, je kleiner man den Wert $c \cdot \sigma$ wählt. Die BfG ist der Ansicht, daß für die Fehlersuche der Wert $2 \cdot 0,3 \text{ m} = 0,6 \text{ m}$ geeignet ist. Die Iteration wird so lange durchgeführt, bis sich die Residuen, bzw. die

Varianz der Gewichtseinheit nicht mehr ändern. Die Schätzwerte β der Iteration m erhält man aus:

$$\hat{\beta}^{(m)} = (X^T W^{(m-1)} X)^{-1} X^T W^{(m-1)} y \quad \text{und} \quad \hat{e}^{(m)} = X \hat{\beta}^{(m)} - y \quad (3-2)$$

Anschließend wird mit Hilfe der studentisierten Residuen für jede Beobachtung eine Testgröße berechnet und mit einem Signifikanzniveau von 95% die Hypothese getestet, ob in dieser Beobachtung ein Ausreißer vorliegt (Koch u. a., 1997). Die Ausreißersuche wird iterativ so lange durchgeführt, bis keine weiteren Ausreißer mehr erkannt werden.

Dabei ist folgendes zu beachten:

Die Meßwerte werden vorab in quadratische Maschen sortiert. Der Algorithmus wird jeweils nur auf eine Masche angewendet. Der Mittelpunkt der Masche wird berechnet.

Im ersten Schritt werden die Extremwerte der Lagekoordinaten pro Masche berechnet und geprüft, ob alle Quadranten der Masche gefüllt sind. Wenn nur 2 Quadranten gefüllt sind, wird versucht, den Mittelpunkt der Masche so zu verschieben, daß alle Werte der ursprünglichen Masche gerade noch in die neuen Maschengrenzen fallen. Die neuen Meßwerte werden in die Stichprobe aufgenommen, wenn sie nicht als Ausreißer markiert sind. Wenn zu wenig Meßwerte in der endgültigen Masche vorhanden sind, werden sie als ungültig markiert. Wir prüfen dazu ab, ob die Redundanz (Anzahl Meßwerte – Anzahl Unbekannte) ≥ 10 ist.

Die Lagekoordinaten werden um die Werte der Mittelpunktskordinaten der Masche gekürzt.

Zu Beginn der inneren Schleife wird die Koeffizientenmatrix X aufgestellt (xko und yko sind die Koordinaten):

```
// Koeffizientenmatrix aufstellen
```

```
//printf("Iteration Nr. : %d\n   Unbekannte = %6d\n   Messdaten = %6d",iteration,nu,n);
    for( i = 0 ; i < n ; ++i )
    {
        X( i, 0 ) = 1.0;
        X( i, 1 ) = xko[ i ];
        X( i, 2 ) = yko[ i ];
        switch( nu )
        {
            case 3:
                break;

            case 4:
                X( i, 3 ) = yko[ i ] * xko[ i ];
                break;
```


case 6:

```
X( i, 3 ) = yko[ i ] * xko[ i ];  
X( i, 4 ) = pow( xko[ i ], 2 );  
X( i, 5 ) = pow( yko[ i ], 2 );  
break;
```

case 10:

```
X( i, 3 ) = yko[ i ] * xko[ i ];  
X( i, 4 ) = pow( xko[ i ], 2 );  
X( i, 5 ) = pow( yko[ i ], 2 );  
X( i, 6 ) = X( i, 4 ) * X( i, 2 );  
X( i, 7 ) = X( i, 5 ) * X( i, 1 );  
X( i, 8 ) = X( i, 4 ) * X( i, 1 );  
X( i, 9 ) = X( i, 5 ) * X( i, 2 );  
break;
```

Damit Rechenungenauigkeiten vermieden werden, werden die Spalten der Matrix X homogenisiert:

```
ColFactor[ 0 ] = 1.0; // Spaltenfaktoren  
bestimmen  
ColFactor[ 1 ] = 1.0 / ( Maschengröße );  
ColFactor[ 2 ] = 1.0 / ( Maschengröße );  
for( i = 3 ; i < nu ; ++i )  
{  
    ColFactor[ i ] = 0.0;  
  
    for( j = 0 ; j < n ; ++j )  
        ColFactor[ i ] += X( j, i );  
    ColFactor[ i ] = n / ColFactor[ i ];  
}  
  
for( i = 1 ; i < nu ; ++i ) // Spalten egalisieren  
{  
    for( j = 0 ; j < n ; ++j )  
        X( j, i ) *= ColFactor[ i ];  
}
```

Dadurch werden die Parameter β ebenfalls um den Faktor ColFactor[i] verändert!

Es folgt die innere Hauptschleife, in der iterativ die Parameter β berechnet werden. Am Ende jeder Iteration wird die Signifikanz der Parameter β getestet und die Spalten der Koeffizientenmatrix X umsortiert! Dies bewirkt, dass Rechenzeit gespart wird und dass die Polynome nicht überparametrisiert sind. Die jeweils verwendete Anzahl von Parametern wird im Protokoll vermerkt. der Benutzer kann daran sehen, ob der Gewässergrund sehr bewegt ist und er besser einen geringeren Modellgrad gewählt hätte, oder ob er eine zu große Maschenweite gewählt hat (Standardabweichung größer als das normale Meßrauschen).

Wenn die Iteration konvergiert ist, wird ein Ausreißertest nach Koch durchgeführt. Alle Meßwerte, die mit der Signifikanz von 95% als potentielle Ausreißer in Frage kommen, werden zusätzlich mit einem geometrischen Kriterium überprüft. Es wird geprüft, ob das Lot auf die Fläche einen Mindestabstand überschreitet.

Das Lot auf die Fläche wird nach dem Newtonschen Nullstellenverfahren berechnet. Der Vektor β wird hier als `betadach ()` bezeichnet:

```
////////////////////////////////////
//                                     BerechneAbstandZurFunktion
//
// Aufgabe: siehe Namen
// In:      int nu,
//          const SimpleArray< int > &Unbekannte
//          const SimpleArray< double > &betadach
//          double xko
//          double yko
//          double zko
//          double *xFuss
//          double *yFuss
// Out:     double
////////////////////////////////////

double Plausibilisierung::BerechneAbstandZurFunktion( int nu, const SimpleArray<
int > &Unbekannte,

               const SimpleArray< double > &betadach,

               double xko, double yko, double zko,

               double *xFuss, double *yFuss ) const
{
double          Abstand, mue, zFuss, fx, fxx, fxy, fyx, fy, fyy,
               G1, G2, G1x, G1y, G2x, G2y, abbruch;
Matrix          a( 2, 2 );
SimpleArray< double > b( 2 ), xm0( 2 ), xm1( 2 );
int             i, j;

Abstand = 0.0;

if( nu == 3 )
{
mue = ( zko - betadach[ 0 ] - betadach[ 1 ] * xko - betadach[ 2 ] * yko ) /
      ( 1 + pow( betadach[ 1 ], 2 ) + pow( betadach[ 2 ], 2 ) );
*xFuss = xko + mue * betadach[ 1 ];
*yFuss = yko + mue * betadach[ 2 ];
zFuss = FunktionswertBerechnen( Unbekannte, betadach, nu, *xFuss, *yFuss );
}
```

```
Abstand = sqrt( pow(( *xFuss ) - xko, 2 ) + pow(( *yFuss ) - yko, 2 ) + pow( zFuss -
zko, 2 ));
}
else
{
xm0[ 0 ] = xko;
xm0[ 1 ] = yko;
do
{
xm0[ 1 ]);
zFuss = FunktionswertBerechnen( Unbekannte, betadach, nu, xm0[ 0 ],

// 1. Ableitung der Fkt. nach x berechnen
fx = 0.0;
for( i = 0 ; i < nu ; ++i )
{
j = Unbekannte[ i ];

switch( j )
{
case 1:
fx += betadach[ i ];
break;

case 3:
fx += betadach[ i ] * xm0[ 1 ];
break;

case 4:
fx += 2 * betadach[ i ] * xm0[ 0 ];
break;

case 6:
fx += 2 * betadach[ i ] * xm0[ 0 ] * xm0[ 1 ];
break;

case 7:
fx += betadach[ i ] * xm0[ 1 ] * xm0[ 1 ];
break;

case 8:
fx += 3 * betadach[ i ] * xm0[ 0 ] * xm0[ 0 ];
break;
}
}

// 2. Ableitung der Fkt. nach x berechnen
fxx = 0.0;
for( i = 0 ; i < nu ; ++i )
```

```
{
j = Unbekannte[ i ];

switch( j )
{
case 4:
    fxx += 2 * betadach[ i ];
    break;

case 6:
    fxx += 2 * betadach[ i ] * xm0[ 1 ];
    break;

case 8:
    fxx += 6 * betadach[ i ] * xm0[ 0 ];
    break;
}
}

// 2. gemischte Ableitung der Fkt. nach x berechnen
fxy = 0.0;
for( i = 0 ; i < nu ; ++i )
{
j = Unbekannte[ i ];

switch( j )
{
case 3:
    fxy += betadach[ i ];
    break;

case 6:
    fxy += 2 * betadach[ i ] * xm0[ 0 ];
    break;

case 7:
    fxy += 2 * betadach[ i ] * xm0[ 1 ];
    break;
}
}
fyx = fxy;

// 1. Ableitung der Fkt. nach y berechnen
fy = 0.0;
for( i = 0 ; i < nu ; ++i )
{
j = Unbekannte[ i ];

switch( j )
```

```
{
case 2:
    fy += betadach[ i ];
    break;

case 3:
    fy += betadach[ i ] * xm0[ 0 ];
    break;

case 5:
    fy += 2 * betadach[ i ] * xm0[ 1 ];
    break;

case 6:
    fy += betadach[ i ] * xm0[ 0 ] * xm0[ 0 ];
    break;

case 7:
    fy += 2 * betadach[ i ] * xm0[ 0 ] * xm0[ 1 ];
    break;

case 9:
    fy += 3 * betadach[ i ] * xm0[ 1 ] * xm0[ 1 ];
    break;
}
```

// 2. Ableitung der Fkt. nach y berechnen

```
fyy = 0.0;
for( i = 0 ; i < nu ; ++i )
{
    j = Unbekannte[ i ];

    switch( j )
    {
        case 5:
            fyy += 2 * betadach[ i ];
            break;

        case 7:
            fyy += 2 * betadach[ i ] * xm0[ 0 ];
            break;

        case 9:
            fyy += 6 * betadach[ i ] * xm0[ 1 ];
            break;
    }
}
```

```
G1 = xm0[ 0 ] - xko + ( zFuss - zko ) * fx;  
G2 = xm0[ 1 ] - yko + ( zFuss - zko ) * fy;  
  
G1x = 1.0 + fx * fx + ( zFuss - zko ) * fxx;  
G1y = fy * fx + ( zFuss - zko ) * fxy;  
  
G2x = fx * fy + ( zFuss - zko ) * fyx;  
G2y = 1.0 + fy * fy + ( zFuss - zko ) * fyy;  
  
a( 0, 0 ) = G1x;  
a( 0, 1 ) = G1y;  
a( 1, 0 ) = G2x;  
a( 1, 1 ) = G2y;  
  
a.Invert();  
  
b[ 0 ] = -G1;  
b[ 1 ] = -G2;  
  
xm1 = a * b;  
xm0[ 0 ] += xm1[ 0 ];  
xm0[ 1 ] += xm1[ 1 ];  
  
abbruch = fabs( xm1[ 0 ] );  
if( fabs( xm1[ 1 ] ) > abbruch )  
    abbruch = fabs( xm1[ 1 ] );  
}  
while( abbruch >= 0.001 );  
  
zFuss = FunktionswertBerechnen( Unbekannte, betadach, nu, xm0[ 0 ], xm0[  
1 ] );  
Abstand = sqrt( pow(( xm0[ 0 ] ) - xko, 2 ) + pow(( xm0[ 1 ] ) - yko, 2 ) +  
pow( zFuss - zko, 2 ));  
  
*xFuss = xm0[ 0 ];  
*yFuss = xm0[ 1 ];  
}  
return Abstand;  
}  
  
/////////////////////////////////////  
//                               FunktionswertBerechnen  
//  
// Aufgabe:  
// In:           const SimpleArray< int > &Unbekannte,  
//               const SimpleArray< double > &betadach  
//               int nu  
//               double xFuss  
//               double yFuss
```

```
//      Out:          double
////////////////////////////////////

double Plausibilisierung::FunktionswertBerechnen( const SimpleArray< int >
&Unbekannte, const SimpleArray< double > &betadach, int nu, double xFuss, double
yFuss ) const
{
int          i, j;
double      x;

x = 0.0;

for( i = 0 ; i < nu ; ++i )
    {
    j = Unbekannte[ i ];

    switch( j )
        {
        case 0:
            x += betadach[ i ];
            break;

        case 1:
            x += betadach[ i ] * xFuss;
            break;

        case 2:
            x += betadach[ i ] * yFuss;
            break;

        case 3:
            x += betadach[ i ] * xFuss * yFuss;
            break;

        case 4:
            x += betadach[ i ] * pow( xFuss, 2 );
            break;

        case 5:
            x += betadach[ i ] * pow( yFuss, 2 );
            break;

        case 6:
            x += betadach[ i ] * pow( xFuss, 2 ) * yFuss;
            break;

        case 7:
            x += betadach[ i ] * pow( yFuss, 2 ) * xFuss;
            break;
        }
    }
}
```

```
        case 8:
            x += betadach[ i ] * pow( xFuss, 3 );
            break;

        case 9:
            x += betadach[ i ] * pow( yFuss, 3 );
            break;
    }
}
```

return x;
}

Wenn ein Meßwert den Mindestabstand überschreitet, wird er als Ausreißer in den Originaldaten markiert und aus dem Vektor $y()$ [Beobachtungsvektor] entfernt.

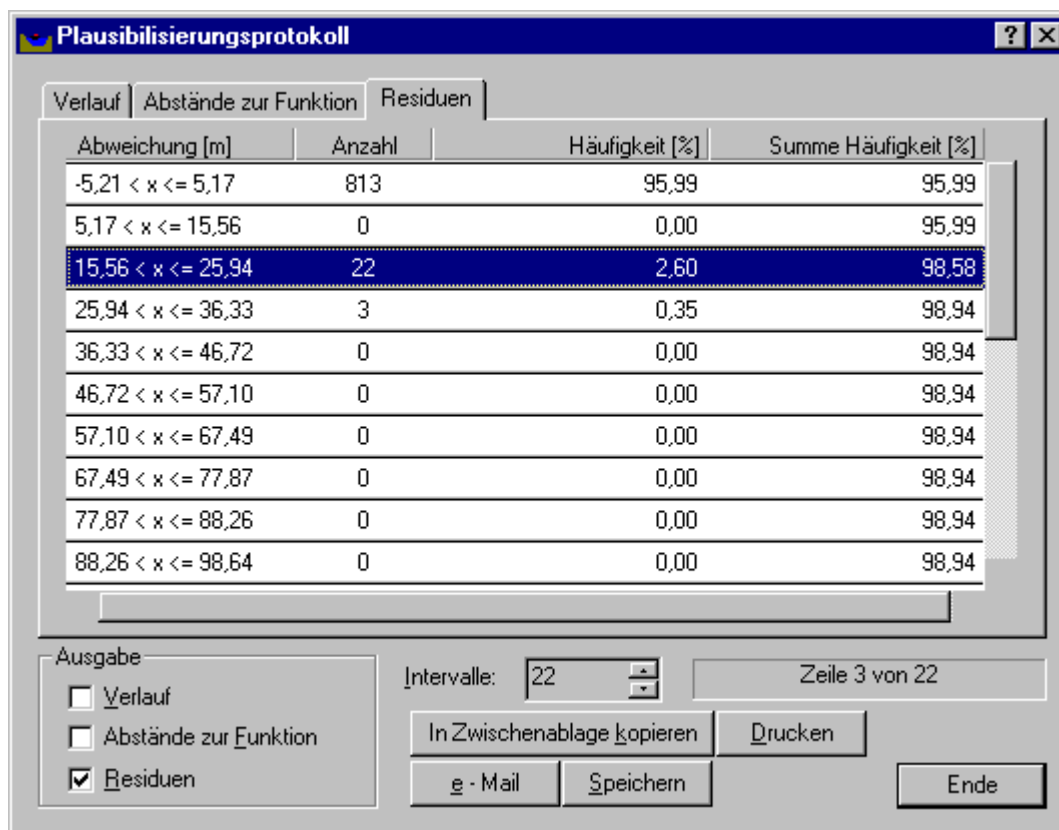
Die äußere Schleife beginnt nun wieder mit dem Aufstellen der Koeffizientenmatrix X .

Das Ganze Verfahren wird so lange fortgeführt, bis in der Masche keine Ausreißer mehr gefunden werden.

Der ganze Rechengang und das Ergebnis wird in einem Protokoll dokumentiert:

Dabei wird in einem Teil des Protokolls der Verlauf des Rechengangs wiedergegeben. Am Anfang stehen die verwendeten Dateien, dann die Steuerungsparameter für die Software, dann für jede Masche die in jeder Iteration erreichte Standardabweichung, die Anzahl der verbliebenen Parameter und die Anzahl der Ausreißer, sowie der Betrag der größten Abweichung.

Weitere Protokollbestandteile sind z.B. eine Statistik der gefundenen Ausreißer, wobei nach Residuen oder Abstand zur Funktion unterschieden wird. Die Liste der Protokollbestandteile sollte ggfs. erweitert werden.

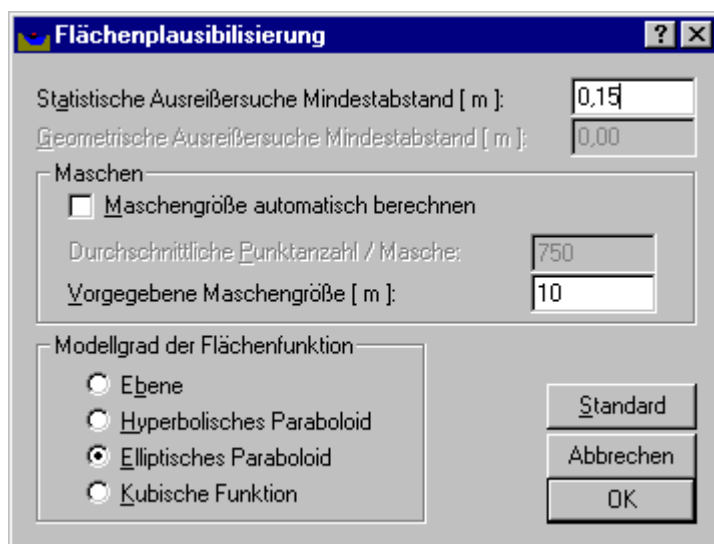


Abweichung [m]	Anzahl	Häufigkeit [%]	Summe Häufigkeit [%]
-5,21 < x <= 5,17	813	95,99	95,99
5,17 < x <= 15,56	0	0,00	95,99
15,56 < x <= 25,94	22	2,60	98,58
25,94 < x <= 36,33	3	0,35	98,94
36,33 < x <= 46,72	0	0,00	98,94
46,72 < x <= 57,10	0	0,00	98,94
57,10 < x <= 67,49	0	0,00	98,94
67,49 < x <= 77,87	0	0,00	98,94
77,87 < x <= 88,26	0	0,00	98,94
88,26 < x <= 98,64	0	0,00	98,94

Ausgabe:
 Verlauf
 Abstände zur Funktion
 Residuen

Intervalle: 22 Zeile 3 von 22
 In Zwischenablage kopieren Drucken
 e-Mail Speichern Ende

Als Benutzeroberfläche müssen die Parameter der Plausibilisierung in einem Dialog eingegeben werden können, z.B:



Flächenplausibilisierung

Statistische Ausreißersuche Mindestabstand [m]: 0,15
 Geometrische Ausreißersuche Mindestabstand [m]: 0,00

Maschen
 Maschengröße automatisch berechnen
 Durchschnittliche Punktzahl / Masche: 750
 Vorgegebene Maschengröße [m]: 10

Modellgrad der Flächenfunktion
 Ebene
 Hyperbolisches Paraboloid
 Elliptisches Paraboloid
 Kubische Funktion

Standard Abbrechen OK

Über diese Realisierung hinaus ist es anzustreben, daß:

- Die Rechenergebnisse für die erreichte Genauigkeit (Standardabweichung der jeweiligen Rastermasche) grafisch in einer Draufsicht angezeigt werden kann,

- Das Rechenfeld durch vorgegebene Grenzpolygone eingeschränkt werden kann,
- Für bestimmte Regionen separate Rechenparameter gewählt werden können,
- die plausibilisierten Daten exportiert werden können.

Aufgestellt:

H. Wirth