



Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C

Dissertationen

Heft Nr. 852

Johannes Bureick

Robuste Approximation von Laserscan-Profilen mit B-Spline-Kurven

München 2020

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325

ISBN 978-3-7696-5264-2

Diese Arbeit ist gleichzeitig veröffentlicht in:

Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover

ISSN 0174-1454, Nr. 355, Hannover 2020



Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C

Dissertationen

Heft Nr. 852

Robuste Approximation von Laserscan-Profilen mit B-Spline-Kurven

Von der Fakultät für Bauingenieurwesen und Geodäsie
der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

Vorgelegt von

Verm.-Ass. Johannes Bureick, M.Sc.

Geboren am 21.06.1986 in Rheda-Wiedenbrück

München 2020

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325

ISBN 978-3-7696-5264-2

Diese Arbeit ist gleichzeitig veröffentlicht in:

Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover

ISSN 0174-1454, Nr. 355, Hannover 2020

Adresse der DGK:



Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften (DGK)

Alfons-Goppel-Straße 11 • D – 80 539 München
Telefon +49 – 331 – 288 1685 • Telefax +49 – 331 – 288 1759
E-Mail post@dgk.badw.de • <http://www.dgk.badw.de>

Prüfungskommission:

Vorsitzender: Prof. Dr.-Ing. habil. Jürgen Müller

Referent: Prof. Dr.-Ing. Ingo Neumann

Korreferenten: Prof. Dr.-Ing. Hans-Berndt Neuner (TU Wien)
Prof. Dr.-Ing. Steffen Schön

Tag der mündlichen Prüfung: 18.12.2019

© 2020 Bayerische Akademie der Wissenschaften, 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

Kurzfassung

Durch den anhaltenden technischen Fortschritt der letzten Jahrzehnte hat sich der Prozess der Erfassung, Analyse und Darstellung von räumlichen dreidimensionalen (3D) Messdaten grundlegend verändert. Flächenhaft erfassende Sensoren, wie z.B. Laserscanner, liefern ein sehr detailliertes Abbild der Realität. Allerdings können die erfassten riesigen 3D-Punktwolken eine signifikante Anzahl an Ausreißern enthalten, die aufgrund der fehlenden Redundanz im polaren Messverfahren nicht eliminiert werden können. Im Bereich der Darstellung von 3D-Messdaten stehen dem Anwender eine Vielzahl von intuitiv benutzbaren und leicht verständlichen Darstellungsmöglichkeiten, z.B. Punktwolken-Viewer, computer-aided design Software oder Umgebungen der virtuellen Realität, zur Verfügung. Die Analyse als Bindeglied zwischen der Erfassung und Darstellung muss auf diese Veränderungen angepasst werden. Der Umfang der Analyse reicht, je nach gewählter Erfassungsmethode, von der Bereinigung, Segmentierung und Transformation der Punktwolken, über die Modellierung bis hin zu weitergehenden Analysen, wie z.B. Zeitreihen- oder Deformationsanalyse.

Bei der Modellierung, also der Interpolation oder Approximation von Punktwolken, wird versucht, die Punktwolke durch eine mathematische Funktion zu beschreiben. Dadurch können zum einen Objekteigenschaften, z.B. Gradienten, oder statistisch gesicherte Qualitätsparameter, z.B. für die Genauigkeit und Zuverlässigkeit, abgeleitet werden. Zum anderen wird die Punktwolke dadurch in ein ressourcensparendes Format transformiert und steht der weiteren computergestützten Verarbeitung zur Verfügung.

Zur Approximation beliebiger Formen haben sich die stückweisen polynomialen Funktionen und hier insbesondere die B-Splines als besonders geeignete mathematische Funktionen herausgestellt. Der Approximationsprozess bei B-Splines gliedert sich in die vier Teilschritte Modellwahl, Parametrisierung, Knotenvektorwahl und Kontrollpunktschätzung. Im Teilschritt der Knotenvektorwahl muss ein multimodales, multivariates, kontinuierliches und nichtlineares Optimierungsproblem gelöst werden. Hierzu werden einerseits deterministische Methoden angewendet, die den Knotenvektor einmalig aus (abgeleiteten) Eigenschaften der zu approximierenden Punktwolke bestimmen. Andererseits werden heuristische Verfahren eingesetzt. Hier erfolgt eine mehrfache zufällige Wahl des Knotenvektors, die anhand eines gewählten Qualitätsmaßes evaluiert wird. Schlussendlich wird der Knotenvektor mit dem höchsten Qualitätsmaß ausgewählt.

In dieser Arbeit werden drei neuartige oder modifizierte Methoden der Knotenvektorwahl vorgestellt. Als neuartige deterministische Methode wird der residuenbasierte iterative Update Algorithmus vorgestellt. Hierbei wird der Knotenvektor iterativ verändert, indem einzelne Knoten anhand der zuvor bestimmten Residuen verschoben werden. Des Weiteren werden zwei modifizierte heuristische Methoden vorgestellt. Zum einen wird eine modifizierte Version einer evolutionären Monte-Carlo Methode vorgestellt. Zum anderen wird eine modifizierte Version eines elitären genetischen Algorithmus präsentiert. Im Rahmen einer Simulation mit normalverteiltem Rauschen wird gezeigt, dass die Version des elitären genetischen Algorithmus im Vergleich zu anderen Methoden zu nahezu optimalen Knotenvektoren führt. Dies spiegelt sich unter anderem im höchsten Anteil bestandener Globaltests und dem geringsten Abstand zwischen geschätztem und wahren Kurvenverlauf wieder.

Außerdem erfolgt in dieser Arbeit eine systematische Untersuchung des Einflusses von Ausreißern auf die Güte der B-Spline-Kurvenapproximation. Dazu werden Datensätze anhand unterschiedlicher Rauschmodelle generiert. Die Rauschmodelle verwenden neben der Normalverteilung die t -Verteilung, die Gaußsche Mischverteilung und zwei Varianten eines einseitigen Rauschens. In der Kontrollpunktschätzung werden drei Schätzer, Methode der kleinsten Quadrate, Huber- und Hampel-Schätzer verwendet und analysiert. Für jedes Rauschmodell und jeden Schätzer wird die Grenze der Belastbarkeit, also der maximal mögliche Anteil an Ausreißern, bestimmt, bei dem noch annehmbare Ergebnisse erzielt werden. Die Grenze der Belastbarkeit variiert je nach Rauschmodell und verwendeten Schätzer. Grundsätzlich werden mit dem Hampel-Schätzer die robustesten Ergebnisse erzielt.

Stichworte: B-Spline-Kurven, Knotenvektor, Kontrollpunktschätzung, Genetischer Algorithmus, robuste Schätzer, Monte-Carlo Simulation

Abstract

Due to the continuing technical progress of the last decades, the process of capturing, analyzing and displaying spatial three-dimensional (3D) measurement data has fundamentally changed. Area-based sensors, such as laser scanners, provide a very detailed image of the reality. However, the captured huge 3D point clouds can contain a significant number of outliers that cannot be eliminated due to the lack of redundancy in the polar measurement procedures. In the field of displaying 3D measurement data, the user has at his disposal a large number of intuitively usable and easily understandable display options. Examples are point cloud viewer, computer-aided design software or virtual reality environments. The analysis, as the link between capture and representation, must be adapted to these changes. The scope of the analysis ranges, depending on the chosen capturing and presentation method, from the cleaning, segmentation and transformation of the point clouds to the modelling and further analyses, e.g. time series or deformation analysis.

When modelling, i.e. interpolating or approximating point clouds, an effort is made to describe the point cloud using a mathematical function. On the one hand, this allows object properties, e.g. gradients, or statistically assured quality parameters, e.g. for accuracy and reliability, to be derived. On the other hand, the point cloud is transformed into a resource-saving format and it is available for further computer-aided processing.

The piecewise polynomial functions and in particular the B-splines have proven to be particularly suitable mathematical functions for the approximation of arbitrary shapes. The approximation process for B-splines is divided into four steps: model selection, parameterization, knot vector selection and control point estimation. In the substep of the knot vector selection multimodal, multivariate, continuous and nonlinear optimization problems must be solved. On the one hand, deterministic methods are used to determine the knot vector once from (derived) properties of the point cloud to be approximated. On the other hand, heuristic methods are used. Here a multiple random selection of the knot vector takes place, which is evaluated on the basis of a selected quality measure. Finally, the knot vector with the highest quality measure is selected.

In this thesis three new or modified methods of knot vector selection process are presented. The residual-based iterative update algorithm is presented as a novel deterministic method. The knot vector is changed iteratively by moving individual knots according to the previously determined residuals. In addition, two modified heuristic methods are presented. On the one hand, a modified version of an evolutionary Monte Carlo method is shown. On the other hand, a modified version of an elitist genetic algorithm is presented. In a simulation with normally distributed noise it is shown that this version of the elitist genetic algorithm leads to nearly optimal knot vectors compared to other methods. This is shown, among others, by the lowest proportion of rejected null hypothesis for the variance of the unit weight and the smallest distance between the estimated and true curve.

In addition, this thesis systematically investigates the influence of outliers on the quality of B-spline curve approximation. For this purpose, data sets are generated using different noise models. In addition to the normal distribution, the noise models use the t -distribution, the Gaussian mixture distribution and two variants of a biased noise. In control point estimation, three estimators, least squares method, Huber and Hampel estimators are used and analyzed. For each noise model and each estimator, the limit of resilience, i.e. the maximum possible proportion of outliers at which acceptable results can still be achieved, is determined. The limit of resilience varies depending on the noise model and estimator used. Basically, the most robust results are obtained with the Hampel estimator.

Keywords: B-Spline-curves, knot vector, control point estimation, genetic algorithm, robust estimator, Monte-Carlo simulation

Inhaltsverzeichnis

1	Einleitung	3
1.1	Motivation und Zielsetzung	4
1.2	Gliederung	5
2	Mathematische Grundlagen	7
2.1	Freiform-Kurven	7
2.1.1	Polynomiale Funktionen	8
2.1.2	Splines	9
2.1.3	Stellgrößen	15
2.2	Schätztheorie	15
2.2.1	Methode der kleinsten Quadrate	16
2.2.2	M-Schätzung	17
2.2.3	Verschiedene (robuste) M-Schätzer	19
2.2.4	Lösungsalgorithmus für die M-Schätzung	22
3	Modellierung und Schätzung von B-Spline-Kurven	25
3.1	Modellwahl	25
3.1.1	Informationskriterien	26
3.1.2	Hypothesentests	28
3.1.3	Structural Risk Minimization	31
3.1.4	Zusammenfassung Modellwahl	32
3.2	Parametrisierung	32
3.2.1	Deterministische Methoden	33
3.2.2	Iterative Methoden	36
3.2.3	Zusammenfassung Parametrisierung	37
3.3	Knotenvektorwahl	38
3.3.1	Deterministische Methoden	38
3.3.2	Heuristische Methoden	40
3.3.3	Zusammenfassung Knotenvektorwahl	42
3.4	Kontrollpunktschätzung	43
3.4.1	Reine Kontrollpunktschätzung	43
3.4.2	Erweiterte Kontrollpunktschätzung	44
3.4.3	Zusammenfassung Kontrollpunktschätzung	44
4	Methodische Innovation Knotenvektorwahl	47
4.1	Residuenbasierter Iterativer Update (RIU) Algorithmus	47
4.1.1	Methodik	48
4.2	Evolutionäre Monte-Carlo (EMC) Methode	51
4.2.1	Methodik	51
4.2.2	Parameterwahl	56
4.3	Elitärer genetischer Algorithmus	56
4.3.1	Methodik	57
4.3.2	Modifikationen	60
4.3.3	Parameterwahl	61

5	Numerische Beurteilung und Validierung der Knotenvektorwahl	63
5.1	Simulation	63
5.1.1	Ablauf der Simulation	63
5.1.2	Testdatensätze	63
5.1.3	Rauschmodell	67
5.2	Konvergenzverhalten der verschiedenen Modifikationen beim EGA	67
5.3	Präzision und Richtigkeit der verschiedenen Methoden zur Knotenvektorwahl	68
5.3.1	Ergebnisse	71
5.3.2	Zusammenfassung	95
5.4	Sensitivitätsanalyse der Kontrollpunktschätzung	97
5.4.1	Ergebnisse	98
5.4.2	Zusammenfassung	103
5.5	Abweichende Modellwahl	104
5.5.1	Ergebnisse	104
5.5.2	Zusammenfassung	108
5.6	Reale Datensätze - Kreuzvalidierung	110
5.6.1	Datensätze	110
5.6.2	Kreuzvalidierung	111
5.6.3	Ergebnisse	111
5.6.4	Zusammenfassung	113
6	Numerische Beurteilung der B-Spline-Approximation bei Ausreißern	115
6.1	Rauschmodelle	115
6.2	Ergebnisse Knotenvektormethoden	119
6.2.1	Zusammenfassung	130
6.3	Sensitivitätsanalyse der Kontrollpunktschätzung bei ausreißerbehafteten Datensätzen	131
6.4	Belastbarkeit der Schätzer	133
6.4.1	Zusammenfassung	136
7	Fazit und Ausblick	139
7.1	Fazit	139
7.2	Ausblick	140
A	Anhang	141
A.1	Simulierte B-spline Kurven	141
A.2	Numerische Beurteilung und Validierung der Knotenvektorwahl	145
A.3	Numerische Beurteilung der B-Spline-Approximation bei Ausreißern	172
	Literaturverzeichnis	183
	Abbildungsverzeichnis	191
	Tabellenverzeichnis	199
	Algorithmenverzeichnis	201
	Lebenslauf	203

1 Einleitung

Durch den anhaltenden technischen Fortschritt der letzten Jahrzehnte hat sich der Prozess der Erfassung, Analyse und Darstellung von räumlichen dreidimensionalen Messdaten grundlegend verändert.

Bei der Erfassung stehen dem Anwender immer kleinere, leichtere und genauere Sensoren zur Verfügung, die gleichzeitig flächenhaft ein immer größeres Messvolumen abdecken können. Die Sensoren können zudem durch die anhaltende Miniaturisierung zu leistungsfähigen und auf die spezielle Aufgabe angepassten Multi-Sensor-Systemen (MSS) fusioniert werden, wodurch der Erfassungsprozess weiter automatisiert werden kann. Flächenhaft erfassende Sensoren, wie z.B. Laserscanner oder (Stereo-)Kameras, liefern ein sehr detailliertes Abbild der Realität. Nach der Erfassung steht dem Anwender eine riesige dreidimensionale Punktwolke zur Verfügung. Das Unsicherheitsbudget bei der flächenhaften Erfassung setzt sich aus einem sensor-basierten, objekt-basierten und geometrie-basierten Anteil zusammen. Die sensor-basierten Unsicherheiten liegen in der Regel über den Unsicherheiten klassischer punktuell erfassender Sensoren, wie z.B. Tachymeter. Die Objektcharakteristik und die Aufnahmegeometrie beeinflussen maßgeblich das Reflexionsverhalten und damit die Genauigkeit der erfassten Punktwolke. Grundsätzlich bedingt die flächenhafte Erfassung somit ein erhöhtes Unsicherheitsbudget und einen höheren Anteil an Ausreißern. Bereits ein geringer Anteil von Ausreißern führt bei einer Punktwolke mit Millionen von Punkten zu einer großen Anzahl an Ausreißern. Da Wiederholungsmessungen bei flächenhaft erfassenden polaren Messmethoden, wie dem Laserscanner, nicht möglich sind, können diese Ausreißer nicht direkt erkannt und eliminiert werden. Dies hat zur Folge, dass die Zuverlässigkeit der erfassten Punkte durch die fehlende Redundanz nicht gegeben ist.

Im Bereich der Darstellung von dreidimensionalen Messdaten stehen dem Anwender eine Vielzahl von intuitiv benutzbaren und leicht verständlichen Darstellungsmöglichkeiten zur Verfügung. Als Beispiele seien hier Punktwolken-Viewer (z.B. CloudCompare), computer-aided design (CAD) Software, Geoinformationssysteme, Building Information Modeling (BIM) oder Umgebungen der Virtuellen Realität aufgezählt.

Die Analyse als Bindeglied zwischen der Erfassung und Darstellung muss auf diese Veränderungen angepasst werden. Der Umfang der Analyse reicht, je nach gewählter Erfassungs- und Darstellungsmethode, von der Bereinigung, Segmentierung und Transformation der Punktwolken, über die Modellierung (Approximation oder Interpolation) bis hin zu weitergehenden Analysen wie z.B. der Zeitreihen- oder Deformationsanalyse. Bei der Modellierung von Punktwolken wird versucht, die Punktwolke durch eine mathematische Funktion zu beschreiben. Im Bereich des Monitorings und der Deformationsanalyse ist eine gute räumliche und zeitliche Modellierung der erfassten Objekte notwendig. Durch die Approximation von Punktwolken können Objekteigenschaften, z.B. Gradienten, und statistisch gesicherte Qualitätsparameter, z.B. der Genauigkeit und Zuverlässigkeit, abgeleitet werden. Dadurch kann wiederum die Robustheit des gesamten Verfahrens gesteigert werden. Außerdem ist es möglich, eine modellierte Punktwolke in ein ressourcensparendes Format¹ zu exportieren und in CAD-Software oder einem Geoinformationssystem einzulesen, darzustellen und zu bearbeiten.

Zur Modellierung beliebiger Formen (Frei-Formen) haben sich die stückweise polynomialen Funktionen als besonders geeignete mathematische Funktionen herausgestellt. Diese zeichnen sich dadurch aus, dass sie in der Lage sind, sowohl einfache als auch komplexe Objekte durch eine einzige mathematische Funktion geometrietreu zu beschreiben. Eine vorherige Segmentierung des Objektes ist

¹z.B. IGES

dadurch nicht notwendig. Zwei bekannte stückweise polynomiale Funktionen sind die Bézier- und die B-Spline-Kurven (bzw. -Flächen). Unter B-Splines fallen die ebenfalls in vielen Fachdisziplinen verbreiteten Non-Uniform Rational B-Splines (NURBS), aus denen die restlichen B-Spline-Kurven und daraus die Bézier-Kurven abgeleitet werden können. Der Fokus in dieser Arbeit liegt auf den non-uniform (und non-rational) B-Spline-Kurven. Diese bieten bei der Approximation trotz einer verhältnismäßig geringen Anzahl an zu bestimmenden Parametern die volle Flexibilität. B-Spline-Kurven bilden die Grundlage für die im Rahmen dieser Arbeit nicht weiter behandelten B-Spline-Flächen. Die folgenden Ausführungen dieses Kapitels gelten allerdings für B-Spline-Kurven und -Flächen, die zusammengefasst als B-Splines bezeichnet werden.

Die Modellierung von Punktwolken mit B-Splines spielt in verschiedenen Anwendungen bzw. Fachdisziplinen, z.B. Reverse Engineering oder Computer Vision, zum Teil schon seit Jahrzehnten eine wichtige Rolle. Hier sei beispielhaft auf de Boor (1972), Dierckx (1993) und Piegel und Tiller (1997) verwiesen. Im geodätischen Kontext wurde die Modellierung von Punktwolken mit B-Splines erst im letzten Jahrzehnt vermehrt in der Literatur beschrieben. Hier sei beispielhaft auf Koch (2009), Harmening und Neuner (2015) und Bureick u. a. (2016b) verwiesen.

Bei der Approximation wird die Punktwolke im Gegensatz zur Interpolation nicht exakt, sondern durch eine geringere Anzahl an Parametern annähernd beschrieben. Der Approximationsprozess mit B-Splines gliedert sich grundsätzlich in vier Teilschritte:

1. Modellwahl (Kapitel 3.1),
2. Parametrisierung (Kapitel 3.2),
3. Knotenvektorwahl (Kapitel 3.3) und
4. Kontrollpunktschätzung (Kapitel 3.4).

Die Funktion der Teilschritte und der aktuelle Stand der Forschung sind jeweils dem angegebenen Kapitel zu entnehmen. Insbesondere die Knotenvektorwahl war in den letzten Jahren Gegenstand vieler Forschungsarbeiten und Veröffentlichungen. Die Approximationsergebnisse der entwickelten Methoden hängen maßgeblich von der Datencharakteristik (z.B. Unsicherheit), den gewünschten Eigenschaften der B-Splines (z.B. Stetigkeit) und den Anforderungen an den Approximationsprozess (z.B. Rechenzeit) ab. Eine Methode, die bei jeglichen Daten optimale Approximationsergebnisse in minimaler Rechenzeit liefert, wurde noch nicht entwickelt. Vergleichsweise wenig Beachtung in der Literatur findet der Umgang mit Ausreißern im Approximationsprozess. Insbesondere bei der Kontrollpunktschätzung sollte das mögliche Auftreten von Ausreißern berücksichtigt werden.

1.1 Motivation und Zielsetzung

Die zu approximierenden Punktwolken können durch verschiedenste Sensoren bzw. MSS erfasst worden sein. Im geodätischen Kontext können dies z.B. terrestrische oder Airborne Laserscanner, Stereo-Kameras, Streifenprojektionssysteme oder Sensoren der Radarinterferometrie sein. Unabhängig vom verwendeten Sensor bzw. MSS weisen die erfassten Punktwolken folgende Charakteristika auf:

- Sie können aus **sehr vielen Punkten** bestehen.
- Die Punkte können **unregelmäßig verteilt** und **unsortiert** gespeichert sein.
- Die Genauigkeit der Punkte folgt dem **Unsicherheitsbudget** des Messprozesses.
- Da keine Wiederholungsmessungen möglich sind, sind die Punkte in der Regel **nicht kontrollierbar**.
- Es können **Ausreißer** und **Datenlücken** auftreten.

- Sie können **komplex geformte Objekte** darstellen.

Diese Charakteristika müssen im Approximationsprozess berücksichtigt werden. Gerade die letzten beiden Charakteristika haben bei Verwendung der in der Literatur beschriebenen Methoden einen signifikanten negativen Einfluss auf die Güte der B-Spline-Approximation. Das Auftreten von Datenlücken kann zum einen zum kompletten Versagen der B-Spline-Approximation führen. Zum anderen können an den Rändern der Datenlücke ungewollte Ausschwingeffekte auftreten. Das Auftreten von Ausreißern hat bei B-Splines einen lokal begrenzten aber dennoch signifikanten Einfluss auf den approximierten B-Spline. Bei der Approximation der Punktwolke eines komplex geformten Objektes treten an Knicken und scharfen Kanten ungewollte Glättungen auf.

Als Folge dieser Vorbemerkungen lassen sich zwei Forschungshypothesen ableiten:

1. *Es kann eine Methode zur Knotenvektorstimmung entwickelt werden, bei der sowohl auftretende Datenlücken als auch komplex geformte Objekte keinen signifikanten negativen Einfluss auf den approximierten B-Spline und die numerische Stabilität der B-Spline-Approximation haben.*
2. *Es ist möglich die Kontrollpunktschätzung so robust zu gestalten, dass der Einfluss von Ausreißern auf den approximierten B-Spline minimal wird.*

Zusammen mit den Vorbemerkungen können die beiden Forschungshypothesen in einem gemeinsamen Ziel zusammengefasst werden:

Die Entwicklung eines Algorithmus zur robusten B-Spline-Approximation einer beliebigen Punktwolke, welcher, trotz auftretender Ausreißer und Datenlücken und unter Berücksichtigung des dem Messprozess zur Erfassung dieser Punktwolke immanenten Unsicherheitsbudgets, die genaue und zuverlässige Approximation eines komplexen Objektes sicherstellt.

In dieser Arbeit werden keine Neuerungen zu den Teilschritten der Modellwahl und der Parametrisierung entwickelt. Die gängigen Methoden zur Erfüllung dieser Teilschritte werden im Stand der Forschung erläutert und fortan als gegeben angesehen. Im Teilschritt der Knotenvektorstimmung werden neue und innovative Algorithmen gezeigt und mit bekannten Verfahren verglichen. In den Teilschritt der Kontrollpunktschätzung werden robuste Schätzer integriert und systematisch analysiert.

1.2 Gliederung

Diese Arbeit ist wie folgt gegliedert. In Kapitel 2 werden die mathematischen Grundlagen zu Freiform-Kurven (Kapitel 2.1) und zur Schätztheorie (Kapitel 2.2) erläutert. Kapitel 3 umfasst die Beschreibung des gesamten Prozesses zur Modellierung und Schätzung von B-Spline-Kurven. Für die vier Teilschritte Modellwahl (Kapitel 3.1), Parametrisierung (Kapitel 3.2), Knotenvektorstimmung (Kapitel 3.3) und Kontrollpunktschätzung (Kapitel 3.4) wird der aktuelle Stand der Forschung dargestellt. Die methodische Innovation dieser Arbeit ist in Kapitel 4 beschrieben. Insgesamt werden drei neue oder modifizierte Algorithmen zur Knotenvektorstimmung, der residuenbasierte iterative Update Algorithmus (Kapitel 4.1), die evolutionäre Monte-Carlo Methode (Kapitel 4.2) und der elitäre genetische Algorithmus (Kapitel 4.3), präsentiert. In Kapitel 5 werden die drei entwickelten und zusätzlich drei bestehenden Algorithmen numerisch beurteilt und validiert. Zu diesem Zweck wird eine Simulation konzipiert (Kapitel 5.1) und daraus vier vergleichende Untersuchungen (Kapitel 5.2 bis 5.5) durchgeführt. Zusätzlich erfolgt in Kapitel 5.6 im Rahmen der Kreuzvalidierung eine vergleichende Analyse der Leistungsfähigkeit bei zwei realen Datensätzen. Kapitel 6 enthält eine numerische Beurteilung der B-Spline-Approximation bei Ausreißern. Diese Beurteilung erfolgt wiederum im Rahmen einer Simulation, die um ausreißerbehafte Rauschmodelle (Kapitel 6.1) erweitert wird. Hierzu erfolgen drei vergleichende Untersuchungen (Kapitel 6.2 bis 6.4). Verglichen werden dabei die verschiedenen Methoden zur Knotenvektorstimmung und die Anwendung verschiedener

Schätzer bei der Kontrollpunktschätzung. Die Arbeit schließt in Kapitel 7 mit einer Zusammenfassung (Kapitel 7.1) und einem Ausblick (Kapitel 7.2).

2 Mathematische Grundlagen

In diesem Kapitel werden die mathematischen Grundlagen zu Freiform-Kurven (Kapitel 2.1) und die für die Approximation von Punktwolken notwendige Schätztheorie (Kapitel 2.2) beschrieben.

Bei den Freiform-Kurven wird auf die *Parameterdarstellung* von Kurven, insbesondere durch polynomiale Funktionen und Splines, fokussiert. Für polynomiale Funktionen (Kapitel 2.1.1) werden die Repräsentationsformen der *Power Basis Form* und der *Bézier Form* erläutert. Für Splines (Kapitel 2.1.2) werden die gängigen Repräsentationsformen (*Truncated*) *Power Basis* (Kapitel 2.1.2.1), *Hermite Basis* (Kapitel 2.1.2.2) und *B-Spline Basis* (Kapitel 2.1.2.3) dargestellt. Aufgrund des Schwerpunktes dieser Arbeit, wird die *B-Spline Basis* am ausführlichsten beschrieben.

Bei der für die Kontrollpunktschätzung wichtigen Schätztheorie werden die wichtigsten Grundlagen der klassischen Methode der kleinsten Quadrate (Kapitel 2.2.1) und der M-Schätzung (Kapitel 2.2.2) erläutert. In Kapitel 2.2.3 wird ein Überblick über verschiedene, häufig bei der M-Schätzung verwendete Schätzer gegeben. Kapitel 2.2.4 umfasst die methodische Beschreibung eines Lösungsalgorithmus für die M-Schätzung.

2.1 Freiform-Kurven

Die mathematischen Funktionen zur Darstellung von Kurven können grundsätzlich in zwei Arten unterschieden werden: die *impliziten Funktionen* und die *Parameterdarstellung* (siehe Piegl und Tiller (1997, S.1ff.)).

Bei impliziten Funktionen wird die Kurve über einen impliziten Zusammenhang zwischen den Koordinaten z.B. der Form $f(x, y) = 0$ modelliert. Als Beispiel verwenden Piegl und Tiller (1997, S.1ff.) den Einheitskreis, der durch folgende *implizite Funktion* vollständig beschrieben werden kann:

$$f(x, y) : x^2 + y^2 - 1 = 0 \quad (2.1)$$

Bei der *Parameterdarstellung* hingegen wird jede Koordinate durch eine explizite Funktion eines unabhängigen Parameters u repräsentiert. Dieser unabhängige Parameter u gibt die Position des Punktes auf der Kurve an und wird im weiteren Verlauf als Ortsparameter¹ bezeichnet. Es ist üblich, u auf den Wertebereich $[0, 1]$ zu normieren. Dies bedeutet, dass $u = 0$ den Anfang der Kurve und $u = 1$ das Ende der Kurve darstellt. Im Folgenden wird diese Konvention übernommen. Beim Beispiel des Einheitskreises lautet die *Parameterdarstellung*:

$$\mathbf{C}(u) = \begin{bmatrix} C_x(u) \\ C_y(u) \end{bmatrix} = \begin{bmatrix} \cos(u) \\ \sin(u) \end{bmatrix}, \quad 0 \leq u \leq \frac{\pi}{2} \quad (2.2)$$

Im Gegensatz zur impliziten Funktion in Formel 2.1 gilt die *Parameterdarstellung* des Einheitskreises in Formel 2.2 nur für den ersten Quadranten.

Diese Arbeit befasst sich ausschließlich mit der *Parameterdarstellung* der Kurven, da diese von den gängigen CAD- und BIM-Formaten, z.B. Drawing Interchange Format (**dxf**, siehe Autodesk (2011)), Initial Graphics Exchange Specification (**igs**, siehe Reid u. a. (2001)), Standard for the exchange of product model data (**stp**, siehe ISO 10303-42 (2014-05)) oder Industry Foundation Classes (**ifc**, siehe ISO 16739 (2016-10)), unterstützt wird. Die Darstellung komplexer Objekte mit *impliziten Funktionen* ist in der Regel sehr mühselig.

¹Engl. location parameter

2.1.1 Polynomiale Funktionen

Eine einfache Möglichkeit zur Modellierung einer Freiform-Kurve bieten die polynomialen Funktionen. Piegl und Tiller (1997, S.6) geben dabei die *Power Basis Form* und die *Bézier Form* als zwei häufig verwendete Repräsentationsformen für polynomiale Funktionen an.

Die *Power Basis Form* einer polynomialen Kurve im dreidimensionalen Fall wird laut Piegl und Tiller (1997, S.6) über

$$\mathbf{C}(u) = [C_x(u), C_y(u), C_z(u)]^T = \sum_{i=0}^p \mathbf{a}_i u^i \text{ mit } \mathbf{a}_i = [a_{x,i}, a_{y,i}, a_{z,i}]^T \quad (2.3)$$

definiert. Der Grad der polynomialen Funktion wird mit p bezeichnet. Die Koeffizienten der polynomialen Funktion finden sich im Vektor \mathbf{a}_i wieder. Dabei hat jede Koordinatenkomponente einen eigenen Koeffizientensatz ($[a_{x,0}, \dots, a_{x,p}]$, $[a_{y,0}, \dots, a_{y,p}]$ und $[a_{z,0}, \dots, a_{z,p}]$).

Die *Bézier Form*, im folgenden auch *Bézier Kurve* genannt, und zugehörige Methoden wurde Ende der 1950er bzw. Anfang der 1960er Jahre von den Mathematikern Paul de Faget de Casteljau (für Citroën) und Pierre Etienne Bézier (für Renault) unabhängig voneinander entwickelt² (Salomon (2006, S.175)). Die *Bézier-Kurve* wird über Formel 2.4 definiert.

$$\mathbf{C}(u) = [C_x(u), C_y(u), C_z(u)]^T = \sum_{i=0}^p B_{i,p}(u) \mathbf{x}_i \text{ mit } \mathbf{x}_i = [x_{x,i}, x_{y,i}, x_{z,i}]^T \quad (2.4)$$

Ein Punkt $\mathbf{C}(u)$ auf der *Bézier-Kurve* wird über die aufsummierten Linearkombinationen aus Bernsteinpolynomen $B_{i,p}$ und den so genannten Kontrollpunkten \mathbf{x}_i berechnet. Die Bernsteinpolynome werden mit Formel 2.5 berechnet (siehe z.B. Salomon (2006, S.179)).

$$B_{i,p}(u) = \binom{p}{i} u^i (1-u)^{p-i} \text{ mit } \binom{p}{i} = \frac{p!}{i!(p-i)!} \quad (2.5)$$

De Casteljau entwickelte einen effizienten, rekursiven Algorithmus zur numerisch stabilen Berechnung eines Punktes auf der *Bézier-Kurve*. Dieser so genannte De-Casteljau-Algorithmus wird im Rahmen dieser Arbeit nicht dargestellt. Für weitere Information zum De-Casteljau-Algorithmus sei an dieser Stelle z.B. auf Piegl und Tiller (1997, S.22ff.) verwiesen.

Die *Power Basis Form* und die *Bézier Form* sind mathematisch äquivalent, allerdings wird die *Bézier Form* als wesentlich eleganter angesehen (siehe z.B. Piegl und Tiller (1997, S.9)). Dies ist zum einen darin begründet, dass die Kontrollpunkte der *Bézier Form* für den Anwender intuitiv besser zu interpretieren sind, während die Koeffizienten der *Power Basis Form* keinen geometrischen Einblick zulassen (Piegl und Tiller (1997, S.9)). Darüber hinaus ist die *Bézier Form* invariant gegenüber einer affinen Transformation (Salomon (2006, S.192); Pieggl und Tiller (1997, S.14f.)). Um die *Bézier-Kurve* zu transformieren, müssen lediglich die Kontrollpunkte transformiert werden. Zum anderen wirken sich bei der numerisch eher instabilen *Power Basis Form* mögliche Rundungsfehler stärker auf das Ergebnis aus (Piegl und Tiller (1997, S.9)).

Eine Erweiterung der *Bézier-Kurven* stellen die *rationalen Bézier-Kurven* dar. Anstatt der Bernsteinpolynome $B_{i,p}$ werden bei *rationalen Bézier-Kurven* rationale Bernsteinpolynome $R_{\mathcal{B},i,p}$ verwendet (siehe Salomon (2006, S.217ff.)):

$$\mathbf{C}(u) = [C_x(u), C_y(u), C_z(u)]^T = \sum_{i=0}^p R_{\mathcal{B},i,p}(u) \mathbf{x}_i. \quad (2.6)$$

²De Casteljau entwickelte die Methoden bereits Ende der 1950er Jahre und somit vor Bézier. Allerdings unterlagen seine Entwicklungen bei Citroën der Geheimhaltung und wurden nicht publiziert. Deshalb wurden die Methoden nach der zweiten Person benannt, die sie entwickelte. (siehe Salomon (2006, S.175))

Die rationalen Bernsteinpolynome $R_{\mathcal{B},i,p}$ werden über

$$R_{\mathcal{B},i,p}(u) = \frac{B_{i,p}(u)w_i}{\sum_{j=0}^n B_{j,p}(u)w_j} \quad (2.7)$$

berechnet. Dadurch erhält jeder Kontrollpunkt zusätzlich ein Gewicht w_i , über das der Anwender die Kurvenform steuern kann.

Salomon (2006, S.217f.) nennt als Vorteile der *rationalen Bézier-Kurve* gegenüber der *Bézier-Kurve* Projektionsinvarianz und eine genauere Kontrolle über die Kurvenform, weist aber gleichzeitig darauf hin, dass *rationale Bézier-Kurven* für viele Anwendungen unnötig kompliziert sind.

Um mit Hilfe von polynomialen Funktionen, unabhängig davon ob in *Power Basis Form* oder in *Bézier Form* repräsentiert, komplexe Formen darzustellen, muss der Grad p erhöht werden. Dies führt aber ab einem bestimmten Grad zu unerwünschten Ausschwingeffekten.

Werden polynomiale Funktionen zur Approximation von Punktwolken verwendet, wirkt sich ein veränderter Punkt (z.B. ein Ausreißer) auf die gesamte Funktion aus. Diese Eigenschaft wird auch als globale Kontrolle bezeichnet.

2.1.2 Splines

Um die negativen Eigenschaften der polynomialen Funktionen bei der Darstellung komplexer Formen zu vermeiden, wurden die so genannten *Splines* entwickelt. Dabei handelt es sich um stückweise polynomiale Funktionen, also, bildlich gesprochen, einer Aneinanderreihung mehrerer polynomialer Funktionen.

Der Begriff *Spline*³ stammt aus dem Schiffsbau und bezeichnet eine elastische Holzleiste, die, eingespannt an beiden Enden, durch Gewichte oder Nägel in eine glatte, harmonische Form gebracht wird. Die Schiffskonstrukteure verwendeten den Spline als „Schablone“, um die Zeichnungen ihrer Schiffe zu fertigen (siehe Farin (2002a) sowie Ezhov u. a. (2018)). Farin (2002a) taxiert die erste namentliche Erwähnung des Begriffs *Spline* auf Mitte des 18. Jahrhunderts und Ezhov u. a. (2018) bezeichnen die oben beschriebene Konstruktion als mechanische Realisierung einer *Spline* Interpolation.

In der heutigen Zeit finden *Splines* vor allem Anwendung in der Interpolation oder Approximation von Punktwolken. Als der relevanteste *Spline* wird dabei der *kubische Spline* bezeichnet (siehe Ezhov u. a. (2018)), bei dem polynomiale Funktionen 3. Grades aneinander gereiht werden. Um einen glatten Verlauf der Splines sicherzustellen, müssen an den Verknüpfungsstellen der polynomialen Funktionen, im weiteren Verlauf Knoten genannt und mit ξ bezeichnet, bestimmte (wählbare) Stetigkeitsbedingungen eingehalten werden.

Wie bei den polynomialen Funktionen gibt es auch bei den *Splines* verschiedene Repräsentationsformen, auch Basis genannt, von denen drei bekannte Vertreter: (Truncated) Power Basis (Kapitel 2.1.2.1), Hermitesche Basis (Kapitel 2.1.2.2) und die B-Spline Basis (Kapitel 2.1.2.3) in den angegebenen Kapiteln dargestellt werden. Die gezeigten Repräsentationsformen sind mathematisch äquivalent. So werden z.B. in de Boor (2001, S.101ff.) und Hoschek und Lasser (1992, S.412ff.) die Transformationen zwischen verschiedenen Repräsentationsformen gezeigt. Allerdings unterscheiden sie sich, wie schon bei den Repräsentationsformen der polynomialen Funktionen erläutert, z.B. hinsichtlich numerischer Stabilität, geometrischer Interpretierbarkeit, Interaktivität und der Auswirkungen einer Transformation.

In Kapitel 2.1.2.4 werden mit den *Smoothing Splines* und den *Penalized Splines* zwei spezielle Splines vorgestellt, die zwar keine eigene Repräsentationsform bilden, aber in der Literatur häufig verwendet werden.

³zu deutsch: Straklatte

2.1.2.1 (Truncated) Power Basis

Wird die Power Basis als Repräsentationsform gewählt, wird der *Spline* über aneinander gereihete polynomiale Funktionen in *Power Basis Form* definiert. Nach Ezhov u. a. (2018) wird ein dreidimensionaler kubischer *Spline* mit *Power Basis* über:

$$\begin{aligned} C_{x,i}(u) &= a_{x,0i} + a_{x,1i}u + a_{x,2i}u^2 + a_{x,3i}u^3, \\ C_{y,i}(u) &= a_{y,0i} + a_{y,1i}u + a_{y,2i}u^2 + a_{y,3i}u^3, \\ C_{z,i}(u) &= a_{z,0i} + a_{z,1i}u + a_{z,2i}u^2 + a_{z,3i}u^3, \end{aligned} \quad (2.8)$$

mit $i = 1, \dots, \kappa$ definiert, wobei κ den Index des letzten Knotens bezeichnet. Der Index i bezeichnet das Intervall $[\xi_{i-1}, \xi_i)$ für das gilt:

$$\xi_{i-1} \leq u < \xi_i. \quad (2.9)$$

Die Intervallgrenzen werden im Knotenvektor ξ festgelegt. Der Knotenvektor ξ enthält $\kappa+1$ Knoten, die in einer monoton steigenden Reihenfolge gespeichert werden:

$$\xi = |\xi_0, \dots, \xi_\kappa| \text{ mit } \xi_{i-1} \leq \xi_i, \quad i \in \{1, \dots, \kappa\}. \quad (2.10)$$

In jedem Intervall $[\xi_{i-1}, \xi_i)$ wird der *Spline* pro Koordinatendimension über die vier Parameter $a_{x,0}, \dots, a_{x,3}$; $a_{y,0}, \dots, a_{y,3}$ bzw. $a_{z,0}, \dots, a_{z,3}$ der kubischen polynomialen Funktion bestimmt. Um einen glatten *Spline* mit zwei kontinuierlichen Ableitungen an allen Stellen zu erhalten, werden für jede Intervallgrenze⁴ in jeder Dimension 3 Bedingungsgleichungen eingeführt. Für die x-Komponenten lauten diese:

$$C_{x,i}(\xi_i) = C_{x,i+1}(\xi_i) \text{ mit } i = 1, \dots, \kappa - 1, \quad (2.11)$$

$$C'_{x,i}(\xi_i) = C'_{x,i+1}(\xi_i) \text{ mit } i = 1, \dots, \kappa - 1, \quad (2.12)$$

$$C''_{x,i}(\xi_i) = C''_{x,i+1}(\xi_i) \text{ mit } i = 1, \dots, \kappa - 1. \quad (2.13)$$

Für die anderen Koordinatenkomponenten werden die Bedingungsgleichungen analog gebildet (siehe auch Ezhov u. a. (2018)).

Eine sehr ähnliche Repräsentationsform wird über die *Truncated Power Basis* erreicht. Bei *Splines* mit *Truncated Power Basis* ist keine explizite Formulierung der Bedingungsgleichungen (siehe Formel 2.11 bis 2.13) notwendig, da sie sich bereits implizit im funktionalen Modell der *Truncated Power Basis* verbirgt. Für weitere Informationen zur *Truncated Power Basis* sei an dieser Stelle auf de Boor (2001, Kap. 8) und Ezhov u. a. (2018) verwiesen.

Splines mit (*Truncated*) *Power Basis* erben die bereits bei der polynomialen Repräsentationsform der *Power Basis* erwähnten Eigenschaften (siehe Kapitel 2.1.1), dass sie numerisch instabil, geometrisch schwer interpretierbar, nur schwer interaktiv steuerbar und nicht affin invariant sind.

2.1.2.2 Hermitesche Basis

Eine Repräsentationsform, die vor allem bei der Interpolation zum Einsatz kommt, sind die *Splines* mit *hermitescher Basis*, im folgenden auch *hermitesche Splines* genannt. Auch hier werden zumeist kubische Polynome verwendet. Nach Salomon (2006, Kap. 4.2) wird ein *hermitescher Spline* über

⁴mit Ausnahme der beiden äußeren Intervallgrenzen

Formel 2.14 definiert.

$$\begin{aligned}
C_x(u) &= (2t^3 - 3t^2 + 1) p_{x,i-1} + (-2t^3 + 3t^2) p_{x,i} + (t^3 - 2t^2 + t) p'_{x,i-1} + (t^3 - t^2) p'_{x,i}, \\
C_y(u) &= (2t^3 - 3t^2 + 1) p_{y,i-1} + (-2t^3 + 3t^2) p_{y,i} + (t^3 - 2t^2 + t) p'_{y,i-1} + (t^3 - t^2) p'_{y,i}, \\
C_z(u) &= (2t^3 - 3t^2 + 1) p_{z,i-1} + (-2t^3 + 3t^2) p_{z,i} + (t^3 - 2t^2 + t) p'_{z,i-1} + (t^3 - t^2) p'_{z,i} \\
\text{mit } t &= \frac{u - \xi_{i-1}}{\xi_i - \xi_{i-1}}
\end{aligned} \tag{2.14}$$

Der Index i bezeichnet auch hier das Intervall $[\xi_{i-1}, \xi_i]$, für das Formel 2.9 gilt. Eine kubische polynomiale Funktion wird in hermitescher Form über den Startpunkt $\mathbf{p}_{i-1} = [p_{x,i-1}, p_{y,i-1}, p_{z,i-1}]$ und den Endpunkt des Intervalls $\mathbf{p}_i = [p_{x,i}, p_{y,i}, p_{z,i}]$, sowie deren zugehörige Tangenten $\mathbf{p}'_{i-1} = [p'_{x,i-1}, p'_{y,i-1}, p'_{z,i-1}]$ bzw. $\mathbf{p}'_i = [p'_{x,i}, p'_{y,i}, p'_{z,i}]$, definiert.

Um für den *hermiteschen Spline* C^0 - bis C^2 -Stetigkeit herzustellen, werden für zwei benachbarte Intervalle $[\xi_{i-1}, \xi_i]$ und $[\xi_{j-1}, \xi_j]$ mit $j = i + 1$ folgende Bedingungsgleichungen eingeführt (siehe Salomon (2006, Kap. 5.1)):

$$\mathbf{p}_i = \mathbf{p}_{j-1} \tag{2.15}$$

$$\mathbf{p}'_i = \mathbf{p}'_{j-1} \tag{2.16}$$

$$\mathbf{p}'_{i-1} + 4\mathbf{p}'_i + \mathbf{p}'_j = 3(\mathbf{p}_j - \mathbf{p}_{i-1}). \tag{2.17}$$

Aufgrund der Repräsentationsform sind *hermitesche Splines* für den Anwender leichter interpretierbar. Bei der Interpolation kann der Anwender die Form des *hermiteschen Splines* durch die Wahl der äußeren Tangenten beeinflussen. Eine ausführliche Beschreibung der (kubischen) *hermiteschen Splines* findet sich in den Sammelwerken von Farin (2002b, Kap. 7.5 und 9) und Salomon (2006, Kap. 4 und 5), die vorrangig auf die Interpolation fokussieren und verschiedene Methoden zur Bestimmung der äußeren Tangenten darstellen.

2.1.2.3 B-Spline Basis

Die eleganteste, effizienteste, numerisch stabilste und am weitesten verbreitete Repräsentationsform für Splines ist die B-Spline Basis (siehe z.B. Piegl und Tiller (1997, Kap. 2)). Namensgebend für die B-Splines (das „B“ in B-Splines steht für „Basis“) sind die Basisfunktionen (siehe z.B. de Boor (2001, S.99)). Die Entwicklung der B-Splines geht zurück auf Schoenberg (1946) und Curry (1946). Ursprünglich wurden die Basisfunktionen über einen mühsamen Ansatz der dividierten Differenz definiert (Farin (2002a)). Mit der von Cox (1972) und de Boor (1972) entwickelten Rekursionsformel (siehe Formel 2.19) zur Berechnung der Basisfunktionen $N_{i,p}(u)$ wurde die B-Spline Basis für die Praxis interessant (Farin (2002a)).

Eine Spline-Kurve mit B-Spline Basis, im folgenden B-Spline-Kurve genannt, wird über Formel 2.18 definiert (Piegl und Tiller (1997, S.81f.)):

$$\mathbf{C}(u) = [C_x(u), C_y(u), C_z(u)]^T = \sum_{i=0}^n N_{i,p}(u) \mathbf{x}_i \text{ mit } \mathbf{x}_i = [x_{x,i}, x_{y,i}, x_{z,i}]^T. \tag{2.18}$$

Der Kurvenpunkt $\mathbf{C}(u)$ einer B-Spline-Kurve wird über die aufsummierten Linearkombinationen aus Basisfunktionen $N_{i,p}(u)$ und Kontrollpunkten \mathbf{x}_i gebildet. Der Parameter n enthält die Anzahl der Kontrollpunkte ($n + 1$), die die B-Spline-Kurve bilden und p steht für den Grad der Basisfunktionen. Die Ordnung der Basisfunktionen entspricht $p + 1$. Die Definition weist somit eine starke Ähnlichkeit zu der Definition einer Bézier-Kurve auf, unterscheidet sich aber maßgeblich in der

Definition der Basisfunktion:

$$N_{i,0}(u) = \begin{cases} 1, & \text{wenn } \xi_i \leq u < \xi_{i+1} \\ 0, & \text{andernfalls} \end{cases}$$

$$N_{i,p}(u) = \frac{u - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(u) + \frac{\xi_{i+p+1} - u}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(u). \quad (2.19)$$

Eingehender Parameter ist neben dem Ortsparameter u und dem Grad der Basisfunktionen p der für das Erscheinungsbild der B-Spline-Kurve sehr wichtige Knotenvektor ξ . Der Knotenvektor ξ bei B-Splines enthält $n + p + 2$ Knoten, die wie in Formel 2.10 in einer monoton steigenden Reihenfolge aufgestellt werden. Dabei kann zwischen den $2 \cdot (p + 1)$ externen Knoten ξ_{ext} und den $n - p$ internen Knoten ξ_{int} unterschieden werden:

$$\xi = \left[\underbrace{\xi_0 \dots \xi_p}_{\xi_{\text{ext},1}} \underbrace{\xi_{p+1} \dots \xi_n}_{\xi_{\text{int}}} \underbrace{\xi_{n+1} \dots \xi_{n+p+1}}_{\xi_{\text{ext},2}} \right] \text{ mit } \xi_{i-1} \leq \xi_i, i \in \{1, \dots, n + p + 1\}, \quad (2.20)$$

$$\xi_{\text{ext}} = [\xi_{\text{ext},1}, \xi_{\text{ext},2}]. \quad (2.21)$$

Am Beginn und Ende des Knotenvektors stehen jeweils $p + 1$ externe Knoten. Die externen Knoten am Beginn des Knotenvektors $\xi_{\text{ext},1}$ werden auf den minimalen Endpunkt des Wertebereichs für den Ortsparameter u , also hier: 0, gesetzt. Die externen Knoten am Ende des Knotenvektors $\xi_{\text{ext},2}$ werden auf den maximalen Endpunkt des Wertebereichs für den Ortsparameter u , also hier: 1, gesetzt. Jeder externe Knoten in ξ_{ext} steht dabei für Beginn oder Ende einer Basisfunktion. Jeder interne Knoten in ξ_{int} steht für das Ende einer und den Beginn einer weiteren Basisfunktion. Die Anzahl externer Knoten in ξ_{ext} beträgt:

$$\kappa_{\text{ext}} = 2 \cdot (p + 1). \quad (2.22)$$

Die Anzahl interner Knoten in ξ_{int} beträgt:

$$\kappa_{\text{int}} = n - p. \quad (2.23)$$

Der letzte Knoten hat somit den Index κ :

$$\kappa = \kappa_{\text{ext}} + \kappa_{\text{int}} - 1 = n + p + 1. \quad (2.24)$$

Abb. 2.1 gibt die Erstellung einer B-Spline-Kurve grafisch wieder. Grundlage für die B-Spline-Kurve bilden die $n + 1$ Basisfunktionen (in schwarz in der oberen Reihe dargestellt). Im dargestellten Beispiel in Abb. 2.1 wurde $n = 3$ und $p = 2$ gewählt, was nach den Formeln 2.22 und 2.23 zu 6 externen und einem internen Knoten führt (in lila oben links und unten rechts dargestellt). Am internen Knoten endet die Basisfunktion $N_{0,2}$ und Basisfunktion $N_{3,2}$ beginnt. Die $n + 1$ Basisfunktionen werden mit der gleichen Anzahl an Kontrollpunkten (in rot dargestellt) multipliziert. Die daraus resultierenden $n + 1$ Linearkombinationen sind in blau in der unteren Reihe von Abb. 2.1 dargestellt. Die Kontrollpunkte ziehen die Basisfunktionen in eine bestimmte Richtung, wodurch die B-Spline-Kurve jede mögliche Form annehmen kann. Durch Aufsummieren der $n + 1$ Linearkombinationen erhält man die B-Spline-Kurve (in grün unten rechts dargestellt). Eine B-Spline-Kurve hat die beiden folgenden wichtigen Eigenschaften (siehe Piegl und Tiller (1997, Kap. 3.2)):

- Lokale Kontrolle: Die Kontrollpunkte haben jeweils nur auf einen eng begrenzten Bereich einen Einfluss. Ein Kontrollpunkt hat nur einen Einfluss auf einen durch maximal $p + 1$ benachbarte Knoten aufgespannten Bereich. Jeder beliebige Knotenspanne zwischen zwei benachbarten Knoten wird durch maximal $p + 1$ Kontrollpunkte beeinflusst.

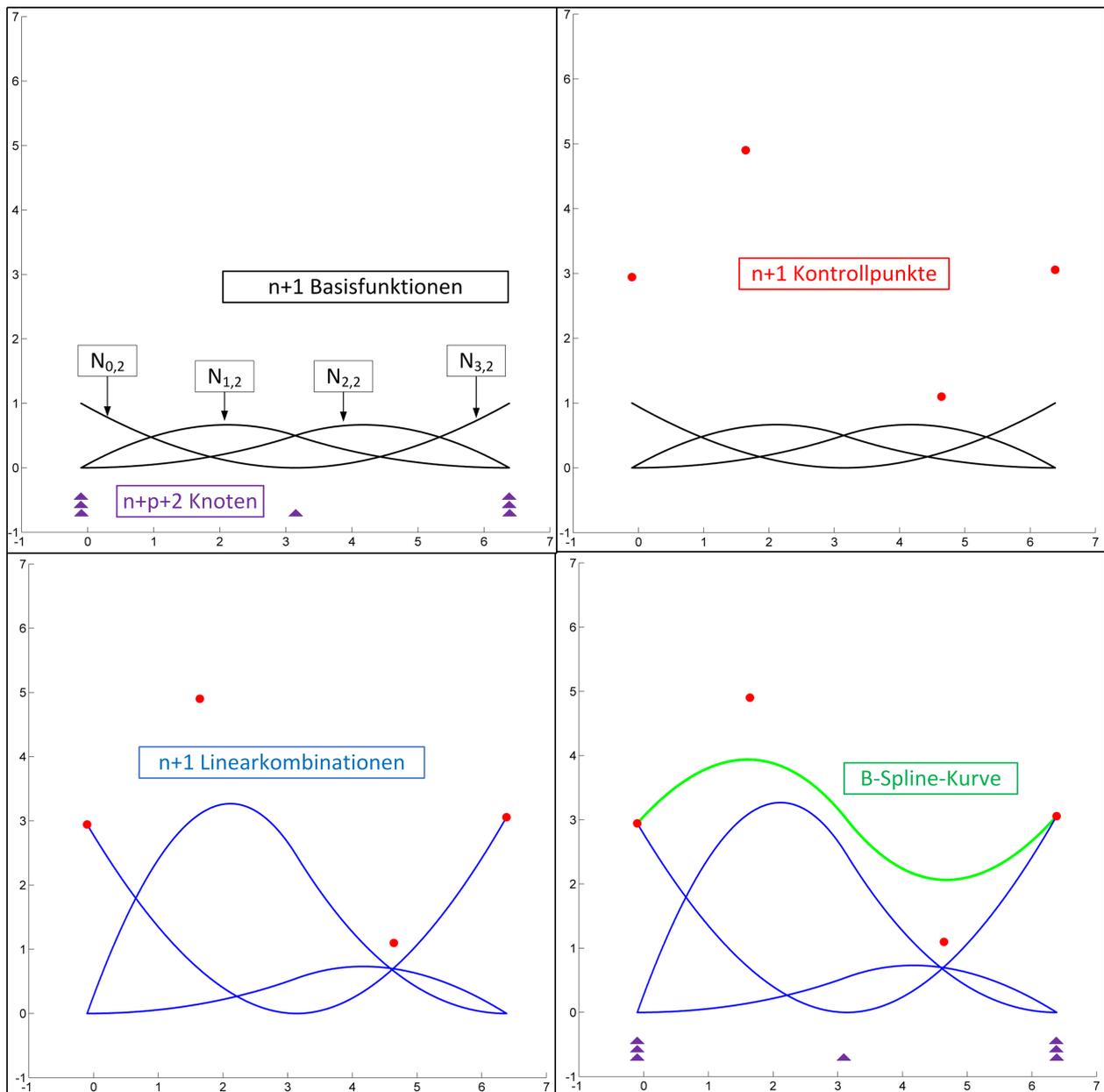


Abbildung 2.1: Erstellung einer B-Spline-Kurve

- Stetigkeit: Die Stetigkeit einer B-Spline-Kurve wird durch die Wahl der Knoten beeinflusst. Generell besitzt die B-Spline-Kurve an jeder Stelle mindestens $p - k$ kontinuierliche Ableitungen. Wobei k für die Multiplizität eines Knoten (also die Anzahl mehrfacher Knoten) an dieser Stelle steht (Siehe hierzu Kapitel 3.3).

Durch passende Wahl der Knoten lässt sich mit einer B-Spline-Kurve die hier nicht weiter beschriebene Repräsentationsform der Bernstein-Bézier Basis, auch Bézier Spline genannt, realisieren. Bei dieser werden polynomiale Funktionen in Bézier Form verknüpft. Durch verschiedene explizit zu modellierende Bedingungsgleichungen lässt sich für die verknüpften Bézier Kurven die gewünschte Stetigkeit erreichen. Allerdings ist die Realisierung einer $C^{\geq 1}$ -Stetigkeit zum einen mühsam herzustellen und zum anderen geht sie einher mit einer verminderten Variabilität des entstehenden Bézier Splines. Für weitere Informationen zur Bernstein-Bézier Basis sei z.B. auf Hoschek und Laser (1992, Kap. 4.1) verwiesen.

Eine Erweiterung der B-Spline-Kurven stellen die Non-uniform rational B-Spline(NURBS)-Kurven

dar. Die Erweiterung erfolgt analog zu der Erweiterung der Bézier-Kurven zu rationalen Bézier-Kurven. NURBS-Kurven werden über Formel 2.25 definiert. Durch die Verwendung von rationalen Basisfunktionen $R_{\mathcal{N},i,p}(u)$ werden die Kontrollpunkte gewichtet.

$$\mathbf{C}(u) = [C_x(u), C_y(u), C_z(u)]^T = \sum_{i=0}^n R_{\mathcal{N},i,p}(u) \mathbf{x}_i \text{ mit } \mathbf{x}_i = [x_{x,i}, x_{y,i}, x_{z,i}]^T \quad (2.25)$$

Die Berechnung von $R_{\mathcal{N},i,p}(u)$ erfolgt über Formel 2.26. Neben den bekannten Basisfunktionen $N_{i,p}(u)$, dem Grad der Basisfunktion p , der Anzahl der Kontrollpunkte $n+1$ und dem Ortsparameter u , gehen die dem jeweiligen Kontrollpunkt zugeordneten Gewichte w_i in die Berechnung mit ein.

$$R_{\mathcal{N},i,p}(u) = \frac{N_{i,p}(u)w_i}{\sum_{j=0}^n N_{j,p}(u)w_j} \quad (2.26)$$

Durch die eingeführten Gewichte kann die letztendliche Form der NURBS-Kurve ohne Veränderung der Kontrollpunkte und der Basisfunktionen beeinflusst werden. Wird ein Gewicht w_i erhöht, führt dies dazu, dass die Kurve näher zu dem zugehörigen Kontrollpunkt gezogen wird. Diese Eigenschaft gibt den NURBS mehr Flexibilität bei interaktiven Modifizierungen im Bereich des Computer-Aided Design. Bei der Approximation von Objekten hingegen werden NURBS seltener eingesetzt. Mögliche Gründe dafür sind ein höherer Rechenaufwand, gelegentlich auftretende numerisch instabile Ergebnisse und dadurch unbrauchbare Ergebnisse.

NURBS sind die Verallgemeinerung aller anderen hier aufgeführten stückweise polynomialen Funktionen. D.h. aus ihnen können alle anderen stückweise polynomialen Funktionen abgeleitet werden.

2.1.2.4 Smoothing und Penalized Splines

Zwei oft verwendete Splines sind die *Smoothing Splines* und die *Penalized Splines*. Sie bilden keine eigene Repräsentationsform, sondern entsprechen darin der (Truncated) Power Basis bzw. der B-Spline Basis. Allerdings unterscheiden sie sich im Erstellungsprozess von „normalen“ Splines.

Die *Smoothing Splines* entsprechen in ihrer Repräsentationsform der Truncated Power Basis. Bei „normalen“ Splines wird bei der Erstellung, also der Interpolation oder Approximation von Punktwolken, der Abstand zu den zu approximierenden Punkten minimiert. Bei *Smoothing Splines* erfolgt dies auch, allerdings wird ein weiterer zu minimierender Term eingeführt. Dieser Term, im Folgenden Glattheitsterm genannt, beschreibt die Glattheit des Splines, indem z.B. bei kubischen *Smoothing Splines* die quadrierte Krümmung (2. Ableitung) des Splines über den gesamten Wertebereich integriert wird (siehe z.B. Schoenberg (1964) und Reinsch (1967)). Über einen zu wählenden Parameter kann der Anwender steuern, ob vorrangig der Abstand zu den zu interpolierenden oder approximierenden Punkten oder der zusätzliche Glattheitsterm minimiert werden soll. *Smoothing Splines* wurden von Schoenberg (1964) und Reinsch (1967) entwickelt und werden ausführlich in de Boor (2001, S.207ff.) beschrieben.

Die *Penalized Splines* sind den *Smoothing Splines* in der Idee und Funktionweise sehr ähnlich, allerdings ist die Repräsentationsform der *Penalized Splines* eine B-Spline Basis. Die *Penalized Splines* wurden für die Approximation verrauschter Datensätze entwickelt (siehe z.B. O'Sullivan (1986, 1988)). Als Glattheitsterm verwendet O'Sullivan (1986, 1988), wie bei den *Smoothing Splines*, eine über den gesamten Wertebereich integrierte quadrierte Ableitung des B-Splines. Der Grad der Glättung kann dabei wiederum über einen wählbaren Parameter gesteuert werden.

Eilers und Marx (1996) schlagen für den Glattheitsterm eine Berechnung über Differenzen zwischen angrenzenden B-Spline-Koeffizienten vor. Dadurch lässt sich die Berechnung des *Penalized Splines* elegant in die Schätzung des B-Splines bzw. dessen Kontrollpunkte integrieren.

2.1.3 Stellgrößen

Die in den vorangegangenen Abschnitten 2.1.1 und 2.1.2 beschriebenen Stellgrößen werden in diesem Kapitel zusammengefasst. Dabei wurde auf die (rationale) Bézier Basis für polynomiale Funktionen und auf die B-Spline Basis für Splines fokussiert. Für die anderen Repräsentationsformen müssen grundsätzlich ähnliche Stellgrößen (z.B. Grad der Funktion p , Ortsparameter u , Knotenvektor ξ) bestimmt werden. Allerdings werden sie zum Teil anders genannt oder unterscheiden sich strukturell aufgrund des mathematischen Modells. So werden die Kontrollpunkte \mathbf{x}_i bei Verwendung der Power Basis durch die Koeffizienten in \mathbf{a} ersetzt.

Der Einfluss der Stellgrößen auf die Bézier und die B-Spline Basis ist in Tabelle 2.1 dargestellt. Jede Stellgröße, die einen Einfluss auf die jeweilige polynomiale Funktion hat, muss im Approxi-

Tabelle 2.1: Einfluss der Stellgrößen auf die stückweise polynomiale Funktionen

Stellgröße/Variable		Bézier	Rat. Bézier	B-Splines	NURBS	Kapitel
Grad der Basisfunktion	p	Ja	Ja	Ja	Ja	
Anzahl der Kontrollpunkte	n	Nein	Nein	Ja	Ja	3.1
Ortsparameter	u	Ja	Ja	Ja	Ja	3.2
Knotenvektor	ξ	Nein	Nein	Ja	Ja	3.3
Kontrollpunkte	\mathbf{x}_i	Ja	Ja	Ja	Ja	3.4
Gewichte	\mathbf{w}_i	Nein	Ja	Nein	Ja	-

mationprozess bestimmt werden. In der letzten Spalte von Tabelle 2.1 ist angegeben, in welchem Kapitel die gängigen Verfahren zur Bestimmung der Stellgröße beschrieben sind. Verfahren zur Wahl der Gewichte werden nicht beschrieben, da die Verwendung von NURBS zur Approximation kaum verbreitet ist und der Fokus dieser Arbeit auf den B-Splines liegt. Bei der gesamten Approximation und zum Teil bereits bei einzelnen Teilschritten handelt es sich um ein multimodales, multivariates, kontinuierliches und nichtlineares Optimierungsproblem. Dies hat zur Folge, dass der Approximationsprozess iterativ durchlaufen werden muss und gute Lösungen häufig nur mithilfe metaheuristischer Verfahren erzielt werden können. Der Fokus dieser Arbeit liegt dabei auf der Knotenvektorstimmung, bei der alle der oben beschriebenen Eigenschaften zutreffen, und der Kontrollpunktschätzung. Die Knotenvektorstimmung ist dabei die wohl entscheidende Stellschraube für die Güte der Approximation. In der Literatur gibt es nur wenige Arbeiten, die bei der Knotenvektorstimmung auftretende Datenlücken berücksichtigen. Die Kontrollpunktschätzung ist auf den ersten Blick ein wohl bekannter und wenig komplizierter Schritt im Approximationsprozess. Von den wenigen Arbeiten, die es in diesem Bereich gibt, gehen aber so gut wie keine auf die Behandlung von Ausreißern in den Daten ein. Im Umfeld der Geodäsie, insbesondere bei der Erfassung mit Laserscannern, sind Datenlücken und Ausreißer aber eher die Regel als die Ausnahme.

2.2 Schätztheorie

Bei der B-Spline-Approximation werden die Kontrollpunkte (siehe hierzu auch Kapitel 3.4) in der Regel als Parameter in einem linearen Gauß-Markov-Modell (GMM) bestimmt (siehe z.B. Piegel und Tiller (1997, S.410f.) oder Koch (2009)). Ein GMM hat nach Jäger (2005, S.98) die Form:

$$\tilde{\mathbf{I}} = \mathbf{1} - \boldsymbol{\varepsilon} = \mathbf{A}\tilde{\mathbf{x}}. \quad (2.27)$$

Dabei lassen sich die wahren Beobachtungen $\tilde{\mathbf{I}}$ durch eine lineare Funktion $\mathbf{A}\tilde{\mathbf{x}}$ der wahren Parameter $\tilde{\mathbf{x}}$ ausdrücken. Die tatsächlichen Beobachtungen \mathbf{I} weichen von den wahren Beobachtungen um die wahre Abweichung $\boldsymbol{\varepsilon}$ ab. Da in der Realität sowohl die wahren Parameter $\tilde{\mathbf{x}}$ als auch die wahren

Abweichungen ε in der Regel unbekannt sind, wird auf die folgende Darstellung mit geschätzten Größen übergegangen:

$$\hat{\mathbf{I}} = \mathbf{I} + \mathbf{v} = \mathbf{A}\hat{\mathbf{x}}. \quad (2.28)$$

Die geschätzten Beobachtungen $\hat{\mathbf{I}}$ können dabei durch eine lineare Funktion $\mathbf{A}\hat{\mathbf{x}}$ der geschätzten Parameter $\hat{\mathbf{x}}$ beschrieben werden. Die Verbesserungen \mathbf{v} sind Schätzwerte für die negativen wahren Abweichungen $-\varepsilon$. Durch ε bzw. \mathbf{v} werden Inkonsistenzen im Falle eines überbestimmten Gleichungssystems (die Anzahl der Beobachtungen m übersteigt die Anzahl der Parameter o) beseitigt. Das klassische Verfahren der Parameterschätzung im GMM ist die Methode der kleinsten Quadrate. Die Grundzüge und wichtigsten Formeln dieser nicht robusten Methode werden in Kapitel 2.2.1 erläutert.

Aus der zweiten Forschungshypothese bzw. dem Ziel dieser Arbeit (siehe Kapitel 1.1) ergibt sich die Notwendigkeit, robuste Verfahren der Parameterschätzung in die Kontrollpunktschätzung zu integrieren. In Kapitel 2.2.2 wird das Prinzip der M-Schätzung erläutert. Dieses erlaubt die Integration verschiedener (robuster) Schätzer in das GMM. In Kapitel 2.2.3 werden verschiedene (robuste) Schätzer vorgestellt. In Kapitel 2.2.4 wird ein Algorithmus zur Berechnung der M-Schätzung beschrieben.

2.2.1 Methode der kleinsten Quadrate

Die Methode der kleinsten Quadrate gilt als das klassische Verfahren zur Parameterschätzung (z.B. Jäger (2005, S.6)). Im Falle normalverteilter Beobachtungen liefert die Methode der kleinsten Quadrate die beste Effizienz aller Schätzverfahren (Jäger (2005, S.6)).

Die Formeln zur Parameterschätzung lassen sich nach Jäger (2005, S.7) aus dem Optimierungsziel

$$\mathbf{v}^T \mathbf{P} \mathbf{v} = (\mathbf{A}\mathbf{x} - \mathbf{I})^T \mathbf{P} (\mathbf{A}\mathbf{x} - \mathbf{I}) \rightarrow \min. \quad (2.29)$$

entwickeln. Die Verbesserungen \mathbf{v} werden dabei durch den durch Umformung von Formel 2.28 gewonnenen Zusammenhang:

$$\mathbf{v} = \mathbf{A}\hat{\mathbf{x}} - \mathbf{I} \quad (2.30)$$

ersetzt. Die Gewichtsmatrix $\mathbf{P} = \sigma_0^2 \boldsymbol{\Sigma}_l^{-1}$ stellt die Inverse der mit dem frei wählbaren⁵ a priori Varianzfaktor σ_0^2 versehenen Kovarianzmatrix $\boldsymbol{\Sigma}_l$ der Beobachtungen dar (Jäger (2005), S.6). Ausmultipliziert ergibt sich:

$$\mathbf{v}^T \mathbf{P} \mathbf{v} = \mathbf{x}^T \mathbf{A}^T \mathbf{P} \mathbf{A} \mathbf{x} - 2 \cdot \mathbf{x}^T \mathbf{A}^T \mathbf{P} \mathbf{I} + \mathbf{I}^T \mathbf{I} \rightarrow \min. \quad (2.31)$$

Das Minimum dieser Formel liegt an der Nullstelle der ersten Ableitung nach den Parametern \mathbf{x} . Es gilt also:

$$\frac{\partial}{\partial \mathbf{x}} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{P} \mathbf{A} \mathbf{x} - 2 \cdot \mathbf{x}^T \mathbf{A}^T \mathbf{P} \mathbf{I} + \mathbf{I}^T \mathbf{I} \right) = \mathbf{A}^T \mathbf{P} \mathbf{A} \mathbf{x} - \mathbf{A}^T \mathbf{P} \mathbf{I} = \mathbf{0}. \quad (2.32)$$

Der geschätzte Parametervektor $\hat{\mathbf{x}}$ ergibt sich dann durch:

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T \mathbf{P} \mathbf{A} \right)^{-1} \mathbf{A}^T \mathbf{P} \mathbf{I}. \quad (2.33)$$

Der a posteriori Varianzfaktor $\hat{\sigma}_0^2$ wird über:

$$\hat{\sigma}_0^2 = \frac{\Omega}{f} = \frac{\mathbf{v}^T \mathbf{P} \mathbf{v}}{m - o} \quad (2.34)$$

⁵frei wählbar so lange \mathbf{P} positiv definit ist

berechnet. Wobei die Differenz zwischen der Anzahl der Beobachtungen m und der Anzahl der Parameter o als Freiheitsgrad f bezeichnet wird. Die Kovarianzmatrix der geschätzten Parameter $\Sigma_{\hat{x}\hat{x}}$ ergibt sich aus:

$$\Sigma_{\hat{x}\hat{x}} = \sigma_0^2 \left(\mathbf{A}^T \mathbf{P} \mathbf{A} \right)^{-1}. \quad (2.35)$$

Die geschätzte Kovarianzmatrix der geschätzten Parameter $\hat{\Sigma}_{\hat{x}\hat{x}}$ ergibt sich aus:

$$\hat{\Sigma}_{\hat{x}\hat{x}} = \hat{\sigma}_0^2 \left(\mathbf{A}^T \mathbf{P} \mathbf{A} \right)^{-1}. \quad (2.36)$$

2.2.2 M-Schätzung

Der Begriff M-Schätzung resultiert aus dem Begriff Maximum-Likelihood-Schätzung. Hierbei wird eine gegebene Dichtefunktion $f(\varepsilon)$ der wahren Messabweichung ε als Zielfunktion der M-Schätzung in Ansatz gebracht (Jäger (2005, S.100)). Diese Dichtefunktion kann mit Hilfe des funktionalen Modells des GMM in Formel 2.27 in Abhängigkeit von den Beobachtungen \mathbf{l} und den Parametern $\tilde{\mathbf{x}}$ dargestellt werden (Jäger (2005, S.100)). Eine in dieser Art parametrisierte Dichtefunktion wird als Likelihood-Funktion bezeichnet (Jäger (2005, S.100)). Es gilt laut Jäger (2005, S.100):

$$\mathcal{L}(\tilde{\mathbf{x}}, \mathbf{l}) =: f(\varepsilon) = f(-(\mathbf{A} \cdot \tilde{\mathbf{x}} - \mathbf{l})) \text{ mit } \varepsilon = -(\mathbf{A} \cdot \tilde{\mathbf{x}} - \mathbf{l}). \quad (2.37)$$

Für normalverteilte Messabweichungen ($\varepsilon \sim N(\mathbf{0}, \Sigma_{\varepsilon\varepsilon})$) lautet die Likelihood-Funktion $\mathcal{L}(\tilde{\mathbf{x}}, \mathbf{l})$:

$$\mathcal{L}(\tilde{\mathbf{x}}, \mathbf{l}) = \frac{1}{\sqrt{m/2\pi \cdot \det\{\Sigma_{\varepsilon\varepsilon}\}}} \cdot \exp\left\{-\frac{1}{2} (\mathbf{A}\tilde{\mathbf{x}} - \mathbf{l}) \cdot \Sigma_{\varepsilon\varepsilon}^{-1} \cdot (\mathbf{A}\tilde{\mathbf{x}} - \mathbf{l})\right\}. \quad (2.38)$$

Die Kovarianzmatrix der Messabweichungen $\Sigma_{\varepsilon\varepsilon}$ entspricht der Kovarianzmatrix der Beobachtungen Σ_{ll} .

Im Falle unterschiedlich genauer oder korrelierter Beobachtungen ist eine Homogenisierung des funktionalen und stochastischen Modells erforderlich, um die Likelihood-Funktion wie in den Formeln 2.37 und 2.38, sowie den folgenden Ausführungen aufstellen zu können (Jäger (2005, S.98ff.)). Durch eine Transformation des Beobachtungsvektors \mathbf{l} und der Designmatrix \mathbf{A} werden unkorrelierte, gleichgenaue, transformierte Beobachtungen $\bar{\mathbf{l}}$ erzeugt. Die Kovarianzmatrix der transformierten Beobachtungen $\bar{\mathbf{l}}$ entspricht der Einheitsmatrix: $\Sigma_{\bar{l}\bar{l}} = \mathbf{I}$ (Jäger (2005, S.99)). Die Homogenisierung erfolgt durch Linksmultiplikation des funktionalen Modells mit der Matrix $\Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}}$ als Transformationsmatrix (Jäger (2005, S.99)):

$$\bar{\mathbf{l}} = \Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}} \cdot \mathbf{l} \quad (2.39)$$

$$\bar{\varepsilon} = \Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}} \cdot \varepsilon \quad (2.40)$$

$$\bar{\mathbf{A}} = \Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}} \cdot \mathbf{A}. \quad (2.41)$$

Die Transformationsmatrix $\Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}}$ lässt sich über die diagonale Matrix $\mathbf{\Lambda}$ der Eigenwerte und die zugehörige Modalmatrix \mathbf{M} der Eigenvektoren berechnen (Jäger (2005, S.99)):

$$\Sigma_{\varepsilon\varepsilon} = \mathbf{M} \cdot \mathbf{\Lambda} \cdot \mathbf{M}^T \quad (2.42)$$

$$\Sigma_{\varepsilon\varepsilon}^{-\frac{1}{2}} = \mathbf{M} \cdot \mathbf{\Lambda}^{-\frac{1}{2}} \cdot \mathbf{M}^T. \quad (2.43)$$

Durch die Transformation geht das GMM in Formel 2.27 in das homogenisierte GMM:

$$\tilde{\bar{\mathbf{l}}} = \bar{\mathbf{l}} - \bar{\varepsilon} = \bar{\mathbf{A}} \tilde{\mathbf{x}} \quad (2.44)$$

über. Für eine weitergehende statistische Analyse erfolgt laut (Jäger (2005, S.99f.)) nach durchgeführter Parameterschätzung eine Rücktransformation der Verbesserungen durch:

$$\mathbf{v} = \boldsymbol{\Sigma}_{\varepsilon\varepsilon}^{+\frac{1}{2}} \cdot \bar{\mathbf{v}}. \quad (2.45)$$

Zum Übergang auf die Schätzung werden die wahren Parameter $\hat{\mathbf{x}}$ und die wahren Abweichungen $\bar{\boldsymbol{\varepsilon}}$ durch die Schätzwerte $\hat{\mathbf{x}}$ und die negativen Verbesserungen $\bar{\mathbf{v}}$ ersetzt (Jäger (2005, S.100)). Somit erhält man als Likelihood-Funktion $\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}})$ der Parameterschätzung für eine beliebige Dichtefunktion der homogenisierten Messabweichungen nach Jäger (2005, S.100f.):

$$\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}}) =: f(-\bar{\mathbf{v}}) = f(-(\bar{\mathbf{A}} \cdot \hat{\mathbf{x}} - \bar{\mathbf{I}})) \text{ mit } \bar{\mathbf{v}} = \bar{\mathbf{A}} \cdot \hat{\mathbf{x}} - \bar{\mathbf{I}}. \quad (2.46)$$

Nach Jäger (2005, S.101) ist die M-Schätzung der Parameter $\hat{\mathbf{x}}_M$ so definiert, dass sie ein Maximum der Likelihood-Funktion $\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}})$ und damit zugleich ein Maximum der Dichtefunktion $f(\bar{\mathbf{v}})$ liefert:

$$\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}}) = f(\bar{\mathbf{v}}) = f(-(\bar{\mathbf{A}} \cdot \hat{\mathbf{x}} - \bar{\mathbf{I}})) = \text{Max}_{|\hat{\mathbf{x}}_M}. \quad (2.47)$$

Durch die Extremalwertforderung in Formel 2.47 ergibt sich der Parametervektor $\hat{\mathbf{x}}_M$ durch Auflösung des folgenden Gleichungssystems:

$$\frac{\partial \mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}})}{\partial \hat{x}_i} = \mathbf{0}. \quad (2.48)$$

Im Falle normalverteilter Messabweichungen lautet die Likelihood-Funktion (vgl. mit Formel 2.38):

$$\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}}) = \frac{1}{\sqrt{2\pi}^m} \cdot \exp\left\{-\frac{1}{2} \sum_{i=1}^m \bar{v}_i^2\right\} = \prod_{i=1}^m \frac{1}{\sqrt{2\pi}} \cdot \exp\left\{-\frac{1}{2} \bar{v}_i^2\right\}. \quad (2.49)$$

Eine Maximierung dieser Likelihood-Funktion entspricht einer Minimierung des Arguments im negativen Exponenten der Likelihood-Funktion (vgl. Jäger (2005, S.102)). Es gilt:

$$\mathcal{L}(\hat{\mathbf{x}}, \bar{\mathbf{I}}) = \text{Max}_{|\hat{\mathbf{x}}_M} \Leftrightarrow \frac{1}{2} \bar{\mathbf{v}}^T \bar{\mathbf{v}} = \text{Min}_{|\hat{\mathbf{x}}_M} \quad (2.50)$$

womit die aus der M-Schätzung hervorgehenden Parameter $\hat{\mathbf{x}}_M$ äquivalent zu denen der Methode der kleinsten Quadrate in Formel 2.33 sind. Ersetzt man nun die spezielle so genannte M-Schätzerfunktion der kleinsten Quadrate Schätzung in Formel 2.50:

$$\rho(\bar{v}_i) = \frac{1}{2} \bar{v}_i^2 \quad (2.51)$$

durch eine beliebige, auch als Verlustfunktion bezeichnete M-Schätzerfunktion:

$$\rho(\bar{v}_i) = \rho(\bar{\mathbf{a}}_i \cdot \hat{\mathbf{x}} - \bar{l}_i), \quad (2.52)$$

erhält man das verallgemeinerte Konzept der M-Schätzung (vgl. Jäger (2005, S.103)). Der Vektor $\bar{\mathbf{a}}_i$ entspricht der i -ten Zeile der Designmatrix $\bar{\mathbf{A}}$.

Die Ableitung der Verlustfunktion nach den Messabweichungen wird als Einflussfunktion $\Psi(\bar{v}_i)$ bezeichnet:

$$\Psi(\bar{v}_i) = \frac{\partial \rho(\bar{v}_i)}{\partial \bar{v}_i}. \quad (2.53)$$

Eine Verlustfunktion ist dann für eine robuste Schätzung geeignet, wenn die zugehörige Einflussfunktion beschränkt ist. Das bedeutet, dass der Einfluss eines Ausreißers nur einen beschränkten

Einfluss auf die geschätzten Parameter hat.

2.2.3 Verschiedene (robuste) M-Schätzer

Im Folgenden werden, mit dem L1-Norm-, dem Huber- und dem Hampel-Schätzer, drei gebräuchliche M-Schätzer mit beschränkter Einflussfunktion vorgestellt. Zum Vergleich wird zunächst der nicht-robuste L2-Norm- bzw. Least-Squares-Schätzer vorgestellt, bei dem die geschätzten Parameter denen der Methode der kleinsten Quadrate entsprechen. Die Verlustfunktionen sollten, damit ein eindeutiges Minimum gefunden werden kann, konvex⁶ und kontinuierlich differenzierbar sein. Die folgenden Formeln und Darstellungen sind an Jäger (2005, Kap. 4.5) angelehnt. Für weitere Informationen und weitere M-Schätzer sei auf diese Quelle verwiesen.

2.2.3.1 L2-Norm-Schätzer

Bei Verwendung des L2-Norm-Schätzers entspricht die M-Schätzung der Methode der kleinsten Quadrate, bei der die quadrierten Verbesserungen minimiert werden. Folglich lautet die zu minimierende Verlustfunktion beim L2-Norm-Schätzer:

$$\rho_{L2}(\bar{v}_i) = \frac{1}{2} \bar{v}_i^2. \quad (2.54)$$

Die dazugehörige Einflussfunktion lautet:

$$\Psi_{L2}(\bar{v}_i) = \bar{v}_i. \quad (2.55)$$

Die graphische Darstellung der Einflussfunktion in Abbildung 2.2 zeigt eindeutig, dass die Einflussfunktion nicht beschränkt ist und der L2-Norm-Schätzer damit kein robuster Schätzer ist. Der

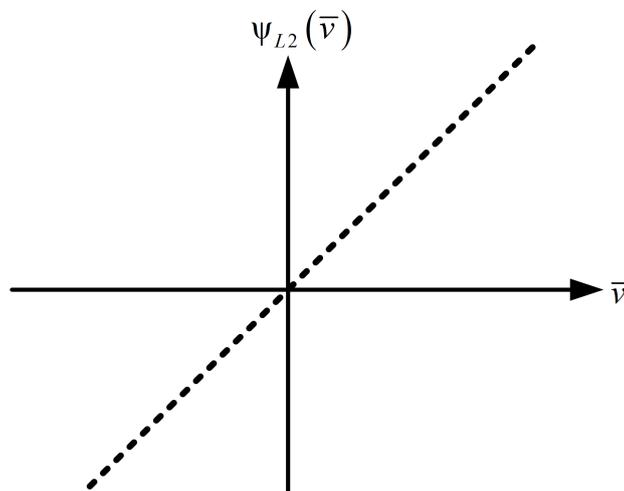


Abbildung 2.2: Einflussfunktion L2-Norm-Schätzer

Einfluss eines Ausreißers wächst proportional zur Größe der Abweichung.

2.2.3.2 L1-Norm-Schätzer

Der L1-Norm-Schätzer gehört, wie der L2-Norm-Schätzer, zu den Ls-Norm-Schätzern, deren allgemeine Verlustfunktion:

$$\rho_{Ls}(\bar{v}_i) = |\bar{v}_i|^s \quad (2.56)$$

⁶Eine Funktion ist konvex, wenn jede Linie zwischen 2 beliebigen Punkten über dem Graph liegt

und die dazugehörige Einflussfunktion:

$$\Psi_{L_s}(\bar{v}_i) = s \cdot \text{sign}(\bar{v}_i) \cdot |\bar{v}_i|^{s-1} \quad (2.57)$$

lauten. Der Ls-Norm-Schätzer ist für Werte: $1 \leq s < 2$ robust (Jäger (2005, S.125)). Beim L1-Norm-Schätzer wird $s = 1$ gesetzt⁷. Dadurch lautet die Verlustfunktion des L1-Norm-Schätzers:

$$\rho_{L1}(\bar{v}_i) = |\bar{v}_i| \quad (2.58)$$

und die dazugehörige Einflussfunktion:

$$\Psi_{L1}(\bar{v}_i) = \text{sign}(\bar{v}_i). \quad (2.59)$$

Die graphische Darstellung der Einflussfunktion in Abbildung 2.3 zeigt, dass alle Beobachtungen, je nach Vorzeichen der Abweichung, auf einen Einfluss von 1 bzw. -1 beschränkt sind. Es ist

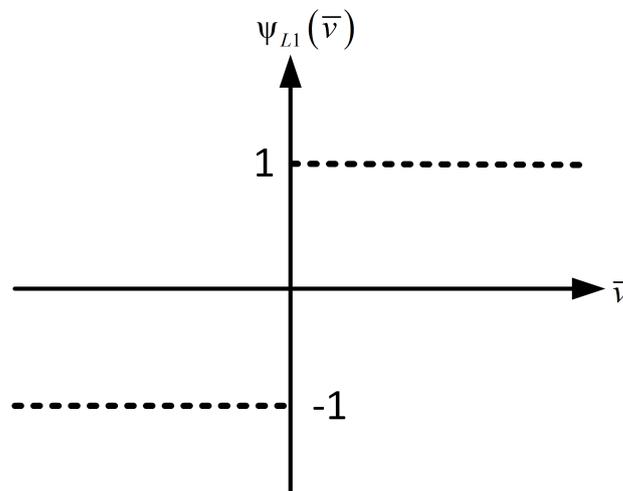


Abbildung 2.3: Einflussfunktion L1-Norm-Schätzer

anzumerken, dass der L1-Norm-Schätzer bei einer Abweichung von $\bar{v}_i = 0$ nicht kontinuierlich differenzierbar ist.

2.2.3.3 Huber-Schätzer

Die zu minimierende Verlustfunktion des auf Huber (1964) zurückgehenden Huber-Schätzers lautet:

$$\rho_H(\bar{v}_i) = \begin{cases} \frac{\bar{v}_i^2}{2} & \text{für } |\bar{v}_i| < c \\ c|\bar{v}_i| - \frac{c^2}{2} & \text{für } |\bar{v}_i| \geq c. \end{cases} \quad (2.60)$$

Die dazugehörige Einflussfunktion lautet:

$$\Psi_H(\bar{v}_i) = \begin{cases} \bar{v}_i & \text{für } |\bar{v}_i| < c \\ c \cdot \text{sign}(\bar{v}_i) & \text{für } |\bar{v}_i| \geq c. \end{cases} \quad (2.61)$$

Durch die graphische Darstellung der Einflussfunktion in Abbildung 2.4 wird schnell ersichtlich, dass der Einfluss eines Ausreißers beschränkt ist. Überschreitet die absolute Verbesserung einer Beobachtung den Schwellwert c , ist der Einfluss dieser Beobachtung auf ebendiesen Schwellwert c beschränkt. Der Schwellwert c muss im Vorhinein nach der erwarteten Größe der Ausreißer gewählt

⁷Beim L2-Norm-Schätzer wird dementsprechend $s = 2$ gesetzt.

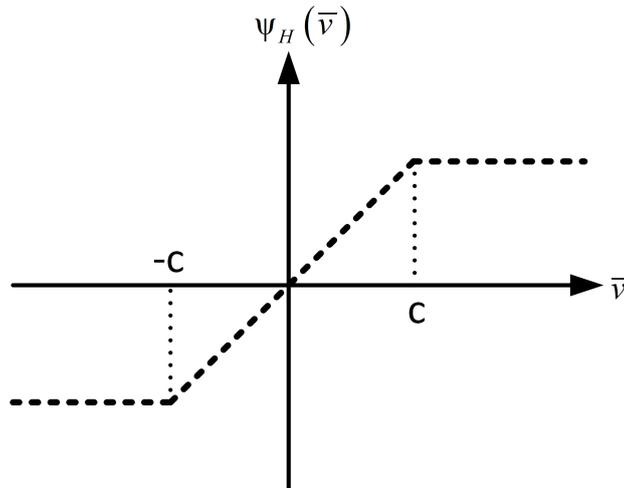


Abbildung 2.4: Einflussfunktion Huber-Schätzer

werden. Caspary (2013, S.64) gibt für technische Messungen, bei denen ein Ausreißeranteil von über 1% selten ist, für c einen Wert zwischen 1.5 und 2.0 als geeignet an. Beobachtungen, deren Verbesserungen kleiner als der Schwellwert c sind, haben einen unbeschränkten Einfluss auf die Parameterschätzung.

2.2.3.4 Hampel-Schätzer

Die zu minimierende Verlustfunktion des, auf Hampel (1974) zurückgehenden, Hampel-Schätzers lautet:

$$\rho_{\text{Ha}}(\bar{v}_i) = \begin{cases} \frac{\bar{v}_i^2}{2} & \text{für } |\bar{v}_i| < a \\ a|\bar{v}_i| - \frac{a^2}{2} & \text{für } a \leq |\bar{v}_i| \leq b \\ ab - \frac{a^2}{2} + (c-b)\frac{a}{2}\left(1 - \left(\frac{c-|\bar{v}_i|}{c-b}\right)^2\right) & \text{für } b \leq |\bar{v}_i| \leq c \\ ab - \frac{a^2}{2} + (c-b)\frac{a}{2} & \text{für } c \leq |\bar{v}_i|. \end{cases} \quad (2.62)$$

Die dazugehörige Einflussfunktion lautet:

$$\Psi_{\text{Ha}}(\bar{v}_i) = \begin{cases} \bar{v}_i & \text{für } |\bar{v}_i| < a \\ a \cdot \text{sign}(\bar{v}_i) & \text{für } a \leq |\bar{v}_i| \leq b \\ a \cdot \frac{c-|\bar{v}_i|}{c-b} \cdot \text{sign}(\bar{v}_i) & \text{für } b \leq |\bar{v}_i| \leq c \\ 0 & \text{für } c \leq |\bar{v}_i|. \end{cases} \quad (2.63)$$

Die graphische Darstellung der Einflussfunktion in Abbildung 2.5 zeigt, dass die Einflussfunktion des Hampel-Schätzers ebenfalls beschränkt ist. Durch die Einführung von insgesamt 3 Schwellwerten a , b und c wird der Einfluss großer Ausreißer noch mehr beschränkt als beim Huber-Schätzer. Beobachtungen mit einer absoluten Abweichung unterhalb des ersten Schwellwertes a haben einen unbeschränkten Einfluss auf die Parameterschätzung. Beobachtungen, deren absolute Abweichung zwischen den Schwellwerten a und b liegt, werden wie beim Huber-Schätzer auf einen Einfluss gleich dem Schwellwert a beschränkt. Der Einfluss von Beobachtungen, deren absolute Abweichung zwischen den Schwellwerten b und c liegt, nimmt ausgehend von einem Einfluss gleich a bis zum Schwellwert c linear ab. Beobachtungen mit einer absoluten Abweichung größer dem Schwellwert c haben keinen Einfluss auf die Schätzung. Hampel (1974) schlägt als geeignete Wahl für die Schwellwerte $a = 2$, $b = 4$ und $c = 8$ vor.

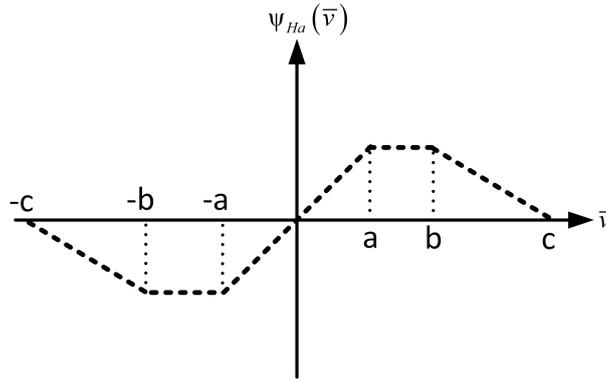


Abbildung 2.5: Einflussfunktion Hampel-Schätzer

2.2.4 Lösungsalgorithmus für die M-Schätzung

Die Einflussfunktionen sind nichtlineare Funktionen, wodurch die Parameter in der M-Schätzung iterativ zu berechnen sind (Holland und Welsch (1977)). Hierzu beschreibt Huber (1981, Kap. 7.8) unter anderem die Methode der iterativ regewichteten kleinsten Quadrate.

Der Ablauf der iterativ regewichteten kleinsten Quadrate ist in Algorithmus 1 dargestellt. In den Algorithmus gehen die Designmatrix $\bar{\mathbf{A}}$ und die Beobachtungen $\bar{\mathbf{I}}$ als Eingangsgrößen ein. Als Ausgangsgröße erhält man die Schätzwerte für die Parameter $\hat{\mathbf{x}}_M$, die geschätzte Kovarianzmatrix der geschätzten Parameter $\hat{\Sigma}_{\hat{x}\hat{x}}$ und die geschätzte a posteriori Standardabweichung $\hat{\sigma}_0$.

Zunächst werden mit Hilfe von Formel 2.33, wobei für die Gewichtsmatrix $\mathbf{P} = \mathbf{I}$ gilt, und Formel 2.30 initiale Schätzwerte für die Parameter und Verbesserungen berechnet. Mit den geschätzten Verbesserungen erfolgt eine vereinfachte Schätzung der a posteriori Standardabweichung:

$$\hat{\sigma}_0 = f_{\text{mad}} \cdot \text{mad}(\bar{\mathbf{v}}) = f_{\text{mad}} \cdot \text{median}(|\bar{\mathbf{v}} - \text{median}(\bar{\mathbf{v}})|) \quad \text{mit } f_{\text{mad}} = 1.4826. \quad (2.64)$$

Im Falle normalverteilter Verbesserungen liefert Formel 2.64 eine unverzerrte Schätzung für σ . In der Literatur gibt es verschiedene leicht abweichende Angaben für den Faktor f_{mad} . So spezifiziert Jäger (2005, S.132) einen Wert von $f_{\text{mad}} = 1.423$, während Holland und Welsch (1977) einen Wert von $f_{\text{mad}} = 1.48$ ansetzen. In der Arbeit wird schlussendlich der in MathWorks[®] (2019b) für die verwendete MATLAB-Funktion *mad* spezifizierte Wert $f_{\text{mad}} = 1.4826$ gewählt. Eine strenge Schätzung für $\hat{\sigma}_0$ ist in Huber (1981, Kap. 7.7) zu finden.

Die in Zeile 7 beginnende while-Schleife wird so lange durchlaufen, bis die Verbesserungen zweier konsekutiver Iterationen nur noch marginale Unterschiede aufweisen. Zur Festlegung eines Schwellwertes ist die Rechengenauigkeit der Software zu beachten. Innerhalb der while-Schleife werden zunächst neue Gewichte berechnet. Die Gewichte sind abhängig von den zuvor geschätzten und durch $\hat{\sigma}_0^{(k)}$ normierten Verbesserungen und der gewählten Einflussfunktion. Mit den neuen Gewichten $w_i^{(k+1)}$ wird die Hauptdiagonale der Gewichtsmatrix $\mathbf{W}^{(k+1)}$ besetzt. Anschließend werden mit der Matrix $\mathbf{W}^{(k+1)}$ neue Schätzwerte für die Parameter, die Verbesserungen und den Varianzfaktor berechnet. Ist die while-Schleife durchlaufen, wird die geschätzte Kovarianzmatrix für die Parameter der M-Schätzung $\hat{\Sigma}_{\hat{x}\hat{x}}$ geschätzt. Hierzu schlägt Jäger (2005, S.131) folgende Formel vor:

$$\hat{\Sigma}_{\hat{x}\hat{x}} = (\bar{\mathbf{A}}^T \mathbf{W}^{(k)} \bar{\mathbf{A}})^{-1} \left(\bar{\mathbf{A}}^T (\mathbf{W}^{(k)})^2 \bar{\mathbf{A}} \right) (\bar{\mathbf{A}}^T \mathbf{W}^{(k)} \bar{\mathbf{A}})^{-1}. \quad (2.65)$$

Mit der Ausgabe der Ausgangsgrößen endet die Methode der iterativ regewichteten kleinsten Quadrate.

Algorithmus 1 Iterativ regewichtete kleinste Quadrate

```

1: procedure IRKQ( $\bar{\mathbf{A}}, \bar{\mathbf{I}}$ )                                     ▷ Designmatrix und Beobachtungen
2:    $\hat{\mathbf{x}}_{\mathbf{M}}^{(0)} = (\bar{\mathbf{A}}^T \bar{\mathbf{A}})^{-1} \bar{\mathbf{A}}^T \bar{\mathbf{I}}$                                        ▷ Initiale Parameterschätzung
3:    $\bar{\mathbf{v}}^{(0)} = \bar{\mathbf{A}} \hat{\mathbf{x}}_{\mathbf{M}}^{(0)} - \bar{\mathbf{I}}$                                                        ▷ Initiale Verbesserungen
4:    $\hat{\sigma}_0^{(0)} = 1,4826 \cdot \text{mad}(\bar{\mathbf{v}}^{(0)})$                                        ▷ Vereinfachte Schätzung des Varianzfaktors
5:    $\bar{\mathbf{v}}^{(1)} = \infty$                                                                      ▷ Für den Einstieg in die while-Schleife
6:    $k = 0$                                                                                  ▷ Zählvariable initialisieren
7:   while  $|\bar{\mathbf{v}}^{(k+1)} - \bar{\mathbf{v}}^{(k)}| > \mathbf{0}$  do                                           ▷ Rechengenauigkeit beachten
8:      $w_i^{(k+1)} = \frac{\Psi(\bar{v}_i^{(k)}/\hat{\sigma}_0^{(k)})}{\bar{v}_i^{(k)}/\hat{\sigma}_0^{(k)}}$  mit  $i = 1, \dots, m$            ▷ Update Gewichtung
9:      $\mathbf{W}^{(k+1)} = \text{diag}([w_1^{(k+1)}, \dots, w_m^{(k+1)}])$ 
10:     $\hat{\mathbf{x}}_{\mathbf{M}}^{(k+1)} = (\bar{\mathbf{A}}^T \mathbf{W}^{(k+1)} \bar{\mathbf{A}})^{-1} \bar{\mathbf{A}}^T \mathbf{W}^{(k+1)} \bar{\mathbf{I}}$                                        ▷ Parameterschätzung
11:     $\bar{\mathbf{v}}^{(k+1)} = \bar{\mathbf{A}} \hat{\mathbf{x}}_{\mathbf{M}}^{(k+1)} - \bar{\mathbf{I}}$                                                        ▷ Verbesserungen
12:     $\hat{\sigma}_0^{(k+1)} = 1,4826 \cdot \text{mad}(\bar{\mathbf{v}}^{(k+1)})$                                        ▷ Vereinfachte Schätzung des Varianzfaktors
13:     $k = k + 1$                                                                                ▷ Zählvariable inkrementieren
14:  end while
15:   $\hat{\Sigma}_{\hat{\mathbf{x}}\hat{\mathbf{x}}} = (\bar{\mathbf{A}}^T \mathbf{W}^{(k)} \bar{\mathbf{A}})^{-1} (\bar{\mathbf{A}}^T (\mathbf{W}^{(k)})^2 \bar{\mathbf{A}})^{-1} (\bar{\mathbf{A}}^T \mathbf{W}^{(k)} \bar{\mathbf{A}})^{-1}$    ▷ Geschätzte Kovarianzmatrix
16:  return  $\hat{\mathbf{x}}_{\mathbf{M}}^{(k)}, \hat{\Sigma}_{\hat{\mathbf{x}}\hat{\mathbf{x}}}, \hat{\sigma}_0^{(k)}$                                        ▷ Rückgabe
17: end procedure

```

3 Modellierung und Schätzung von B-Spline-Kurven

In den Kapiteln 3.1 bis 3.4 wird der Approximationsprozess zur Modellierung und Schätzung von B-Spline-Kurven dargestellt. Dieser Approximationsprozess lässt sich grundsätzlich in die folgenden vier Arbeitsschritte einteilen:

1. Modellwahl (Kapitel 3.1),
2. Parametrisierung (Kapitel 3.2),
3. Knotenvektorwahl (Kapitel 3.3) und
4. Kontrollpunktschätzung (Kapitel 3.4).

In den angegebenen Kapiteln werden die in der Literatur bekannten Methoden zur Erfüllung dieser vier Arbeitsschritte erläutert. Im Approximationsprozess soll eine beliebige dreidimensionale Punktwolke der Form:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 \\ \vdots \\ \mathbf{Q}_r \end{bmatrix} = \begin{bmatrix} Q_{x,1} & Q_{y,1} & Q_{z,1} \\ \vdots & \vdots & \vdots \\ Q_{x,r} & Q_{y,r} & Q_{z,r} \end{bmatrix} \quad (3.1)$$

approximiert werden. Der Beobachtungsvektor \mathbf{l} enthält folglich die zu approximierende Punktwolke, bestehend aus r Punkten:

$$\mathbf{l} = [Q_{x,1} \ \dots \ Q_{x,r} \ Q_{y,1} \ \dots \ Q_{y,r} \ Q_{z,1} \ \dots \ Q_{z,r}]^T. \quad (3.2)$$

D.h. die Anzahl der Beobachtungen ist $m = d \cdot r = 3 \cdot r$, wobei d die Dimension der Punktwolke bezeichnet.

3.1 Modellwahl bei B-Spline-Kurven

B-Spline-Kurven (und -Flächen) zeichnen sich dadurch aus, dass sich das mathematische Modell mit steigendem Grad der Basisfunktion oder steigender Anzahl an Kontrollpunkten nicht grundlegend ändert. Aus Formel 2.18 ist ersichtlich, dass der lineare Zusammenhang zwischen B-Spline-Kurve und Kontrollpunkten auch bei einer Erhöhung des Basisfunktionsgrades p oder der Anzahl der Kontrollpunkte $n + 1$ bestehen bleibt.

Trotz dieser Tatsache ist die Wahl von p und n entscheidend für die Qualität der Approximation. Große p und n führen einerseits dazu, dass die B-Spline-Kurve die gegebenen Messdaten sehr gut approximiert, es aber andererseits zu Ausschwingeffekten zwischen den Messdaten, einer geringeren Glättung der Messdaten und zu einem erhöhten Rechenaufwand kommt. Es muss also eine Abwägung zwischen p und n sowie der Güte der Approximation geben. Bei dieser Abwägung gilt die Maßgabe: „So viele wie nötig, so wenige wie möglich“ (Burnham und Anderson (2004a, Kap. 1.2.6); Harmening und Neuner (2016)). In der Praxis mögen p und n oftmals durch Vorwissen über das zu modellierende Objekt oder Erfahrung festgelegt werden. Fundierter, aber gleichzeitig auch rechenaufwendiger, ist die Anwendung von

- Informationskriterien (Kapitel 3.1.1),
- Hypothesentests (Kapitel 3.1.2) oder
- Structural Risk Minimization (Kapitel 3.1.3)

für die Festlegung der Modell-Komplexität.

3.1.1 Informationskriterien

Die Güte einer Approximation kann z.B. mithilfe der Residuenquadratsumme Ω quantifiziert werden. Die Grundidee eines Informationskriteriums ist es, die Anzahl der Parameter (im Falle von B-Spline-Kurven: p und n) in einen Strafterm zu überführen. Die Summe aus Ω und dem Strafterm ergibt das Informationskriterium. Es wird anschließend die Parameteranzahl gewählt, bei der das Informationskriterium minimal ist.

Die beiden verbreitetsten Informationskriterien sind das Akaike (AIC) und das Bayesian (BIC) Informationskriterium, die sich in der Berechnung des Strafterms unterscheiden. Trotz der offenkundigen Ähnlichkeit von AIC und BIC beruhen sie auf unterschiedlichen Prinzipien (Burnham und Anderson (2004b); Harmening und Neuner (2016)).

Im Fall korrelierter Beobachtungen ist vor der Anwendung des AIC und BIC eine Dekorrelation mittels Cholesky-Zerlegung, wie in Zhao u. a. (2019) beschrieben, notwendig.

3.1.1.1 Akaike Informationskriterium (AIC)

Das AIC verwendet den Kullback-Leibler(KL)-Abstand¹ als fundamentale Basis für die Modellwahl. Durch den von Akaike (1998b)² publizierten Zusammenhang zwischen dem geschätzten relativen KL-Abstand, der maximierten log-Likelihood der geschätzten Parameter $\hat{\mathbf{x}}$ unter Vorliegen der Beobachtungen \mathbf{l} und der Anzahl der Parameter K konnte das AIC nach Formel 3.3 definiert werden:

$$AIC = -2 \log(\mathcal{L}(\hat{\mathbf{x}}|\mathbf{l})) + 2K. \quad (3.3)$$

Sind die Abweichungen normalverteilt und haben eine konstante Varianz, so kann das AIC direkt über Formel 3.4 berechnet werden:

$$AIC = m \log(\hat{\sigma}^2) + 2K \text{ mit } \hat{\sigma}^2 = \frac{\sum \hat{\epsilon}^2}{m}. \quad (3.4)$$

Die Anzahl der Beobachtungen m entspricht dabei auch der Anzahl der Residuen ϵ , die aus der kleinsten Quadrate Schätzung der Parameter resultieren. Im Strafterm $2K$ gibt K die Anzahl der zu schätzenden Parameter an. Für B-Spline-Kurven müssen die $n + 1$ d -dimensionalen Kontrollpunkte und der a posteriori Varianzfaktor $\hat{\sigma}_0^2$ als Parameter geschätzt werden (Harmening und Neuner (2016)). Somit ergibt sich K bei B-Spline-Kurven über:

$$K = d(n + 1) + 1. \quad (3.5)$$

Beim AIC handelt es sich um ein asymptotisch effizientes Kriterium. Diesem unterliegt die Annahme, dass sich unter den zur Auswahl stehenden Modellen nicht das wahre Modell befindet, da das ursächliche Phänomen unendlich komplex ist und nicht in allen Facetten in einer endlichen Stichprobe (in diesem Fall der zu approximierenden Punktwolke) widergespiegelt wird. Asymptotisch effiziente Kriterien tendieren dazu bei wachsendem Stichprobenumfang, also größeren Punktwolken, komplexere Modelle als optimal zu identifizieren, da mit wachsendem Stichprobenumfang mehr Details des unendlich komplexen Phänomens ersichtlich werden (siehe Harmening und Neuner (2016)).

¹auch KL-Divergenz, KL-Information oder KL-Entropie genannt

²Dabei handelt es sich um einen Nachdruck der ursprünglichen Veröffentlichung aus dem Jahr 1973

Da die Modellwahl und damit die Informationskriterien nicht im primären Fokus dieser Arbeit stehen, sie aber bei der Bestimmung der B-Splines eine gewichtige Rolle spielen, werden an dieser Stelle einige Literaturhinweise genannt.

McQuarrie und Tsai (1998) sowie Burnham und Anderson (2004b) sind Standardwerke zur Modellwahl. Es werden die mathematischen Grundlagen der Informationskriterien (AIC, BIC und weiterer) ausführlich hergeleitet und dargestellt. Harmening und Neuner (2016) orientieren sich bei der Übertragung der Informationskriterien auf B-Splines an der Notation dieser Standardwerke und fassen die wichtigsten Herleitungsschritte zusammen. Auch in anderen Arbeiten (z.B. Harmening und Neuner (2014) oder Gálvez u. a. (2015)) wurden die Informationskriterien für die Bestimmung der optimalen Parameteranzahl bei B-Splines eingesetzt.

3.1.1.2 Bayesian Informationskriterium (BIC)

Das BIC wurde Ende der 1970er Jahre zeitgleich von Akaike (1998a)³ und Schwarz (1978)⁴ vorgestellt. Während Akaike (1998a) sein Informationskriterium für die Modellwahl bei linearer Regression herleitete, leitete Schwarz (1978) sein Informationskriterium für die Modellwahl aus der Exponentialfamilie her.

Das BIC wählt dasjenige Modell unter den zur Auswahl stehenden Modellen aus, das a posteriori die höchste Wahrscheinlichkeit hat. Die a posteriori Wahrscheinlichkeit wird über das Bayes-Theorem berechnet (siehe Claeskens und Hjort (2008, S.79) sowie Harmening und Neuner (2016)):

$$P(M_j|\mathbf{l}) = \frac{P(M_j)}{f(\mathbf{l})} \underbrace{\int f(\mathbf{l}|M_j, \mathbf{x}_j)\pi(\mathbf{x}_j|M_j)d\mathbf{x}_j}_{\lambda_j(\mathbf{l})}. \quad (3.6)$$

Die a posteriori Wahrscheinlichkeit $P(M_j|\mathbf{l})$ für ein bestimmtes Modell M_j unter den gegebenen Beobachtungen \mathbf{l} ist dabei von folgenden drei Größen abhängig:

- Der a priori Wahrscheinlichkeit für ein bestimmtes Modell $P(M_j)$. Diese ist in der Regel unbekannt und wird deshalb für alle Modelle als gleich angenommen. Dadurch hat diese Größe keinen Einfluss auf den relativen Vergleich zwischen den verschiedenen Modellen.
- Der a priori Wahrscheinlichkeit der Beobachtungen $f(\mathbf{l})$. Diese Größe ist für alle Modelle gleich und hat deshalb ebenfalls keinen Einfluss auf die a posteriori Wahrscheinlichkeiten.
- Der Randverteilung $\lambda_j(\mathbf{l})$ eines bestimmten Modells, die aus der Wahrscheinlichkeit $f(\mathbf{l}|M_j, \mathbf{x}_j)$ und der a priori Dichte $\pi(\mathbf{x}_j|M_j)$ des Parameters \mathbf{x}_j bei gegebenem Modell M_j bestimmt wird. Dies ist die entscheidende Größe, die bestimmt werden muss:

$$\lambda_j(\mathbf{l}) = \int f(\mathbf{l}|M_j, \mathbf{x}_j)\pi(\mathbf{x}_j|M_j)d\mathbf{x}_j = \int \mathcal{L}_j(\mathbf{x}_j)\pi(\mathbf{x}_j|M_j)d\mathbf{x}_j. \quad (3.7)$$

Daraus wird die exakte Definition des BIC hergeleitet:

$$BIC = 2 \log(\lambda_{n,j}(\mathbf{l})). \quad (3.8)$$

Da eine geschlossene Berechnung der Randverteilung im Allgemeinen nicht möglich ist, kann sie mit Hilfe der Methode von Laplace approximiert werden:

$$BIC \approx -2 \log(\mathcal{L}(\hat{\mathbf{x}}|\mathbf{l})) + \log(m)K. \quad (3.9)$$

Analog zu Formel 3.4 kann das BIC über Formel 3.10 genähert berechnet werden:

$$BIC \approx m \log(\hat{\sigma}^2) + \log(m)K \text{ mit } \hat{\sigma}^2 = \frac{\sum \hat{\epsilon}^2}{m}. \quad (3.10)$$

³Dabei handelt es sich um einen Nachdruck der ursprünglichen Veröffentlichung aus dem Jahr 1978

⁴deshalb auch Schwarz Information Criterion (SIC) genannt

Der Strafterm K bei B-Splines wird wiederum über Formel 3.5 berechnet.

Beim BIC handelt es sich um ein asymptotisch konsistentes Kriterium. Dessen unterliegt die Annahme, dass das ursächliche Phänomen endlich komplex und in der Menge der zur Auswahl stehenden Modelle enthalten ist. Im Gegensatz zu den asymptotisch effizienten Kriterien tendieren asymptotisch konsistente Kriterien dazu, bei wachsendem Stichprobenumfang weniger komplexe Modelle als optimal zu identifizieren.

3.1.2 Hypothesentests

Neben den Informationskriterien können bei der Modellwahl grundsätzlich statistische Hypothesentests zum Einsatz kommen. Der Globaltest (Kapitel 3.1.2.1) bietet die Möglichkeit, die Eignung eines Modells zu überprüfen. Mit Vuong's (Kapitel 3.1.2.2) oder Clarke's Test (Kapitel 3.1.2.3) können zwei Modelle auf signifikante Unterschiede untersucht werden. Während die Verwendung des Globaltests für die Modellwahl bei B-Splines in der Praxis wenig geeignet ist, werden Vuong's und Clarke's Test für die Modellwahl bei B-Splines eingesetzt.

3.1.2.1 Globaltest

Der in der Ausgleichsrechnung verwendete Globaltest ist die abschließende Kontrolle der Ausgleichung. Im Falle der B-Spline-Approximation findet der Globaltest nach der Kontrollpunkt-schätzung statt. Die Standardabweichung der Gewichtseinheit a priori σ_0 wird mit der a posteriori bestimmten Standardabweichung der Gewichtseinheit $\hat{\sigma}_0$ verglichen. Für ein Gauß-Markov-Modell beschreibt Niemeier (2008, Kap. 4.5.3) das folgende Vorgehen.

Die Testgröße t wird über:

$$t = \frac{f \cdot \hat{\sigma}_0^2}{\sigma_0^2} \quad (3.11)$$

bestimmt. Wobei die Anzahl der Freiheitsgrade mit f bezeichnet wird. Das Quantil wird über die Chi-Quadrat-Verteilung berechnet. Bei einem einseitigen Globaltest wird es über:

$$\text{Quantil : } \chi_{f,1-\alpha}^2 \quad (3.12)$$

berechnet. Die Irrtumswahrscheinlichkeit wird mit α angegeben.

Bei einem einseitigen Globaltest wird die Nullhypothese angenommen, wenn die Testgröße kleiner oder gleich dem Quantil ist (Formel 3.13). Die Nullhypothese des einseitigen Globaltests besagt, dass die a posteriori die a priori Standardabweichung der Gewichtseinheit nicht signifikant übersteigt.

$$H_0 \text{ annehmen : } t \leq \chi_{f,1-\alpha}^2 \quad (3.13)$$

Ist die Testgröße allerdings größer als das Quantil, wird die Nullhypothese verworfen:

$$H_0 \text{ verwerfen : } t > \chi_{f,1-\alpha}^2 \quad (3.14)$$

Alternativ kann hier auch ein zweiseitiger Globaltest angesetzt werden (siehe Koch (2004, S.313)). Dazu werden zwei Quantile über:

$$\begin{aligned} \text{Quantil 1 : } & \chi_{f,\frac{\alpha}{2}}^2 \\ \text{Quantil 2 : } & \chi_{f,1-\frac{\alpha}{2}}^2 \end{aligned} \quad (3.15)$$

berechnet. Bei einem zweiseitigen Globaltest wird die Nullhypothese verworfen, wenn die Testgröße signifikant größer oder kleiner als das Quantil ist (Formel 3.16). Die Nullhypothese des zweiseitigen Globaltests besagt, dass die a posteriori und die a priori Standardabweichung der Gewichtseinheit

nicht signifikant abweichen.

$$H_0 \text{ annehmen} : \chi_{f, \frac{\alpha}{2}}^2 \leq t \leq \chi_{f, 1-\frac{\alpha}{2}}^2. \quad (3.16)$$

Ist die Testgröße kleiner als Quantil 1 oder größer als Quantil 2, wird die Nullhypothese verworfen:

$$H_0 \text{ verwerfen} : t < \chi_{f, \frac{\alpha}{2}}^2 \text{ oder } t > \chi_{f, 1-\frac{\alpha}{2}}^2. \quad (3.17)$$

Wird die Nullhypothese H_0 verworfen, so kann dies verschiedene Ursachen haben (siehe z.B. Nie-meier (2008, S.170)):

- σ_0^2 gilt nicht für die tatsächlichen Messbedingungen,
- das funktionale Modell ist nicht zutreffend oder
- das stochastische Modell ist nicht zutreffend.

Bei der B-Spline-Approximation kann eine zu gering oder zu groß gewählte Anzahl an Kontrollpunkten zu einem unzutreffenden funktionalen Modell führen. Weitere Ursache kann eine unzulängliche Parametrisierung (siehe Kapitel 3.2) oder Knotenvektorwahl (siehe Kapitel 3.3) sein.

Das stochastische Modell ist dann unzutreffend, wenn es entweder zu pessimistisch oder zu optimistisch gewählt wurde. Bei einem einseitigen Globaltest führt nur letzteres zu einem Verwerfen der Nullhypothese.

Unter der Annahme, dass keine groben Fehler vorliegen, die Parametrisierung und Knotenvektorwahl angemessen durchgeführt wurde, sowie ein zutreffendes stochastisches Modell gewählt wurde, wäre das korrekte Modell dasjenige mit der geringsten Anzahl an Parametern (Kontrollpunkten) und einer angenommenen Nullhypothese. Auch bei einem zweiseitigen Globaltest ist damit zu rechnen, dass bei mehreren Modellen die Nullhypothese angenommen wird.

Aufgrund der oben beschriebenen vielfältigen Möglichkeiten, die zu einem Verwerfen der Nullhypothese führen können, ist der Globaltest für die Modellwahl bei B-Splines wenig geeignet.

3.1.2.2 Vuong's Test

Vuong's Test wurde ursprünglich von Vuong (1989) entwickelt. In Alkhatib u. a. (2018), Zhao u. a. (2018) und Zhao (2019) wurde Vuong's Test verwendet, um verschiedene B-Spline-Modelle auf signifikante Unterschiede zu untersuchen. Die Idee des Vuong Test ist es zu überprüfen, ob die geschätzten Parameter $\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{(0)}$ und $\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{(1)}$ der Maximum-Likelihood-Schätzung sehr kleine oder sehr große Differenzen zwischen den logarithmierten Likelihoods $\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{(0)}|\mathbf{1}))$ und $\log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{(1)}|\mathbf{1}))$ hervorrufen. Dazu wird die Differenz mit folgendem Term normiert:

$$\hat{\omega} = \sqrt{E - F}, \quad (3.18)$$

mit

$$E = \frac{1}{m} \sum_{i=1}^m [\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{(0)}|l_i)) - \log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{(1)}|l_i))]^2 \quad (3.19)$$

und

$$F = \left[\frac{1}{m} \sum_{i=1}^m [\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{(0)}|l_i)) - \log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{(1)}|l_i))] \right]^2. \quad (3.20)$$

Die Testgröße T_V wird damit über:

$$T_V = \frac{\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{2(0)}|\mathbf{1})) - \log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{2(1)}|\mathbf{1}))}{\hat{\omega} \cdot \sqrt{m}} \quad (3.21)$$

berechnet und folgt bei einer hohen Anzahl an Beobachtungen annähernd der Standardnormalverteilung. Um größere Modelle gegenüber kleineren Modellen zu bestrafen, schlägt Vuong (1989) vor, in die Berechnung der Testgröße einen Strafterm aus einem Informationskriterium zu integrieren. Integriert man den Strafterm des BICs wird die Testgröße T_V wie folgt berechnet:

$$T_V = \frac{(\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{2(0)}|\mathbf{1})) - \frac{1}{2} \cdot K^{(0)} \cdot \log(m)) - (\log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{2(1)}|\mathbf{1})) - \frac{1}{2} \cdot K^{(1)} \cdot \log(m))}{\hat{\omega} \cdot \sqrt{m}} \quad (3.22)$$

wobei die Anzahl der Parameter $K^{(0)}$ und $K^{(1)}$ der Modelle nach Formel 3.5 berechnet werden. Die Testgröße T_V wird gegenüber 2 Quantilen der Normalverteilung:

$$C_1 = C_{\alpha/2}^{\mathcal{N}(0,1)} \quad (3.23)$$

und

$$C_2 = C_{1-\alpha/2}^{\mathcal{N}(0,1)} \quad (3.24)$$

getestet, wobei α für das Signifikanzniveau steht. Dadurch können in der Testentscheidung 3 Hypothesen angenommen werden:

1. Die beiden Modelle unterscheiden sich nicht signifikant (Nullhypothese):

$$C_1 \leq T_V \leq C_2. \quad (3.25)$$

2. Modell 1 ist signifikant besser als Modell 2 (Alternativhypothese 1):

$$T_V < C_1. \quad (3.26)$$

3. Modell 2 ist signifikant besser als Modell 1 (Alternativhypothese 2):

$$T_V > C_2. \quad (3.27)$$

3.1.2.3 Clarke's Test

Clarke (2003) entwickelte einen Hypothesentest, der nicht auf der Annahme einer normalverteilten Testgröße basiert. In Alkhatib u. a. (2018) wurde dieser Hypothesentest auf verschiedene B-Spline-Flächen-Modelle angewendet. Die Idee bei Clarke's Test ist es, anders als bei Vuong's Test, nicht die Differenzen der Likelihoods der gesamten Beobachtungen, sondern die Differenzen zwischen den logarithmierten Likelihoods bei jeder Beobachtung l_i zu berechnen:

$$d_i = \log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{2(0)}|l_i)) - \log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{2(1)}|l_i)) \text{ mit } i = 1, \dots, m. \quad (3.28)$$

Anschließend wird die Anzahl der Elemente in d_i mit $i = 1, \dots, m$ bestimmt, die einen positiven Wert aufweisen. Diese Anzahl wird mit T_{C1} bezeichnet und dient im Folgenden als Testgröße. Unterscheiden sich die beiden Modelle nicht signifikant, wird erwartet, dass T_{C1} der halben Anzahl der Beobachtungen entspricht. T_{C1} folgt der Binomial-Verteilung, deren Parameter der Anzahl der Beobachtungen m und der a priori Wahrscheinlichkeit von 0.5 für eine positive Differenz entsprechen. Clarke (2007) schlägt analog zu Vuong vor, größere Modelle gegenüber kleineren Modellen mit einem Strafterm zu bestrafen. Die Berechnung der Differenzen aus Formel 3.28 ändert sich

durch die Integration des BIC-Strafterms zu:

$$d_i = (\log(\mathcal{L}(\hat{\mathbf{x}}^{(0)}, \hat{\sigma}^{2(0)} | l_i)) - \frac{1 \cdot K^{(0)}}{2 \cdot m} \cdot \log(m)) - (\log(\mathcal{L}(\hat{\mathbf{x}}^{(1)}, \hat{\sigma}^{2(1)} | l_i)) - \frac{1 \cdot K^{(1)}}{2 \cdot m} \cdot \log(m)) \quad (3.29)$$

mit $i = 1, \dots, m$,

wobei die Anzahl der Parameter $K^{(0)}$ und $K^{(1)}$ der Modelle nach Formel 3.5 berechnet wird. Die Testgröße T_{Cl} wird gegenüber 2 Quantilen der Binomialverteilung:

$$B_1 = B_{\alpha/2}^{B(0.5, 3 \cdot s \cdot t)} \quad (3.30)$$

und

$$B_2 = B_{1-\alpha/2}^{B(0.5, 3 \cdot s \cdot t)} \quad (3.31)$$

getestet, wobei α wiederum für das Signifikanzniveau steht. Analog zu Vuong's Test können in der Testentscheidung 3 Hypothesen angenommen werden:

1. Die beiden Modelle unterscheiden sich nicht signifikant (Nullhypothese):

$$B_1 \leq T_{\text{Cl}} \leq B_2. \quad (3.32)$$

2. Modell 1 ist signifikant besser als Modell 2 (Alternativhypothese 1):

$$T_{\text{Cl}} < B_1. \quad (3.33)$$

3. Modell 2 ist signifikant besser als Modell 1 (Alternativhypothese 2):

$$T_{\text{Cl}} > B_2. \quad (3.34)$$

3.1.3 Structural Risk Minimization

Einen anderen Ansatz zur Modellwahl bei B-Splines haben Harmening und Neuner (2016) entwickelt. Sie bestimmen die optimale Anzahl an Parametern einer B-Spline-Kurve mit Hilfe von Structural Risk Minimization (SRM). In Harmening und Neuner (2017) erweitern sie diesen Ansatz auf B-Spline-Flächen. Im Folgenden soll nur die Grundidee des Ansatzes vermittelt werden. Für eine ausführlichere Darstellung und eine theoretische Herleitung sei auf die beiden Quellen Harmening und Neuner (2016) sowie Harmening und Neuner (2017) verwiesen.

SRM stammt aus der Statistical Learning Theory (SLT). In der SLT wird diejenige Funktion \tilde{g} aus einer Menge von Funktionen gesucht, die den Zusammenhang zwischen einem Input-Vektor und einem Output-Vektor optimal beschreibt. Die Qualität mit der eine Funktion diesen Zusammenhang beschreibt bzw. approximiert, wird mithilfe einer Verlustfunktion, die als wahres Risiko bezeichnet wird, gemessen. Der Erwartungswert des wahren Risikos R_{wahr} wird durch die optimale Funktion \tilde{g} minimiert.

Da R_{wahr} nicht berechnet werden kann, wird dieser durch den Erwartungswert des empirischen Risikos R_{emp} approximiert. R_{wahr} und R_{emp} sind über eine Ungleichung miteinander verknüpft, in die neben der Stichprobengröße auch die Vapnik-Chervonenkis-Dimension (VC-Dimension) eingeht. Da die VC-Dimension nur in den wenigsten Fällen bekannt ist, muss diese empirisch bestimmt werden. Für die empirische Bestimmung der VC-Dimension haben Harmening und Neuner (2016) einen Klassifikator für B-Spline-Kurven sowie in Harmening und Neuner (2017) einen Klassifikator für B-Spline-Flächen entwickelt. Es wird dann die Funktion oder im Falle von B-Splines das Modell g_{opt} als optimal gewählt, bei dem das approximierte R_{wahr} am kleinsten ist. In Harmening und Neuner (2016) sowie Harmening und Neuner (2017) haben die Autoren anhand verschiedener simulierter

B-Splines mit unterschiedlicher Kontrollpunktanzahl und unterschiedlichem Grad der Basisfunktion gezeigt, dass die VC-Dimension von g_{opt} proportional mit der Anzahl der Kontrollpunkte verknüpft ist. In einem Vergleich zwischen der simulierten Soll-Kontrollpunktanzahl und der über die verschiedenen Methoden bestimmten Kontrollpunktanzahl zeigen die Autoren, dass mit SRM, je nach Form der B-Splines, vergleichbare bis bessere Ergebnisse als mit den Informationskriterien erzielt werden.

3.1.4 Zusammenfassung Modellwahl

In der Modellwahl bei B-Spline-Kurven wird versucht, den optimalen Basisfunktionsgrad p sowie die optimale Anzahl an Kontrollpunkten $n + 1$ zu bestimmen. Für diese Aufgabe sind drei Verfahren bekannt.

Das erste und zugleich bekannteste Verfahren ist die Anwendung eines Informationskriteriums. Hierbei kommt vor allem das AIC und das BIC zum Einsatz, wobei das BIC bei einem großem Stichprobenumfang komplexere Modelle stärker bestraft als das AIC.

Das zweite Verfahren ist die Anwendung eines Hypothesentests. Hier wurden in Zusammenhang mit B-Spline Vuong's und Clarke's Test verwendet. Bei beiden Tests wird ein Strafterm integriert, der komplexere Modelle bestraft.

Das dritte Verfahren bestimmt die optimale Anzahl an Parametern über Structural Risk Minimization. Hierzu erfolgt eine empirische Berechnung der VC-Dimension anhand eines Klassifikators für B-Splines. Die in diesem Kapitel beschriebenen und zuvor zusammengefassten Verfahren sowie

Tabelle 3.1: Zusammenfassung Modellwahl bei B-Splines

Methode		Literatur
Informationskriterium	AIC	Akaike (1998b); McQuarrie und Tsai (1998); Burnham und Anderson (2004a); Harmening und Neuner (2016)
	BIC	Akaike (1998a); Schwarz (1978); Claeskens und Hjort (2008); Harmening und Neuner (2016)
Hypothesentest	Globaltest	Niemeier (2008); Koch (2004)
	Vuong's Test	Vuong (1989); Alkhatib u. a. (2018); Zhao u. a. (2018); Zhao (2019)
	Clarke's Test	Clarke (2003); Alkhatib u. a. (2018)
Structural Risk Minimization		Harmening und Neuner (2016, 2017)

einige wichtige Quellen dazu sind in Tabelle 3.1 aufgelistet. Die dort aufgelisteten Quellen stellen nur eine gewisse Auswahl dar. Ein konkreter Bezug zu B-Splines ist in den Arbeiten Harmening und Neuner (2016, 2017); Alkhatib u. a. (2018); Zhao u. a. (2018) und Zhao (2019) gegeben. Die restlichen Arbeiten beschreiben die Modellwahl allgemein.

3.2 Parametrisierung

Ziel der Parametrisierung ist es, für jeden Punkt der zu approximierenden Punktwolke einen Ortsparameter \bar{u} zu bestimmen⁵. Um kenntlich zu machen, dass ein Punkt zur Approximation der B-Spline-Kurve verwendet wird, wird sein Ortsparameter mit \bar{u} anstatt mit u bezeichnet. Ein beliebiger Punkt auf der B-Spline-Kurve wird weiterhin über den Ortsparameter u ausgegeben. Da

⁵Bei den, in dieser Arbeit nicht näher behandelten, B-Spline-Flächen müssen für jeden Punkt zwei Ortsparameter bestimmt werden

in dieser Arbeit die Konvention übernommen wird, die Ortsparameter auf einen Bereich zwischen 0 und 1 zu normieren, gilt:

$$0 \leq \bar{u} \leq 1. \quad (3.35)$$

Die Parametrisierung im Falle von Kurven kann als ein eindimensionales Problem mit vergleichsweise einfachen Methoden gelöst werden. Bei der im Folgenden nicht dargestellten Flächenparametrisierung ist die Aufgabenstellung deutlich komplizierter und macht die Anwendung fortgeschrittener Methoden (z.B. Coons (1967); Ma und Kruth (1995); Farin (2002b); Harmening und Neuner (2015)) notwendig.

Im Folgenden werden nur Methoden vorgestellt, die für die Approximation geeignet sind. Methoden, die nur zur Parametrisierung bei der Interpolation geeignet sind (z.B. die von Lim (1999) entwickelte universelle Parametrisierungsmethode), werden im Rahmen dieser Arbeit nicht genauer vorgestellt.

Der gesamte Vektor der Ortsparameter einer Punktwolke wird bei Kurven mit $\bar{\mathbf{u}}$ zusammengefasst:

$$\bar{\mathbf{u}} = \begin{bmatrix} \bar{u}_1 & \dots & \bar{u}_r \end{bmatrix}. \quad (3.36)$$

Bei den im Folgenden vorgestellten Methoden kann grob zwischen deterministischen Methoden (siehe Kapitel 3.2.1) und iterativen Methoden (siehe Kapitel 3.2.2) unterschieden werden. Deterministische Methoden verwenden eine oder mehrere, zum Teil auch abgeleitete Eigenschaften der zu approximierenden Punktwolke, um daraus den Ortsparametervektor $\bar{\mathbf{u}}$ zu berechnen. Diese Berechnung erfolgt einmalig und ist in der Regel wenig rechenintensiv. Allerdings erfordern die deterministischen Methoden in der Regel eine sortierte Punktwolke. Die iterativen Methoden starten mit einer initialen Parametrisierung, z.B. über eine deterministische Methode berechnet, und verbessern die Parametrisierung iterativ und abwechselnd zur Kontrollpunktschätzung. Diese Ansätze erfordern zwar einen höheren Rechenaufwand, können dafür aber zumeist auch unsortierte Punktwolken verarbeiten.

3.2.1 Deterministische Methoden

3.2.1.1 Gleichabständig

Bei der gleichabständigen Parametrisierung wird den r Punkten der sortierten Punktwolke ein inkrementierender Integer-Wert k , beginnend mit 0, zugewiesen (siehe Piegler und Tiller (1997, S. 364)). Der Ortsparameter \bar{u}_k setzt sich aus dem Quotient von k und dem Integer-Wert des letzten Punktes, also $r - 1$, zusammen (siehe Formel 3.37).

$$\bar{u}_k = \frac{k - 1}{r - 1} \text{ mit } k = 1, \dots, r \quad (3.37)$$

Um mit Blick auf die anderen Parametrisierungsmethoden eine vergleichbare Notation zu verwenden, kann dieser Ansatz auch wie folgt dargestellt werden. Der Abstand Δ_k zwischen den Datenpunkten wird auf 1 gesetzt (Formel 3.38).

$$\Delta_k = 1 \quad (3.38)$$

Die Abstände werden zu einer Gesamtlänge d_{sum} aufsummiert (Formel 3.39),

$$d_{\text{sum}} = \sum_{k=1}^{r-1} \Delta_k \quad (3.39)$$

wobei d_{sum} verwendet wird, um den Ortsparametervektor $\bar{\mathbf{u}}$ auf einen Bereich zwischen 0 und 1 zu normieren (Formel 3.40).

$$\bar{u}_k = \bar{u}_{k-1} + \frac{\Delta_{k-1}}{d_{\text{sum}}} \text{ mit } k = 2, \dots, r \text{ und } \bar{u}_1 = 0 \quad (3.40)$$

Diese Parametrisierungsmethode wird von Piegl und Tiller (1997, S.364) im Fall von nicht gleich-abständigen Punktwolken nicht empfohlen, da unerwünschte fehlerhafte Formen, wie Schleifen, hervorgerufen werden können.

3.2.1.2 Chordal

Bei der chordalen Parametrisierungsmethode wird die Sehnenlänge zwischen benachbarten Punkten für die Parametrisierung verwendet (Piegl und Tiller (1997, S.364f.)). In der wiederum sortierten Punktwolke, bestehend aus r Punkten, werden die $r - 1$ Sehnenlängen zwischen den benachbarten Punkten berechnet. Die Abstände Δ_k werden somit durch die Sehnenlänge definiert (siehe Formel 3.41).

$$\Delta_k = |\mathbf{Q}_{k+1} - \mathbf{Q}_k| \quad (3.41)$$

Die weitere Berechnung erfolgt mit den Formeln 3.39 und 3.40.

Piegl und Tiller (1997, S.365) bezeichnen die chordale Methode als die am meisten verwendete Methode zur Parametrisierung. Im Allgemeinen liefert diese Methode adäquate Ergebnisse.

3.2.1.3 Zentripetal

Als dritte Parametrisierungsmethode schlagen Piegl und Tiller (1997, S.365) die von Lee (1989) entwickelte zentripetale Parametrisierungsmethode vor, bei der die Quadratwurzel der Sehnenlänge berechnet und zugeordnet wird. Die Methode ist dabei identisch zu der im vorherigen Abschnitt beschriebenen chordalen Parametrisierungsmethode mit dem Unterschied, dass anstatt der Sehnenlänge die Quadratwurzel der Sehnenlänge zur Berechnung der Abstände Δ_k verwendet wird (siehe Formel 3.42).

$$\Delta_k = \sqrt{|\mathbf{Q}_{k+1} - \mathbf{Q}_k|} \quad (3.42)$$

Die weitere Berechnung erfolgt mit den Formeln 3.39 und 3.40.

In der Literatur wird die zentripetale Parametrisierungsmethode als einfache, aber gleichzeitig sehr gute Methode zur Parametrisierung klassifiziert.

An dieser Stelle sei auf die Arbeit von Fang und Hung (2013) hingewiesen, die die zentripetale Parametrisierungsmethode um einen zusätzlichen Term erweitern, durch den Richtungsänderungen hintereinander liegender Punkte berücksichtigt werden.

3.2.1.4 Affin invariant

Die affin invariante Parametrisierungsmethode wurde von Foley und Nielson (1989) sowie Nielson und Foley (1989) entwickelt und ebenfalls in Hoschek und Lasser (1992, S.195f.) beschrieben. Bei der affin invarianten Parametrisierung wird der Abstand Δ_k zwischen zwei Punkten \mathbf{Q}_k und \mathbf{Q}_{k+1} über

$$\Delta_k = M[\mathbf{P}](\mathbf{Q}_k, \mathbf{Q}_{k+1}) = \sqrt{(\mathbf{Q}_k - \mathbf{Q}_{k+1})\mathbf{Q}_\sigma(\mathbf{Q}_k - \mathbf{Q}_{k+1})^T} \quad (3.43)$$

berechnet. Formel 3.43 wurde ursprünglich in Nielson (1987) entwickelt und wird deshalb auch als Nielson-Metrik bezeichnet.

Die Koeffizientenmatrix \mathbf{Q}_σ wird über die Standardabweichungen⁶ der gegebenen (hier zweidimensionalen) Punktwolke berechnet:

$$\sigma_x = \frac{\sum_{i=1}^r (Q_{x,i} - \bar{Q}_x)^2}{r}, \quad \sigma_y = \frac{\sum_{i=1}^r (Q_{y,i} - \bar{Q}_y)^2}{r}, \quad \sigma_{xy} = \frac{\sum_{i=1}^r (Q_{x,i} - \bar{Q}_x)(Q_{y,i} - \bar{Q}_y)}{r} \quad (3.44)$$

mit

$$\bar{Q}_x = \frac{\sum_{i=1}^r Q_{x,i}}{r}, \quad \bar{Q}_y = \frac{\sum_{i=1}^r Q_{y,i}}{r}. \quad (3.45)$$

\mathbf{Q}_σ wird dann schlussendlich über

$$\mathbf{Q}_\sigma = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} = \begin{bmatrix} \frac{\sigma_y}{d_\sigma} & \frac{-\sigma_{xy}}{d_\sigma} \\ \frac{-\sigma_{xy}}{d_\sigma} & \frac{\sigma_x}{d_\sigma} \end{bmatrix} \quad (3.46)$$

mit $d_\sigma = \sigma_x \sigma_y - (\sigma_{xy})^2$ aufgestellt. Die weitere Berechnung erfolgt mit den Formeln 3.39 und 3.40.

3.2.1.5 Foley-Methode

Diese von Foley und Nielson (1989) entwickelte und ebenfalls in Hoschek und Lasser (1992, S.195f.) sowie Farin (2002b, S.163) beschriebene Parametrisierungsmethode berücksichtigt neben den Abständen auch Winkeländerungen. Diese Methode wird als „Foley-Methode“ bezeichnet.

Zunächst werden die Winkel $\hat{\Theta}_k$ berechnet:

$$\hat{\Theta}_k = \min(\pi - \Theta_k, \frac{\pi}{2}). \quad (3.47)$$

Der in Formel 3.47 eingehende Winkel Θ_k (siehe Abbildung 3.1) wird nach Lim (1999) über

$$\Theta_k = \pi - \arccos \left[\frac{d_{k-1}^2 + d_k^2 - M^2[\mathbf{P}](\mathbf{Q}_{k-1}, \mathbf{Q}_{k+1})}{2d_k d_{k-1}} \right] \quad (3.48)$$

berechnet, wobei $M^2[\mathbf{P}](\mathbf{Q}_{k-1}, \mathbf{Q}_{k+1})$ die quadrierte Nielson-Metrik zwischen Punkt \mathbf{Q}_{k-1} und \mathbf{Q}_{k+1} bezeichnet (siehe Formel 3.43). Ursprünglich wurde für die Methode vorgeschlagen, die Nielson-Metrik (siehe Formel 3.43) für d_k einzusetzen (siehe Formel 3.49).

$$d_k = \sqrt{(\mathbf{Q}_k - \mathbf{Q}_{k+1}) \mathbf{Q}_\sigma (\mathbf{Q}_k - \mathbf{Q}_{k+1})^T} \quad (3.49)$$

Nach Hoschek und Lasser (1992, S.196) ist es auch möglich, die Sehnenlänge (Formel 3.41) oder die Wurzel der Sehnenlänge (Formel 3.42) analog zu Formel 3.49 für d_k einzusetzen.

Die Abstände Δ_k werden schlussendlich über

$$\Delta_1 = d_1 \left[1 + \frac{3}{2} \frac{\hat{\Theta}_2 d_2}{d_1 + d_2} \right], \quad (3.50)$$

$$\Delta_k = d_k \left[1 + \frac{3}{2} \frac{\hat{\Theta}_k d_{k-1}}{d_{k-1} + d_k} + \frac{3}{2} \frac{\hat{\Theta}_{k+1} d_{k+1}}{d_k + d_{k+1}} \right], \quad \text{für } k = 2, \dots, r-2 \quad (3.51)$$

⁶Warum in der Literatur (Hoschek und Lasser (1992, S.195f.)) der Begriff Standardabweichung und dessen Formelzeichen und nicht der Begriff Varianz und dessen Formelzeichen verwendet wird und warum bei der Berechnung der Standardabweichungen bzw. Varianz der Nenner r und nicht wie üblich $r-1$ lautet, konnte im Rahmen dieser Arbeit nicht nachvollzogen werden.

$$\Delta_{r-1} = d_{r-1} \left[1 + \frac{3}{2} \frac{\hat{\Theta}_{r-1} d_{r-2}}{d_{r-2} + d_{r-1}} \right] \quad (3.52)$$

berechnet. Die weitere Berechnung erfolgt mit den Formeln 3.39 und 3.40.

Die aus Foley und Nielson (1989) adaptierte Abbildung 3.1 stellt die für die Berechnung entscheidenden Einflussgrößen dar.

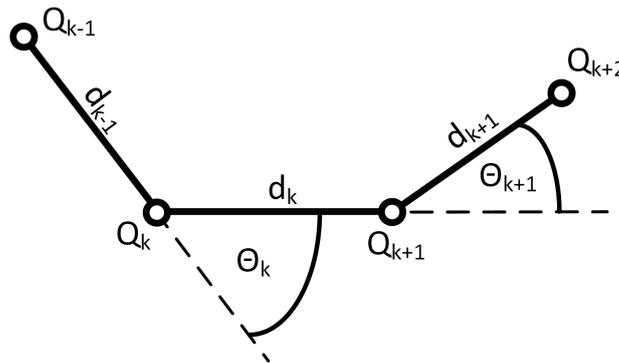


Abbildung 3.1: Einflussgrößen der Foley-Methode (adaptiert aus Foley und Nielson (1989))

3.2.2 Iterative Methoden

In der Literatur wurden zahlreiche Ansätze, im Rahmen dieser Arbeit als iterative Methoden⁷ bezeichnet, entwickelt, bei denen die Ortsparameter mit den Kontrollpunkten gemeinsam bestimmt wurden. Das Ziel dieser Ansätze ist es, im Falle von B-Spline-Kurven jeweils den Ortsparameter \bar{u}_i zu finden, bei dem der Abstand zwischen dem ausgeglichenen Punkt auf der B-Spline-Kurve $\mathbf{C}(\bar{u}_i)$ und dem zugehörigen Datenpunkt \mathbf{Q}_i minimal wird. Im Folgenden sollen die Grundzüge verschiedener Ansätze, die dieses nicht-lineare Optimierungsproblem lösen, erläutert werden.

Sarkar und Menq (1991) beschreiben einen Algorithmus, bei dem Kontrollpunkte und Ortsparameter von B-Spline-Kurven und -Flächen gemeinsam bestimmt werden. Nach Modellwahl und Knotenvektorwahl im Vorhinein, formulieren Sarkar und Menq (1991) eine explizite Fehlerfunktion, die dann mit Hilfe eines modifizierten Levenberg-Marquardt-Algorithmus und einer „Active set Strategie“ minimiert wird. Als Vorteil führen Sarkar und Menq (1991) an, dass die Kontrollpunktschätzung nur einmal nach der Optimierung der Ortsparameter durchgeführt werden muss und nicht nach jeder Iteration.

In Hoschek (1988) wird ein Ansatz vorgestellt, bei dem zunächst eine initiale Parametrisierung (z.B. mit der chordalen Parameterisierung) erfolgt. Anschließend wird der Knotenvektor bestimmt und es werden die Kontrollpunkte geschätzt, womit die initiale B-Spline-Kurve bestimmt ist. An jedem initialen Ortsparameter \bar{u}_i wird der Abweichungsvektor ϵ_i zwischen Originalpunkt \mathbf{Q}_i und geschätztem Punkt $\mathbf{C}(\bar{u}_i)$ auf der B-Spline-Kurve, sowie die erste Ableitung bzw. Tangente der B-Spline-Kurve bestimmt (siehe Abbildung 3.2). Die Länge des auf die Tangente projizierten Abweichungsvektors Δc_i bildet nach Division durch die Gesamtlänge d_{sum} (aus Formel 3.39) den Korrekturterm zum initialen Ortsparameter \bar{u}_i (siehe Formel 3.53):

$$\bar{u}_{i,\text{neu}} = \bar{u}_i + \frac{\Delta c_i}{d_{\text{sum}}}. \quad (3.53)$$

⁷Diese Ansätze werden in der Literatur vielfach auch als „Parameter-Optimierung“ bzw. im Englischen als „parameter optimization“ bezeichnet

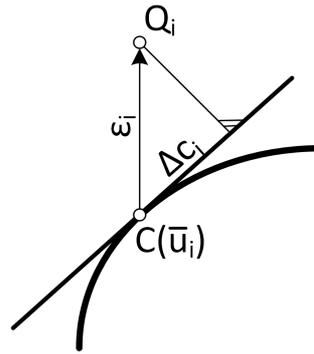


Abbildung 3.2: Projizierter Abweichungsvektor bei der Hoschek-Methode (adaptiert aus Hoschek (1988))

Mit den korrigierten Ortsparametern \bar{u}_{neu} wird dieser Prozess wiederholt. Ziel dieser Methode ist es, dass der Abweichungsvektor ϵ_i senkrecht auf der B-Spline-Kurve steht. Hoschek (1988) schlägt vor, den Winkel zwischen Abweichungsvektor und Tangente als Abbruchkriterium einzuführen und den iterativen Prozess abzubrechen, wenn die maximale Abweichung dieses Winkels von 90° unter einem bestimmten Schwellwert liegt.

Ma und Kruth (1995) beschreiben einen ähnlichen Ansatz, der sowohl für B-Spline- als auch NURBS-Kurven und Flächen geeignet ist. Nach einer initialen Bestimmung der Ortsparameter (z.B. mit der chordalen Parametrisierung) werden diese iterativ verbessert. Dazu werden die zu approximierenden Punkte nach erfolgter Ausgleichung und Bestimmung der initialen B-Spline-Kurve (basierend auf der initialen Parametrisierung) auf eben diese, im Folgenden Basis-Kurve genannt, projiziert. Die initialen Ortsparameter werden anschließend durch die Ortsparameter der auf die Basis-Kurve projizierten Punkte ersetzt. Mit den verbesserten Ortsparametern wird die Ausgleichung erneut durchgeführt und eine neue, als Basis-Kurve fungierende B-Spline-Kurve bestimmt. Dieser Prozess wird dann so lange wiederholt, bis ein geeignetes Abbruchkriterium erfüllt wird.

Lai und Lu (1996) beschreiben einen Algorithmus, bei dem Kontrollpunkte und Ortsparameter von NURBS-Flächen⁸ gemeinsam bestimmt werden. Während die Modellwahl, die Knotenvektorwahl und im Falle von NURBS die Bestimmung der Gewichte der Kontrollpunkte im Vorhinein durchgeführt werden, werden die Ortsparameter nach einer initialen Bestimmung (z.B. mit der chordalen Parametrisierung, siehe 3.2.1.2) iterativ verbessert. Mit jeder Iteration werden die Kontrollpunkte neu geschätzt. Mit den dadurch gewonnenen verbesserten Ortsparametern werden wiederum die Kontrollpunkte neu geschätzt. Dieser Prozess wird so lange wiederholt, bis der durchschnittliche Abstand zwischen $C(\bar{u}_i)$ und Q_i einen bestimmten, zu Beginn gewählten Schwellwert unterschreitet.

3.2.3 Zusammenfassung Parametrisierung

Ziel der Parametrisierung ist es, für jeden Punkt der zu approximierenden Punktwolke einen Ortsparameter zu berechnen. Die in Tabelle 3.2 zusammengefassten Ansätze zur Berechnung der Ortsparameter können grundsätzlich in zwei Gruppen unterteilt werden. Unter die wenig rechenaufwendigen deterministischen Methoden fallen die häufig verwendete chordale und zentripetale Parametrisierung.

Die iterativen Methoden erfordern in der Regel einen höheren Rechenaufwand, erzielen aber durch die iterative Verbesserung des Ortsparametervektors auch bessere Ergebnisse.

⁸Der Ansatz kann analog auf B-Spline-Kurven und -Flächen angewendet werden.

Tabelle 3.2: Zusammenfassung der Parametrisierung bei B-Splines

Methoden	Literatur	
Deterministisch	Gleichabständig	Piegl und Tiller (1997, S.364)
	Chordal	Piegl und Tiller (1997, S.364f.)
	Zentripetal	Lee (1989); Piegl und Tiller (1997, S.365)
	Affin invariant	Foley und Nielson (1989); Nielson und Foley (1989); Hoschek und Lasser (1992, S.195f.)
	Foley-Methode	Foley und Nielson (1989); Hoschek und Lasser (1992, S.195f.); Farin (2002b, S.163)
Iterativ	Hoschek (1988); Sarkar und Menq (1991); Ma und Kruth (1995); Lai und Lu (1996)	

3.3 Knotenvektorwahl

Wie bereits in Kapitel 2.1.2.3 beschrieben, besteht der Knotenvektor ξ aus $n + p + 2$ monoton steigenden Knoten (siehe Formel 2.20). Beginn und Ende des Knotenvektors wird durch die externen Knoten ξ_{ext} (siehe Formel 2.20 und 2.21) gebildet. Die $p + 1$ externen Knoten am Anfang des Knotenvektors erhalten den Wert 0 und die $p + 1$ externen Knoten am Ende des Knotenvektors erhalten den Wert 1. Die optimale Platzierung der $n - p$ dazwischen liegenden und für das Erscheinungsbild des B-Splines wichtigen internen Knoten ξ_{int} hingegen gestaltet sich deutlich schwieriger, da es sich dabei um ein multimodales und multivariates, kontinuierliches, nichtlineares Optimierungsproblem handelt (siehe Dierckx (1993) und Gálvez u. a. (2015)). Das Finden einer Lösung für dieses so genannte „Knoten-Optimierungs-Problem“ wird zusätzlich erschwert, wenn die zu approximierende Punktwolke größere Datenlücken enthält. Die Lage der Knoten bestimmt nämlich unter anderem auch die Stetigkeit der B-Spline-Kurve. Eine B-Spline-Kurve ist an einer bestimmten Stelle mindestens $p - k$ -fach differenzierbar, wobei k für die Multiplizität eines Knoten (also die Anzahl mehrfacher Knoten) an dieser Stelle steht. D.h. eine B-Spline-Kurve mit dem Basisfunktionsgrad $p = 3$ hat an der Stelle $u = 0.5$ nur C^1 -Stetigkeit, wenn zwei Knoten $\xi_i = \xi_{i+1} = 0.5$ existieren und die Multiplizität an dieser Stelle somit $k = 2$ beträgt (zum Thema Stetigkeit siehe auch Piegl und Tiller (1997)). Bei der Approximation von Datensätzen mit großen Datenlücken kommt eine der Stetigkeit eng verwandte Problematik vor. Werden mehr als p Knoten innerhalb der Datenlücke platziert, enthält die Designmatrix (siehe Formel 3.58) eine Spalte mit Nullen und die Normalgleichungsmatrix bei der Kontrollpunktschätzung wird singular.

In den letzten Jahren bzw. Jahrzehnten wurden viele verschiedene Algorithmen zu Bestimmung des Knotenvektors entwickelt. Alle Algorithmen müssen dabei zwei Aufgaben bewältigen:

1. Die Qualität der Approximation soll möglichst gut sein.
2. Die numerische Stabilität bei der Kontrollpunktschätzung (siehe Kapitel 3.4) in Form einer gut konditionierten Normalgleichungsmatrix muss gegeben sein.

Die Algorithmen können grundsätzlich in deterministische und heuristische Algorithmen untergliedert werden. Die hier im Folgenden beschriebenen Algorithmen für B-Spline-Kurven lassen sich zumeist direkt oder durch einfache Adaption auch auf B-Spline-Flächen anwenden. Wie in Kapitel 3.2 werden dabei nur Algorithmen vorgestellt, die auch für die Approximation geeignet sind.

3.3.1 Deterministische Methoden

Bei deterministischen Algorithmen treten immer reproduzierbare Zustände auf, d.h. bei gleichen Eingangsgrößen werden identische Ausgangsgrößen erhalten. Im Falle der Knotenvektorwahl bedeu-

tet dies, dass aus der zu approximierenden Punktwolke eine oder mehrere Eingangsgrößen gewählt oder abgeleitet werden und nur anhand dieser Eingangsgrößen der Knotenvektor als Ausgangsgröße bestimmt wird. Deterministische Algorithmen zeichnen sich bei der Knotenvektorwahl durch eine geringe Rechenzeit und in der Regel gut konditionierte Normalgleichungsmatrizen aus. Allerdings führen die erstellten Knotenvektoren, insbesondere im Fall von komplexen Punktwolken, in der Regel zu einer geringeren Qualität der Approximation.

3.3.1.1 Cardinal B-Splines

Das denkbar einfachste Verfahren zur Wahl des Knotenvektors ist die gleichabständige Verteilung der internen Knoten über den Ortsparameter-Bereich zwischen 0 und 1. B-Splines mit gleichabständigen Knoten werden als Cardinal B-Splines bezeichnet (de Boor (2001, S.89)). Der j -te interne Knoten wird wie folgt berechnet:

$$\xi_{\text{int},j-1} = \frac{j}{(n-p)+1} \text{ mit } j = 1, \dots, n-p. \quad (3.54)$$

3.3.1.2 Knoten-Platzierungs-Methode (PT1)

Eine oft verwendete und zitierte Methode zur Bestimmung des Knotenvektors ist die Knoten-Platzierungs-Methode⁹ (PT1) aus Pieggl und Tiller (1997, S.412). Diese Methode kann auch als Standardverfahren zur Bestimmung des Knotenvektors bezeichnet werden. Bei PT1 werden die internen Knoten in Abhängigkeit der in der vorhergehenden Parametrisierung bestimmten Ortsparameter berechnet. Ziel dieser Methode ist es, dass in jedem Bereich zwischen zwei Knoten, Knotenspanne genannt, mindestens ein Ortsparameter aus $\bar{\mathbf{u}}$ liegt. Dadurch ergibt sich bei der Kontrollpunktschätzung (siehe 3.4) eine gut konditionierte Normalgleichungsmatrix.

Nach Pieggl und Tiller (1997, S. 412) wird der j -te interne Knoten über Formelapparat 3.55 berechnet, wobei r wieder für die Anzahl der zu approximierenden Punkte steht.

$$\begin{aligned} d &= \frac{r}{n-p+1} \\ i &= \text{int}(jd) \\ \alpha &= jd - i \\ \xi_{\text{int},j-1} &= (1-\alpha)\bar{u}_{i-1} + \alpha\bar{u}_i \text{ mit } j = 1, \dots, n-p \end{aligned} \quad (3.55)$$

Ein ähnlicher Ansatz wird in Ma und Kruth (1995) vorgestellt. Die Anzahl der Ortsparameter, die zur Berechnung eines Knotens herangezogen werden, kann bei Ma und Kruth (1995) über eine wählbare Konstante gesteuert werden. Bei PT1 werden konstant zwei Ortsparameter zur Berechnung eines Knotens herangezogen.

In Pieggl und Tiller (2000) wird eine modifizierte Variante der „Knoten-Platzierungs-Methode“ beschrieben. Auch bei diesem Algorithmus werden die Knoten in Abhängigkeit der Ortsparameter gewählt.

3.3.1.3 Dominante Punkte

Ein Algorithmus, bei dem in der zu approximierenden Punktwolke dominante Punkte selektiert werden, wird von Park und Lee (2007) beschrieben. Dazu wird eine initiale Basiskurve durch die Punktwolke geschätzt. Die Ortsparameter, bei denen die Basiskurve überdurchschnittlich gekrümmt ist, werden in einer Liste gespeichert. Zusammen mit den Endpunkten der Punktwolke werden alle bzw. ein Teil der Punkte dieser Liste verwendet, um den Knotenvektor zu bestimmen. Dies erfolgt über eine Adaption der bei der Interpolation von B-Spline-Kurven zum Einsatz kommenden Average-Methode (siehe z.B. auch Pieggl und Tiller (1997, S.365)). Iterativ werden anschließend

⁹Im Englischen als *knot placement technique* bezeichnet

weitere dominante Punkte hinzugefügt, der Knotenvektor neu bestimmt und die Kurve erneut geschätzt. Die dominanten Punkte werden mit Hilfe eines „Shape-Index“ in den Segmenten mit der größten Abweichung zwischen Punktwolke und Kurve bestimmt und hinzugefügt. Der Prozess wiederholt sich, bis die Anzahl der dominanten Punkte der Anzahl der internen Knoten entspricht. Abschließend wird der Knotenvektor mit der finalen Menge an dominanten Punkten unter Verwendung der Adaption der Average-Methode bestimmt.

3.3.2 Heuristische Methoden

Heuristische Algorithmen liefern aufgrund von zufällig gewählten Variablen oder Eingangsgrößen (im Mittel) eine Ausgangsgröße. Im Falle der Knotenvektorwahl bedeutet dies, dass der Knotenvektor als Eingangsgröße zufällig gewählt und die Qualität des Ergebnisses aufgrund eines gewählten Qualitätsmaßes, der Fitnessfunktion, quantifiziert wird. Über die Fitnessfunktion wird für jeden Knotenvektor ein Fitnesswert berechnet. Dieser Prozess wird dann mehrfach wiederholt und schlussendlich wird die Eingangsgröße als Ausgangsgröße gewählt, die das beste Ergebnis, also den besten Fitnesswert geliefert hat. Da dieser Prozess sehr rechenaufwendig ist, wurden zahlreiche Algorithmen entwickelt, die schon mit einer geringen Anzahl an Wiederholungen zu sehr guten Ergebnissen kommen. Trotzdem sind heuristische Algorithmen rechenaufwendiger als deterministische Algorithmen. Allerdings liefern sie im Falle der Knotenvektorwahl auch deutlich bessere Ergebnisse. Ein häufig verwendete Fitnessfunktion ist die aus der Kontrollpunktschätzung resultierende Berechnung der Residuenquadratsumme Ω (siehe Formel 2.34 oder 4.1), wobei Ω dann direkt als Fitnesswert fungiert. Vom Sprachgebrauch her ist ein hoher Fitnesswert besser als ein niedriger Fitnesswert. Wählt man Ω als Fitnesswert, muss man dieses Verhältnis per Definition umdrehen, da ein kleines Ω besser als ein großes Ω ist.

3.3.2.1 Elitist clonal selection algorithm

In Gálvez u. a. (2015) wird der so genannte „elitist clonal selection algorithm“, im weiteren CSA genannt, vorgestellt. Dieser Algorithmus imitiert das menschliche Immunsystem, wobei die zu approximierende Punktwolke das Antigen und die Knotenvektoren die Antikörper darstellen. Nach einer initialen, auf der Gleichverteilung beruhenden, Wahl einer bestimmten Anzahl von Knotenvektoren (Population \mathcal{P}_N) werden diese verwendet, um die Punktwolke zu approximieren. Im Anschluss werden dann diejenigen Knotenvektoren geklont, die den besten Fitnesswert erzielt haben. Die geklonten Knotenvektoren werden dann durch Mutation, d.h. in diesem Ansatz eine kleine zufällige Änderung an beliebiger Stelle, leicht modifiziert (Population \mathcal{P}_M) und wiederum verwendet, um die Punktwolke zu approximieren. Wiederum basierend auf den Fitnesswerten werden die besten Knotenvektoren für die nächste Generation (neue Population \mathcal{P}_N) ausgewählt und dieser Prozess über eine bestimmte Anzahl an Generationen wiederholt. Um sowohl gute Ergebnisse zu behalten, als auch eine Konvergenz in ein lokales Minimum zu vermeiden, werden der Population \mathcal{P}_M die besten Knotenvektoren der vorherigen Generation und eine gewisse Anzahl zufällig über die Gleichverteilung gewählten Knotenvektoren hinzugefügt.

3.3.2.2 Estimation-of-distribution-Ansätze

Die Estimation-of-distribution-Ansätze haben die Grundidee, dass die bisher besten Knotenvektoren dazu verwendet werden, eine (Art) Knoten-Verteilung zu berechnen und kommende Knotenvektoren aus dieser Verteilung zu ziehen.

Zhao u. a. (2011) stellen einen Ansatz vor, bei dem die Knotenpositionen durch eine gaußsche Mischverteilung beschrieben werden. Diese Mischverteilung wird iterativ aus den besten zuvor generierten Knotenvektoren berechnet. Knotenvektoren der folgenden Generationen werden aus der geschätzten Mischverteilung gezogen.

Initial wird der Knotenvektor mit der PT1 (siehe Kapitel 3.3.1.2) berechnet. Den berechneten Knoten wird jeweils ein Intervall zugewiesen, dessen untere Grenze der Mittelwert aus dem aktuellen

und dem nächst kleineren Knoten und dessen obere Grenze der Mittelwert aus dem aktuellen und dem nächst größeren Knoten gebildet wird. In den Intervallen wird jeweils ein Knoten zufällig gezogen und zu einem Knotenvektor zusammengefügt. Dieser Vorgang wird mehrfach wiederholt und die so generierten Knotenvektoren bilden die initiale Population (Population \mathcal{P}_N). Die Knotenvektoren dieser Population werden verwendet, um jeweils eine ausgleichende B-Spline-Kurve zu berechnen. Auf Basis der Fitnessfunktion wird für jeden Knotenvektor der Population \mathcal{P}_N ein Fitnesswert berechnet. Basierend auf diesen Fitnesswerten werden die besten Knotenvektoren ausgewählt und der Population \mathcal{P}_M zugeordnet. Auf Basis von \mathcal{P}_M werden mit Hilfe einer *k-means*-Clustering und dem Expectation-Maximization-Algorithmus die Parameter einer gaußschen Mischverteilung geschätzt. Basierend auf dieser Verteilung werden nun neue Knotenvektoren generiert, die die neue Population \mathcal{P}_N bilden. Dieser Prozess wird so lange wiederholt, bis ein zuvor gewähltes (aber nicht näher erläutertes) Stop-Kriterium erreicht wird.

Einen ähnlichen Ansatz verfolgen Bureick u. a. (2016a). In dem Ansatz wird zunächst der gesamte Bereich, in dem Knoten liegen dürfen, in kleine Intervalle unterteilt. Jedem dieser Intervalle wird eine Wahrscheinlichkeit zugeordnet. Zu Beginn des Algorithmus werden allen Intervallen die gleiche Wahrscheinlichkeit zugeordnet. Im weiteren wird die Gesamtheit der den Intervallen zugeordneten Wahrscheinlichkeiten mit \mathbf{w} bezeichnet. Mit Hilfe eines Resampling-Schrittes, wie er z.B. bei Partikelfiltern verwendet wird, werden aus \mathbf{w} neue Knotenvektoren generiert. Die Knotenvektoren mit dem besten Fitnesswert werden in einer Bestenliste gespeichert. Nach einer bestimmten Anzahl an Generationen wird \mathbf{w} aus den Knotenvektoren der Bestenliste berechnet¹⁰. Dazu wird die Wahrscheinlichkeit für ein Intervall erhöht, in dem ein Knoten eines Knotenvektors der Bestenliste platziert ist. Die Erhöhung erfolgt proportional zum Fitnesswert. Um ein Konvergieren des Algorithmus in ein lokales Optimum zu verhindern, wird ein bestimmter kleiner Prozentsatz der neuen Knotenvektoren aus einem \mathbf{w} mit gleichen Wahrscheinlichkeiten wie zu Beginn des Algorithmus generiert. Eine modifizierte Version dieses Ansatzes, im weiteren evolutionäre Monte-Carlo (EMC)-Methode genannt, wird in Kapitel 4.2 ausführlich beschrieben.

3.3.2.3 Genetische Algorithmen

Yoshimoto u. a. (1999) stellen einen Ansatz vor, bei dem sowohl die Größe des Knotenvektors als auch die Position der Knoten gemeinsam bestimmt werden. Dazu verwenden sie einen genetischen Algorithmus, der in seiner Funktionsweise die Evolution in der Natur imitiert. Dazu wird der gesamte Bereich, in dem Knoten liegen dürfen, in kleine Intervalle unterteilt. Die Anzahl dieser Intervalle wird als Gen-Länge bezeichnet. Jedes dieser Intervalle wird binär codiert und kann somit entweder den Wert 0 oder 1 annehmen. Nimmt ein Intervall den Wert 1 an, wird an dieser Stelle ein (interner) Knoten gesetzt. Nimmt es den Wert 0 an, wird an dieser Stelle entsprechend kein Knoten gesetzt. Der Binär-String, der alle binär codierten Intervalle enthält, wird als Chromosom bezeichnet und repräsentiert den Knotenvektor. Zunächst werden zufällig binäre Chromosomen erstellt, wobei die absolute Anzahl an Einsen (und somit auch der Nullen) über eine wählbare relative Konstante gesteuert werden kann. Die Gesamtheit der erzeugten Chromosomen bildet die initiale Population (Population \mathcal{P}_N). Mit jedem Chromosom bzw. Knotenvektor wird eine ausgleichende B-Spline-Kurve geschätzt und das Ergebnis mit einer Fitnessfunktion bewertet. Yoshimoto u. a. (1999) verwenden das AIC als Fitnessfunktion (siehe Kapitel 3.1.1.1). Aus der Population werden über einen Turniermodus die Chromosomen mit den besten Fitnesswerten selektiert (Population \mathcal{P}_M). Aus \mathcal{P}_M wird durch Rekombination und Mutation¹¹ eine neue Population generiert. Rekombination bezeichnet das Zusammenfügen von Teilen zweier Elternchromosomen zu einem neuen Chromosom. Mutation bedeutet bei diesem Ansatz das zufällige Ändern eines Chromosoms an beliebiger Stelle. Diese Population bildet die neue Population \mathcal{P}_N . Dieser Prozess wird so lange wiederholt, bis das

¹⁰In Bureick u. a. (2016a) sind noch zwei weitere Varianten beschrieben, bei denen \mathbf{w} entweder aus den Ortsparametern oder der Krümmung berechnet wird. Die damit erzielten Ergebnisse sind aber deutlich schlechter als mit der Variante, bei der die Knotenvektoren der Bestenliste zur Berechnung von \mathbf{w} verwendet werden.

¹¹Die Häufigkeit von Rekombination und Mutation kann über wählbare Wahrscheinlichkeiten gesteuert werden.

Ergebnis konvergiert, d.h. der Fitnesswert des besten Chromosoms der aktuellen Population \mathcal{P}_N über mehrere Generationen nicht mehr besser ist als der Fitnesswert des besten Chromosoms der vorherigen Population \mathcal{P}_N .

Sarfraz und Raza (2001) adaptieren den oben beschriebenen genetischen Algorithmus, in dem sie signifikante Punkte in der Punktwolke identifizieren und an diesen Stellen im Binär-String eine 1 und damit einen Knoten setzen.

Da die oben beschriebenen genetischen Algorithmen durch die Unterteilung des gesamten Bereichs in kleine Intervalle ein kontinuierliches Problem in ein diskretes Problem umwandeln, entstehen Diskretisierungsfehler, deren Größenordnung umgekehrt proportional zur Gen-Länge ist.

Dieses Problems nahmen sich die Autoren an und stellen in Yoshimoto u. a. (2003) eine modifizierte Variante vor, bei der die Knotenpositionen nicht über einen Binär-String repräsentiert werden, sondern direkt als reale Zahlen gespeichert werden. Dies bedeutet, dass die Knotenpositionen als kontinuierliche Variablen behandelt werden. Die Wahrscheinlichkeiten für Rekombination und Mutation werden in Yoshimoto u. a. (2003) für jedes „Elternchromosomenpaar“ bzw. Chromosom individuell in Abhängigkeit vom Fitnesswert berechnet und als Fitnessfunktion wird das BIC verwendet.

Spiriti u. a. (2013) stellen ebenfalls einen genetischen Algorithmus vor, bei dem die Knotenpositionen als kontinuierliche Variablen behandelt werden. Während Yoshimoto u. a. (2003) die Mutation eines Knotens im gesamten Ortsparameterbereich zwischen 0 und 1 zulassen, begrenzen Spiriti u. a. (2013) die Mutation eines Knotens auf einen Bereich zwischen den nächstgelegenen Knoten. Aus diesem Grunde erzielt der genetische Algorithmus nach Spiriti u. a. (2013) deutlich bessere Resultate.

In Bureick u. a. (2019) wurde ein elitärer genetischer Algorithmus vorgestellt. Dieser wird im Detail in Kapitel 4.3 erläutert und im Rahmen dieser Arbeit an zwei Stellen modifiziert (siehe Kapitel 4.3.2).

3.3.3 Zusammenfassung Knotenvektorwahl

Ziel der Knotenvektorwahl ist es, die Verknüpfungsstellen zwischen den polynomialen Funktionen möglichst optimal für die zu approximierende Punktwolke zu bestimmen.

Die in Tabelle 3.3 zusammengefassten Methoden zur Knotenvektorwahl lassen sich in zwei Gruppen unterteilen. Die deterministischen Methoden verwenden eine oder mehrere auch abgeleitete Eigenschaften der zu approximierenden Punktwolke, um einmalig einen Knotenvektor zu berechnen. Aus diesem Grund sind sie wenig rechenaufwendig. Die in verschiedene Untergruppen aufteilbaren heuristischen Methoden erzeugen viele verschiedene Lösungen, um schlussendlich die beste Lösung als finalen Knotenvektor zu wählen. Sie sind in der Regel deutlich rechenaufwendiger als die deterministischen Methoden, führen aber zumeist auch zu deutlich besseren Approximationsergebnissen. Es ist sehr schwierig eine allgemeingültige Aussage zu treffen, welche der erwähnten Methoden

Tabelle 3.3: Zusammenfassung Knotenvektorwahl bei B-Splines

	Methoden	Literatur
Deterministisch	Cardinal B-Spline	de Boor (2001, S.89)
	Knoten-Platzierungs-Methode	Piegl und Tiller (1997, S.412)
	Dominante Punkte	Park und Lee (2007)
Heuristisch	Elitist clonal selection algorithm	Gálvez u. a. (2015)
	Estimation-of-distribution	Zhao u. a. (2011); Bureick u. a. (2016a)
	Genetische Algorithmen	Yoshimoto u. a. (1999, 2003); Spiriti u. a. (2013); Bureick u. a. (2019)

„die Beste“ ist. Die Leistungsfähigkeit der Methoden hängt zum Teil von der bereits in Kapitel 1 erwähnten Datencharakteristik sowie den gestellten Anforderungen an den B-Spline und den Approximationsprozess ab.

Unter Datencharakteristik fällt zum einen die Komplexität der zu approximierenden Punktwolke bzw. dessen zugrunde liegenden Objektes. Insbesondere Sprünge, Kanten und Knicke führen vor allem bei den deterministischen Methoden zu unerwünschten Approximationsergebnissen. Zum anderen fällt auch die Unsicherheit der zu approximierenden Punktwolke unter die Datencharakteristik. Die Leistungsfähigkeit einiger Methoden wird stark vom Rauschen der zu approximierenden Punktwolke beeinflusst.

Eine häufig gestellte Anforderung an eine B-Spline-Kurve ist die Einhaltung einer bestimmten Stetigkeit. So gibt es Methoden, bei denen die Generation multipler Knoten verhindert wird, um für die gesamte B-Spline-Kurve eine bestimmte Stetigkeit zu garantieren.

Eine häufig gestellte Anforderung an den Approximationsprozess ist die Einhaltung einer bestimmten Rechenzeit. Soll der Approximationsprozess echtzeitfähig oder nahezu echtzeitfähig sein, scheidet der Einsatz heuristischer Methoden in der Regel aus.

3.4 Kontrollpunktschätzung

Im abschließenden Schritt der B-Spline-Approximation wird die Lage der Kontrollpunkte geschätzt. Beim klassischen Ansatz wird einzig die Position der Kontrollpunkte in einem linearen GMM geschätzt. Der klassische und im Rahmen dieser Arbeit verfolgte Ansatz ist in Kapitel 3.4.1 beschrieben. Andere Ansätze erweitern die Kontrollpunktschätzung um Nebenbedingungen oder integrieren die Optimierung weiterer Stellgrößen in die Kontrollpunktschätzung. Diese erweiterten Ansätze sind in Kapitel 3.4.2 erläutert.

3.4.1 Reine Kontrollpunktschätzung

Sind die Ortsparameter $\bar{\mathbf{u}}$ und der Knotenvektor $\boldsymbol{\xi}$ im Vorhinein bestimmt, so kann die optimale Schätzung der Position der Kontrollpunkte in einem linearen GMM wie in Formel 2.28 erfolgen. Der zu schätzende Parametervektor $\hat{\mathbf{x}}$ setzt sich dabei aus den $n + 1$ Kontrollpunkten zusammen:

$$\hat{\mathbf{x}} = [x_{x,0} \ \dots \ x_{x,n} \ x_{y,0} \ \dots \ x_{y,n} \ x_{z,0} \ \dots \ x_{z,n}]^T. \quad (3.56)$$

Im Falle von 3D-Punktwolken entspricht die Anzahl der Unbekannten $o = 3 \cdot (n + 1)$.

Der Beobachtungsvektor \mathbf{l} hat weiterhin die in Formel 3.2 dargestellte Form und besteht aus der zu approximierenden Punktwolke. Bei dreidimensionalen Daten hat der Beobachtungsvektor weiterhin die Länge $m = 3 \cdot r$.

\mathbf{A} entspricht bei der Kontrollpunktschätzung einer Blockdiagonalmatrix, die aus einer der Dimension entsprechenden Anzahl an identischen Teildesignmatrizen $\mathbf{A}_{\text{einzel}}$ zusammengesetzt wird. Im dreidimensionalen Fall lautet \mathbf{A} :

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{\text{einzel}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{\text{einzel}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{einzel}} \end{bmatrix} \quad (3.57)$$

Bei B-Spline Kurven wird $\mathbf{A}_{\text{einzel}}$ mit jeweils einer Basisfunktion $N_{i,p}(\bar{u})$ besetzt:

$$\mathbf{A}_{\text{einzel}} = \begin{bmatrix} N_{0,p}(\bar{u}_1) & \dots & N_{n,p}(\bar{u}_1) \\ \vdots & & \vdots \\ N_{0,p}(\bar{u}_r) & \dots & N_{n,p}(\bar{u}_r) \end{bmatrix} \quad (3.58)$$

In der Literatur erfolgt die Lösung des angegebenen GMM zur Kontrollpunktschätzung beinahe ausschließlich mit der Methode der kleinsten Quadrate (siehe Kapitel 2.2.1). Beispielhaft sei hierzu auf Pieggl und Tiller (1997, S.410f.) oder Koch (2009) verwiesen.

Die Anwendung der M-Schätzung im Zusammenhang mit der Kontrollpunktschätzung von B-Splines ist bisher kaum beschrieben. So wird lediglich in Bureick u. a. (2016a) die Kontrollpunktschätzung mithilfe des Huber-Schätzers beschrieben.

Koch und Kargoll (2013) verwenden einen Expectation-Maximization-Algorithmus (EM-Algorithmus) für eine robuste Kontrollpunktschätzung einer B-Spline-Fläche. Der beschriebene EM-Algorithmus führt zu einer Schätzung wie bei der Methode der iterativ regewichteten kleinsten Quadrate (Kapitel 2.2.4). Die Gewichte für die Beobachtungen werden im EM-Algorithmus als fehlende Beobachtung behandelt. In Koch und Kargoll (2013) wird angenommen, dass die Beobachtungen der t -Verteilung unterliegen. Dies führt dazu, dass neben den Parametern (Kontrollpunkten) und dem a posteriori Varianzfaktor auch der Freiheitsgrad der t -Verteilung geschätzt wird. Dadurch ergibt sich eine adaptive robuste Schätzung, bei der vorab keine Annahme über die Höhe des Ausreißeranteils getroffen werden muss.

Xu u. a. (2018) verwenden diesen EM-Algorithmus für die robuste Kontrollpunktschätzung für B-Spline-Kurven. Die B-Spline-Kurven sollen linienhafte Laserscannmessungen in einem Tunnel approximieren. Die gemessenen Profile enthalten neben den Messungen zu den gesuchten Tunnelwänden auch unerwünschte Messungen, wie z.B. zu Kabeln oder Schienen. Mit einer geeigneten Modellwahl wird der Einfluss dieser unerwünschten Messungen durch den verwendeten EM-Algorithmus nahezu eliminiert.

3.4.2 Erweiterte Kontrollpunktschätzung

Verschiedene Ansätze erweitern die in Kapitel 3.4.1 beschriebene reine Kontrollpunktschätzung um Nebenbedingungen oder um die Optimierung weiterer Stellgrößen.

Die in Kapitel 2.1.2.4 beschriebenen Smoothing oder Penalized Splines werden in der Regel über eine Nebenbedingung, die den Glattheitsterm enthält, modelliert. Dazu sei hier z.B. auf de Boor (2001, S.207ff.), O'Sullivan (1986, 1988), sowie Elfving und Andersson (1987) verwiesen.

Ferguson u. a. (1988) führen eine Nebenbedingung ein, um die Konvexität einer B-Spline-Fläche zu kontrollieren.

Bei anderen Ansätzen werden mit der Kontrollpunktschätzung weitere Stellgrößen der B-Splines optimiert. So stellen Schmitt und Neuner (2015) einen nicht linearen Ansatz vor, bei dem ausgehend von einer Bézier-Kurve iterativ Knoten bzw. Kontrollpunkte hinzugefügt werden. Abwechselnd werden dann Kontrollpunkte (wie in Kapitel 3.4.1) und die Knotenposition des neuen Knotens geschätzt.

Laurent-Gengoux und Mekhilef (1993) stellen für die Approximation von NURBS eine Kostenfunktion auf, in die neben den Kontrollpunkten der Knotenvektor, die Ortsparameter und die Gewichte der Kontrollpunkte als Parameter eingehen. Diese Kostenfunktion wird mit Hilfe der Polak-Ribière-Technik minimiert und somit Kontrollpunkte, Knotenvektor, Ortsparameter und Gewichte der Kontrollpunkte gemeinsam optimiert.

3.4.3 Zusammenfassung Kontrollpunktschätzung

Ziel der Kontrollpunktschätzung ist es, die Kontrollpunkte der B-Spline-Kurven optimal im Hinblick auf das gewählte Optimierungskriterium zu schätzen bzw. zu bestimmen.

Die, in Tabelle 3.4 zusammengefassten Ansätze der Kontrollpunktschätzung lassen sich in zwei Gruppen unterteilen. Bei der reinen Kontrollpunktschätzung werden einzig die Kontrollpunkte ohne weitere (Neben-)Bedingungen geschätzt. Die klassische Schätzung nach der Methode der kleinsten Quadrate ist dabei die am häufigsten verwendete Methode. Einige wenige Ansätze verwenden alternative Schätzer oder Algorithmen, um z.B. die Robustheit gegenüber Ausreißern zu steigern. Bei der erweiterten Kontrollpunktschätzung werden zum einen Nebenbedingungen in die Kontroll-

punktschätzung eingeführt. Diese Nebenbedingungen sollen bestimmte gewünschte Eigenschaften der B-Spline-Kurve sicherstellen. Zum anderen wird in die Kontrollpunktschätzung die Optimierung weiterer Stellgrößen der B-Spline-Approximation integriert.

Tabelle 3.4: Zusammenfassung Kontrollpunktschätzung bei B-Splines

	Methode	Literatur
Reine	Methode der kleinsten Quadrate	Piegl und Tiller (1997, S.410f.); Koch (2009)
	M-Schätzung	Bureick u. a. (2016a)
	Weitere	Koch und Kargoll (2013); Xu u. a. (2018)
Erweiterte	Nebenbedingungen	O’Sullivan (1986, 1988); Elfving und Andersson (1987); de Boor (2001, S.207ff.)
	Gleichzeitige Optimierung mehrerer Stellgrößen	Laurent-Gengoux und Mekhilef (1993); Schmitt und Neuner (2015)

4 Methodische Innovation Knotenvektorwahl

Im Rahmen dieser Arbeit werden drei neuartige Methoden zur Knotenvektorwahl präsentiert. Die erste präsentierte Methode (Kapitel 4.1) ist eine deterministische Methode, die im Rahmen dieser Arbeit erstmalig veröffentlicht wird. Die zweite präsentierte Methode (Kapitel 4.2) ist eine modifizierte Version der in Bureick u. a. (2016a) vorgestellten evolutionären Monte-Carlo Methode. Die dritte Methode (Kapitel 4.3) ist eine leicht modifizierte Variante des in Bureick u. a. (2019) vorgestellten elitären genetischen Algorithmus.

Die in diesem Kapitel beschriebenen Algorithmen werden in Kapitel 5 und 6 auf Basis numerischer Untersuchungen analysiert, evaluiert und mit weiteren Methoden zur Knotenvektorwahl verglichen. Folgende Vorbemerkungen und Definitionen gelten für alle nachfolgend beschriebenen Methoden. Ein Individuum besteht aus den internen Knoten ξ_{int} eines Knotenvektors. Für die Kontrollpunktschätzung werden die internen Knoten ξ_{int} , also ein Individuum, mit den festen externen Knoten ξ_{ext} zum Knotenvektor ξ zusammengefügt. Es muss dabei sichergestellt werden, dass ξ in Form der Formel 2.20 vorliegt.

Als Fitnesswert zur Beurteilung der Güte eines Individuums dient die Residuenquadratsumme Ω aus der Kontrollpunktschätzung (siehe hierzu auch Kapitel 3.4):

$$\Omega = \mathbf{v}^T \mathbf{P} \mathbf{v}. \quad (4.1)$$

Die Kontrollpunktschätzung zur Bestimmung des Fitnesswertes wird für jedes Individuum separat durchgeführt und ist nicht mit der auf die Knotenvektorwahl folgenden Kontrollpunktschätzung zur finalen Bestimmung der Kontrollpunkte zu verwechseln. Bei der Kontrollpunktschätzung zur Bestimmung des Fitnesswertes wird im Rahmen dieser Arbeit durchgängig die Methode der kleinsten Quadrate und Formel 4.1 zur Berechnung des Fitnesswertes verwendet. Dies gilt auch für Kapitel 6, wenn bei der Kontrollpunktschätzung zur finalen Bestimmung der Kontrollpunkte ein robuster Schätzer verwendet wird. Die robusten Schätzer werden also nicht in die Kontrollpunktschätzung zur Bestimmung des Fitnesswertes integriert. Zum einen ist dies in einer erheblichen Reduktion des Rechenaufwandes begründet, da dadurch die Anwendung eines iterativen Lösungsalgorithmus (siehe Kapitel 2.2.4) für jedes Individuum in jeder Iteration vermieden wird. Zum anderen steht bei Anwendung eines iterativen Lösungsalgorithmus kein adäquater vergleichbarer Fitnesswert zur Verfügung, der einen fairen Vergleich verschiedener Individuen zulässt.

Für die Kontrollpunktschätzung (zur Bestimmung des Fitnesswertes) werden neben dem Knotenvektor zusätzlich die Anzahl der Kontrollpunkte $n + 1$, der Grad der Basisfunktion p , der Beobachtungsvektor \mathbf{l} , die Gewichtsmatrix \mathbf{P} und der Ortsparametervektor \mathbf{u} benötigt. Folglich werden diese Größen den beschriebenen Methoden zur Knotenvektorwahl zu Beginn als Eingangsgrößen übergeben.

4.1 Residuenbasierter Iterativer Update (RIU) Algorithmus

Der neu entwickelte residuenbasierte iterative Update (RIU) Algorithmus ist ein deterministischer Algorithmus, der mit geringer Rechenzeit gute Ergebnisse liefert. Der RIU-Algorithmus kann dabei sowohl als alleinige Methode zur Knotenvektorbestimmung als auch in Kombination mit einem heuristischen Ansatz verwendet werden. So ist es denkbar, dass der RIU-Algorithmus als deterministische Methode bei der Initialisierung des elitären genetischen Algorithmus (siehe Kapitel 4.3.1.1) verwendet wird.

4.1.1 Methodik

Beim RIU-Algorithmus wird der Knotenvektor iterativ verbessert. Die Anzahl der Iterationen ist auf die Anzahl der zu bestimmenden internen Knoten $n - p$ begrenzt. Dadurch kann jeder Knoten theoretisch einmal modifiziert werden. Eine detaillierte Begründung zur Wahl dieses Abbruchkriteriums erfolgt am Ende dieses Kapitels. In jeder Iteration wird nur der Wert eines einzelnen Knotens verändert. Der neue Knotenwert wird aufgrund der geschätzten Verbesserungen aus der Kontrollpunktschätzung der vorherigen Iteration berechnet. Hierbei wird angenommen, dass die Approximationsgüte besonders gesteigert wird, wenn ein Knoten an der Position der maximalen geschätzten Verbesserung platziert wird. Um mit dem RIU-Algorithmus auch multiple Knoten generieren zu können, wird ein gerade veränderter Knoten für eine bestimmte Anzahl an Iterationen gesperrt. Ein gesperrter Knoten kann nicht verändert werden.

Der RIU-Algorithmus erhält neben den zu Beginn des Kapitels 4 beschriebenen Eingangsgrößen als einzige zusätzliche Eingangsgröße die (ganzzahlige) maximale Anzahl benachbarter Ausreißer k_{med} . Diese Größe wird dazu benötigt, um den Algorithmus robust gegenüber Ausreißern zu gestalten. Die Größe k_{med} ist in der Regel im Vorhinein nicht bekannt und muss dann abgeschätzt oder iterativ gewählt werden. Ausgangsgröße des RIU-Algorithmus ist der Knotenvektor $\mathcal{I}_{\text{best}}$. Der

Algorithmus 2 Ablauf Residuenbasierter Iterativer Update Algorithmus

Input: k_{med} ▷ Maximale Anzahl benachbarter Ausreißer
Input: $n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}$
Output: $\mathcal{I}_{\text{best}}$ ▷ Interner Knotenvektor

- 1: **procedure** RIU($k_{\text{med}}, n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}, \mathcal{I}_{\text{best}}$)
- 2: $i_{\text{fest}} = \min(n - p - 1, p)$ ▷ Anzahl der Iterationen, die ein geänderter Knoten gesperrt ist
- 3: $M_{\text{fest}} = \emptyset$ ▷ Initialisiere Menge für gesperrte Knoten
- 4: $\Omega_{\text{best}} = \infty$ ▷ Initialisiere besten Fitnesswert
- 5: $i = 1$ ▷ Initialisiere Laufvariable
- 6: Generiere ein Individuum \mathcal{I}_{det} mit der PT1-Methode ▷ Kapitel 3.3.1.2
- 7: Füge alle Knoten von \mathcal{I}_{det} der Menge M_{frei} zu
- 8: **while** $i \leq (n - p)$ **do**
- 9: Führe alle Elemente von M_{fest} und M_{frei} zum Individuum \mathcal{I} zusammen
- 10: Führe die Kontrollpunktschätzung durch ▷ Kapitel 3.4
- 11: Berechne Ω für das Individuum \mathcal{I} ▷ Formel 4.1
- 12: **if** $\Omega < \Omega_{\text{best}}$ **then** ▷ Wenn aktueller Fitnesswert besser als der bisher beste
- 13: $\Omega_{\text{best}} = \Omega$ ▷ Speichere neuen besten Fitnesswert
- 14: $\mathcal{I}_{\text{best}} = \mathcal{I}$ ▷ Speichere neues bestes Individuum
- 15: **end if**
- 16: Berechne für jeden Punkt in \mathbf{Q} eine Gesamtverbesserung \mathbf{v}_{xyz} ▷ Formel 4.3
- 17: Berechne \mathbf{v}_{filt} über Medianfilterung von \mathbf{v}_{xyz} unter Verwendung von k_{med} ▷ Formel 4.4
- 18: Finde den Ortsparameter \bar{u}_{max} des Punktes mit der größten Verbesserung in \mathbf{v}_{filt}
- 19: Entferne den Knoten mit dem minimalen Abstand zu \bar{u}_{max} aus M_{frei}
- 20: Füge \bar{u}_{max} der Menge M_{fest} hinzu
- 21: **if** $|M_{\text{fest}}| > i_{\text{fest}}$ **then**
- 22: Verschiebe zuerst hinzugefügtes Element aus M_{fest} zu M_{frei}
- 23: **end if**
- 24: $i = i + 1$ ▷ Erhöhe Laufvariable
- 25: **end while**
- 26: **Return:** $\mathcal{I}_{\text{best}}$ ▷ Ergebnisausgabe
- 27: **end procedure**

Ablauf des RIU-Algorithmus ist als Pseudocode in Algorithmus 2 beschrieben oder graphisch in Abbildung 4.1 dargestellt.

Zu Beginn des RIU-Algorithmus wird der ganzzahlige Wert der Variable i_{fest} berechnet. i_{fest} gibt die Anzahl der Iterationen an, die ein geänderter Knoten für weitere Veränderungen gesperrt ist. Der Wert für i_{fest} entspricht dem Minimum aus dem Grad der Basisfunktion p und dem Wert $n - p - 1$. Diese Wahl von i_{fest} lässt zum einen die Möglichkeit zu, multiple Knoten zu erzeugen. Zum anderen wird dadurch vermieden, alle internen Knoten zu sperren. Die Menge, in der die gesperrten Knoten gespeichert werden, wird mit M_{fest} bezeichnet. Diese wird als leere Menge initialisiert. Nach Initialisierung des besten Fitnesswertes Ω_{best} und der Laufvariable i wird in der ersten Iteration ein Individuum \mathcal{I}_{det} mit der PT1-Methode (siehe Kapitel 3.3.1.2) erzeugt. Zur Initialisierung wurde die PT1-Methode gewählt, da sie wenig rechenaufwendig ist und numerisch stabile Ergebnisse liefert. Alle internen Knoten des Individuums \mathcal{I}_{det} werden der Menge M_{frei} zugewiesen. Die Menge M_{frei} ist die Menge, in der die veränderbaren Knoten gespeichert werden.

Zu Beginn der while-Schleife wird abgefragt, ob die Laufvariable i den Wert $(n-p)$ überschreitet und somit das Abbruchkriterium erfüllt. Ist dies nicht der Fall, werden alle Knoten, die in den Mengen M_{fest} und M_{frei} gespeichert sind, zu einem Individuum \mathcal{I} zusammengefügt. Mit dem Knotenvektor \mathcal{I} und dem Ortsparametervektor $\bar{\mathbf{u}}$ wird die Designmatrix für die Kontrollpunktschätzung aufgestellt (siehe Kapitel 3.4 und Formel 2.19, 3.57 und 3.58). Mit dem Beobachtungsvektor \mathbf{l} und der Gewichtsmatrix \mathbf{P} erfolgt die Parameterschätzung nach der Methode der kleinsten Quadrate (Formel 2.33). Mit den geschätzten Parametern werden die Verbesserungen \mathbf{v} nach Formel 2.30 geschätzt. Aus den Verbesserungen wird anhand der Formel 4.1 die Residuenquadratsumme Ω berechnet, die als Fitnesswert für das aktuelle Individuum \mathcal{I} dient. Unterschreitet der aktuelle Fitnesswert Ω den bisher besten Fitnesswert Ω_{best} , so wird Ω zu Ω_{best} und Individuum \mathcal{I} zum besten Individuum $\mathcal{I}_{\text{best}}$. Der Verbesserungsvektor \mathbf{v} hat bei einer zu approximierenden dreidimensionalen Punktwolke die Form:

$$\mathbf{v} = [v_{x,1}, \dots, v_{x,r}, v_{y,1}, \dots, v_{y,r}, v_{z,1}, \dots, v_{z,r}]^T. \quad (4.2)$$

Mit der folgenden Formel wird für jeden Punkt \mathbf{Q}_j der Punktwolke \mathbf{Q} eine Gesamt-Verbesserung $v_{xyz,j}$ ermittelt (siehe hierzu auch Formel 3.1 und 3.2):

$$v_{xyz,j} = \sqrt{v_{x,j}^2 + v_{y,j}^2 + v_{z,j}^2} \quad \text{mit } j = 1, \dots, r. \quad (4.3)$$

Die Gesamt-Verbesserungen \mathbf{v}_{xyz} werden mit einem Medianfilter gefiltert¹, um einen negativen Einfluss einzelner Ausreißer zu minimieren. Der Medianfilter wird häufig in der digitalen Bildbearbeitung eingesetzt und ist z.B. in Burger und Burge (2015) beschrieben. Auf die hier vorliegende Aufgabenstellung angewendet, berechnet der Medianfilter für jede Gesamtverbesserung $v_{xyz,j}$ den Median aus den umliegenden, also in der Filterregion liegenden, Gesamtverbesserungen. Die gefilterte Gesamtverbesserung wird mit $v_{\text{filt},j}$ bezeichnet:

$$v_{\text{filt},j} = \text{median}(v_{xyz,j-k_{\text{med}}}, \dots, v_{xyz,j+k_{\text{med}}}) \quad \text{mit } j = 1, \dots, r. \quad (4.4)$$

Die Größe der Filterregion wird über den Wert $k_{\text{med}} \in \mathbb{Z}\{0, \dots, \lfloor \frac{r-1}{2} \rfloor\}$ gesteuert. An der Medianberechnung nehmen $2 \cdot k_{\text{med}} + 1$ Werte teil. In den Randbereichen, also für $j - k_{\text{med}} < 1$ oder $j + k_{\text{med}} > r$, bei denen es keine zugehörigen Werte in \mathbf{v}_{xyz} gibt, wird die Filterregion mit Nullen aufgefüllt. Für die konkrete Berechnung wurde die MATLAB-Funktion `medfilt1` der Signal Processing Toolbox (Version 8.0 und MATLAB R2018a) verwendet (für weitere Informationen siehe MathWorks[®] (2019c)).

Anschließend wird der Punkt mit der größten Verbesserung in \mathbf{v}_{filt} gesucht. Dessen Ortsparameterwert wird in \bar{u}_{max} gespeichert. Zwischen \bar{u}_{max} und allen Knoten in M_{frei} wird die absolute Differenz berechnet. Der Knoten mit der minimalen absoluten Differenz wird aus M_{frei} entfernt. Der Wert \bar{u}_{max} wird der Menge M_{fest} zugewiesen. Übersteigt die Anzahl der Elemente in M_{fest} den Wert i_{fest} , so wird das zuerst hinzugefügte Element aus M_{fest} in M_{frei} verschoben. Mit Erhöhung der

¹dabei sollte \mathbf{v}_{xyz} wie \mathbf{l} sortiert vorliegen

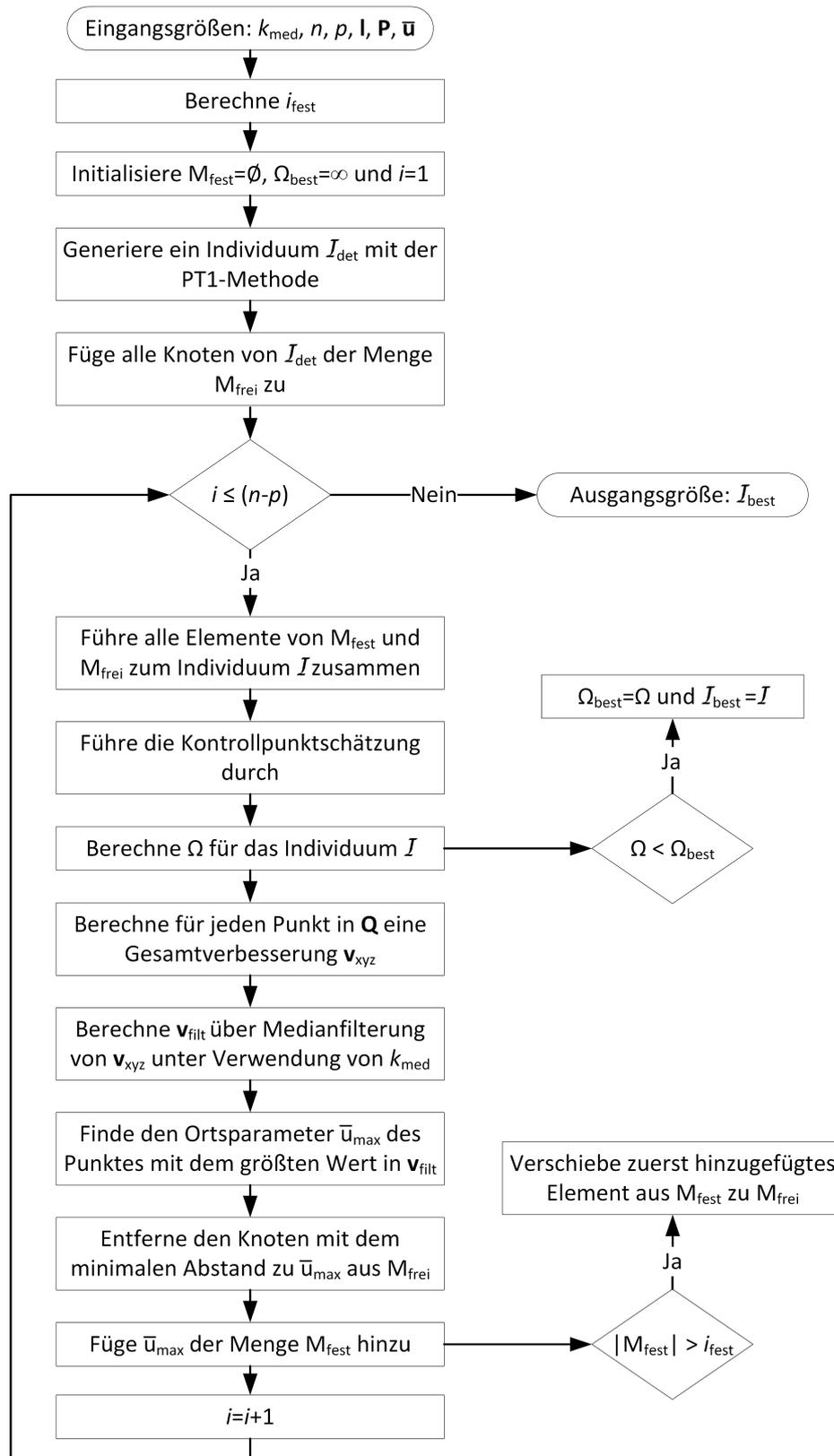


Abbildung 4.1: Ablauf Residuenbasierter Iterativer Update Algorithmus

Laufvariable kehrt der Algorithmus wieder zum Anfang der while-Schleife zurück. Dieser Ablauf wird bis zum Erfüllen des Abbruchkriteriums wiederholt. Die Entscheidung, die Iterationsanzahl als Abbruchkriterium zu verwenden und die maximale Iterationsanzahl auf die Anzahl der zu wäh-

lenden internen Knoten $n - p$ zu begrenzen, basiert auf der Verwendung der größten Verbesserung in \mathbf{v}_{filt} zur Bestimmung eines neuen Knotens. Bei der erstmaligen Modifikation eines Knotens wird die größte Verbesserung in der Regel durch einen unzulänglichen Knoten bzw. Knotenvektor verursacht. Bei der wiederholten Modifikation eines Knotens wird die größte Verbesserung in der Regel durch eine zufällige Abweichung verursacht. Die Verschiebung eines Knotens auf diese (zufällig große) Verbesserung hat in der Regel zur Folge, dass sich die Residuenquadratsumme Ω für dieses Individuum sprunghaft erhöht und auch für die nachfolgenden Individuen erhöht bleibt.

Dieser Algorithmus wurde zunächst in leicht abgewandelter Form entwickelt. Anstatt bei der Initialisierung ein Individuum mit der vollen Anzahl interner Knoten und der PT1-Methode zu generieren, wurde zunächst ein Individuum ohne interne Knoten generiert (also eine Bézier-Kurve). Anschließend wurde ein Knoten an der Position der maximalen Verbesserung hinzugefügt und die Kontrollpunkte und Verbesserungen neu geschätzt. Dies wurde bis zum Erreichen der gewünschten Anzahl interner Knoten wiederholt. Dieses Vorgehen hat sich aber als wenig vorteilhaft herausgestellt, da die Approximationsgüte, zumindest bei den untersuchten Kurven, zum Teil deutlich unter der Approximationsgüte der PT1-Methode lag. Ein Grund hierfür ist, dass die jeweils besten verfügbaren Knotenvektoren mit einer unterschiedlichen Anzahl an Knoten komplett voneinander abweichen können. Dies trifft insbesondere dann zu, wenn mindestens einer der beiden Knotenvektoren nicht die benötigte Anzahl an Knoten aufweist². Siehe hierzu auch die Untersuchung in Kapitel 5.5. D.h. die Wahl des ersten internen Knoten kann zu der Iteration die optimale Wahl gewesen sein. Bei der Wahl des zweiten internen Knotens ist der bereits fixierte erste interne Knoten nicht mehr die optimale Wahl und dadurch wird auch die Wahl des zweiten internen Knoten nicht optimal. Um den optimalen Knotenvektor zu erhalten, müssen beide Knoten gemeinsam³ optimiert werden.

4.2 Evolutionäre Monte-Carlo (EMC) Methode

Die Grundidee der EMC-Methode liegt wie bereits in Kapitel 3.3.2.2 beschrieben in der Schätzung einer Wahrscheinlichkeitsdichte aus den Knotenvektoren, mit denen die bis dahin besten Fitnesswerte (siehe Formel 4.1) erzielt wurden. Aus der geschätzten Wahrscheinlichkeitsdichte werden in der Folge neue Knotenvektoren generiert. Dieser Prozess wird bis zum Erreichen eines Abbruchkriteriums wiederholt. Der in Bureick u. a. (2016a) beschriebene Ablauf wird im Rahmen dieser Arbeit präzisiert und modifiziert. Diese modifizierte EMC-Methode wird im folgenden Kapitel 4.2.1 beschrieben.

4.2.1 Methodik

Der Ablauf der modifizierten EMC-Methode ist als Pseudocode in Algorithmus 3 oder graphisch in Abbildung 4.2 dargestellt. Zu Beginn wird die Spanne der Ortsparameter an jedem Ortsparameter \bar{u}_j mit $j = 1, \dots, r$ unterteilt (siehe auch Formel 3.36). Jedem der $r - 1$ Intervalle wird eine Wahrscheinlichkeit zugewiesen. Jedem Intervall

$$I_j = \{x \in \mathbb{R} | \bar{u}_j \leq x < \bar{u}_{j+1}\} \text{ mit } j = 1, \dots, r - 1 \quad (4.5)$$

wird zu Beginn eine identische Wahrscheinlichkeit $w_{\text{eq},j}$ zugewiesen. Alle Wahrscheinlichkeiten werden in dem Vektor

$$\mathbf{w}_{\text{eq}} = [w_{\text{eq},1}, \dots, w_{\text{eq},r-1}] \text{ mit } w_{\text{eq},1} = \dots = w_{\text{eq},r-1} \quad (4.6)$$

²Man spricht im Englischen von *Underfitting*

³Dies wird auch als *global* bezeichnet

Algorithmus 3 Ablauf Evolutionäre Monte-Carlo Methode

Input: i_{\max} ▷ Gesamtanzahl Iterationen; $i_{\max} \geq 1$
Input: i_{chance} ▷ Anzahl Iterationen mit gleichgewichtetem \mathbf{w} ; $i_{\text{chance}} \geq 1$
Input: N ▷ Anzahl Individuen
Input: N_r ▷ Anzahl Individuen die aus \mathbf{w}_{eq} gezogen werden
Input: N_b ▷ Anzahl Individuen in der Population $\mathcal{P}_{\text{best}}$
Input: $n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}$
Output: $\mathcal{I}_{\text{best}}$ ▷ Bestes Individuum

- 1: **procedure** EMC($i_{\max}, i_{\text{chance}}, N, N_r, N_b, n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}, \mathcal{I}_{\text{best}}$)
- 2: Teile die Ortsparameterspanne an jedem Wert von $\bar{\mathbf{u}}$ ▷ Es entstehen $r - 1$ Intervalle
- 3: Weise jedem Intervall das gleiche Gewicht zu
- 4: Speichere alle Gewichte in \mathbf{w}_{eq}
- 5: Setze $\mathbf{w} = \mathbf{w}_{\text{eq}}$
- 6: Setze $i = 1$ ▷ Initialisierung Iterationszähler
- 7: **while** $i \leq i_{\max}$ **do**
- 8: **if** $i > i_{\text{chance}}$ **then**
- 9: Berechne \mathbf{w} aus $\mathcal{P}_{\text{best}}$ ▷ Kapitel 4.2.1.1
- 10: **end if**
- 11: Ziehe $N - N_r$ Individuen zufällig anhand von \mathbf{w} ▷ Kapitel 4.2.1.2
- 12: Speichere die $N - N_r$ Individuen in der Population \mathcal{P}_N
- 13: Ziehe N_r Individuen zufällig anhand von \mathbf{w}_{eq} ▷ Kapitel 4.2.1.2
- 14: Speichere die N_r Individuen in \mathcal{P}_N
- 15: Berechne Ω für jedes Individuum in \mathcal{P}_N ▷ siehe Formel 4.1
- 16: **if** $i = 1$ **then**
- 17: Verschiebe die N_b Individuen mit dem kleinsten Ω aus \mathcal{P}_N in $\mathcal{P}_{\text{best}}$
- 18: **else**
- 19: Verschiebe die N_b Individuen mit dem kleinsten Ω aus \mathcal{P}_N in \mathcal{P}_{all}
- 20: Verschiebe alle Individuen aus $\mathcal{P}_{\text{best}}$ in \mathcal{P}_{all}
- 21: Verschiebe die N_b Individuen mit dem kleinsten Ω aus \mathcal{P}_{all} in $\mathcal{P}_{\text{best}}$
- 22: **end if**
- 23: Lösche \mathcal{P}_N und \mathcal{P}_{all}
- 24: $i = i + 1$ ▷ Erhöhe Iterationszähler
- 25: **end while**
- 26: Wähle das Individuum $\mathcal{I}_{\text{best}}$ mit dem kleinsten Ω aus $\mathcal{P}_{\text{best}}$ ▷ Bester Knotenvektor
- 27: **Return:** $\mathcal{I}_{\text{best}}$ ▷ Ergebnisausgabe
- 28: **end procedure**

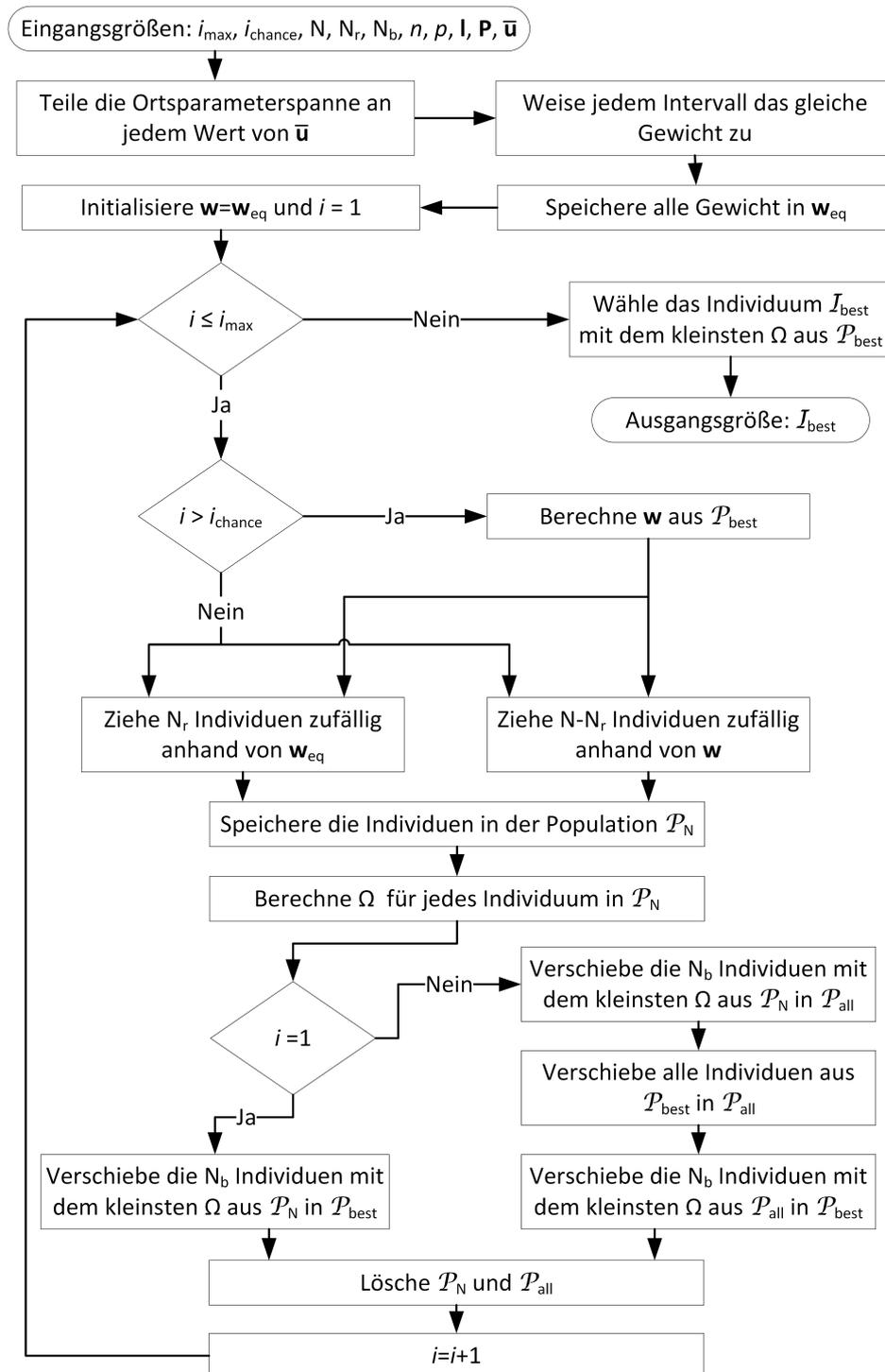


Abbildung 4.2: Ablauf Evolutionäre Monte-Carlo Methode

gespeichert. Die Wahrscheinlichkeiten werden normiert, in dem $w_{\text{eq},j} = \frac{1}{r-1}$ gewählt wird, so dass:

$$\sum_{j=1}^{r-1} w_{\text{eq},j} = 1 \quad (4.7)$$

gilt. Zu Beginn der EMC-Methode gilt:

$$\mathbf{w} = \mathbf{w}_{\text{eq}}, \quad (4.8)$$

wobei \mathbf{w} der Vektor mit den Wahrscheinlichkeiten ist, anhand dem ein Großteil der neuen Individuen generiert wird.

Beim Eintritt in die while-Schleife wird zunächst geprüft, ob die aktuelle Iterationszahl i den Wert i_{chance} übersteigt⁴. Ist dies nicht der Fall, bleibt \mathbf{w} mit gleichen Wahrscheinlichkeiten besetzt. Andernfalls wird \mathbf{w} auf Basis der Population $\mathcal{P}_{\text{best}}$ berechnet (siehe Kapitel 4.2.1.1). $\mathcal{P}_{\text{best}}$ enthält die Individuen, mit denen bisher die besten Fitnesswerte, also im Rahmen dieser Arbeit die kleinsten Residuenquadratsummen Ω (siehe Formel 4.1) erzielt wurden.

4.2.1.1 Berechnung der Wahrscheinlichkeit \mathbf{w}

Für die konkrete Berechnung von \mathbf{w} werden zunächst alle Knoten der Individuen von $\mathcal{P}_{\text{best}}$ in einem Vektor \mathcal{K} gespeichert. Der Vektor \mathcal{K} hat die Länge $N_b \cdot (n - p)$, wobei N_b der Anzahl der Individuen in $\mathcal{P}_{\text{best}}$ entspricht. Anschließend wird \mathcal{K} verwendet, um daraus eine Kerndichte \hat{f}_h zu schätzen. Die Kerndichteschätzung wurde von Rosenblatt (1956) und Parzen (1962) eingeführt. Für eine beliebige Position t wird die Kerndichte über:

$$\hat{f}_h(t) = \frac{1}{N_b \cdot (n - p) \cdot \mathcal{B}} \sum_{j=1}^{N_b \cdot (n-p)} K\left(\frac{t - \mathcal{K}_j}{\mathcal{B}}\right) \quad (4.9)$$

geschätzt. Der Parameter \mathcal{B} gibt die Bandbreite der Kerndichteschätzung an. Als Kernel-Funktion $K(\cdot)$ wird eine normalverteilte Kernel-Funktion der Form:

$$K(t_{\mathcal{K}}) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t_{\mathcal{K}}^2} \quad \text{mit } t_{\mathcal{K}} = \frac{t - \mathcal{K}_j}{\mathcal{B}} \quad (4.10)$$

gewählt. Die Bandbreite \mathcal{B} steuert die Glättung der geschätzten Kerndichte. Grundsätzlich ist die Wahl der Bandbreite \mathcal{B} eine entscheidende Stellschraube der EMC-Methode. Ziel ist es, klare Maxima in der Kerndichte zu erhalten, um in der Folge möglichst viele gute Knotenvektoren zu generieren. Gleichzeitig soll die Kerndichte in der direkten Nachbarschaft dieser Maxima ebenfalls leicht erhöht werden, um eine weitere Evolution der Knotenvektoren zuzulassen. Zudem soll \mathcal{B} für jegliche zu approximierende Punktwolken gelten. Die genannten Anforderungen werden erfüllt, wenn man \mathcal{B} mit der durchschnittlichen Intervallbreite gleichsetzt (Formel 4.11).

$$\mathcal{B} = \frac{1}{r - 1}. \quad (4.11)$$

Für die konkrete Berechnung der Kerndichte \hat{f}_h wurde die MATLAB-Funktion *ksdensity* der Statistics and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) verwendet (siehe MathWorks[®] (2019a)). Anhand der geschätzten Kerndichte \hat{f}_h lässt sich die kumulierte Kerndichte \hat{c}_h berechnen:

$$\hat{c}_h(t) = \int_0^t \hat{f}_h(x) dx. \quad (4.12)$$

Das neue Gewicht w_j für das Intervall I_j wird über

$$w_j = \hat{c}_h(\bar{u}_{j+1}) - \hat{c}_h(\bar{u}_j) \quad \text{mit } j = 1, \dots, r - 1 \quad (4.13)$$

berechnet.

Das hier beschriebene Vorgehen zur Berechnung von \mathbf{w} unterscheidet sich von dem in Bureick u. a. (2016a). Dort werden die neuen Wahrscheinlichkeiten w_j direkt aus $\mathcal{P}_{\text{best}}$ berechnet und zusätzlich über eine Funktion, in die der inverse Fitnesswert Ω des jeweiligen Individuums eingeht, skaliert. Hier ergibt sich die Schwierigkeit, eine allgemein gültige Skalierungsfunktion zu finden,

⁴Bei der Wahl von i_{chance} muss gelten: $i_{\text{chance}} \geq 1$

die bei allen realistischen Datensätzen zu einer guten Performance der Methode in Bureick u. a. (2016a) führt. Der oben beschriebene Ablauf führt zu einer deutlich stabileren Performance bei den unterschiedlichsten Datensätzen.

4.2.1.2 Ziehung neuer Individuen

Die Ziehung bzw. Generierung neuer Individuen aus \mathbf{w} oder \mathbf{w}_{eq} erfolgt mit Hilfe einer Resampling-Technik, die z.B. Anwendung in Partikelfiltern findet (siehe z.B. Simon (2006, S.466ff.)). Die folgende Beschreibung der Resampling-Technik orientiert sich an der Beschreibung von Simon (2006, S.467). Die nachfolgenden Formeln sind für die Variable \mathbf{w} gegeben. Für die Berechnung mit \mathbf{w}_{eq} ist \mathbf{w} entsprechend durch \mathbf{w}_{eq} zu ersetzen.

Im ersten Schritt der Resampling-Technik werden aus \mathbf{w} die kumulierten Gewichte \mathbf{c}_w berechnet:

$$c_{w,j} = \sum_{i=1}^j w_i. \quad (4.14)$$

Mithilfe einer Division durch das kumulierte Gewicht des letzten Intervalls $c_{w,r-1}$ werden die kumulierten Gewichte \mathbf{c}_w auf einen Bereich zwischen 0 und 1 normiert:

$$c_{\text{norm},j} = c_{w,j}/c_{w,r-1} \text{ mit } j = 1, \dots, r-1. \quad (4.15)$$

Anschließend werden für jedes neu zu bestimmende Individuum zufällig $n-p$ Realisierungen der Zufallsvariable Z aus der Gleichverteilung $\mathcal{U}(0,1)$ generiert:

$$Z \sim \mathcal{U}(0,1), \quad (4.16)$$

wobei 0 die untere und 1 die obere Grenze der Gleichverteilung $\mathcal{U}(0,1)$ sind. Für jede Realisierung z_i mit $i = 1, \dots, n-p$ wird dann der Intervallindex j bestimmt, für den gilt:

$$c_{\text{norm},j-1} < z_i \text{ und } c_{\text{norm},j} \geq z_i. \quad (4.17)$$

Der gewählte Intervallindex j wird im Vektor \mathcal{J} gespeichert:

$$\mathcal{J}_i = j. \quad (4.18)$$

Dieses Vorgehen hat zur Folge, dass Intervalle mit einem geringen Gewicht weniger wahrscheinlich ausgewählt werden als Intervalle mit einem hohen Gewicht. Aus den gezogenen Intervallen bzw. Intervallindizes \mathcal{J} muss in der Folge ein neues Individuum generiert werden. Denkbar wäre es, den Individuen z.B. die Mitte des Intervalls oder eine der Intervallgrenzen zuzuweisen. Dadurch wäre aber die „Diversität“ der entstehenden Individuen eingeschränkt. Deshalb wird über die untere $\bar{u}_{\mathcal{J}_i}$ und obere Intervallgrenze $\bar{u}_{\mathcal{J}_i+1}$ der interne Knoten ξ_i des Individuums wie folgt bestimmt:

$$\xi_i = \bar{u}_{\mathcal{J}_i} + z \cdot \frac{1}{\bar{u}_{\mathcal{J}_i+1} - \bar{u}_{\mathcal{J}_i}} \text{ mit } i = 1, \dots, n-p. \quad (4.19)$$

z ist eine weitere Realisierung der gleichverteilten Zufallsvariable Z (siehe Formel 4.16). Die so generierten Individuen bilden die neue Population \mathcal{P}_N . Die Berechnung in Formel 4.19 kann auch als eine Art der Mutation verstanden werden.

Die dauerhafte Ziehung von N_r Individuen über den gleichgewichteten Vektor \mathbf{w}_{eq} soll verhindern, dass die EMC-Methode in ein lokales Optimum konvergiert. So werden fortwährend Knotenvektoren generiert, die nicht von den vorigen Knotenvektoren beeinflusst sind.

4.2.1.3 Selektion und Ausgabe

Für jedes Individuum in \mathcal{P}_N wird die Kontrollpunktschätzung durchgeführt und der Fitnesswert Ω berechnet. Anschließend werden \mathcal{P}_N und $\mathcal{P}_{\text{best}}$ zur Gesamtpopulation \mathcal{P}_{all} zusammengefügt. Aus \mathcal{P}_{all} werden die N_b Individuen mit dem besten Fitnesswert selektiert und als $\mathcal{P}_{\text{best}}$ bezeichnet. Anschließend werden \mathcal{P}_N und \mathcal{P}_{all} gelöscht. Nach dem Inkrementieren des Iterationszählers kehrt der Algorithmus zurück zum Beginn der while-Schleife. Dieser Ablauf wird nun so lange wiederholt, bis die maximale Anzahl an Iterationen i_{max} überschritten ist. Abschließend wird aus $\mathcal{P}_{\text{best}}$ dasjenige Individuum $\mathcal{I}_{\text{best}}$ gewählt, welches den besten Fitnesswert hat. Mit der Ausgabe von $\mathcal{I}_{\text{best}}$ endet die EMC-Methode.

Bei der EMC-Methode, sowie bei den noch folgenden heuristischen Methoden, wird wie oben beschrieben die Iterationszahl als Abbruchkriterium verwendet. Alternative Abbruchkriterien werden im Rahmen dieser Arbeit nicht verwendet. Denkbar wäre es z.B., die while-Schleife zu beenden, sobald der Globaltest der Kontrollpunktschätzung (siehe Kapitel 3.1.2.1) bestanden wird⁵. In der Untersuchung in Kapitel 5.2 wird so ähnlich vorgegangen. Eine andere Möglichkeit wäre es, den Algorithmus zu beenden, wenn sich das beste Individuum in $\mathcal{P}_{\text{best}}$ oder die gesamte Population $\mathcal{P}_{\text{best}}$ über mehrere Iterationen nicht mehr signifikant ändert.

4.2.2 Parameterwahl

Tabelle 4.1 enthält die gewählten Parameterwerte für die EMC-Methode.

Tabelle 4.1: Gewählte Parameter bei der EMC-Methode

Parameter	Wert
i_{max}	100
i_{chance}	15
N	200
N_r	40
N_b	100

Die Werte für die Parameter i_{max} , i_{chance} und N entsprechen den in Bureick u. a. (2016a) gewählten Werten. Als Anteil der Knotenvektoren, die über den gleichgewichteten Vektor \mathbf{w}_{eq} gezogen werden, hat sich bei einer im Rahmen dieser Arbeit nicht dargestellten systematischen Untersuchung verschiedener Datensätze ein Wert von ca. 20% als optimal herausgestellt. Daraus folgt in Abhängigkeit zum gewählten Wert für N ($= 200$), für $N_r = 40$. Die mit dem Parameter N_b spezifizierte Größe der Population $\mathcal{P}_{\text{best}}$ wurde aufgrund der im Vergleich zu Bureick u. a. (2016a) modifizierten Berechnung (siehe Kapitel 4.2.1.1) vergrößert. Hier wurden bei der oben erwähnten systematischen Untersuchung mit einem Wert von ca. 100 die besten Ergebnisse erzielt.

4.3 Elitärer genetischer Algorithmus

Der elitäre genetische Algorithmus ist ein heuristisches Verfahren zur Knotenvektorwahl und wurde erstmalig in Bureick u. a. (2019) für die Knotenvektorwahl bei B-Spline-Kurven beschrieben. Der elitäre genetische Algorithmus aus Bureick u. a. (2019) wird in Kapitel 4.3.1 beschrieben und wird fortan als EGAori bezeichnet. In Kapitel 4.3.2 werden zwei zusätzliche Modifikationen des EGAori vorgestellt. Die modifizierte Version dieses Algorithmus wird in der Folge mit EGAmo bezeichnet.

⁵Um eine Endlos-Schleife zu vermeiden, sollte zur Sicherheit ein zweites Abbruchkriterium eingeführt werden, das auf der Iterationszahl basiert.

4.3.1 Methodik

Der in Bureick u. a. (2019) beschriebene EGAori ist als Pseudocode in Algorithmus 4 oder graphisch in Abbildung 4.3 zusammengefasst. Die Kernpunkte des EGAori werden in den folgenden Unterkapiteln 4.3.1.1 bis 4.3.1.5 detailliert erläutert.

Algorithmus 4 Ablauf elitärer genetischer Algorithmus

Input: i_{\max}	▷ Gesamtanzahl Iterationen
Input: N	▷ Anzahl Individuen
Input: N_b	▷ Anzahl der zu selektierenden Individuen
Input: N_c	▷ Anzahl der geklonten Individuen
Input: N_e	▷ Anzahl der elitären Individuen
Input: N_r	▷ Anzahl der zufälligen Individuen
Input: r_{rc}	▷ Prozentsatz Einzelpunkt-Crossover
Input: $n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}$	
Output: $\mathcal{I}_{\text{best}}$	▷ Bester Knotenvektor
1: procedure EGA($i_{\max}, N, N_b, N_c, N_e, N_r, r_{rc}, n, p, \mathbf{l}, \mathbf{P}, \bar{\mathbf{u}}, \mathcal{I}_{\text{best}}$)	
2: Generiere $N - 1$ Individuen zufällig aus $\mathcal{U}(0, 1)$	
3: Generiere ein Individuum mit deterministischer Methode ▷ Kapitel 4.3.1.1	
4: Speichere alle generierten Individuen in \mathcal{P}_N	
5: Berechne Ω für jedes Individuum in \mathcal{P}_N ▷ siehe Formel 4.1	
6: Setze $i = 1$ ▷ Initialisierung Iterationszähler	
7: while $i \leq i_{\max}$ do	
8: Wähle die N_e Individuen mit dem kleinsten Ω aus \mathcal{P}_N aus und kopiere sie in \mathcal{P}_e	
9: Wähle die N_b Individuen mit dem kleinsten Ω aus \mathcal{P}_N aus und kopiere sie in $\mathcal{P}_{\text{best}}$	
10: Lösche die Population \mathcal{P}_N	
11: Klone die Individuen in $\mathcal{P}_{\text{best}}$ und speichere sie in \mathcal{P}_c ▷ Kapitel 4.3.1.2	
12: Verschiebe zufällig $r_{rc} \cdot N_c$ Individuen aus \mathcal{P}_c in \mathcal{P}_{rc}	
13: Führe Einzelpunkt-Crossover in \mathcal{P}_{rc} durch ▷ Kapitel 4.3.1.3	
14: Verschiebe alle Individuen aus \mathcal{P}_{rc} und die restlichen aus \mathcal{P}_c in \mathcal{P}_M	
15: Mutiere die Individuen in \mathcal{P}_M ▷ Kapitel 4.3.1.4	
16: Berechne Ω für jedes Individuum in \mathcal{P}_M ▷ siehe Formel 4.1	
17: Verschiebe die $N - (N_e + N_r)$ Individuen mit dem kleinsten Ω aus \mathcal{P}_M in \mathcal{P}_N	
18: Generiere N_r Individuen zufällig, berechne ihr Ω und füge sie \mathcal{P}_N zu ▷ Kapitel 4.3.1.5	
19: Verschiebe alle Individuen aus \mathcal{P}_e in \mathcal{P}_N ▷ Kapitel 4.3.1.5	
20: Lösche die Populationen $\mathcal{P}_M, \mathcal{P}_e, \mathcal{P}_{\text{best}}$	
21: $i = i + 1$ ▷ Erhöhe Iterationszähler	
22: end while	
23: Wähle das Individuum $\mathcal{I}_{\text{best}}$ mit dem kleinsten Ω aus \mathcal{P}_N ▷ Bester Knotenvektor	
24: Return: $\mathcal{I}_{\text{best}}$ ▷ Ergebnisausgabe	
25: end procedure	

4.3.1.1 Initialisierung

Der EGA beginnt mit der initialen Generierung einer Population. Hierzu wird ein Teil der Individuen, wie bei heuristischen Verfahren üblich, zufällig durch Ziehung aus der Gleichverteilung $\mathcal{U}(0, 1)$ generiert. Insgesamt $N - 1$ Individuen werden so erstellt. Im Falle größerer Datenlücken besteht das Risiko, dass durch die Ziehung aus der Gleichverteilung alle generierten Individuen (also Knotenvektoren) zu Singularitäten bei der Kontrollpunktschätzung führen und folglich für alle Ω keine reelle Zahl sondern NaN⁶ liefern. Um dieses Risiko zu umgehen, wird ein Individuum über eine

⁶engl. „Not a Number“

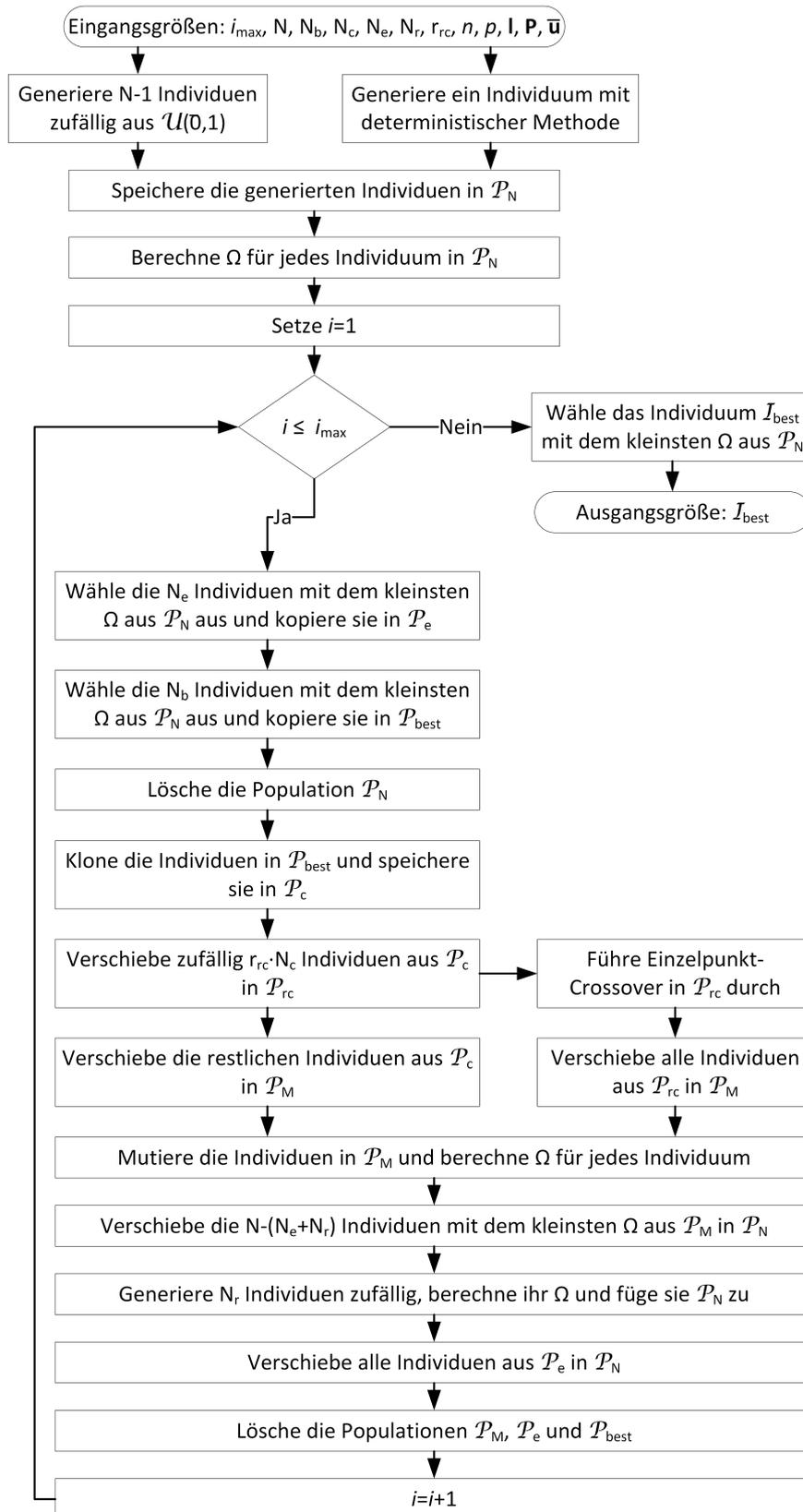


Abbildung 4.3: Ablauf elitärer genetischer Algorithmus

deterministische Methode generiert. Die deterministische Methode vermeidet sicher Singularitäten bei der Kontrollpunktschätzung und führt zu einem reellen Ω . Als deterministische Methode wird die PT1-Methode von Piegel und Tiller (1997, S.412) (siehe auch Kapitel 3.3.1.2) verwendet. Für

alle generierten Individuen wird die Kontrollpunktschätzung durchgeführt und ihr Fitnesswert Ω berechnet.

4.3.1.2 Klonen und Selektion

Nach Eintritt in die while-Schleife, die so lange ausgeführt wird bis eine bestimmte Iterationsanzahl i_{\max} überschritten wird, werden zunächst die N_e Individuen mit dem kleinsten Ω aus \mathcal{P}_N identifiziert und in die Population \mathcal{P}_e kopiert. Die Population \mathcal{P}_e wird auch elitäre Population genannt und bleibt innerhalb der Iteration unverändert (siehe hierzu auch Kapitel 4.3.1.5).

Im folgenden Schritt wird eine weitere Population $\mathcal{P}_{\text{best}}$ erstellt, die die N_b Individuen mit dem kleinsten Ω aus \mathcal{P}_N enthält⁷. Für das nachfolgende Klonen müssen die Individuen in $\mathcal{P}_{\text{best}}$ allerdings in sortierter Reihenfolge⁸ vorliegen. Die Individuen von $\mathcal{P}_{\text{best}}$ werden nach dem von Gálvez u. a. (2015) dargestellten Schlüssel geklont. Dabei wird die Anzahl der Klone $N_{\text{ind},j}$ für jedes Individuum von $\mathcal{P}_{\text{best}}$ über die folgende Formel ermittelt:

$$N_{\text{ind},j} = \text{round}[(\mathcal{V}_j + 1) \cdot 0.1 \cdot N] \text{ mit } j = 1, \dots, N_b. \quad (4.20)$$

Um für die besten Individuen eine höhere Klonrate zu erreichen, stellen Gálvez u. a. (2015) den Vektor \mathcal{V} wie folgt auf:

$$\mathcal{V} = [\mathcal{V}_1, \mathcal{V}_2, \mathcal{V}_3, \mathcal{V}_4, \dots, \mathcal{V}_{N_b}] = [5, 3, 1, 0, \dots, 0]. \quad (4.21)$$

Wendet man die Formeln 4.20 und 4.21 an, wird das beste Individuum in $\mathcal{P}_{\text{best}}$ sechsmal mehr geklont als das Individuum mit dem geringsten Fitnesswert in $\mathcal{P}_{\text{best}}$. Die Gesamtanzahl der geklonten Individuen wird über

$$N_c = \sum_{j=1}^{N_b} N_{\text{ind},j} \quad (4.22)$$

berechnet. Dieses Vorgehen beim Klonen wurde in Bureick u. a. (2019) übernommen. Die geklonten Individuen werden in der Population \mathcal{P}_c gespeichert.

4.3.1.3 Rekombination

Aus der Population \mathcal{P}_c werden zufällig $r_{\text{rc}} \cdot N_c$ Individuen ausgewählt und in die Population \mathcal{P}_{rc} verschoben. Die Individuen in \mathcal{P}_{rc} werden mit Hilfe des zufälligen Einzelpunkt-Crossovers zu neuen Individuen kombiniert. Das zufällige Einzelpunkt-Crossover ist eine spezielle Form des Uniform-Crossovers. In Bureick u. a. (2019) wurde festgestellt, dass durch diese Form der Rekombination die Wahrscheinlichkeit für die Erzeugung multipler Knoten erhöht wird. Der Ablauf des zufälligen punktweisen Crossovers ist in Abbildung 4.4 dargestellt.

Zunächst wird bei beiden Eltern-Individuen zufällig jeweils eine Indexzahl \mathcal{J}_1 und \mathcal{J}_2 (im Beispiel in Abbildung 4.4: $\mathcal{J}_1 = 2$ und $\mathcal{J}_2 = 4$) gezogen. Eltern-Individuum 1 überträgt alle Knoten auf Kind-Individuum 1 mit Ausnahme des Knotens mit der Indexnummer \mathcal{J}_1 . Anstatt diesem wird der Knoten mit der Indexnummer \mathcal{J}_2 von Eltern-Individuum 2 auf Kind-Individuum 1 übertragen. Kind-Individuum 2 erhält analog alle Knoten von Eltern-Individuum 2 mit Ausnahme des Knotens mit der Indexnummer \mathcal{J}_2 . Anstatt diesem wird der Knoten mit der Indexnummer \mathcal{J}_1 von Eltern-Individuum 1 übertragen.

Die neu kombinierten Individuen aus \mathcal{P}_{rc} , sowie die restlichen Individuen aus \mathcal{P}_c werden in die Population \mathcal{P}_M verschoben.

⁷Für $N_e = N_b$ gilt $\mathcal{P}_e = \mathcal{P}_{\text{best}}$.

⁸Individuen sind von kleinem zu großem Ω sortiert.

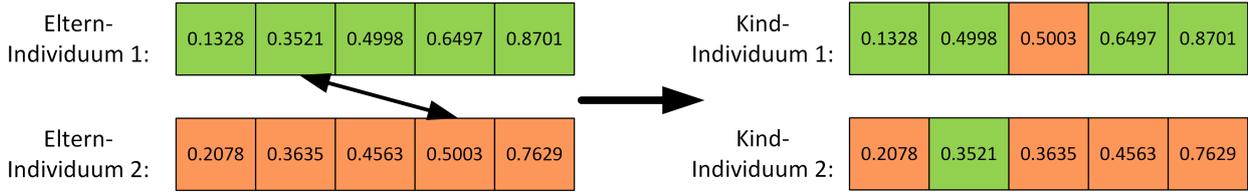


Abbildung 4.4: Schematischer Ablauf des zufälligen Einzelpunkt-Crossovers (adaptiert aus Bureick u. a. (2019))

4.3.1.4 Mutation

Alle Individuen in der Population \mathcal{P}_M werden mit dem folgenden Mechanismus mutiert. Zunächst wird zufällig ein Knoten ξ_i jedes Individuums ausgewählt. Die Mutation erfolgt gemäß:

$$\xi_i^* = \xi_i - \left(\frac{\Delta}{2}\right) (z - 0.5) \text{ mit } \Delta = \min\{\xi_i, 1 - \xi_i\}. \quad (4.23)$$

z stellt wiederum eine Realisierung der gleichverteilten Zufallsvariable $Z \sim \mathcal{U}(0, 1)$ dar. Dieser Mutationsmechanismus wurde in Bureick u. a. (2019) in leicht veränderter Form aus Gálvez u. a. (2015) adaptiert. Gálvez u. a. (2015) ermitteln Δ über $\Delta = \min\{\xi_i/2, 1 - \xi_i\}$. Diese Halbierung von Δ für Knoten, die näher an 0 liegen, im Vergleich zu den Knoten, die näher an 1 liegen, wird in Gálvez u. a. (2015) nicht erklärt und hat keinen ersichtlichen Grund. Deshalb vermuten Bureick u. a. (2019) an dieser Stelle einen Übertragungs- oder Formelfehler und ändern den Mutationsmechanismus zu der in Formel 4.23 dargestellten Form. Er stellt sicher, dass keine Knoten außerhalb der Knotenspanne zwischen 0 und 1 generiert werden. Um die monoton steigende Reihenfolge der Knoten sicherzustellen, müssen die mutierten Knotenvektoren sortiert werden. Anschließend wird für jeden Knotenvektor in \mathcal{P}_M die Kontrollpunktschätzung durchgeführt und der zugehörige Fitnesswert Ω berechnet.

4.3.1.5 Elitismus und Exploration

Die in Gálvez u. a. (2015) vorgestellten Eigenschaften des Elitismus und der Exploration wurden für den EGA in Bureick u. a. (2019) adaptiert. Die Eigenschaft Elitismus besagt, dass die besten bisher erzielten Ergebnisse gespeichert werden und nicht durch Rekombination oder Mutation verloren gehen. So wurde bereits zu Beginn der while-Schleife (siehe Alg. 4 Zeile 8) sichergestellt, dass die besten N_e Ergebnisse der Population \mathcal{P}_N in die Population \mathcal{P}_e überführt wurden.

Die Explorationseigenschaft besagt, dass der Algorithmus zusätzlich Individuen zieht, die nicht von den vorherigen Ergebnissen abhängig sind. Dadurch soll vermieden werden, dass der Algorithmus in ein lokales Optimum konvergiert. Diese Eigenschaft wird im konkreten Fall durch das Ziehen von N_r komplett zufälligen Individuen erreicht. Diese zufällig gezogenen Individuen bilden mit den Individuen der elitären Population \mathcal{P}_e und den $N - (N_e + N_r)$ besten Individuen der Population \mathcal{P}_M die neue Population \mathcal{P}_N . Anschließend kehrt der Algorithmus wieder an den Beginn der while-Schleife (siehe Alg. 4 Zeile 7) zurück. Ist die Iterationsanzahl i_{\max} erreicht, wird das beste Individuum $\mathcal{I}_{\text{best}}$ aus der Population \mathcal{P}_N ausgewählt und mit der Rückgabe von $\mathcal{I}_{\text{best}}$ endet der EGA. Auch hier gilt die bereits in Kapitel 4.2.1.3 geführte Diskussion zu alternativen Abbruchkriterien.

4.3.2 Modifikationen

Im Rahmen dieser Arbeit wird der EGAori an zwei Stellen modifiziert. Zum einen wird bei der Initialisierung (Kapitel 4.3.1.1) der neu entwickelte RIU-Algorithmus (Kapitel 4.1) als deterministische Methode anstatt der PT1-Methode von Piegler und Tiller (1997, S.412) verwendet. Zum

anderen wird der Mutations-Mechanismus in folgender Weise leicht verändert:

$$\xi_i^* = \xi_i - \left(f_{\text{mut}} \cdot \frac{\Delta}{2} \right) (z - 0.5) \text{ mit } \Delta = \min \{ \xi_i, 1 - \xi_i \}. \quad (4.24)$$

Durch den eingeführten Faktor f_{mut} wird die maximal mögliche Veränderung durch die Mutation variiert. Hierbei haben numerische Untersuchungen ergeben, dass es vorteilhaft ist, den Faktor f_{mut} , beginnend mit dem Startwert 1 mit steigender Iterationsanzahl i zu minimieren:

$$f_{\text{mut}} = \frac{i_{\text{max}} - i + 1}{i_{\text{max}}}. \quad (4.25)$$

Siehe hierzu auch die Untersuchung in Kapitel 5.2.

4.3.3 Parameterwahl

Tabelle 4.2 enthält die gewählten Parameterwerte für den EGA. Da die in Kapitel 4.3.2 beschriebenen Modifikationen keinen Einfluss auf die Parameterwerte haben, können die in Bureick u. a. (2019) gewählten Parameterwerte verwendet werden. Die in Bureick u. a. (2019) gewählten Parameterwerte entsprechen, mit Ausnahme des Parameters r_{rc} , den für den CSA in Gálvez u. a. (2015) spezifizierten Werten. Der optimale Wert für r_{rc} wurde in Bureick u. a. (2019) im Zuge einer umfangreichen Untersuchung bestimmt. In dieser Untersuchung wurden verschiedene Testkurven approximiert. Dabei wurde der Prozentsatz für verschiedene Crossover-Formen (Einpunkt-Crossover, Zweipunkt-Crossover und zufälliges Einzelpunkt-Crossover) systematisch variiert. Anschließend wurde analysiert, wie viele Iterationen der EGA mit den jeweils gewählten Prozentsätzen benötigt, bis die Residuenquadratsumme Ω , als Fitnesswert, einen bestimmten Schwellwert unterschreitet. Hier führte eine Prozentsatz von 20 % bis 50 % für zufälliges Einzelpunkt-Crossover zu einer signifikanten Reduktion der benötigten Iterationszahl. Bei den anderen Crossover-Formen hingegen war keine signifikante Änderung der benötigten Iterationszahl festzustellen.

Tabelle 4.2: Gewählte Parameter beim EGA

Parameter	Wert
i_{max}	100
N	100
N_{b}	10
N_{c}	190
N_{e}	10
N_{r}	20
r_{rc}	0.3

5 Numerische Beurteilung und Validierung der Knotenvektorstichwahl

In diesem Kapitel soll die Performance der verschiedenen Methoden zur Knotenvektorstichwahl evaluiert werden. Dazu werden die verschiedenen Methoden zunächst auf neun unterschiedliche Testdatensätze angewendet. Die Testdatensätze sind mit einem normalverteilten, unkorrelierten Rauschen überlagert. Die Evaluation erfolgt zunächst auf Basis simulierter (synthetischer) Daten, um einen Vergleich zu den Soll-Daten ziehen zu können. Dadurch ist es möglich, die Richtigkeit der erzielten Approximationsergebnisse zu beurteilen. Die Simulation und die Testdatensätze werden in Kapitel 5.1 erläutert. In den Kapiteln 5.2 bis 5.5 sind die Ergebnisse verschiedener Untersuchungen auf Basis der simulierten Daten dargestellt. In Kapitel 5.6 werden zwei reale Datensätze mit den verschiedenen Methoden approximiert und im Rahmen einer Kreuzvalidierung verglichen. Dadurch soll die Praxistauglichkeit der entwickelten Methoden demonstriert werden.

5.1 Simulation

5.1.1 Ablauf der Simulation

Abbildung 5.1 stellt den grundsätzlichen Ablauf der durchgeführten Simulationen dar. Dieser Ablauf gilt auch für die in Kapitel 6 dargestellten Untersuchungen, die um die in Kapitel 6.1 beschriebenen Rauschmodelle II bis V erweitert wurden. Für die in Kapitel 5 dargestellten Untersuchungen wurde nur Rauschmodell I verwendet (siehe Kapitel 5.1.3).

Nach der Wahl des Rauschmodells wird ein Datensatz (siehe Kapitel 5.1.2) ausgewählt. Für den gewählten Datensatz werden, wie in Kapitel 5.1.2 beschrieben, die Parameter der Modellwahl festgelegt. Anschließend werden, wie in Kapitel 5.1.2 beschrieben, äquidistante Ortsparameterwerte generiert und an jedem dieser Ortsparameter ein Punkt ausgegeben. Die so generierte (unverrauschte) Punktwolke wird im Folgenden auch als Soll-Punktwolke bezeichnet. Anhand des gewählten Rauschmodells wird ein Rauschen erzeugt (siehe Kapitel 5.1.3 und 6.1). Die Summe der Soll-Punktwolke und des Rauschens bildet die Ist-Punktwolke. Die Ist-Punktwolke wird jeweils unter Verwendung der zu untersuchenden Knotenvektormethoden approximiert. Bei einigen Untersuchungen wird bei den heuristischen Methoden neben dem Knotenvektor der letzten Iteration auch der jeweils beste Knotenvektor jeder Iteration weiter untersucht (vgl. Kapitel 5.2 und 5.3). Mit dem oder den erzielten Knotenvektoren werden die Kontrollpunkte geschätzt. Je nach Untersuchung erfolgt die Schätzung neben dem MdkQ-Schätzer (Kapitel 5) auch mit dem Huber- und Hampel-Schätzer (Kapitel 5.6 und 6). Auf Basis der geschätzten Kontrollpunkte werden mit der ausgeglichenen Punktwolke und dem Globaltest weitere Größen berechnet. Die Ergebnisse der Kontrollpunktschätzung werden gespeichert. Um vertrauenswürdige Ergebnisse zu erhalten, wird dieser Prozess 500 mal wiederholt. Die Analyse der Ergebnisse wird in den Kapiteln 5.2 bis 5.5, sowie Kapitel 6 präsentiert.

5.1.2 Testdatensätze

Bei den verwendeten Testdatensätzen handelt es sich um neun verschiedene Datensätze (siehe Abbildung 5.2), die zum Teil schon in der Literatur zur Evaluation von B-Spline Approximationen herangezogen wurden. Die Datensätze 1-3 sind häufig verwendete Testfunktionen für die B-Spline Approximation. Nach der erstmaligen Verwendung in Yoshimoto u. a. (1999) wurden diese Daten-

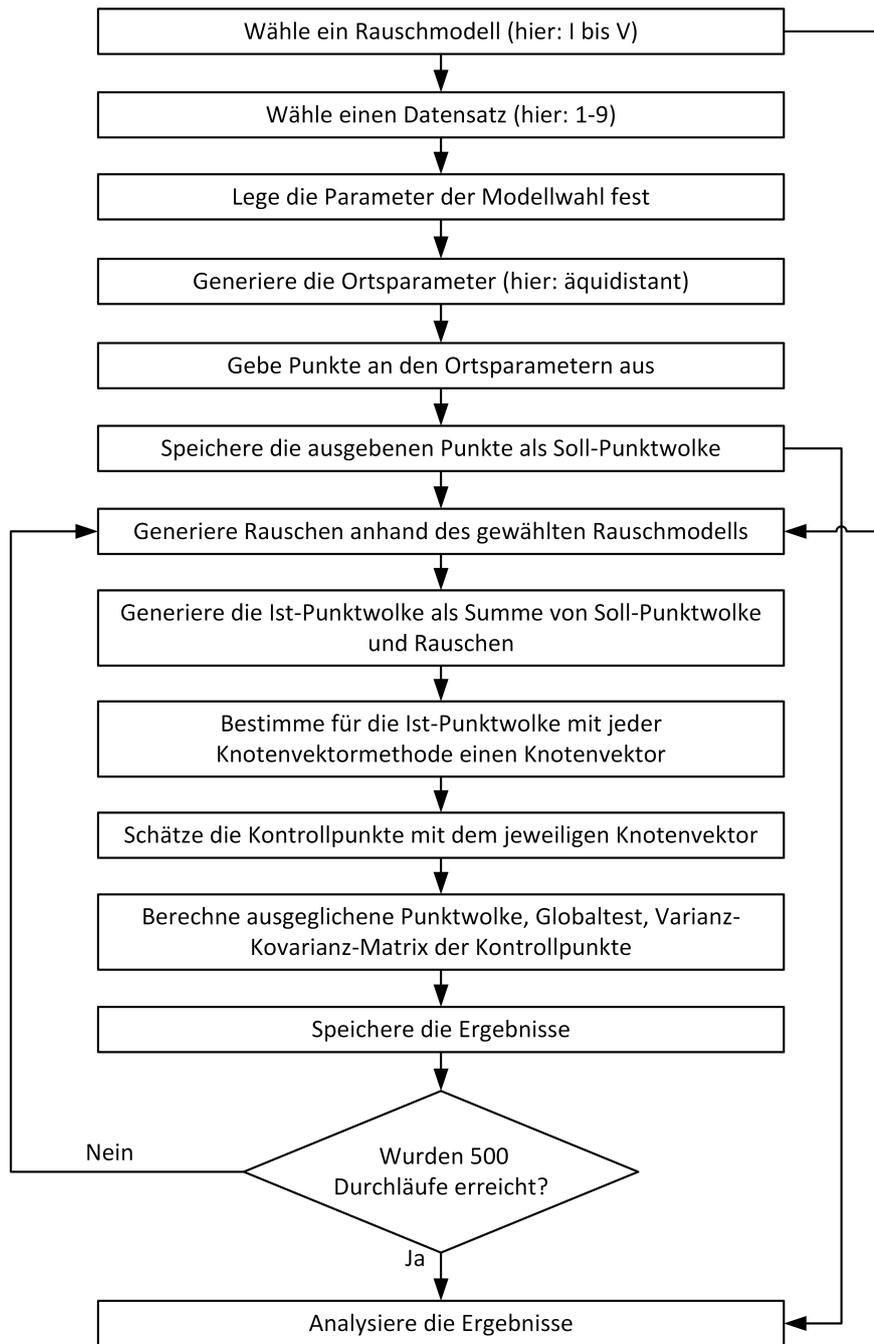


Abbildung 5.1: Ablauf der Simulationen

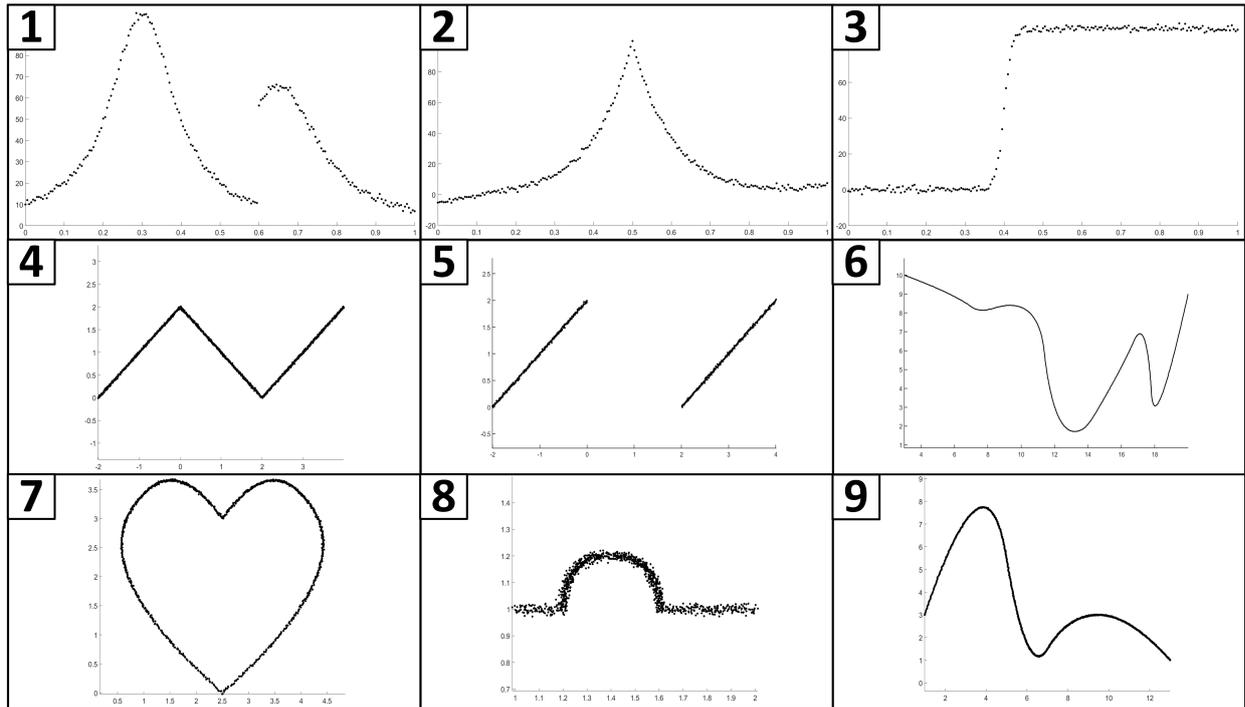


Abbildung 5.2: Verwendete Datensätze zur Evaluation der B-Spline-Kurven Approximation

sätze auch in weiteren Arbeiten verwendet (siehe z.B. Gálvez u. a. (2015); Bureick u. a. (2016a, 2019)). Datensatz 1 wird über

$$\phi_1(u) = \begin{cases} \frac{1}{0.01+(u-0.3)^2} & \text{for } u < 0.6 \\ \frac{1}{0.015+(u-0.65)^2} & \text{for } u \geq 0.6 \end{cases} \quad (5.1)$$

berechnet und enthält als besondere Schwierigkeit an der Stelle $u = 0.6$ einen Sprung. Für eine optimale Approximation wird an dieser Stelle ein dreifacher Knoten benötigt.

Datensatz 2 wird über

$$\phi_2(u) = \frac{100}{e^{|10u-5|}} + \frac{(10u-5)^5}{500} \quad (5.2)$$

berechnet und enthält als besondere Schwierigkeit an der Stelle $u = 0.5$ eine Spitze. Für eine optimale Approximation wird an dieser Stelle ein doppelter Knoten benötigt.

Datensatz 3 repräsentiert eine Stufenfunktion und wird über

$$\phi_3(u) = \frac{90}{1 + e^{-100(u-0.4)}} \quad (5.3)$$

berechnet. Die Schwierigkeit bei diesem Datensatz liegt in der Modellierung des steilen Anstiegs bei $u = 0.4$. Für eine optimale Approximation müssen in diesem Bereich vier eng zusammen liegende Knoten platziert werden. Bei den Testdatensätzen 4-9 handelt es sich um synthetische B-Spline-Kurven, deren Parameter den Tabellen A.1-A.6 zu entnehmen sind. Datensatz 4 und 5 enthalten jeweils zwei doppelte Knoten an den beiden Kanten. Bei Datensatz 5 wurden keine Datenpunkte zwischen den beiden Kanten ausgegeben. Der Datensatz 6 wurde bereits in Schmitt und Neuner (2015) zur Evaluation von B-Spline-Kurven Approximationen verwendet. Hier sind zwar keine multiplen Knoten erforderlich, allerdings müssen bei diesem Datensatz insgesamt 7 interne Knoten in unregelmäßigen Abständen platziert werden. Der herzförmige Datensätze 7 wurde bereits

in Bureick u. a. (2019) verwendet. Die optimale Approximation des oberen Knickes erfordert einen doppelten Knoten. Die B-Spline-Kurve beginnt und endet im unteren Bereich. Deshalb ist für die Modellierung des sichtbaren Knickes im unteren Bereich kein doppelter Knoten notwendig. Bei Datensatz 8 müssen wie bei Datensatz 6 ebenfalls 7 interne Knoten platziert werden. Für eine optimale Approximation wird zu Beginn und Ende des Halbbogens jeweils ein doppelter Knoten benötigt. Datensatz 9 wurde ebenfalls bereits in Bureick u. a. (2019) verwendet. Hierbei handelt es sich um einen vergleichsweise einfachen Datensatz. Die internen Knoten sind äquidistant über die gesamte Knotenspanne verteilt. Insgesamt sollen die ausgewählten Datensätze ein großes Spektrum möglicher Punktwolken abdecken, um eine möglichst umfassende Evaluation der verschiedenen Methoden zur Knotenvektorstwahl zu ermöglichen.

Zur Generierung der Beobachtungen aus den Testfunktionen und den synthetischen B-Spline-Kurven wird zunächst ein äquidistanter Ortsparametervektor $\bar{\mathbf{u}}$ mit r Punkten erzeugt:

$$\bar{\mathbf{u}} = [\bar{u}_1 \dots \bar{u}_r] = [0 \dots 1]. \quad (5.4)$$

Bei den Testfunktionen in den Datensätzen 1-3 wird der Soll-Beobachtungsvektor $\tilde{\mathbf{I}}$ mit einer ein-dimensionalen Punktwolke besetzt, die sich durch Einsetzen der Ortsparameter in die jeweilige Formel 5.1-5.3:

$$\tilde{\mathbf{I}} = [\phi(\bar{u}_1), \dots, \phi(\bar{u}_r)]^T \quad (5.5)$$

ergibt. Bei den Datensätzen 4-9 werden die Ortsparameter in $\bar{\mathbf{u}}$ und die weiteren, die B-Spline-Kurve definierenden Parameter p , n , ξ und \mathbf{x} in Formel 2.18 eingesetzt. Die Dimension der dadurch generierten (unverrauschten) Punktwolke $\mathbf{C} = [\mathbf{C}(\bar{u}_1), \dots, \mathbf{C}(\bar{u}_r)]^T$ entspricht der Dimension der gegebenen Kontrollpunkte \mathbf{x} . Bei den Datensätzen 4-9 handelt es sich um zweidimensionale Punktwolken. Der Soll-Beobachtungsvektor $\tilde{\mathbf{I}}$ ergibt sich somit durch:

$$\tilde{\mathbf{I}} = [C_x(\bar{u}_1), \dots, C_x(\bar{u}_r), C_y(\bar{u}_1), \dots, C_y(\bar{u}_r)]^T. \quad (5.6)$$

Die Eigenschaften der Datensätze sind in Tabelle 5.1 zusammengefasst. Die für die Approximation benötigten Parameter der Modellwahl entsprechen bei den Testdatensätzen 4-9 den gewählten und somit bekannten Parametern der synthetischen B-Spline-Kurven. Bei den Testdatensätzen 1-3 sind die Parameter der Modellwahl im Vorhinein unbekannt. In diesem Fall wurde der Grad der Basisfunktionen auf $p = 3$ fixiert und experimentell der minimale Wert für n bestimmt, bei dem der Globaltest (siehe Kapitel 3.1.2.1) bestanden werden kann.

Tabelle 5.1: Eigenschaften der Datensätze und der Modellwahl

Datensatz	r	Dimension	m	n	p
1	201	1	201	11	3
2	201	1	201	8	3
3	201	1	201	7	3
4	1001	2	2002	6	2
5	670	2	1340	6	2
6	2000	2	4000	9	2
7	1001	2	2002	6	2
8	1001	2	2002	9	2
9	1001	2	2002	5	2

5.1.3 Rauschmodell

Für die weiteren Untersuchungen werden die in Kapitel 5.1.2 beschriebenen Soll-Beobachtungsvektoren $\tilde{\mathbf{I}}$ mit normalverteiltem und unkorreliertem Rauschen

$$\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_{\text{noise}}^2 \mathbf{I}) \quad (5.7)$$

überlagert. Das normalverteilte Rauschen wird mit dem MATLAB-Zufallsgenerator *normrnd* der Statistics and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) generiert. Im Folgenden wird dieses Rauschen als Rauschmodell I bezeichnet. Die zu approximierenden Punkte der Punktwolke bzw. die Beobachtungen ergeben sich dann über:

$$\mathbf{l} = \tilde{\mathbf{I}} + \boldsymbol{\epsilon}. \quad (5.8)$$

Dieser Vorgang wird für jeden Testdatensatz 500 mal wiederholt. Für die Datensätze 1 bis 3 beträgt $\sigma_{\text{noise}} = 1$. Für die Datensätze 4, 5, sowie 7 bis 9 wird $\sigma_{\text{noise}} = 0.01$ gesetzt. Bei Datensatz 6 beträgt $\sigma_{\text{noise}} = 0.005$. Die simulierten Datensätze sind generell einheitslos. Die gewählten Werte für σ_{noise} müssen grundsätzlich in Relation zum Wertebereich der Funktionen (siehe Abbildung 5.2) betrachtet werden.

5.2 Konvergenzverhalten der verschiedenen Modifikationen beim EGA

Zunächst soll untersucht werden, ob die in Kapitel 4.3.2 beschriebenen Modifikationen des EGA zu einer beschleunigten Konvergenz führen. Die jeweils 500 simulierten Datensätze des Rauschmodells I (siehe Kapitel 5.1.3) werden durch die vier Ausprägungen des EGA approximiert. Die Ausprägungen entstehen durch das jeweilige An- oder Ausschalten der beiden in Kapitel 4.3.2 beschriebenen Modifikationen. Der Ortsparametervektor $\bar{\mathbf{u}}$ und die Wahl von p und n sind aus der Simulation bekannt und werden als gegeben angesehen. Die maximale Iterationsanzahl i_{max} wird auf 100 gesetzt. Für jede Ausprägung des EGA wird zu jeder Iteration i der bisher erzielte beste Fitnesswert Ω gespeichert und ausgegeben. Anschließend wird berechnet, bei welcher Iteration i mit dem Fitnesswert Ω der Globaltest bestanden wird (siehe Kapitel 3.1.2.1).

Tabelle 5.2 gibt den Mittelwert der Iteration an, bei dem der Globaltest erstmals bestanden wurde. Insgesamt führt die Verwendung des RIU-Algorithmus bei der Initialisierung dazu, dass sich die mittlere benötigte Iterationsanzahl bis zum Bestehen des Globaltests um ca. 21 % reduziert. Durch die Verwendung des veränderten Mutations-Mechanismus verringert sich die mittlere Iterationsanzahl um ca. 11 %. Beide Modifikationen kombiniert führen zu einer Reduktion um ca. 29 %. Allerdings gibt es bei den Datensätzen erhebliche Unterschiede.

Bei Datensatz 3 führt die Verwendung des RIU-Algorithmus bei der Initialisierung zu einer erhöhten Iterationsanzahl. Dies ist evtl. dadurch zu erklären, dass die Initialisierung durch den RIU-Algorithmus zunächst zu einer Konvergenz in ein lokales Optimum führt. Erst mit steigender Iterationsanzahl konvergiert der so modifizierte EGA in das globale Optimum. Bei den Datensätzen 1, 5, 6 und 9 sind die Unterschiede zwischen den Initialisierungs-Varianten des EGA gering. Bei den Datensätzen 2, 4, 7 und 8 führt die Verwendung des RIU-Algorithmus zu einer signifikanten Reduktion der Iterationsanzahl. Allgemein bringt die Verwendung des RIU-Algorithmus bei der Initialisierung dann einen signifikanten Vorteil, wenn der zu approximierende Datensatz abrupte Richtungswechsel z.B. durch Knicke, Kanten oder Ecken enthält. Die Modellierung abrupter Richtungswechsel erfordert multiple Knoten und diese können durch den RIU-Algorithmus im Gegensatz zur PT1-Methode generiert werden.

Der veränderte Mutations-Mechanismus führt bei den Datensätzen 1-4, 7 und 9 zu einer geringen Reduktion der Iterationsanzahl. Bei Datensatz 6 ist eine signifikante Reduktion der Iterationsanzahl zu beobachten. Bei den Datensätzen 5 und 8 ist hingegen teilweise eine leicht erhöhte Iterationsanzahl festzustellen. Generell sind hier keine speziellen Eigenschaften der Datensätze erkennbar, auf

die der veränderte Mutations-Mechanismus einen besonderen Einfluss hat.

Tabelle 5.2: Mittlere Iterationsanzahl, bei der der Globaltest bei den verschiedenen Varianten des EGA bestanden wird.

Initialisierung		PT1	PT1	RIU	RIU
Mutation		Konstant	Abnehmend	Konstant	Abnehmend
Datensatz	1	26.28	23.93	25.44	24.96
	2	17.14	17.07	11.77	11.64
	3	15.39	14.25	23.18	21.01
	4	27.62	24.69	6.42	6.42
	5	4.44	4.54	4.57	4.41
	6	73.01	58.28	74.35	59.87
	7	36.70	32.60	9.85	9.67
	8	10.90	11.63	8.26	9.11
	9	8.25	7.90	8.06	7.82
Summe		219.73	194.89	171.90	154.91

Diese Untersuchungen zeigen, dass der in der Initialisierung und dem Mutationsmechanismus modifizierte EGA grundsätzlich eine schnellere Konvergenz aufweist und bereits bei geringerer Iterationsanzahl zu einem Bestehen des Globaltests führt.

5.3 Präzision und Richtigkeit der verschiedenen Methoden zur Knotenvektorwahl

In diesem Abschnitt wird die Präzision und Richtigkeit der verschiedenen Methoden zur Knotenvektorwahl evaluiert. Es werden insgesamt 6 Methoden verglichen. Alle Methoden wurden bereits zuvor beschrieben und sind in Tabelle 5.3 zusammengefasst.

Tabelle 5.3: Vergleichene Methoden der Knotenvektorwahl

Abk.	Bezeichnung			Art	Herkunft	Kapitel
PT1	Knot placement technique			deterministisch	Piegl und Tiller (1997)	3.3.1.2
RIU	Residuenbasiertes iteratives Update			deterministisch	neu	4.1.1
CSA	Elitist Clonal Selection Algorithm			heuristisch	Gálvez u. a. (2015)	3.3.2.1
EMC	Evolutionäre Monte-Carlo Methode			heuristisch	Bureick u. a. (2016a) modifiziert	4.2
EGAori	Elitärer Genetischer	Algorithmus		heuristisch	Bureick u. a. (2019)	4.3
EGAmo	Elitärer Genetischer	Algorithmus		heuristisch	Bureick u. a. (2019) modifiziert	4.3.2

Wie in Kapitel 5.2 beschrieben ist der Ortsparametervektor $\bar{\mathbf{u}}$ und die Wahl von p und n aus der Simulation bekannt. Diese Parameter werden als gegeben angenommen. Mit allen Methoden der Knotenvektorwahl werden Knotenvektoren für die 500 simulierten Datensätze des Rauschmodells I (siehe Kapitel 5.1.3) bestimmt. Bei den heuristischen Verfahren wird die maximale Anzahl

an Iterationen $i_{\max} = 100$ gewählt. Mit den bestimmten Knotenvektoren wird die Schätzung der Kontrollpunkte $\hat{\mathbf{x}}$ mit der Methode der kleinsten Quadrate (siehe Kapitel 2.2.1 und 3.4) durchgeführt. Der mit Formel 2.34 berechnete a posteriori Varianzfaktor der Gewichtseinheit $\hat{\sigma}_0^2$ aus der Kontrollpunktschätzung wird verwendet, um den Globaltest (siehe Kapitel 3.1.2.1) zu berechnen. Der Prozentsatz der in den 500 Simulationsdurchläufen bestandenen Globaltests kann als wichtige Kennziffer zur Evaluierung der Präzision und Richtigkeit der verschiedenen Methoden zur Knotenvektorwahl herangezogen werden. Da das stochastische Modell bekannt ist und durch die Verwendung von Rauschmodell I keine groben Fehler oder Ausreißer in der Simulation vorliegen, ist in diesem Fall ein bestandener Globaltest ein Indiz für ein zutreffendes funktionales Modell in der B-Spline-Approximation. Da in der Simulation die korrekte Modellwahl und Parametrisierung verwendet wird, wird das funktionale Modell einzig durch den Knotenvektor signifikant beeinflusst. Das bedeutet hier, dass ein verworfener Globaltest nur durch einen nicht optimalen Knotenvektor verursacht werden kann.

Zur Evaluierung der Konvergenz der heuristischen Verfahren wird die Kontrollpunktschätzung mit den besten Knotenvektoren nach jeder Iteration durchgeführt. Mit den geschätzten Kontrollpunkten $\hat{\mathbf{x}}$ können die ausgeglichenen Beobachtungen $\hat{\mathbf{I}}$ über:

$$\hat{\mathbf{I}} = \mathbf{A}\hat{\mathbf{x}} \quad (5.9)$$

berechnet werden. Anschließend können die ausgeglichenen Beobachtungen $\hat{\mathbf{I}}$ mit den Soll-Beobachtungen $\tilde{\mathbf{I}}$ verglichen werden. Dieser Vergleich erfolgt auf Basis der Punkte, d.h. bei zweidimensionalen Beobachtungen müssen $\hat{\mathbf{I}}$ und $\tilde{\mathbf{I}}$ in ausgeglichene Punktwolke $\hat{\mathbf{Q}}$ und Soll-Punktwolke $\tilde{\mathbf{Q}}$ transformiert werden. Hier exemplarisch für $\hat{\mathbf{Q}}$ gezeigt:

$$\hat{\mathbf{Q}} = \begin{bmatrix} \hat{l}_1 & \hat{l}_{r+1} \\ \vdots & \vdots \\ \hat{l}_r & \hat{l}_{2r} \end{bmatrix} \quad (5.10)$$

Zwischen ausgeglichener und Soll-Punktwolke wird die Differenz-Matrix $\Delta\mathbf{Q}$ berechnet:

$$\Delta\mathbf{Q} = \begin{bmatrix} \Delta Q_{x,1} & \Delta Q_{y,1} \\ \vdots & \vdots \\ \Delta Q_{x,r} & \Delta Q_{y,r} \end{bmatrix} = \tilde{\mathbf{Q}} - \hat{\mathbf{Q}}. \quad (5.11)$$

Die gemittelte euklidische Distanz zwischen $\tilde{\mathbf{Q}}$ und $\hat{\mathbf{Q}}$ ergibt den Mean Absolute Error (MAE):

$$MAE = \frac{1}{r} \sum_{k=1}^r \sqrt{\Delta Q_{x,k}^2 + \Delta Q_{y,k}^2}. \quad (5.12)$$

In den nachfolgenden Ergebnissen wird anstatt des MAE der kumulierte Absolute Error (kAE) dargestellt. Der kAE kann mit folgender Formel berechnet werden:

$$kAE = r \cdot MAE = \sum_{k=1}^r \sqrt{\Delta Q_{x,k}^2 + \Delta Q_{y,k}^2}. \quad (5.13)$$

Bei den eindimensionalen Datensätzen 1-3 sind $\hat{\mathbf{Q}}$, $\tilde{\mathbf{Q}}$ und $\Delta\mathbf{Q}$ Spaltenvektoren. Die Formel zur Berechnung des MAE ändert sich dort zu:

$$MAE = \frac{1}{r} \sum_{k=1}^r \sqrt{\Delta Q_k^2} \quad (5.14)$$

und die Formel des kAE zu:

$$kAE = \sum_{k=1}^r \sqrt{\Delta Q_k^2}. \quad (5.15)$$

Die über die Formeln 5.10 bis 5.15 berechneten MAE oder kAE sind ein Maß für die Richtigkeit der Approximation. Im folgenden Kapitel 5.3.1 werden für die Datensätze jeweils acht Kennziffern des kAE dargestellt, die aus den 500 Wiederholungen der Simulation gewonnen werden. Die Kennziffern sind *Minimum* (Min.), *Maximum* (Max.), *Mittelwert* (Mittel), *Median*, *Modus*, *Standardabweichung* (Std.), sowie die *untere* (\downarrow KI (95%)) und *obere* (\uparrow KI (95%)) *Intervallgrenze* des Konfidenzintervalls (Irrtumswahrscheinlichkeit $\alpha = 5\%$). Alternativ könnte die untere Intervallgrenze mit $KI_{l,\alpha/2=0.025}$ und die obere Intervallgrenze mit $KI_{u,1-\alpha/2=0.975}$ bezeichnet werden. Alle Kennziffern sind beispielhaft in Abbildung 5.3 dargestellt. Die Berechnung von *Minimum*,

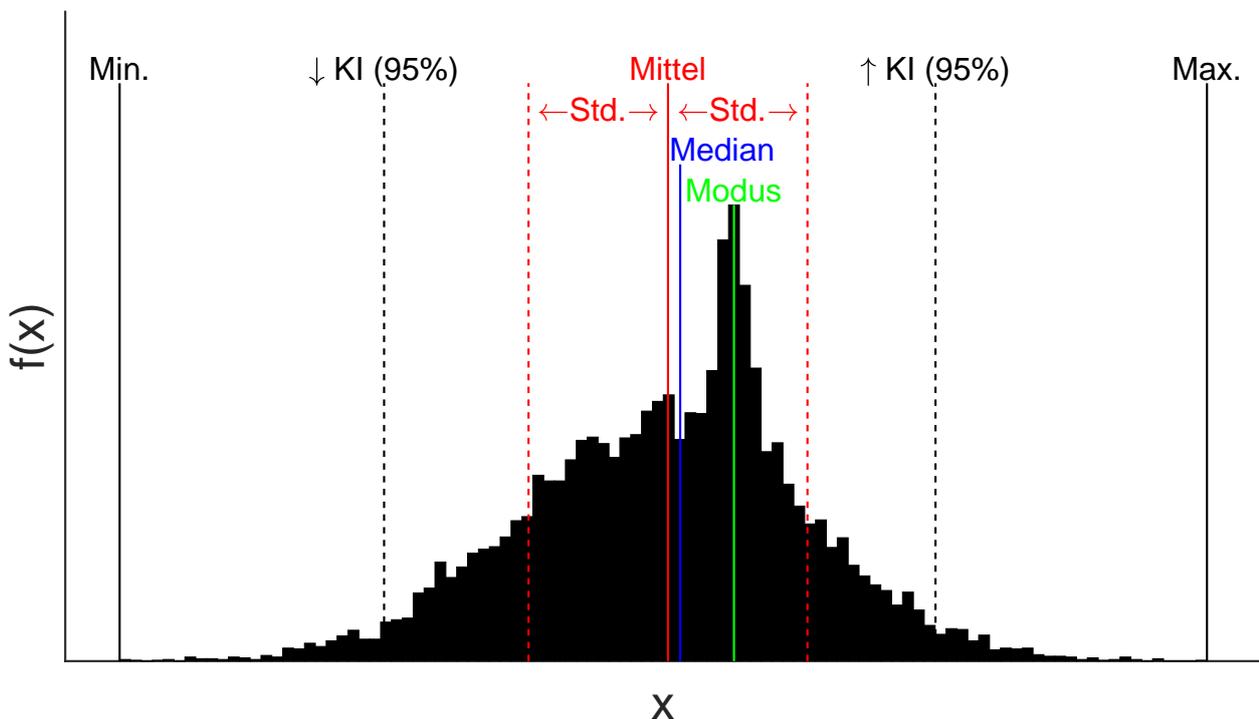


Abbildung 5.3: Kennziffern einer Stichprobe

Maximum, *Mittelwert*, *Median* und *Standardabweichung* ist allgemein bekannt und wird an dieser Stelle nicht angegeben. Der *Modus* bezeichnet den häufigsten Wert einer Stichprobe. Da die kAE-Werte eine kontinuierliche Zufallsvariable darstellen, wird die Kerndichte der kAE-Werte durch die MATLAB-Funktion *ksdensity* approximiert (siehe MathWorks[®] (2019a) und Kapitel 4.2.1.1). Der kAE-Wert mit der höchsten Kerndichte wird dann als *Modus* bezeichnet. Eine andere Möglichkeit ist es, aus den gegebenen Daten ein Histogramm zu erstellen und die Klasse mit der größten Häufigkeit als *Modus* zu wählen. Dieses Vorgehen ist zwar aufgrund seiner Anschaulichkeit in Abbildung 5.3 dargestellt, wird aber aufgrund hoher Diskretisierungsfehler hier nicht angewendet. Zur Bestimmung der Konfidenzintervallgrenzen wird wie in Alkhatib u. a. (2009) und den dort angegebenen Quellen vorgegangen. Zunächst werden die kAE-Werten der Größe nach sortiert. Die äußeren $(\alpha/2)\%$ der sortierten kAE-Werte werden an beiden Seiten abgeschnitten. Das Minimum und Maximum der verbleibenden Daten geben *untere* und *obere Intervallgrenze* des Konfidenzintervalls an.

5.3.1 Ergebnisse

5.3.1.1 Globaltest

In Tabelle 5.4 sind die Prozentsätze der bestandenen Globaltests ($\alpha = 5\%$) für die verschiedenen Datensätze und Methoden der Knotenvektorstimmung dargestellt. Da die Modellwahl gegeben ist, wurde der einseitige Globaltest aus Kapitel 3.1.2.1 verwendet. Bei den heuristischen Verfahren beruhen die Ergebnisse auf dem Knotenvektor der letzten Iteration. Bei den Datensätzen 4 bis 9 ist zusätzlich zum Vergleich der mit dem wahren Knotenvektor erzielte Prozentsatz bestandener Globaltests angegeben.

Tabelle 5.4: Anteil bestandener Globaltest

DS.	Prozentsatz bestandener Globaltest ($\alpha = 5\%$)						Wahr
	PT1	RIU	CSA	EMC	EGAori	EGAmo	
1	0.0 %	0.0 %	54.4 %	26.8 %	95.0 %	95.6 %	-
2	0.0 %	1.0 %	79.4 %	92.6 %	93.2 %	93.8 %	-
3	0.0 %	1.2 %	94.4 %	91.2 %	92.4 %	93.4 %	-
4	0.0 %	64.6 %	1.2 %	79.4 %	93.2 %	95.0 %	94.4 %
5	0.0 %	94.2 %	96.2 %	96.8 %	96.6 %	96.6 %	95.4 %
6	0.0 %	0.0 %	0.0 %	0.0 %	74.4 %	90.6 %	95.6 %
7	0.0 %	0.0 %	0.6 %	65.2 %	89.8 %	96.4 %	96.0 %
8	0.2 %	30.8 %	73.2 %	94.4 %	94.8 %	95.2 %	95.2 %
9	5.6 %	7.0 %	0.8 %	40.4 %	94.8 %	94.8 %	94.8 %

Grundsätzlich erzielen der EGAori und insbesondere der EGAmo bei allen Datensätzen einen konstanten und hohen Prozentsatz bestandener Globaltests. Im Falle des EGAmo liegt der Prozentsatz bei den meisten Datensätzen mit ca. 95 % im Bereich des Prozentsatzes, der unter Verwendung des wahren Knotenvektors erzielt wird. Dies ist auch der Prozentsatz, der bei dem verwendeten Rauschmodell und der angesetzten Irrtumswahrscheinlichkeit α zu erwarten ist. Einzig bei Datensatz 6 ist mit 90.6 % ein geringfügig kleinerer Prozentsatz an bestandenen Globaltests festzustellen. Unter Verwendung des EGAori werden bei den Datensätzen 1-5 und 8-9 vergleichbare Prozentzahlen erzeugt. Bei Datensatz 7 und insbesondere bei Datensatz 6 wird eine signifikant geringere Prozentzahl als beim EGAmo erzielt. Bei den anderen Methoden wird mit PT1 ein konstant niedriger Prozentsatz an bestandenen Globaltests erzielt. Lediglich bei Datensatz 8 und 9 gibt es Simulationen, bei denen der Globaltest bestanden wurde. Mit RIU, dem CSA und der EMC-Methode werden in den Datensätzen sehr heterogene Prozentzahlen bestandener Globaltests erzeugt, d.h. die Performance dieser Methoden zur Knotenvektorstimmung ist stark von dem jeweiligen Datensatz abhängig.

5.3.1.2 kAE

Datensatz 1 In Tabelle 5.5 sind die Kennziffern des kAE für Datensatz 1 dargestellt. In den ersten beiden Zeilen sind die Ergebnisse der beiden deterministischen Verfahren PT1 und RIU dargestellt. Anschließend folgen die Ergebnisse der vier heuristischen Verfahren zu den ausgewählten Iterationen 1, 25, 50, 75 und 100. Pro Iteration und Kennziffer ist der jeweils geringste kAE-Wert fett markiert. Um einen besseren Eindruck über die erwartbaren kAE-Werte zu vermitteln, sind in der letzten blau markierten Zeile die kAE-Werte dargestellt, die mit den rohen Beobachtungen \mathbf{I} bzw. \mathbf{Q} erzielt werden. Dazu wird in Formel 5.11 $\hat{\mathbf{Q}}$ durch \mathbf{Q} ersetzt. Bei eindimensionalen Daten (Datensatz 1-3) folgt die Verteilung der kAE-Werte der rohen Beobachtungen einer Halb-Normalverteilung. Die beiden deterministischen Verfahren PT1 und RIU erzielen bei diesem Datensatz ein sehr ähnliches Ergebnis. RIU weist sogar gegenüber PT1 einen leicht erhöhten kAE auf. Somit starten die heuristischen Methoden EGAori und EGAmo mit einer vergleichbaren Initialisierung. Die mit den beiden Versionen des EGA erzielten kAE-Werte schrumpfen deutlich schneller als die der EMC-Methode

und des CSA. Auch in der finalen Iteration 100 weisen die mit EGAori und EGAmođ erzielten Ergebnisse einen deutlich kleineren mittleren kAE auf als die Ergebnisse von EMC und CSA. Die mit dem EGAmođ erzielten Ergebnisse weisen einen geringfügig kleineren kAE auf als die Ergebnisse des EGAori. Die Konfidenzintervalle der mit dem CSA und der EMC-Methode erzielten kAE sind in späteren Iterationen (ab Iteration 25) deutlich größer als die Konfidenzintervalle der kAE-Werte, die mit EGAori oder EGAmođ erzielt wurden. Bei Verwendung des CSA und der EMC-Methode ist zu beobachten, dass der Mittelwert des kAE größer als der Median und dieser wiederum größer als der Modus des kAE ist. Dies spricht für eine rechtsschiefe Verteilung der kAE-Werte, die durch eine signifikante Anzahl an Simulationsdurchläufen hervorgerufen wird, bei denen im Vergleich zu den besten Durchläufen ein deutlich höherer kAE-Wert erzielt wurde.

Im Vergleich mit den kAE-Werten der rohen Beobachtungen zeigt sich, dass die kAE-Werte der beiden deterministischen Verfahren PT1 und RIU diese Werte deutlich übersteigen. Das bedeutet, dass die Verwendung einer deterministischen Methode bei diesem Datensatz zu einer Verringerung der Richtigkeit führt. Die Kennziffern der heuristischen Verfahren liegen in Iteration 100 zum Teil deutlich unter denen der rohen Punktwolke. Das bedeutet, dass die Richtigkeit durch die Approximation mithilfe einer heuristischen Methode bei diesem Datensatz gesteigert wird. An dieser Stelle erfolgt

Tabelle 5.5: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 1

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	766.334	794.175	779.104	779.050	777.906	4.803	769.852	788.436
-	RIU	747.777	819.832	799.616	803.323	804.915	12.495	764.816	814.416
1	CSA	238.650	771.741	483.884	485.226	482.457	88.512	305.759	647.157
	EMC	206.237	676.542	451.582	447.618	442.401	80.357	294.597	598.933
	EGAori	249.162	712.127	488.951	489.619	482.195	91.128	306.945	658.277
	EGAmođ	264.892	784.977	499.945	498.880	531.306	95.148	329.843	694.485
25	CSA	39.304	474.066	153.934	128.829	65.654	91.587	46.011	264.949
	EMC	104.716	377.655	248.552	253.293	261.290	52.580	142.358	345.280
	EGAori	32.430	133.963	60.236	59.546	60.111	11.650	40.756	85.377
	EGAmođ	33.176	100.893	58.357	57.561	55.959	10.470	39.708	80.210
50	CSA	34.724	452.143	132.952	63.678	52.592	92.197	40.509	246.240
	EMC	60.421	293.196	164.673	162.194	156.241	47.805	78.894	252.476
	EGAori	32.783	84.617	54.663	54.020	53.026	8.992	38.569	72.852
	EGAmođ	32.286	84.056	53.627	52.943	52.665	8.712	38.186	75.008
75	CSA	34.450	436.891	130.641	61.558	51.555	90.539	40.310	243.491
	EMC	50.820	257.299	116.628	108.183	87.970	40.060	63.745	217.114
	EGAori	31.760	83.896	53.326	52.756	52.251	8.593	38.138	71.735
	EGAmođ	32.285	82.576	52.230	51.632	51.020	8.168	37.374	72.361
100	CSA	34.321	252.899	129.308	60.197	51.481	89.542	40.133	242.570
	EMC	49.449	238.201	103.333	92.624	80.841	36.163	56.842	186.216
	EGAori	31.766	83.999	52.708	52.340	51.841	8.168	38.332	69.267
	EGAmođ	32.234	80.365	51.460	51.243	51.016	7.659	37.507	66.633
-	Roh	133.363	182.145	160.359	160.490	160.840	7.956	144.528	175.368

der Hinweis, dass bei den heuristischen Verfahren auch ein Anstieg der kAE-Werte im Verlauf der Iterationen möglich ist. So steigen z.B. der minimale und maximale kAE-Wert des EGAori zwischen Iteration 75 und Iteration 100 leicht an. Da der Fitnesswert Ω bei allen verwendeten heuristischen Verfahren nicht ansteigen kann, erscheint dieses Verhalten zunächst überraschend. Allerdings ist dieses Verhalten aufgrund des Unterschieds zwischen Fitnesswert (Abstand zur Ist-Punktwolke) und kAE-Wert (Abstand zur Soll-Punktwolke) zu erklären und plausibel. In Abbildung 5.4 sind die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 1 dargestellt. Bei den heuristischen Methoden ist das Ergebnis der letzten Iteration dargestellt. Zusätzlich werden die bereits in Tabelle 5.5 angegebenen Kennziffern *Mittelwert*, *Median* und *Modus* in Abbildung 5.4 dargestellt. Da der *Modus* über Kerndichteschätzung bestimmt wurde, muss dieser nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen. Mit den Methoden PT1 (Abbildung 5.4a), EGAori (Abbildung 5.4e) und EGAmođ (Abbildung 5.4f) werden relativ symmetrische Ver-

teilungen der kAE -Werte erzielt. RIU (Abbildung 5.4b) erzeugt eine asymmetrische, linksschiefe Verteilung der kAE -Werte. Mit dem CSA (Abbildung 5.4c) wird eine bimodale Verteilung der kAE -Werte erzielt. Dies bedeutet, dass der CSA in beinahe der Hälfte der Simulationsdurchläufe in ein lokales Optimum konvergiert ist. Die EMC-Methode (Abbildung 5.4d) erzeugt eine asymmetrische, rechtsschiefe Verteilung der kAE -Werte. Diese rechtsschiefe Verteilung wird ebenfalls durch Simulationsdurchläufe hervorgerufen, bei denen die EMC-Methode in ein lokales Optimum konvergiert ist. Im Vergleich zum CSA ist die Anzahl dieser Simulationsdurchläufe deutlich geringer.

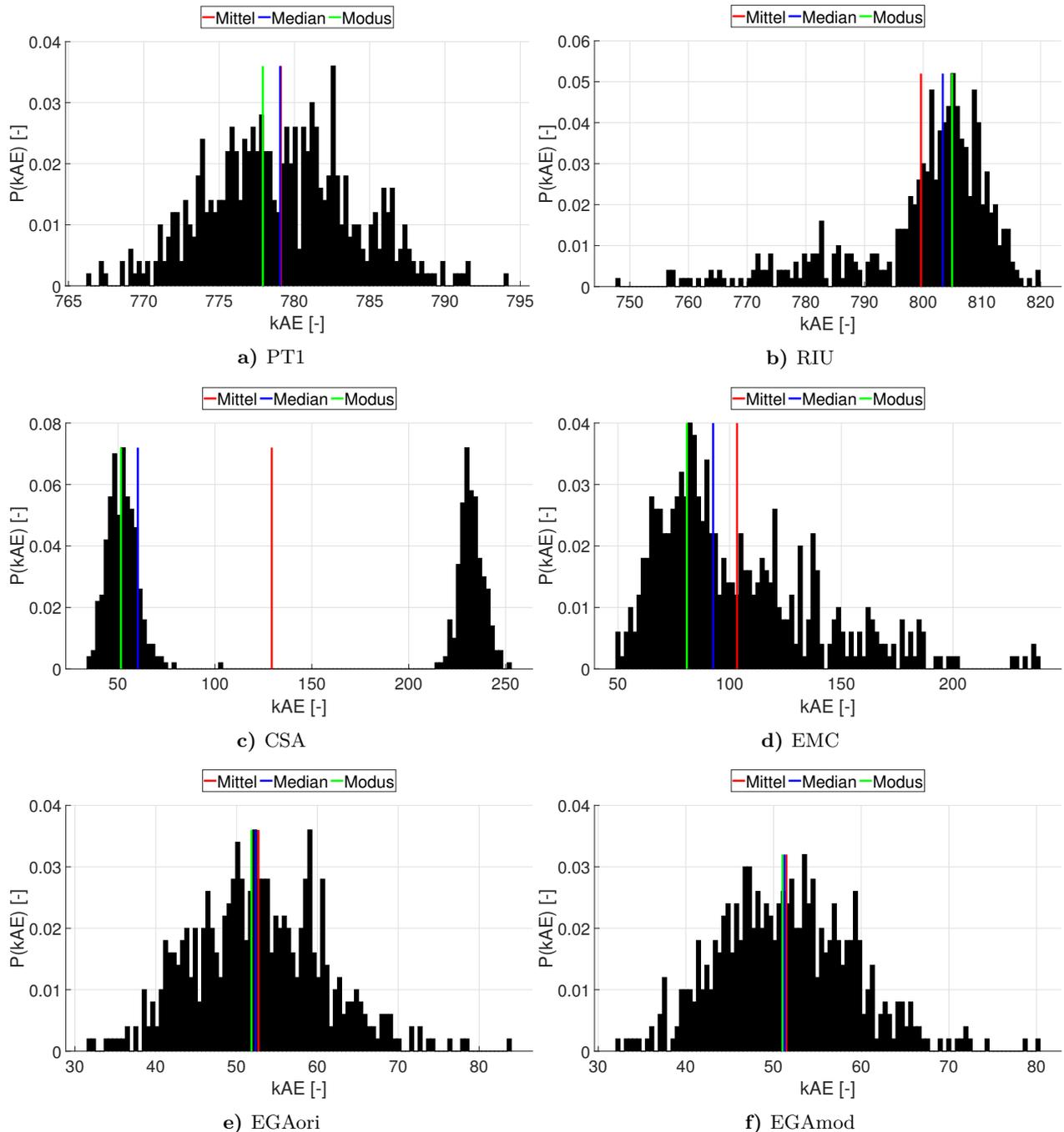


Abbildung 5.4: Histogramm der Verteilung der kAE -Werte bei Datensatz 1. Auf der x-Achse sind die kAE -Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE -Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.5 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

In den Abbildungen 5.5, 5.6, 5.7 und 5.8 sind die Kennziffern *Minimum*, *Maximum*, *Mittelwert*, *Median* und das *Konfidenzintervall* der mit den verschiedenen heuristischen Methoden erzielten kAE-Werte graphisch pro Iteration dargestellt. Auf eine Darstellung der *Standardabweichung* und des *Modus* wird aus Gründen der Übersichtlichkeit verzichtet.

Die mit dem CSA erzielten kAE in Abbildung 5.5 weisen eine hohe Schwankungsbreite auf, die sich in einem breiten Konfidenzband und einem großen Abstand zwischen minimalen und maximalen kAE-Wert widerspiegelt. Signifikant ist dabei die Abweichung zwischen Mittelwert und Median, die (wie oben bereits beschrieben) auf eine asymmetrische, bimodale Verteilung der kAE-Werte zurückzuführen ist.

In Abbildung 5.6 sind die mit der EMC-Methode erzielten kAE-Werte dargestellt. Diese weisen im

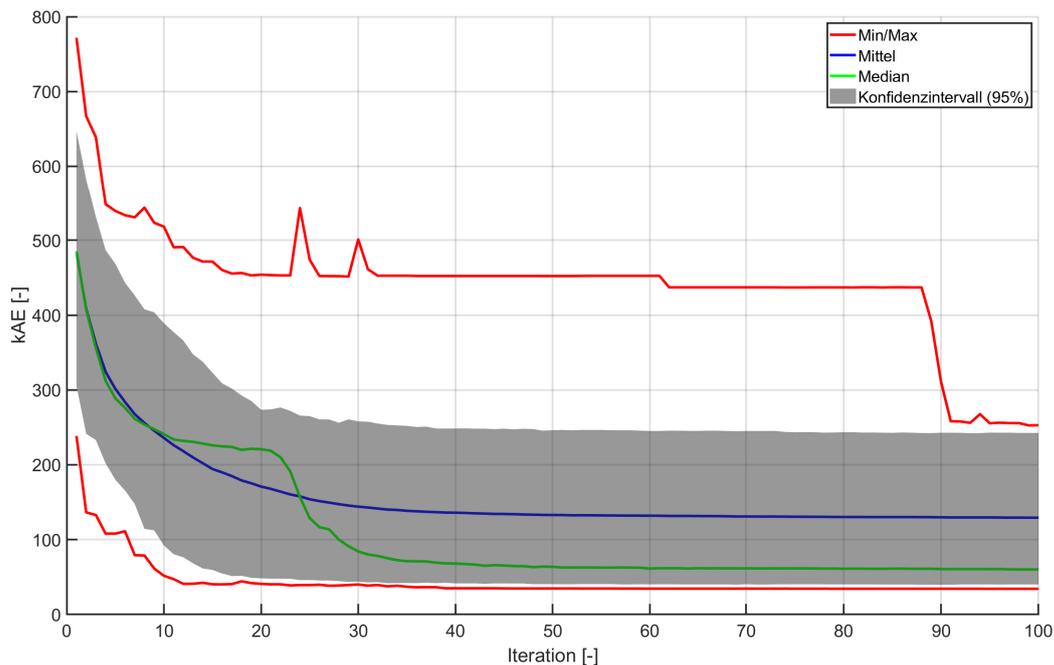


Abbildung 5.5: Graphische Darstellung des kAE des CSA bei Datensatz 1

Vergleich zu den mit dem CSA erzielten Ergebnissen eine kleinere Schwankungsbreite auf. Auch hier kann festgestellt werden, dass die Differenz zwischen Mittelwert und Median wesentlich kleiner ist. Dennoch kann man auch hier eine asymmetrische, rechtsschiefe Verteilung der kAE-Werte feststellen. Bei der EMC-Methode ist auf den in Iteration 16 auftretenden Knick hinzuweisen. Ab dieser Iteration werden die Knoten nicht mehr aus der gleichgewichteten Wahrscheinlichkeit \mathbf{w}_{eq} , sondern aus der Wahrscheinlichkeit \mathbf{w} , gezogen. Die Wahrscheinlichkeit \mathbf{w} wird aus den bisher besten Knotenvektoren berechnet (siehe Kapitel 4.2.1.1).

Die Abbildungen 5.7 und 5.8 stellen die mit dem EGAori und dem EGAmođ erzielten kAE-Werte dar. Beide Abbildungen ähneln sich bei diesem Datensatz stark, deshalb gelten die folgenden Aussagen für beide Methoden. Das Konfidenzintervall der mit dem EGA erzielten kAE-Werte ist deutlich kleiner als das der zuvor beschriebenen Methoden. Mittelwert und Median liegen eng zusammen. Dies spricht für eine symmetrische Verteilung der kAE-Werte. Ab Iteration 33 beim EGAori und ab Iteration 28 beim EGAmođ sind nur noch geringfügige Änderungen in den Kennziffern des kAE zu erkennen.

In Abbildung 5.9 sind die Mittelwerte des kAE der verschiedenen Methoden bei Datensatz 1 dargestellt. Für den EGAori und den EGAmođ ist im Vergleich zu den anderen Methoden eine zügige Konvergenz zu beobachten. In Iteration 100 wird bei beiden Methoden ein mittlerer kAE von ca. 50 erreicht. Der EGAmođ weist zwischen Iteration 5 und 25 einen geringfügig kleineren mittleren

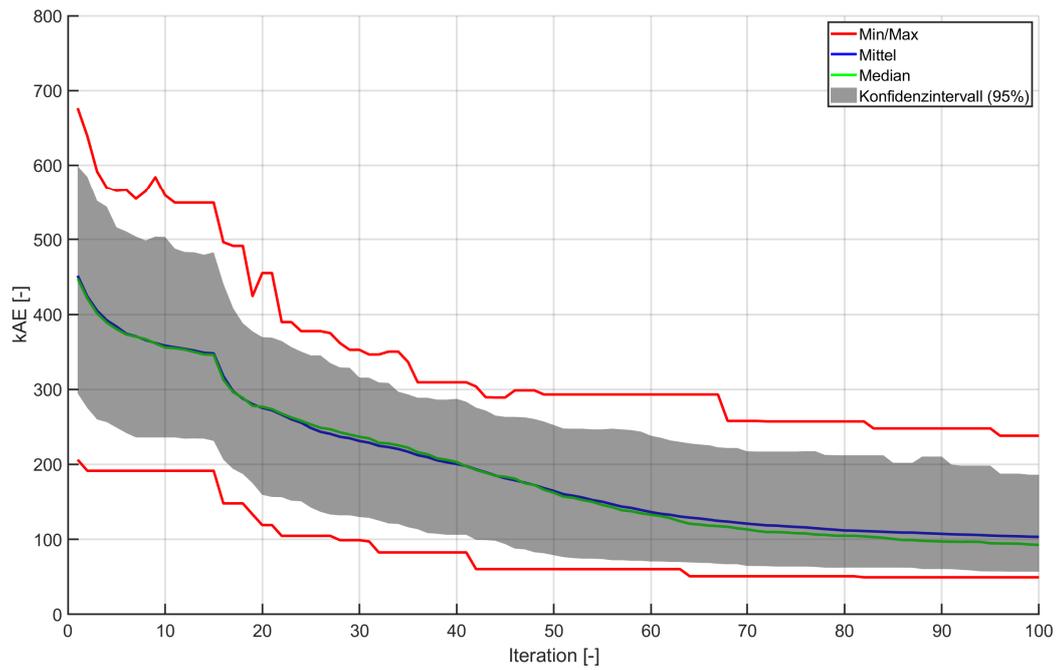


Abbildung 5.6: Graphische Darstellung des kAE der EMC bei Datensatz 1

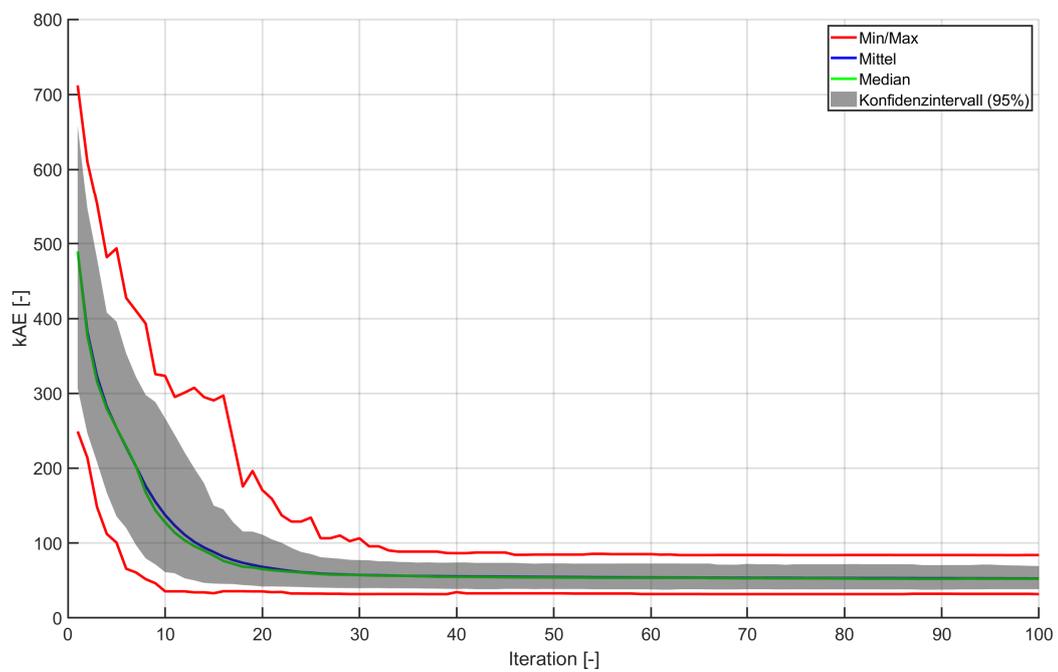


Abbildung 5.7: Graphische Darstellung des kAE des EGAori bei Datensatz 1

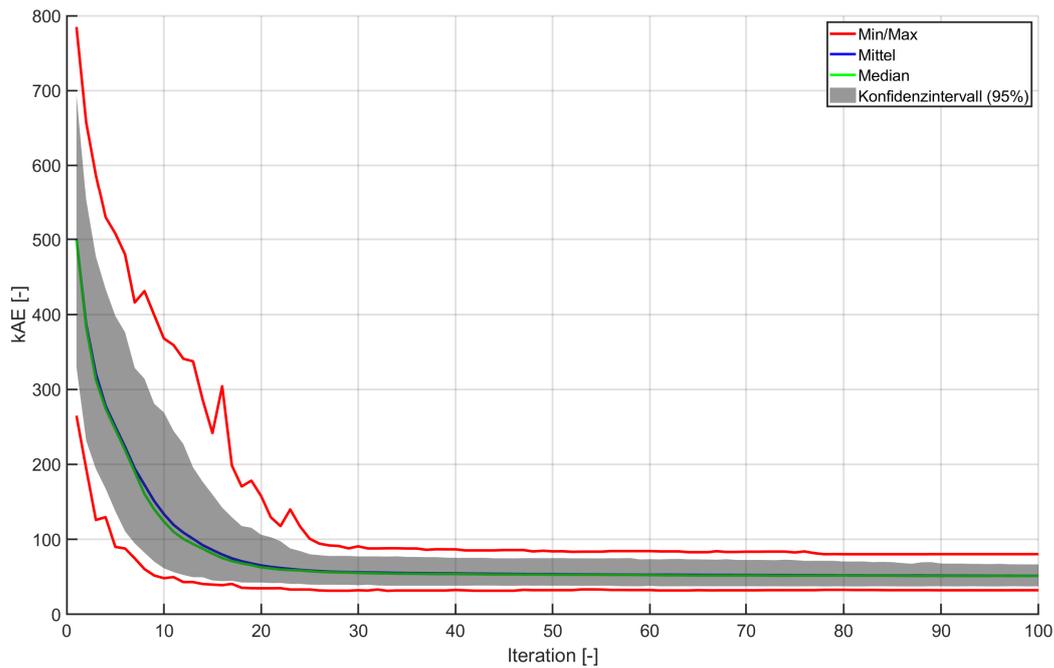


Abbildung 5.8: Graphische Darstellung des kAE des EGAmod bei Datensatz 1

kAE auf als der EGAori. Sowohl CSA als auch die EMC-Methode erzielen ab Iteration 3 in allen Iterationen einen deutlich höheren mittleren kAE. Der mittlere kAE des CSA scheint zu einem Wert von ca. 130 zu konvergieren, was auf eine Konvergenz zu einem lokalen Minimum hindeutet. In Abbildung 5.10 ist die Häufigkeitsverteilung der durch die verschiedenen Methoden im Rahmen der Simulation erstellten internen Knoten dargestellt. Die internen Knoten sind in verschiedenen

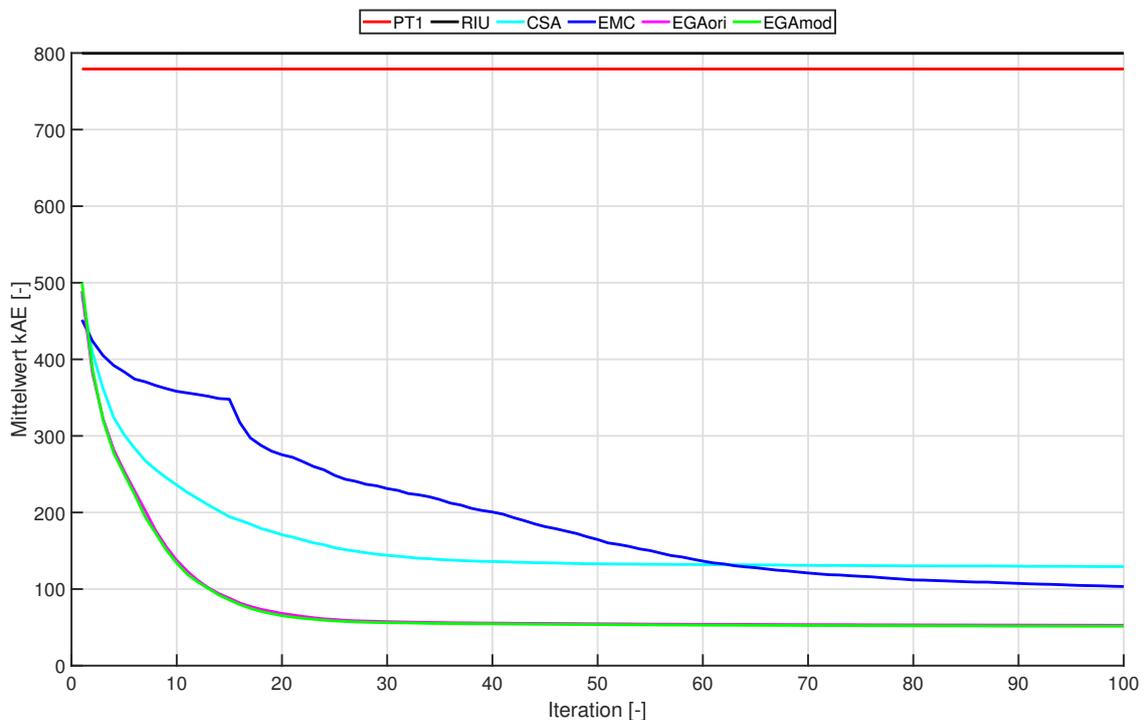


Abbildung 5.9: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 1

Farben dargestellt. Abbildung 5.10a zeigt die Häufigkeitsverteilung der mit PT1 erstellten internen Knoten. Da in diese Methode einzig die in den 500 Simulationsdurchläufen unveränderten Ortspareparameter eingehen, ist auch der durch PT1 erstellte Knotenvektor in allen Simulationsdurchläufen identisch¹. In Datensatz 1 sind die Knoten bei PT1 nahezu äquidistant verteilt. In Abbildung 5.10b ist die Häufigkeitsverteilung der mit RIU erstellten internen Knoten dargestellt. Gegenüber PT1 ist eine höhere Knotenkonzentration bei 0.3 und 0.6 zu erkennen. Zwischen 0.35 und 0.58 sind hingegen keine Knoten platziert. Die Häufigkeitsverteilung der durch den CSA erzielten Knoten in Abbildung 5.10c enthält ein Maximum bei 0.60 und vier kleinere Maxima bei 0.23, 0.30, 0.37 und 0.75. Das deutlich höhere Maximum bei 0.6 deutet auf die Platzierung eines multiplen Knotens an dieser Stelle hin. Im Bereich zwischen 0 und 0.2 wurden ebenfalls Knoten gewählt. Abbildung 5.10d zeigt die Häufigkeitsverteilung der mit der EMC-Methode erzielten internen Knoten. Auch hier ist ein Maximum bei 0.60 zu sehen. Im Gegensatz zum CSA sind bei der EMC-Methode die restlichen Knoten ohne klares Maximum verteilt. Sie konzentrieren sich in einem Bereich zwischen 0.2 und 0.4 sowie zwischen 0.65 und 0.82. Die Abbildungen 5.10e und 5.10f zeigen die Häufigkeitsverteilung der mit dem EGAori und dem EGAmo erzielten internen Knoten. Beide Abbildungen sind fast identisch. Tendenziell ähneln die beiden Abbildungen dem Ergebnis des CSA, allerdings sind das große Maximum sowie die vier kleineren Maxima stärker ausgeprägt. Auch kommen beim EGAori und beim EGAmo keine Knoten in dem Bereich zwischen 0 und 0.2 vor. Da bei diesem Datensatz die wahren Positionen der internen Knoten unbekannt sind, kann auf Basis von Abbildung 5.10 keine Aussage über die Richtigkeit der erstellten internen Knoten getroffen werden.

Datensatz 2 Tabelle 5.6 zeigt die Kennziffern des kAE für Datensatz 2. Der RIU-Algorithmus erreicht bei diesem Datensatz einen deutlich geringeren kAE als PT1. Dadurch startet der EGAmo mit einem deutlich kleineren kAE und konvergiert schneller als die anderen heuristischen Verfahren. Bei diesem Datensatz weist der CSA ein deutlich breiteres Konfidenzintervall auf als die anderen heuristischen Verfahren. Die Werte für Mittelwert, Median und Modus in Iteration 100 deuten wiederum auf eine rechtsschiefe Verteilung der kAE-Werte hin. Bei den anderen heuristischen Verfahren ist dies in geringerem Maße auch zu beobachten.

Die kAE-Kennziffern bei Verwendung des PT1 übersteigen die der rohen Beobachtungen sehr deutlich. Hier ist wiederum eine Verringerung der Richtigkeit festzustellen. Bei Verwendung des RIU ergeben sich kAE-Kennziffern in der Größenordnung der kAE-Kennziffern der rohen Beobachtungen. Mit den heuristischen Verfahren werden in Iteration 100 hingegen deutlich geringere kAE-Werte erzielt als mit den rohen Beobachtungen. Das bedeutet, dass die Richtigkeit auch bei diesem Datensatz durch die Verwendung einer heuristischen Methode gesteigert wird.

Abbildung 5.11 unterstützt die Vermutung einer rechtsschiefen Verteilung der kAE bei den heuristischen Methoden (Abbildungen 5.11c-5.11f). Die Verwendung des CSA (Abbildung 5.11c) führt wie in Datensatz 1 zu einer bimodalen Verteilung der kAE-Werte. Allerdings ist die Anzahl der Simulationsdurchläufe, bei denen der CSA in ein lokales Optimum konvergiert, deutlich geringer als bei Datensatz 1. Mit PT1 (Abbildung 5.11a) wird eine relativ symmetrische Verteilung der kAE-Werte erzielt. Der RIU-Algorithmus (Abbildung 5.11b) führt zu einer bimodalen Verteilung der kAE-Werte. Die graphische Darstellungen der kAE-Werte pro Iteration für Datensatz 2 sind im Anhang in den Abbildungen A.1, A.2, A.3 und A.4 zu finden. Die Kernaussagen lassen sich bereits Tabelle 5.6 und Abbildung 5.11 entnehmen.

Abbildung 5.12 zeigt den Mittelwert des kAE der verschiedenen Methoden. Durch die gute Initialisierung des EGAmo konvergiert dieser schon ab der 8. Iteration. In der letzten Iteration 100 erreichen EMC-Methode, EGAori und EGAmo einen vergleichbaren mittleren kAE. Der Mittelwert des durch den CSA erzielten kAE liegt leicht über dem der anderen heuristischen Verfahren. In Abbildung 5.13 sind die Häufigkeitsverteilungen der internen Knoten bei Datensatz 2 dargestellt. Der PT1 erzeugt, wie bei Datensatz 1, äquidistante interne Knoten (siehe Abbildung 5.13a). Bei der

¹Dadurch lässt sich aus dieser Abbildung sehr leicht die bei diesem Datensatz benötigte Anzahl der zu bestimmenden internen Knoten ablesen

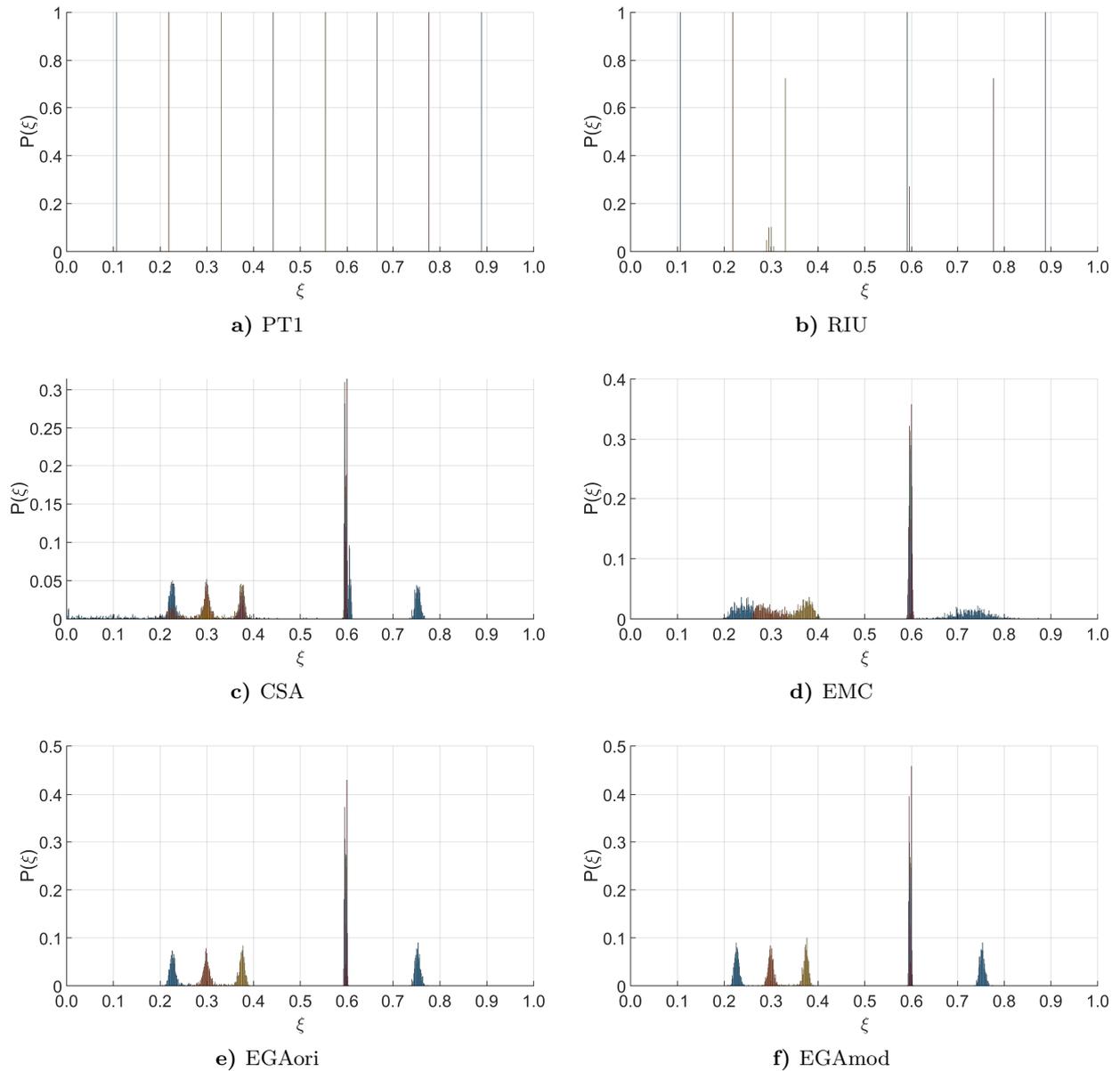


Abbildung 5.10: Histogramm der Verteilung der acht internen Knoten bei Datensatz 1. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

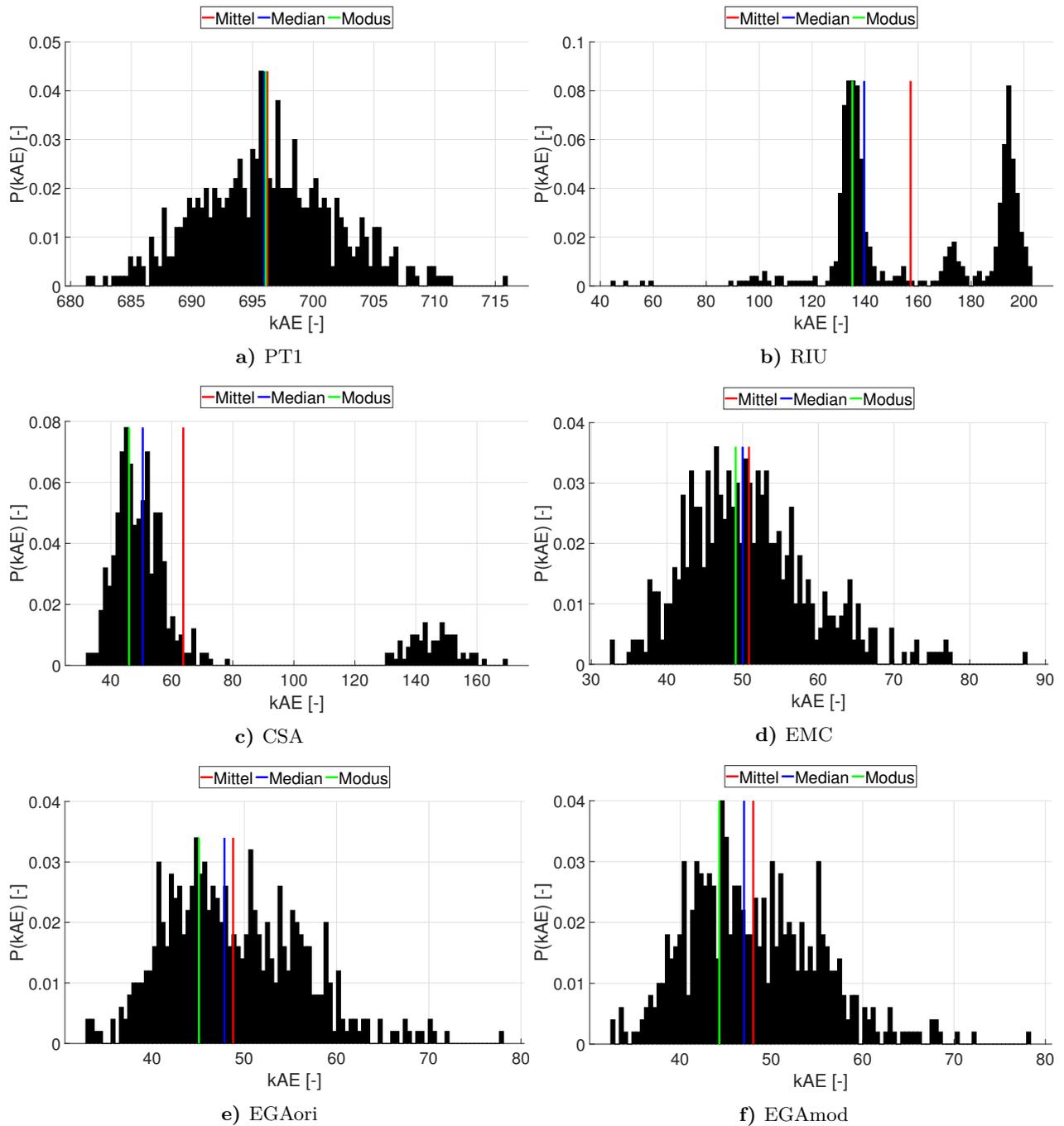
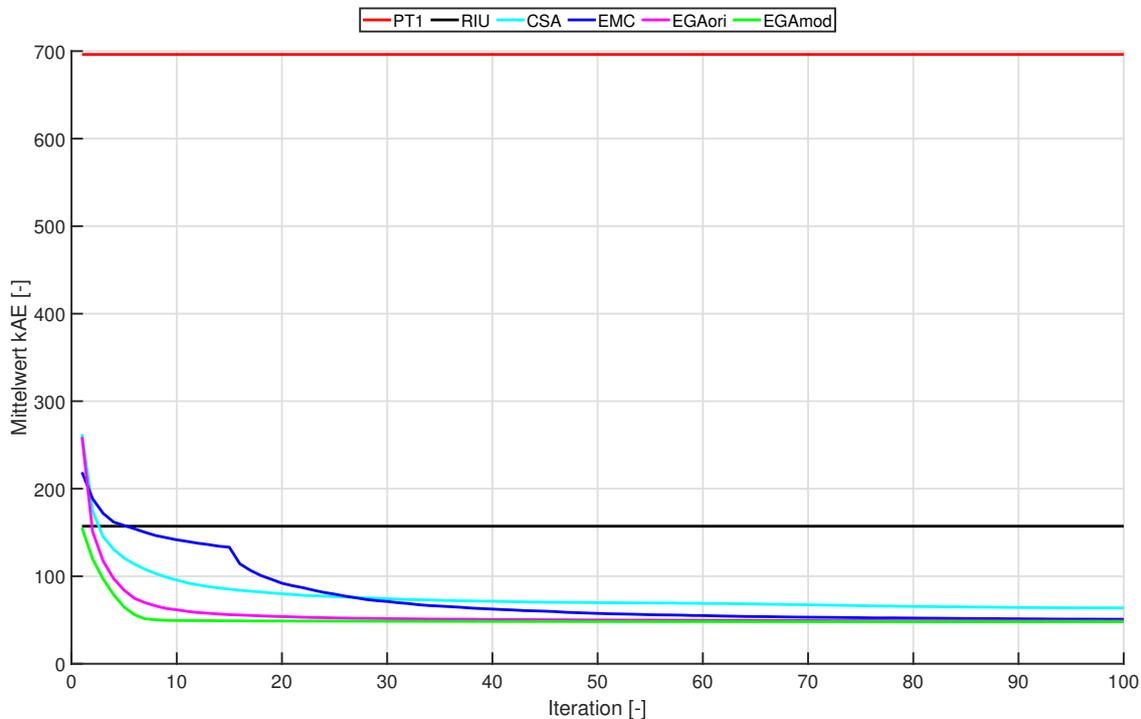


Abbildung 5.11: Histogramm der Verteilung der kAE-Werte bei Datensatz 2. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.6 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

Tabelle 5.6: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 2

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	681.409	715.996	696.214	695.961	696.082	5.650	685.694	707.896
-	RIU	44.710	202.721	157.178	139.638	135.197	31.158	100.139	199.496
1	CSA	70.255	554.253	262.408	252.357	240.203	83.638	115.556	440.870
	EMC	77.446	379.466	218.886	215.277	205.878	57.163	114.552	332.473
	EGAori	73.123	533.275	259.129	252.873	224.068	78.651	122.585	433.678
	EGAmo	44.710	202.721	155.349	139.025	135.260	31.857	96.495	199.430
25	CSA	33.656	178.470	76.745	55.794	51.064	41.360	39.931	159.303
	EMC	35.998	130.638	79.550	79.112	77.937	17.091	50.070	116.335
	EGAori	32.977	86.780	52.356	51.913	53.048	8.465	38.730	71.504
	EGAmo	32.422	77.179	48.646	47.875	44.315	7.343	37.094	65.307
50	CSA	32.742	168.497	69.825	52.374	47.432	39.600	37.920	157.269
	EMC	32.689	109.258	57.543	55.554	53.702	11.528	38.704	82.447
	EGAori	33.354	77.487	50.031	48.949	46.211	7.550	37.830	67.647
	EGAmo	32.425	78.658	48.261	47.284	44.331	7.222	36.384	64.581
75	CSA	33.609	169.386	66.356	51.047	46.156	37.264	37.477	154.845
	EMC	32.556	93.404	52.629	51.585	50.143	9.070	38.329	73.492
	EGAori	32.643	78.176	49.145	47.978	45.252	7.317	37.591	65.226
	EGAmo	32.410	78.303	48.098	47.094	44.224	7.221	36.353	64.167
100	CSA	32.802	169.358	63.767	50.475	45.981	35.546	36.955	152.939
	EMC	32.656	87.572	50.837	50.004	49.083	8.285	37.726	69.937
	EGAori	32.900	78.089	48.783	47.844	45.075	7.279	37.142	64.906
	EGAmo	32.410	78.356	48.008	47.002	44.293	7.186	36.430	64.186
-	Roh	133.844	190.559	160.333	160.332	159.873	8.478	144.855	177.007

**Abbildung 5.12:** Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 2

RIU-Methode in Abbildung 5.13b ist eine Konzentration der internen Knoten bei 0.5 ersichtlich. Die Häufigkeitsverteilung der durch die heuristischen Methoden erstellten internen Knoten (vergleiche Abbildungen 5.13c bis 5.13f) sind sehr ähnlich. Bei 0.5 ist jeweils das absolute Maximum und bei 0.32 und 0.68 sind zwei kleinere und vergleichsweise breite Maxima. Beim CSA (Abbildung 5.13c) werden über den gesamten Wertebereich vereinzelt Knoten gewählt. Dies ist eine mögliche Erklärung für die asymmetrische Verteilung der kAE-Werte. Bei EMC-Methode, EGAori und EGAmo ist dies nicht zu beobachten. Insgesamt sind das absolute Maximum und die beiden kleineren Maxima beim EGAmo (Abbildung 5.13f) am schärfsten ausgeprägt.

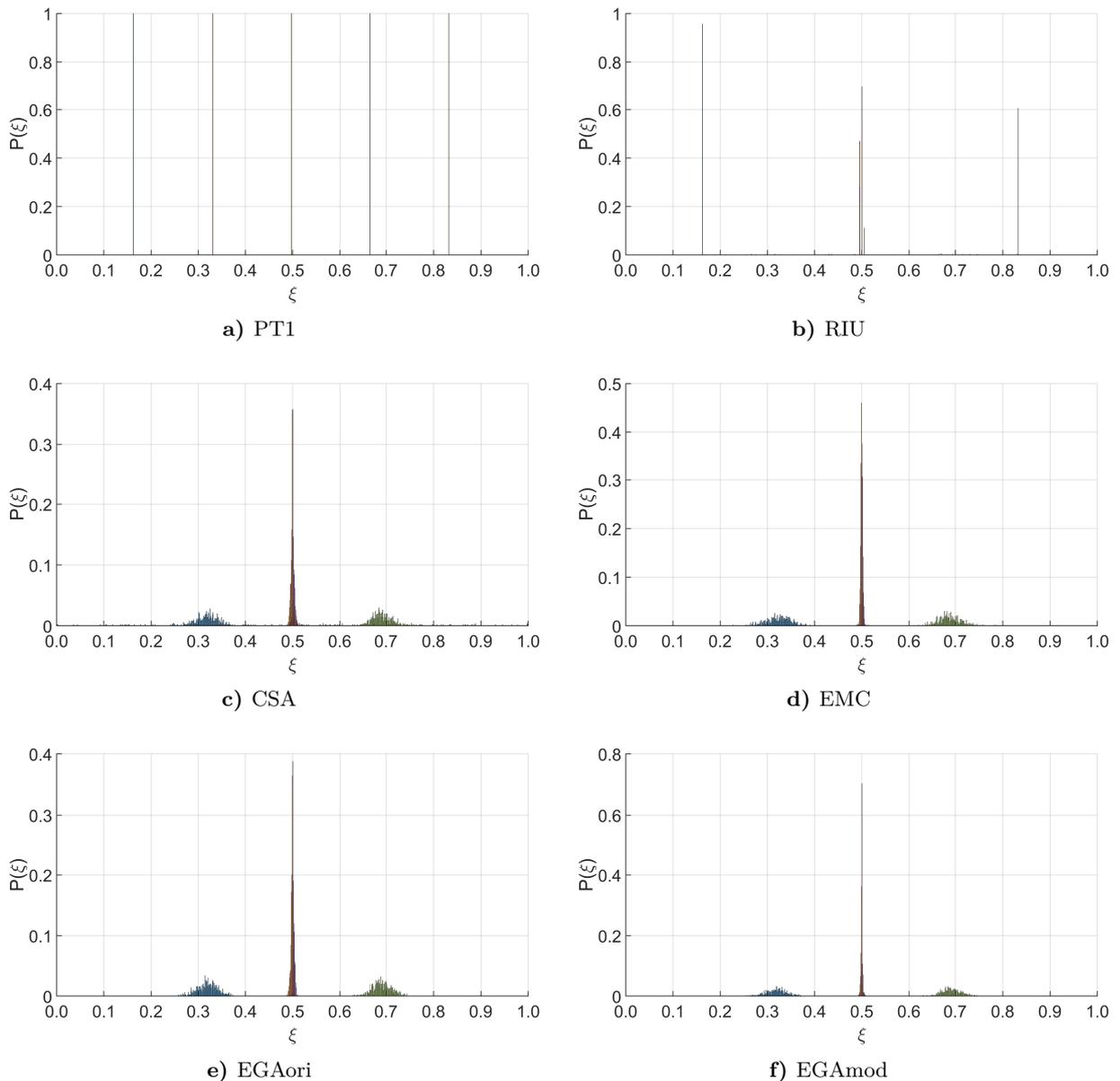


Abbildung 5.13: Histogramm der Verteilung der fünf internen Knoten bei Datensatz 2. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

Datensatz 3 In Tabelle 5.7 sind die Kennziffern des kAE für Datensatz 3 dargestellt. Der RIU erzielt einen deutlich kleineren kAE als PT1. Aus diesem Grund startet der EGAmo gegenüber dem

EGAori und den anderen heuristischen Methoden mit einem deutlich kleineren kAE in Iteration 1. Allerdings ändert sich dies in den späteren Iterationen. Ab Iteration 25 erzielt der EGAmo die höchsten Werte aller heuristischen Methoden bei nahezu allen Kennziffern des kAE. Bei Iteration 100 sind die Unterschiede zwischen den heuristischen Verfahren aber nur noch gering. Eine mögliche Erklärung für dieses Verhalten ist in der Initialisierung des EGAmo durch den RIU zu finden. Der RIU führt zu einer zunächst zügigen Konvergenz in ein lokales Optimum. Erst mit steigender Iterationszahl überwindet der EGAmo dieses lokale Optimum und konvergiert zum globalen Optimum. Bei diesem Datensatz erzielt der CSA den geringsten kAE. Auch das Konfidenzintervall des CSA ist bei diesem Datensatz am kleinsten.

Auch bei diesem Datensatz liegen die kAE-Kennziffern bei Verwendung des PT1 deutlich über denen bei Verwendung der rohen Beobachtungen. Somit kommt es bei Verwendung des PT1 zu einer Verringerung der Richtigkeit. Die kAE-Kennziffern des RIU weisen gegenüber denen der rohen Beobachtungen auf eine deutlich größere Schwankungsbreite hin. Während das Mittel der kAE-Werte beim RIU über dem Mittelwert bei Verwendung der rohen Beobachtungen liegt, liegen Median und Modus beim RIU unter den korrespondierenden Werten der rohen Beobachtungen. Bei den heuristischen Verfahren sind in Iteration 100 wie bei den beiden vorangegangenen Datensätzen gegenüber den rohen Beobachtungen deutlich geringere kAE-Kennziffern festzustellen. Somit kann auch hier die Richtigkeit durch die Verwendung einer heuristischen Methode gesteigert werden.

Die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 3 sind im Anhang

Tabelle 5.7: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 3

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	1473.769	1501.370	1487.939	1487.780	1487.626	5.186	1478.452	1498.480
-	RIU	63.231	677.268	178.282	114.029	98.910	159.835	70.241	588.850
1	CSA	71.370	959.464	593.226	620.605	656.505	161.318	203.118	856.323
	EMC	60.969	826.995	512.856	542.104	581.919	151.226	160.740	772.418
	EGAori	91.312	959.984	597.693	624.634	657.569	158.604	208.127	865.346
	EGAmo	63.231	639.524	172.734	113.774	98.907	150.582	70.241	585.001
25	CSA	18.424	594.423	49.265	43.612	42.215	50.833	28.220	68.389
	EMC	28.669	84.126	51.390	51.049	52.446	9.991	33.559	71.576
	EGAori	21.580	80.874	45.149	44.194	40.396	9.662	28.518	65.534
	EGAmo	28.624	81.017	53.515	53.052	51.983	8.592	38.652	70.610
50	CSA	18.144	77.214	42.442	42.053	41.414	8.189	27.212	59.515
	EMC	19.342	76.928	45.415	44.948	42.223	8.772	31.016	64.877
	EGAori	18.489	80.304	43.683	42.826	39.910	9.229	27.923	63.300
	EGAmo	24.525	80.186	50.205	49.884	48.807	8.943	34.723	68.955
75	CSA	17.946	76.392	41.644	41.192	41.128	7.873	26.946	58.353
	EMC	19.264	73.235	43.985	43.314	40.844	8.480	29.510	61.682
	EGAori	18.782	80.459	43.044	42.117	40.032	9.000	27.452	61.670
	EGAmo	22.399	80.597	47.009	45.881	44.540	8.963	31.558	64.853
100	CSA	17.992	76.603	41.275	40.717	40.542	7.813	27.144	58.642
	EMC	19.874	72.985	43.443	42.534	40.053	8.278	29.510	61.301
	EGAori	18.509	80.319	42.516	41.401	40.219	8.727	27.425	60.127
	EGAmo	19.916	80.203	43.567	42.805	41.863	8.552	29.559	62.439
-	Roh	137.866	184.008	160.892	160.791	160.390	7.920	145.545	178.080

in Abbildung A.5 zu finden. Die heuristischen Methoden erzielen relativ symmetrische, tendenziell leicht rechtsschiefe Verteilungen der kAE-Werte. Wie bei Datensatz 2 sind die graphische Darstellungen der kAE-Werte pro Iteration für Datensatz 3 im Anhang in den Abbildungen A.6, A.7, A.8 und A.9 zu finden.

Abbildung 5.14 zeigt die Entwicklung des mittleren kAE der verschiedenen Methoden über die Iterationen für den Datensatz 3. Hier wird die anhand von Tabelle 5.7 geschilderte Entwicklung des EGAmo mit der zügigen Konvergenz in ein lokales Optimum deutlich. Insgesamt liegen die Mittelwerte der heuristischen Verfahren in Iteration 100 eng zusammen.

Die Häufigkeitsverteilungen der internen Knoten bei den verschiedenen Methoden ist in Abbildung

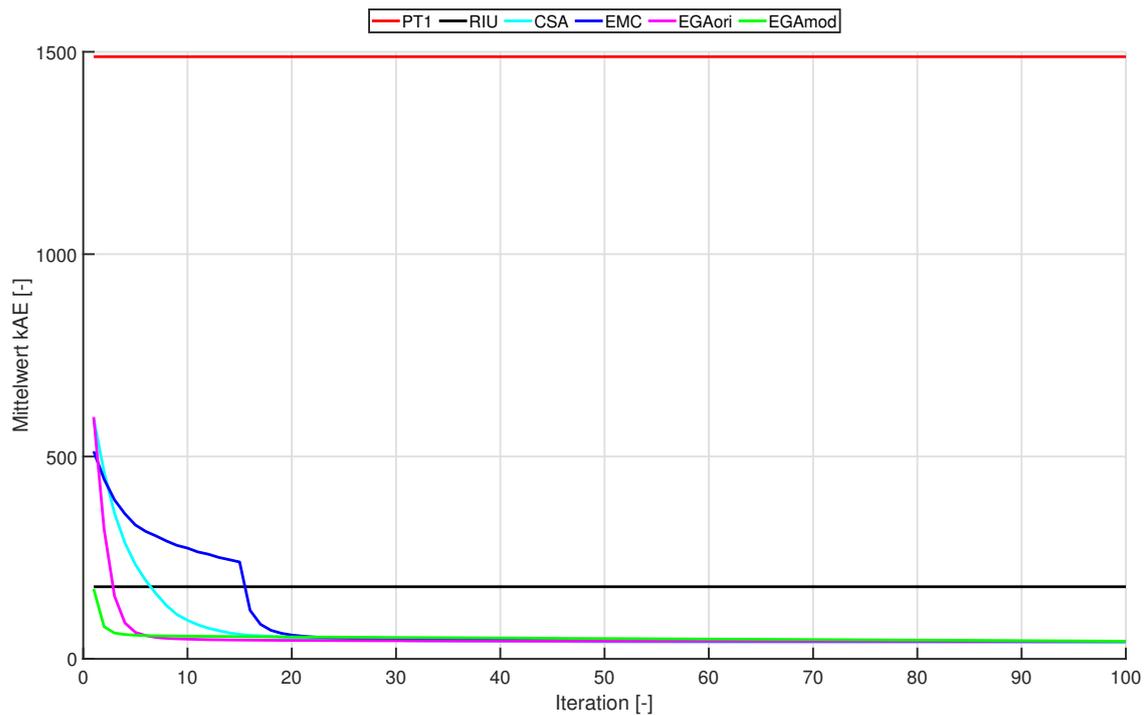


Abbildung 5.14: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 3

5.15 zu sehen. Bei PT1 (Abbildung 5.15a) werden wieder konstant gleichabständige interne Knoten gewählt. Beim RIU in Abbildung 5.15b erfolgt eine Knotenkonzentration in zwei Maxima um 0.4. Bei den heuristischen Verfahren in den Abbildungen 5.15c bis 5.15f sind die internen Knoten in einem Wertebereich zwischen 0.35 und 0.45 konzentriert. Am Rand dieses Wertebereichs sind jeweils zwei hohe Maxima ausgeprägt. In der Mitte des Wertebereichs um 0.4 sind beim CSA und EGAmoD zwei klar ausgeprägte Maxima zu erkennen. Bei der EMC-Methode und dem EGAori verschwimmen diese beiden Maxima deutlich mehr.

Datensatz 4 Tabelle 5.8 enthält die Kennziffern des kAE für Datensatz 4. Die Daten der Datensätze 4 bis 9 sind aus synthetischen B-Spline Kurven generiert. Der Knotenvektor der synthetischen B-Spline Kurve ist bekannt und wird im folgenden als wahrer Knotenvektor bezeichnet. Die vorletzte grün markierte Zeile in Tabelle 5.8 stellt die kAE-Kennziffern dar, die mit dem wahren Knotenvektor erzielt worden sind. Die letzte blau markierte Zeile stellt wiederum die kAE-Kennziffern dar, die mit den rohen Beobachtungen erzielt werden. Im Falle von zweidimensionalen Daten (Datensatz 4-9) folgt deren Verteilung der Rayleigh-Verteilung. RIU erzielt bei diesem Datensatz einen deutlich kleineren kAE-Wert als PT1. Somit startet der EGAmoD im Vergleich zu den anderen heuristischen Verfahren mit einer besseren Initialisierung. Ab Iteration 25 konvergiert der EGAmoD und erzielt am Ende nur geringfügig größere kAE-Werte als mit dem wahren Knotenvektor. Der EGAori erzielt geringfügig größere kAE-Werte als der EGAmoD. Die EMC-Methode erzielt ebenfalls kAE-Werte in einer ähnlichen Größenordnung, wobei das Konfidenzintervall bei der EMC-Methode deutlich größer ist als beim EGAori oder EGAmoD. Der CSA erzielt bei diesem Datensatz deutlich größere kAE-Werte. Im Mittelwert, Median und Modus werden im Vergleich zum EGAmoD um den Faktor 8 erhöhte kAE-Werte erzielt. Der Mittelwert des kAE scheint beim CSA zu einem Wert von 8 zu konvergieren. Dementsprechend weisen die durch den CSA erzielten kAE-Werte ein deutlich breiteres Konfidenzintervall auf. Besonders interessant ist hier, dass Mittelwert, Median und Modus beim CSA in Iteration 100 auf eine leicht linksschiefe Verteilung der kAE-Werte hinweisen. Eine Faustregel besagt, dass bei einer linksschiefen Verteilung in der Regel: $\text{Modus} > \text{Median} > \text{Mittelwert}$

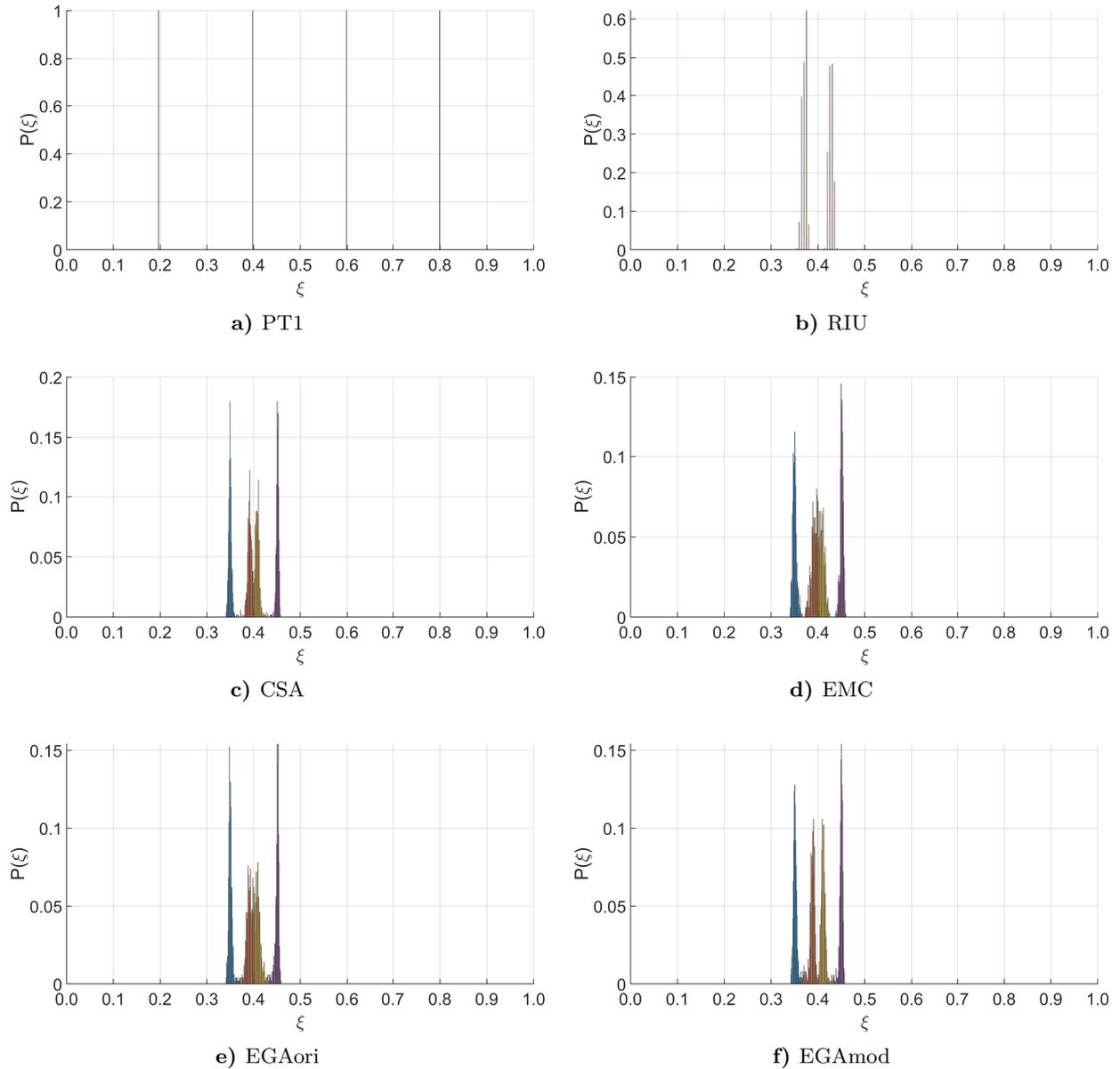


Abbildung 5.15: Histogramm der Verteilung der vier internen Knoten bei Datensatz 3. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

gilt². Die Gültigkeit dieser Faustregel wird in Hippel (2005) diskutiert. Die erhöhten kAE-Werte scheinen beim CSA in diesem Datensatz durch grundsätzliche Schwierigkeiten beim Erreichen der optimalen Approximation begründet zu sein.

Die bei Verwendung des PT1 erzielten kAE-Werte liegen deutlich über den kAE-Werten, die mit den rohen Beobachtungen erzielt werden. Bei allen anderen Methoden unterschreiten die kAE-Kennziffern die korrespondierenden Werte der rohen Beobachtungen zum Teil deutlich. Hier kann also wiederum die Richtigkeit gesteigert werden.

Tabelle 5.8: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 4

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95%)	↑ KI (95%)
-	PT1	54.276	54.647	54.469	54.469	54.470	0.060	54.350	54.597
-	RIU	0.552	5.964	2.283	2.130	1.689	0.954	0.932	4.663
1	CSA	9.750	58.029	38.199	39.100	43.756	10.360	16.896	54.126
	EMC	4.359	59.078	33.399	33.803	36.340	9.113	12.428	49.497
	EGAori	10.127	55.768	38.664	39.414	41.508	9.853	17.277	54.470
	EGAmo	0.552	5.964	2.283	2.130	1.689	0.954	0.932	4.663
25	CSA	1.878	13.912	8.121	8.414	8.951	2.582	2.864	12.599
	EMC	1.274	17.337	7.191	7.013	6.884	2.945	2.276	13.776
	EGAori	0.820	4.808	1.695	1.622	1.588	0.479	0.943	2.773
	EGAmo	0.596	1.823	1.102	1.100	1.100	0.192	0.751	1.506
50	CSA	1.878	13.912	8.062	8.322	8.961	2.578	2.864	12.536
	EMC	0.629	8.890	2.168	1.768	1.414	1.204	0.959	5.333
	EGAori	0.691	2.409	1.367	1.361	1.397	0.253	0.889	1.882
	EGAmo	0.530	1.796	1.098	1.097	1.077	0.191	0.740	1.479
75	CSA	1.878	13.912	8.033	8.273	8.875	2.588	2.840	12.577
	EMC	0.511	7.312	1.673	1.392	1.216	0.886	0.837	4.014
	EGAori	0.689	2.011	1.269	1.268	1.258	0.209	0.851	1.683
	EGAmo	0.524	1.785	1.097	1.102	1.081	0.190	0.748	1.486
100	CSA	1.878	13.912	8.027	8.270	8.873	2.597	2.840	12.577
	EMC	0.528	7.463	1.526	1.310	1.174	0.765	0.787	3.571
	EGAori	0.691	1.914	1.225	1.231	1.245	0.201	0.832	1.614
	EGAmo	0.536	1.796	1.097	1.101	1.081	0.191	0.739	1.479
-	Wahr	0.502	1.713	1.018	1.016	1.006	0.198	0.608	1.405
-	Roh	11.999	13.098	12.536	12.540	12.549	0.203	12.145	12.951

Die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 4 sind, wie bei Datensatz 3, im Anhang in Abbildung A.10 zu finden. Wie bei den vorherigen Datensätzen sind die graphische Darstellungen der kAE-Werte pro Iteration für Datensatz 4 im Anhang in den Abbildungen A.11, A.12, A.13 und A.14 zu finden.

Abbildung 5.16 zeigt die Entwicklung des mittleren kAE der verschiedenen Methoden über die Iterationen. Im Unterschied zu den Abbildungen bei den vorherigen Datensätzen ist zusätzlich der mittlere kAE dargestellt, der mit dem wahren Knotenvektor erzielt wurde. Aufgrund der guten Initialisierung durch den RIU-Algorithmus erreicht der EGAmo bereits ab der 7. Iteration Konvergenz. Der Mittelwert liegt ab dieser Iteration sehr nahe am Mittelwert, der durch den wahren Knotenvektor erzielt wurde. Die EMC-Methode und insbesondere der EGAori nähern sich in späteren Iterationen dem Mittelwert des EGAmo an. Wie bereits oben beschrieben, scheint der CSA ab der 10. Iteration zu einem mittleren kAE-Wert um 8 zu konvergieren.

In Abbildung 5.17 sind die Häufigkeitsverteilungen der internen Knoten bei den verschiedenen Methoden dargestellt. Auch hier sind im Unterschied zu den Abbildungen der vorherigen Datensätze die Positionen der wahren internen Knoten dargestellt. Zwei vertikal übereinander liegende Symbole weisen auf einen multiplen wahren Knoten hin. Die durch PT1 erstellten internen Knoten sind in allen Simulationsdurchläufen äquidistant verteilt und deshalb weit von den wahren Knotenpositionen entfernt (siehe Abbildung 5.17a). Die durch RIU erstellten internen Knoten liegen sehr nahe

²Bei einer rechtsschiefen Verteilung gilt in der Regel: Modus < Median < Mittelwert

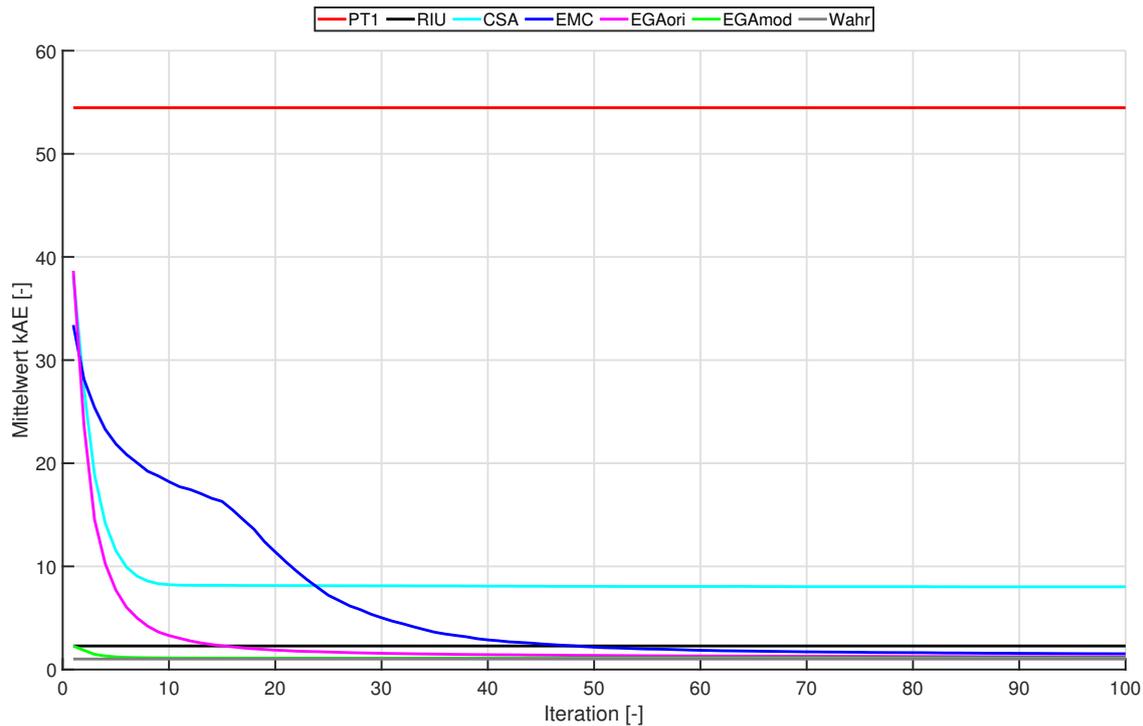


Abbildung 5.16: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4

an den wahren Knotenpositionen (siehe Abbildung 5.17b). Beide doppelten Knoten werden durch die RIU-Methode generiert. Die durch den CSA generierten Knotenpositionen in Abbildung 5.17c liegen in einem breiten Band um die wahren Knotenpositionen herum. Die Breite des Bandes weist nur auf einer geringe Anzahl an generierten doppelte Knoten hin. Bei der EMC-Methode liegen die generierten internen Knoten in einem im Vergleich zum CSA wesentlich schmalen Band um die wahren Knotenpositionen herum (siehe Abbildung 5.17d). Die Verteilung deutet auf einen hohen Prozentsatz an generierten doppelten Knoten hin. Beim EGAori liegen die generierten internen Knoten auf beiden Seiten knapp neben den wahren Knotenpositionen (siehe Abbildung 5.17e). Die Verteilung deutet auf einen geringen Prozentsatz von generierten doppelten Knoten hin. Allerdings werden durch die schmalen Maxima jeweils eng beieinander liegende Knoten erzeugt, die sich ähnlich wie die doppelten Knoten auf die Approximation auswirken. Die Knotenverteilung mit dem EGAmo in Abbildung 5.17f ähnelt sehr der Knotenverteilung des RIU-Algorithmus. Beim EGAmo wird im Vergleich zum RIU ein noch höherer Prozentsatz an Knoten nahe der wahren Knotenpositionen erzeugt. Dies deutet auf einen noch höheren Prozentsatz an generierten doppelten Knoten hin.

Datensatz 5 Datensatz 5 wurde aus derselben synthetischen B-Spline Kurve erzeugt wie Datensatz 4, mit dem Unterschied, dass nahezu alle zwischen den internen Doppelknoten liegenden Datenpunkte entfernt wurden. Dadurch soll die Performance der verschiedenen Methoden bei einer Datenlücke untersucht werden. Die Kennziffern der durch die verschiedenen Methoden erzielten kAE-Werte sind in Tabelle 5.9 dargestellt. RIU erzielt wiederum einen deutlich geringeren kAE als PT1. Die durch RIU erzielten kAE-Werte kommen sogar näher an die kAE-Werte des wahren Knotenvektors heran als die heuristischen Methoden in Iteration 100. Mit Ausnahme des PT1 erzielen alle Methoden einen geringen kAE, der sehr nahe an dem mit dem wahren Knotenvektor erzielten kAE liegt. Der Grund für die gute Approximation bei diesem Datensatz ist in der generierten Datenlücke zu finden. In der Datenlücke kann die approximierende B-Spline-Kurve jegliche Form annehmen. D.h. viele verschiedene Knotenvektoren liefern gute Approximationen. So kann die B-

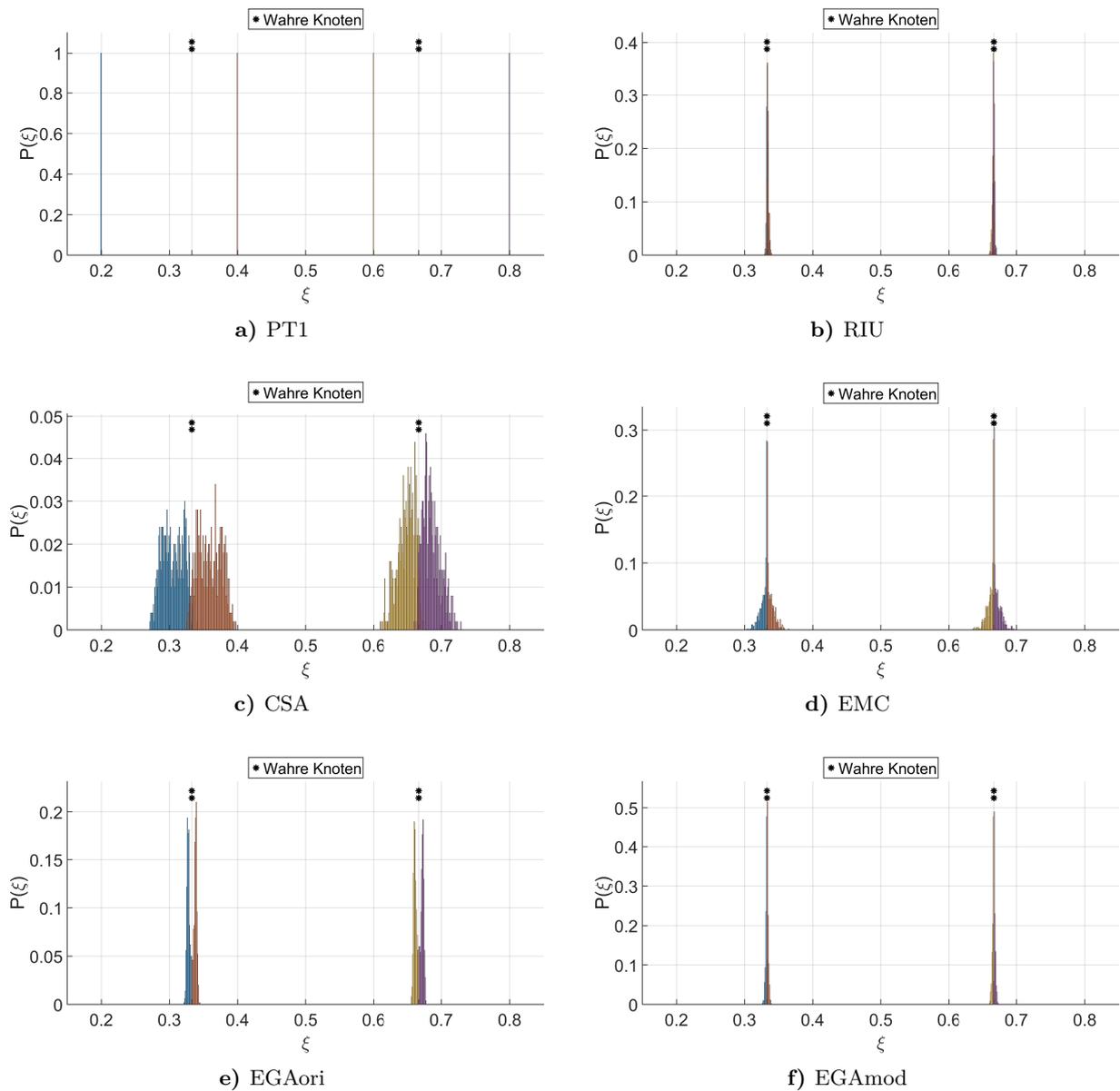


Abbildung 5.17: Histogramm der Verteilung der vier internen Knoten bei Datensatz 4. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt. (Zur besseren Darstellung wurde nicht der gesamte Wertebereich zwischen 0 und 1 dargestellt, sondern nur ein Teilausschnitt. In nicht dargestellten Bereichen ist die relative Häufigkeit $P(\xi) = 0$.)

Spline Kurve im Bereich der Datenlücke sowohl durch zwei doppelte Knoten, so wie in Datensatz 4 geschehen, als auch z.B. durch einen S-förmigen Verlauf modelliert werden.

Im Vergleich zu den kAE-Werten, die mit den rohen Beobachtungen erzielt werden, erzielen mit Ausnahme des PT1 alle Methoden deutlich geringere kAE-Werte. Somit wird bei Verwendung dieser Methoden auch hier die Richtigkeit gesteigert.

Die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 5 sind, wie bei den

Tabelle 5.9: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 5

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	110.041	110.522	110.249	110.245	110.240	0.083	110.086	110.420
-	RIU	0.339	1.356	0.859	0.847	0.798	0.167	0.556	1.192
1	CSA	0.396	1.349	0.888	0.888	0.952	0.164	0.594	1.232
	EMC	0.767	28.452	8.088	7.242	5.596	5.071	1.100	19.504
	EGAori	0.393	1.397	0.891	0.879	0.834	0.166	0.610	1.249
	EGAmo	0.387	1.356	0.886	0.885	0.953	0.166	0.599	1.233
25	CSA	0.364	1.374	0.895	0.886	0.960	0.170	0.601	1.258
	EMC	0.418	1.355	0.874	0.866	0.780	0.171	0.566	1.246
	EGAori	0.368	1.384	0.905	0.909	0.947	0.167	0.613	1.267
	EGAmo	0.412	1.374	0.903	0.914	0.965	0.166	0.614	1.260
50	CSA	0.364	1.374	0.895	0.893	0.963	0.170	0.607	1.254
	EMC	0.417	1.375	0.881	0.874	0.786	0.172	0.566	1.245
	EGAori	0.369	1.384	0.905	0.907	0.948	0.168	0.615	1.270
	EGAmo	0.412	1.375	0.903	0.915	0.968	0.166	0.613	1.255
75	CSA	0.364	1.374	0.891	0.891	0.951	0.170	0.592	1.254
	EMC	0.417	1.378	0.881	0.874	0.787	0.172	0.576	1.241
	EGAori	0.369	1.383	0.905	0.910	0.949	0.168	0.618	1.267
	EGAmo	0.413	1.375	0.902	0.912	0.971	0.167	0.607	1.251
100	CSA	0.391	1.374	0.892	0.891	0.952	0.169	0.592	1.254
	EMC	0.417	1.376	0.882	0.877	0.803	0.172	0.574	1.241
	EGAori	0.369	1.383	0.905	0.910	0.950	0.168	0.618	1.265
	EGAmo	0.412	1.375	0.903	0.912	0.972	0.167	0.607	1.248
-	Wahr	0.325	1.282	0.790	0.775	0.704	0.162	0.502	1.115
-	Roh	7.890	8.963	8.404	8.412	8.439	0.176	8.062	8.710

Datensätzen zuvor, im Anhang in Abbildung A.15 zu finden. Alle Methoden erzeugen relativ symmetrisch verteilte kAE-Werte. Auch die graphische Darstellungen der kAE-Werte pro Iteration für Datensatz 5 sind im Anhang in den Abbildungen A.16, A.17, A.18 und A.19 zu finden.

Der Vergleich der kAE-Mittelwerte in Abbildung 5.18 zeigt, dass RIU, CSA, EGAori und EGAmo sehr nahe am Ergebnis des wahren Knotenvektors liegen. Die EMC-Methode konvergiert in Iteration 16 ebenfalls zu diesem Mittelwert. Die Häufigkeitsverteilung der generierten internen Knoten in Abbildung 5.19 weicht bei diesem Datensatz grundsätzlich ab. Durch die Datenlücke erzeugt der PT1 keine gleichabständigen Knoten (siehe Abbildung 5.19a). Die Knotenspanne, in der die Datenlücke liegt, wird beim PT1 vergrößert. RIU erzeugt zwei Maxima an den wahren Knotenpositionen (siehe Abbildung 5.19b). Zusätzlich werden zwei weitere Maxima bei 0.13 und 0.87 erzeugt. Bei den heuristischen Verfahren (Abbildung 5.19c-5.19f) sind die Knoten jeweils über die komplette Spanne verteilt. Bei den wahren Knotenpositionen ergeben sich bei allen heuristischen Methoden mehr oder weniger ausgeprägte Maxima.

Datensatz 6 In Tabelle 5.10 sind die Kennziffern des kAE der verschiedenen Methoden für Datensatz 6 dargestellt. PT1 und RIU erzielen bei diesem Datensatz identische Ergebnisse. Dadurch ist die Initialisierung von EGAori und EGAmo bei diesem Datensatz identisch. Von den heuristischen Verfahren erzielt der EGAmo final die geringsten kAE-Werte. Jedoch weichen insbesondere Maximum, Mittelwert und obere Konfidenzintervallgrenze beim EGAmo von den durch den wahren Knotenvektor erzielten kAE-Werten ab. Der EGAori erzielt kAE-Werte in ähnlicher Größenordnung wie der EGAmo. CSA und insbesondere die EMC-Methode erzielen bei diesem Datensatz

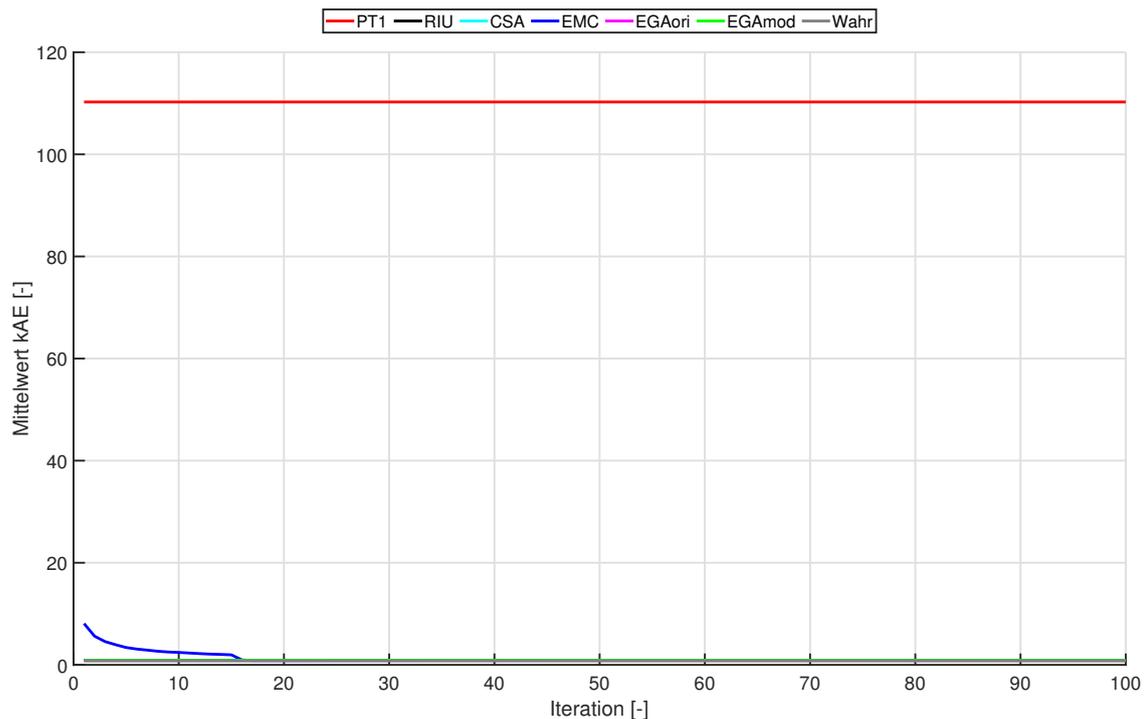


Abbildung 5.18: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 5

in der finalen Iteration deutlich erhöhte kAE-Werte.

Im Vergleich mit den kAE-Werten der rohen Beobachtungen ergibt sich bei diesem Datensatz ein anderes Bild. Sowohl die beiden deterministischen Verfahren PT1 und RIU als auch die beiden heuristischen Verfahren CSA und EMC führen zu zum Teil deutlich größeren kAE-Kennziffern als die rohen Beobachtungen. Das heißt, es kommt bei diesen Methoden und diesem Datensatz zu einer Verringerung der Richtigkeit. Die kAE-Kennziffern des EGAori und EGAmo liegen in Mittel, Median und Modus deutlich unter den korrespondierenden kAE-Kennziffern der rohen Beobachtungen. Der maximale kAE-Wert überschreitet auch bei EGAori und EGAmo den maximalen kAE-Wert, der mit den rohen Beobachtungen erzielt wird. Es kommt also hier nicht bei allen Simulationsdurchläufen zu einer Steigerung der Richtigkeit.

In Abbildung 5.20 sind die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 6 dargestellt. Wie oben bereits erwähnt, erzielen PT1 (Abbildung 5.20a) und RIU (Abbildung 5.20b) bei diesem Datensatz identische kAE-Werte. Die kAE-Werte sind symmetrisch verteilt. Beim CSA (Abbildung 5.20c) liegen in ca. 15 %-20 % der Simulationsdurchgänge die kAE-Werte signifikant über den restlichen kAE-Werten. Die kAE-Verteilung bei der EMC-Methode (Abbildung 5.20d) ist relativ symmetrisch. Bei der kAE-Verteilung der beiden EGA (Abbildung 5.20e und 5.20f) weichen ca. 5 % der Simulationsdurchläufe signifikant von den restlichen Durchläufen ab. In diesen Durchläufen wird ein erhöhter kAE-Wert von ca. 65 erzielt. In diesen Fällen kann festgestellt werden, dass EGAori und EGAmo in ein lokales Minimum konvergiert sind.

Die graphische Darstellungen der kAE-Werte pro Iteration sind für diesen Datensatz im Anhang in den Abbildungen A.20, A.21, A.22 und A.23 zu finden.

In Abbildung 5.21 ist die Entwicklung des mittleren kAE bei den verschiedenen Epochen dargestellt. EGAori und EGAmo erzielen deutlich geringere kAE-Werte als die anderen Verfahren, jedoch erreichen sie nicht den durch den wahren Knotenvektor erzielten mittleren kAE-Wert. Zwischen Iteration 5 und 20 ist zu erkennen, dass der EGAmo bei diesem Datensatz geringfügig schneller konvergiert. Bei identischer Initialisierung ist dies auf die Verwendung des modifizierten Mutations-Mechanismus zurückzuführen. Die Verteilung der generierten internen Knoten ist in Abbildung 5.22

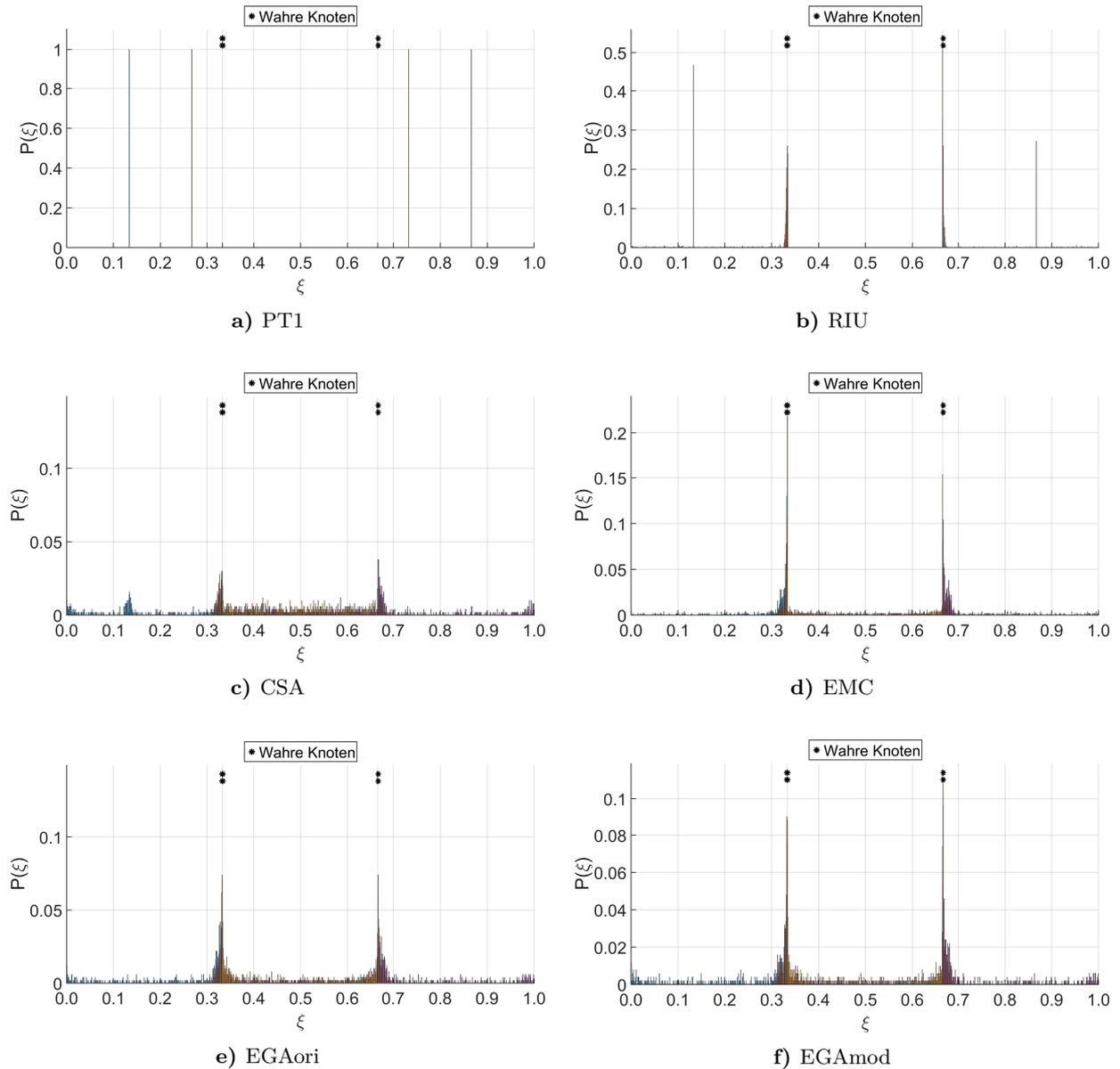


Abbildung 5.19: Histogramm der Verteilung der vier internen Knoten bei Datensatz 5. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

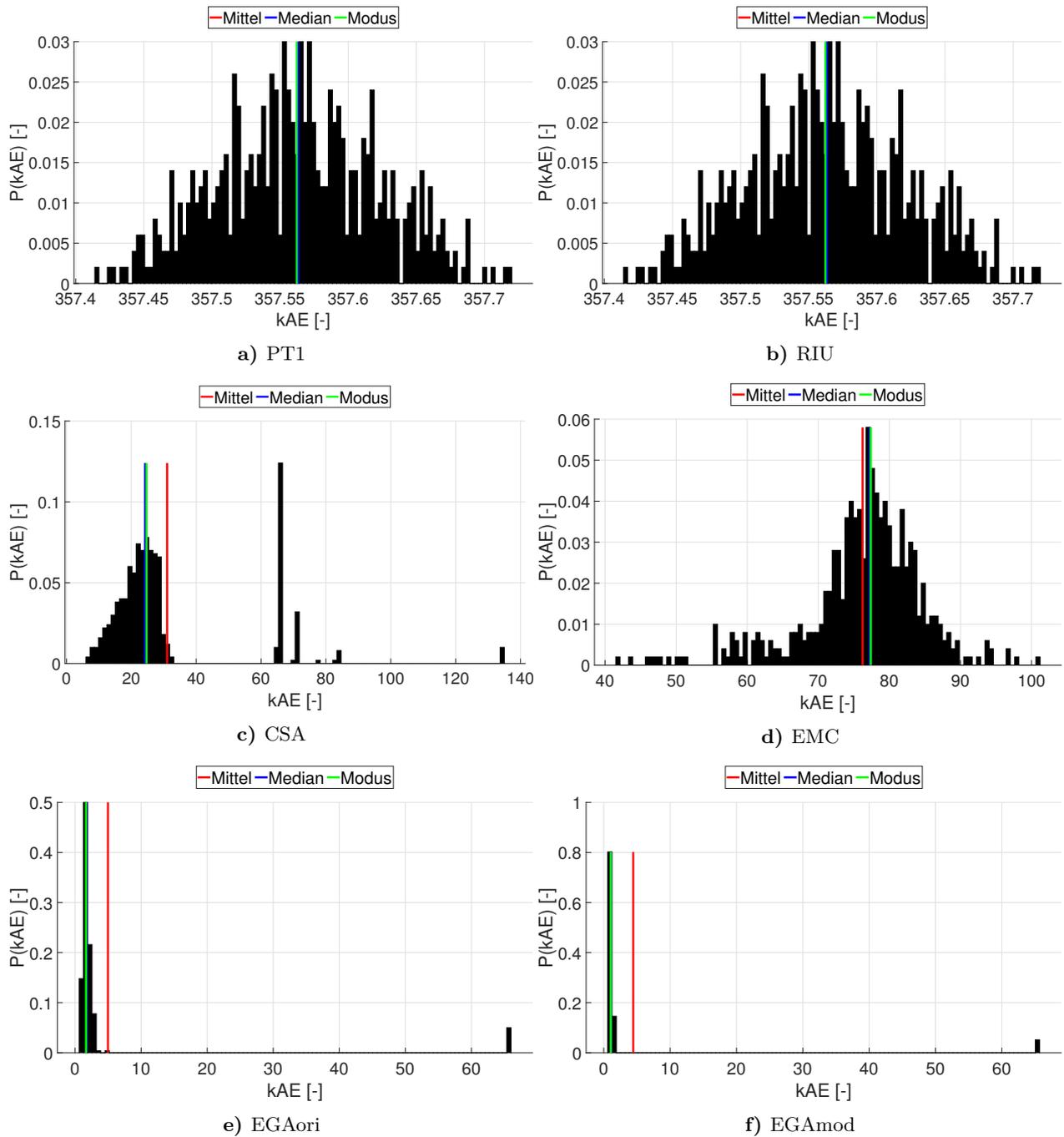
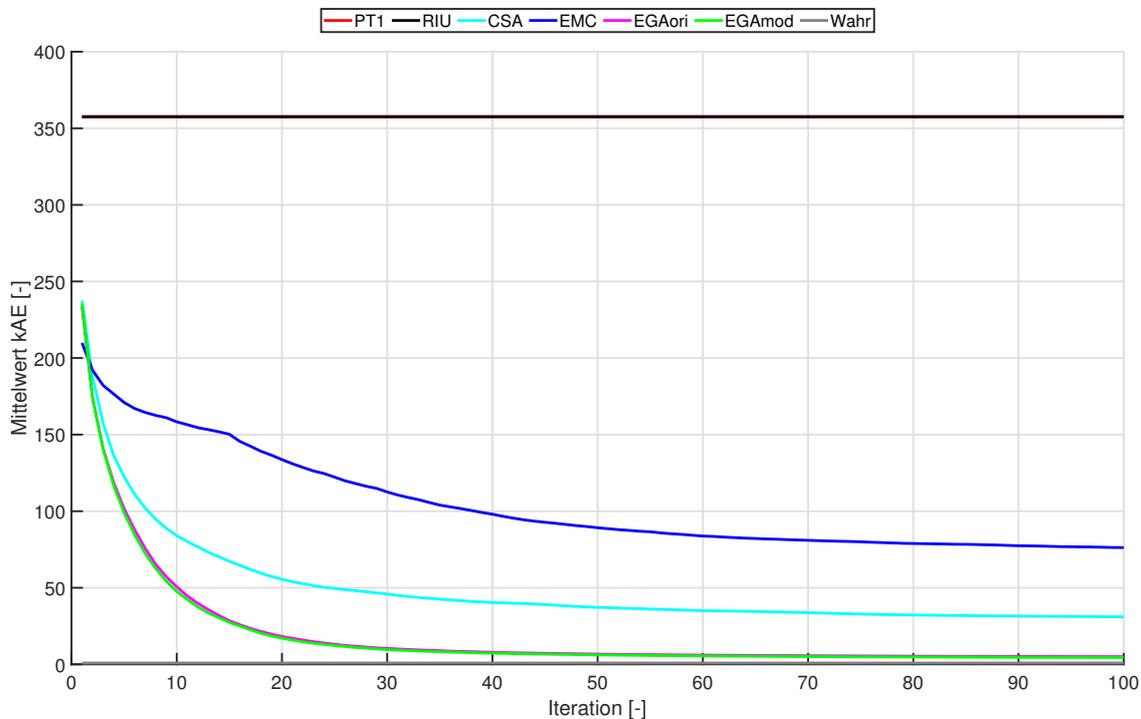


Abbildung 5.20: Histogramm der Verteilung der kAE-Werte bei Datensatz 6. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.10 angegebenen Wert. Dieser über Kern-dichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

Tabelle 5.10: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 6

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	357.417	357.720	357.563	357.563	357.562	0.059	357.450	357.673
-	RIU	357.417	357.720	357.563	357.563	357.562	0.059	357.450	357.673
1	CSA	120.465	392.770	237.546	233.537	217.878	44.618	149.781	336.409
	EMC	85.437	322.585	209.891	211.331	213.201	36.489	129.199	277.593
	EGAori	100.226	358.214	233.595	231.132	230.239	43.230	154.639	330.172
25	EGAmo	114.389	380.181	235.554	233.026	229.435	44.009	154.685	331.391
	CSA	12.881	139.754	49.553	30.817	24.923	29.347	15.946	116.088
	EMC	56.905	164.810	122.238	122.359	117.090	16.586	88.379	152.162
50	EGAori	1.752	71.914	13.044	9.136	7.492	13.816	3.410	67.080
	EGAmo	1.665	68.877	12.308	8.512	7.197	13.697	2.899	66.781
	CSA	6.397	137.928	37.246	25.649	24.682	26.346	11.627	84.744
75	EMC	55.250	120.105	89.186	89.340	89.743	10.264	65.071	109.410
	EGAori	1.102	67.056	6.674	3.393	2.843	13.664	1.595	65.717
	EGAmo	1.093	66.120	6.167	2.756	2.290	13.992	1.386	65.642
100	CSA	6.397	136.526	32.831	24.728	24.630	23.019	10.058	83.938
	EMC	45.866	101.117	80.015	80.039	78.860	8.477	58.485	95.615
	EGAori	0.779	66.149	5.460	2.182	2.034	13.851	1.139	65.645
100	EGAmo	0.871	65.715	4.907	1.552	1.468	14.206	0.996	65.522
	CSA	6.397	134.358	31.040	24.206	24.685	21.194	10.035	71.068
	EMC	41.853	101.117	76.230	77.248	77.391	8.393	55.545	89.717
100	EGAori	0.692	65.973	4.970	1.731	1.658	13.934	1.042	65.617
	EGAmo	0.707	65.618	4.438	1.089	1.064	14.301	0.814	65.443
	-	Wahr	0.513	1.332	0.861	0.858	0.860	0.145	0.591
-	Roh	12.132	13.026	12.525	12.515	12.492	0.143	12.217	12.821

**Abbildung 5.21:** Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 6

dargestellt. PT1 und RIU generieren bei diesem Datensatz identische äquidistante Knotenvektoren (siehe Abbildung 5.22a und 5.22b). Beim CSA (Abbildung 5.22c) und der EMC-Methode (Abbildung 5.22d) liegen die Knoten um die wahren Knotenpositionen verteilt. Allerdings weisen sie jeweils eine größere Schwankungsbreite auf. Beim CSA ist diese Schwankungsbreite geringer als bei der EMC-Methode. Bei der EMC-Methode sind bei 0.55 und 0.60 keine Maxima zu finden, obwohl dort wahre interne Knoten positioniert sind. EGAori (Abbildung 5.22e) und EGAmo (Abbildung 5.22f) generieren schmale und stark ausgeprägte Maxima an den Positionen der wahren Knoten. An der Stelle 0.66 wird ebenfalls ein geringer Prozentsatz interner Knoten positioniert, obwohl dort kein wahrer interner Knoten liegt. Diese internen Knoten könnten aus den Simulationsdurchläufen mit hohem kAE-Wert stammen und ein lokales Minimum repräsentieren.

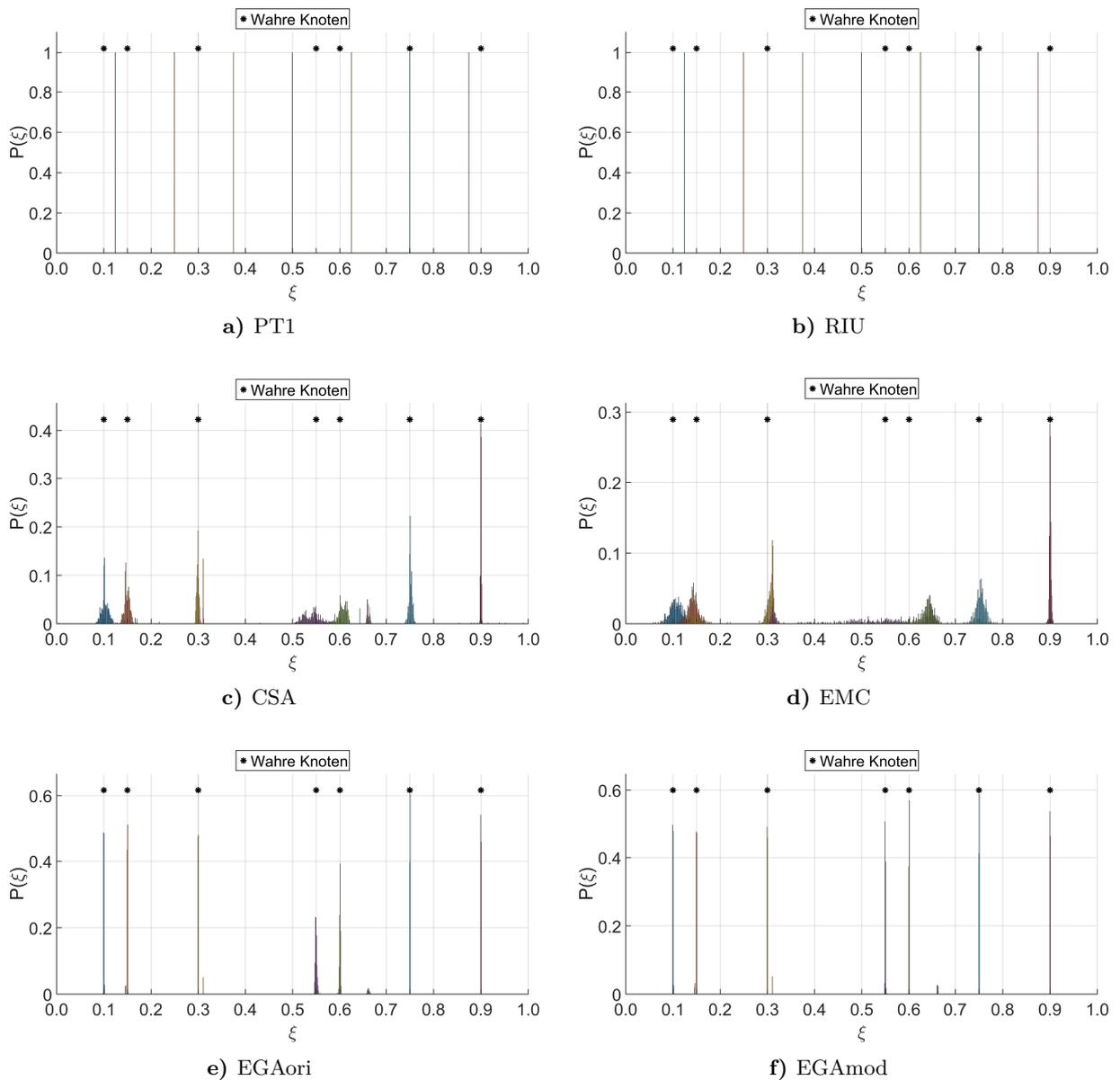


Abbildung 5.22: Histogramm der Verteilung der sieben internen Knoten bei Datensatz 6. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

Datensatz 7 und 8 Die Ergebnisse für die Datensätze 7 und 8 sind im Anhang zu finden. Generell ähneln insbesondere die Ergebnisse für Datensatz 7 denen, die bei Datensatz 4 erzielt wurden. Für Datensatz 7 wird auf Tabelle A.7 und die Abbildungen A.24 bis A.30 verwiesen. Für Datensatz 8 wird auf Tabelle A.8 und die Abbildungen A.31 bis A.37 verwiesen.

Datensatz 9 Tabelle 5.11 zeigt die Kennziffern des kAE für Datensatz 9. PT1 und RIU erzielen bei diesem Datensatz nahezu identische und insgesamt sehr geringe kAE-Werte. Dadurch starten der EGAori und der EGAmo ebenfalls mit geringeren kAE-Werten in Iteration 1. Während der EGAori und EGAmo spätestens bei Iteration 50 Konvergenz erreichen und nahe an den Ergebnissen des wahren Knotenvektors liegen, weichen die durch den CSA und die EMC-Methode erzielten kAE-Werte auch bei der letzten Iteration 100 deutlich von den Ergebnissen des wahren Knotenvektors ab. Die kAE-Werte des CSA und der EMC-Methode übersteigen sogar die durch PT1 erzielten kAE-Werte.

Bei diesem Datensatz führen grundsätzlich alle Methoden zu einer Steigerung der Richtigkeit. Lediglich die kAE-Kennziffern des CSA liegen zum Teil in der Größenordnung, die mit den rohen Beobachtungen erzielt wurden.

Die Histogramme der kAE-Werte der verschiedenen Methoden für Datensatz 9 sind, wie bei den

Tabelle 5.11: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 9

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	3.642	4.305	3.900	3.888	3.878	0.114	3.700	4.153
-	RIU	1.330	4.305	3.859	3.884	3.878	0.318	3.647	4.153
1	CSA	5.105	147.083	49.122	44.975	35.250	23.816	15.530	102.382
	EMC	8.216	101.756	35.764	33.808	25.859	15.321	13.911	73.545
	EGAori	3.642	4.305	3.900	3.888	3.878	0.114	3.700	4.153
	EGAmo	1.330	4.305	3.859	3.884	3.878	0.318	3.647	4.153
25	CSA	1.969	16.141	11.638	12.213	12.509	2.640	5.048	15.740
	EMC	1.262	16.693	8.762	8.571	7.406	3.533	2.649	14.782
	EGAori	0.605	2.308	1.217	1.195	1.161	0.260	0.734	1.749
	EGAmo	0.524	2.276	1.202	1.194	1.181	0.247	0.769	1.734
50	CSA	1.969	16.149	11.452	12.063	12.437	2.645	4.851	15.735
	EMC	0.741	16.693	4.890	3.739	2.240	3.387	1.154	13.362
	EGAori	0.605	1.879	1.122	1.107	1.086	0.224	0.710	1.600
	EGAmo	0.480	1.790	1.108	1.098	1.102	0.216	0.731	1.552
75	CSA	1.969	16.149	11.305	11.941	12.452	2.701	4.604	15.720
	EMC	0.680	14.483	4.373	3.272	1.882	3.184	1.005	11.930
	EGAori	0.602	1.818	1.100	1.092	1.091	0.208	0.718	1.557
	EGAmo	0.480	1.740	1.085	1.069	1.039	0.212	0.693	1.560
100	CSA	1.969	16.149	11.127	11.841	12.446	2.785	4.325	15.694
	EMC	0.658	14.483	4.164	3.137	1.817	3.039	0.979	11.579
	EGAori	0.546	1.821	1.090	1.085	1.088	0.208	0.707	1.557
	EGAmo	0.479	1.719	1.075	1.068	1.056	0.209	0.695	1.544
-	Wahr	0.391	1.704	0.954	0.947	0.929	0.216	0.564	1.393
-	Roh	11.905	13.240	12.556	12.567	12.552	0.216	12.125	12.955

Datensätzen zuvor, im Anhang in Abbildung A.38 zu finden. Auch die graphischen Darstellungen der kAE-Werte pro Iteration sind für Datensatz 9 im Anhang in den Abbildungen A.39, A.40, A.41 und A.42 zu finden.

Abbildung 5.23 verdeutlicht das bereits oben beschriebene Verhalten der Methoden bei diesem Datensatz. Durch die gute Initialisierung des EGAori und des EGAmo konvergieren beide Algorithmen ab Iteration 25 zum Ergebnis des wahren Knotenvektors. Die EMC-Methode erreicht in der letzten Iteration einen mittleren kAE in der Größenordnung des durch PT1 erzielten kAE. Der durch den CSA erzielte mittlere kAE liegt dagegen deutlich über dem kAE der EMC-Methode. Bei diesem Datensatz scheint der CSA in ein lokales Optimum konvergiert zu sein. In Abbildung

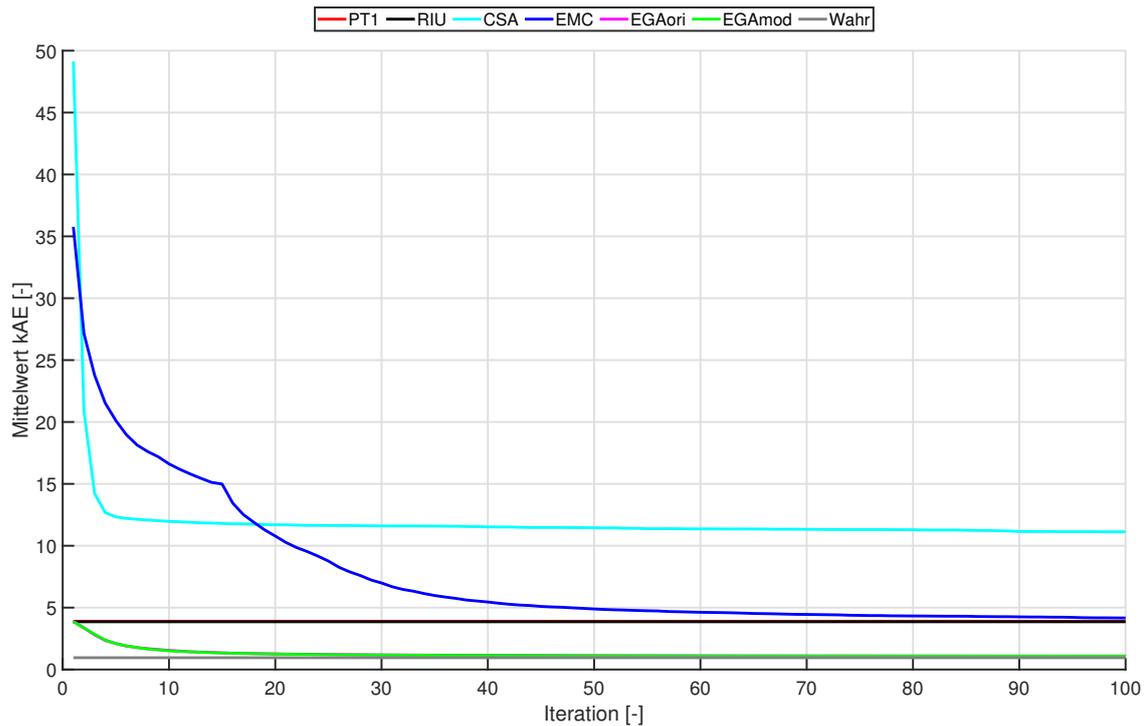


Abbildung 5.23: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 9

5.24 sind die Häufigkeitsverteilungen der internen Knoten dargestellt. In Abbildung 5.24a wird der Grund für die gute Approximation dieses Datensatzes durch den PT1 ersichtlich. Zufällig liegen die durch den PT1 erstellten Knoten sehr nahe an den wahren Knotenpositionen. Beim RIU in Abbildung 5.24b zeigt sich ein sehr ähnliches Bild. Beim CSA in Abbildung 5.24c werden bei 0.25 ebenfalls häufig interne Knoten platziert. Hier ist ein scharf ausgeprägtes Maximum zu erkennen. Bei 0.5 ist ebenfalls ein Maximum zu erkennen. Dieses ist aber deutlich weniger ausgeprägt und wesentlich breiter. An der wahren Knotenposition 0.75 ist beim CSA kein Maximum ausgeprägt. Bei der EMC-Methode sind ebenfalls zwei Maxima bei 0.25 und 0.50 ausgeprägt. Im Vergleich zum CSA sind beide Maxima deutlich schärfer ausgeprägt, wobei das Maximum bei 0.50 ebenfalls breiter ist als bei 0.25. An der wahren Knotenposition 0.75 ist bei der EMC-Methode allenfalls ein Maximum zu erahnen. Beim EGAori (Abbildung 5.24e) und EGAmo (Abbildung 5.24f) sind die beiden Maxima bei 0.25 und 0.50 scharf ausgeprägt. Das Maximum an der wahren Knotenposition 0.75 ist deutlich erkennbar, aber nicht ganz so scharf ausgeprägt wie die anderen beiden Maxima.

5.3.2 Zusammenfassung

Die in Kapitel 5.3.1 dargestellten Ergebnisse lassen folgende Schlussfolgerungen zu:

- Von den sechs untersuchten Methoden zur Knotenvektorwahl erzielt der EGAmo bei den analysierten (normalverteilten) Datensätzen in den 500 Wiederholungen grundsätzlich den höchsten Prozentsatz bestandener Globaltests. In der gezeigten Simulation wird ein verworfener Globaltest nur durch einen nicht optimalen Knotenvektor verursacht.
- Ebenfalls erzeugt der EGAmo insgesamt die Knotenvektoren, die nach der Kontrollpunkt-schätzung zu den geringsten kAE-Werten führen. Der verwendete kAE kann in der Untersuchung als Maß für die Richtigkeit der Approximation angesehen werden.
- Die angegebenen kAE-Werte der rohen Beobachtungen dienen der Einordnung der mit den verschiedenen Methoden erzielten kAE-Kennziffern. Durch eine adäquate B-Spline Approxi-

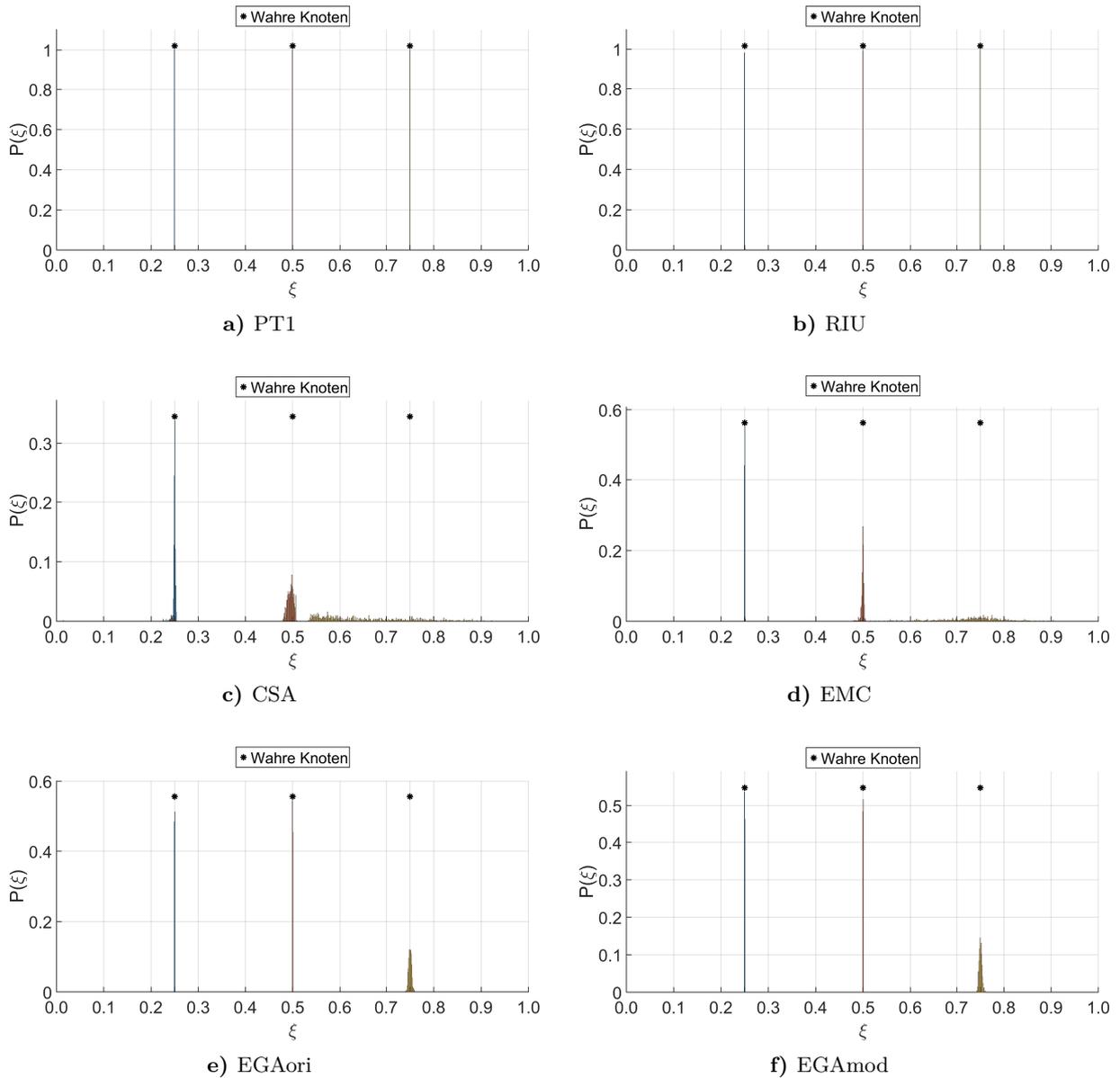


Abbildung 5.24: Histogramm der Verteilung der drei internen Knoten bei Datensatz 9. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

mation sollten die kAE-Kennziffern der rohen Beobachtungen mindestens unterboten werden. Dies ist bei den untersuchten Datensätzen durchgängig nur bei Verwendung von EGAori und EGAmo der Fall.

- Die Unterschiede zwischen den Methoden zur Knotenvektorschätzung variieren je nach untersuchtem Datensatz. RIU und EGAmo führen insbesondere bei Datensätzen mit diskontinuierlichem Kurvenverlauf zu vergleichsweise kleinen kAE-Werten. Da der EGAmo den RIU-Algorithmus zur Initialisierung verwendet, ist dies mit der Fähigkeit des RIU-Algorithmus zu erklären, doppelte bzw. multiple Knoten zu erzeugen.
- Größere Datenlücken wie bei Datensatz 5 erlauben Variationen des Knotenvektors bei vergleichbaren kAE-Werten. Heuristische Verfahren konvergieren dann generell nach wenigen Iterationen zu kleinen kAE-Werten. Mit Ausnahme des PT1 wird bei diesem Datensatz von allen Methoden ein hoher Prozentsatz bestandener Globaltests erzielt.
- EGAori und EGAmo erzielen, mit Ausnahme des Datensatzes 6, vergleichsweise symmetrisch verteilte kAE-Werte. Dies kann als Indikator für die Präzision angesehen werden. Nur bei Datensatz 6 ist eine signifikante Anzahl an Simulationsdurchläufen (ca. 5 %) festzustellen, bei denen EGAori und EGAmo in ein lokales Optimum konvergiert sind. Ein offensichtlicher Grund, warum dies bei Datensatz 6 passiert, ist nicht ersichtlich.
- Die Variation der durch den EGAmo erzeugten internen Knoten ist bei den untersuchten Datensätzen insgesamt im Vergleich zu den anderen heuristischen Verfahren geringer. Dies kann ebenfalls als Indikator für die Präzision angesehen werden. Zusätzlich korrespondieren die erstellten internen Knoten mit den Positionen der wahren Knoten in den Datensätzen 4 bis 9.

5.4 Sensitivitätsanalyse der Kontrollpunktschätzung

In diesem Kapitel soll der Einfluss der verschiedenen Knotenvektormethoden auf die Kontrollpunktschätzung untersucht werden. Exemplarisch wird hier Datensatz 9 untersucht. Bei diesem Datensatz erzielen die sechs Knotenvektormethoden die ähnlichsten Ergebnisse. Dies lässt eine einfache und besser interpretierbare Darstellung der Ergebnisse zu.

Im folgenden Kapitel 5.4.1 werden unter anderem zwei verschiedene Konfidenzellipsen dargestellt. Beide Konfidenzellipsen werden aus einer Varianz-Kovarianz-Matrix:

$$\Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \quad (5.16)$$

berechnet. Im ersten Fall wird aus den 500 Realisierungen für jeden Kontrollpunkt \mathbf{x}_j über Formel 5.17 eine empirische Varianz-Kovarianz-Matrix $\Sigma_{\text{emp},j}$ berechnet (siehe z.B. Niemeier (2008)):

$$\Sigma_{\text{emp},j} = \begin{bmatrix} \frac{1}{500-1} \sum_{i=1}^{500} (x_{x,j,i} - \bar{x}_{x,j})^2 & \frac{1}{500-1} \sum_{i=1}^{500} (x_{x,j,i} - \bar{x}_{x,j}) \cdot (x_{y,j,i} - \bar{x}_{y,j}) \\ \frac{1}{500-1} \sum_{i=1}^{500} (x_{y,j,i} - \bar{x}_{y,j}) \cdot (x_{x,j,i} - \bar{x}_{x,j}) & \frac{1}{500-1} \sum_{i=1}^{500} (x_{y,j,i} - \bar{x}_{y,j})^2 \end{bmatrix}, \quad (5.17)$$

wobei $\bar{\mathbf{x}}_j = [\bar{x}_{x,j}, \bar{x}_{y,j}]$ der Mittelwert der 500 Realisierungen des Kontrollpunktes \mathbf{x}_j ist.

Im zweiten Fall werden die Varianz-Kovarianz-Matrizen aus dem GMM der Kontrollpunktschätzung betrachtet. Diese werden über die 500 Simulationsdurchläufe gemittelt und im Folgenden mit $\Sigma_{\text{GMM},j}$ bezeichnet.

Nach Niemeier (2008) werden die große und kleine Halbachse A_F und B_F , sowie der Richtungswinkel Θ_F der Helmertschen Fehlerellipse aus einer Varianz-Kovarianz-Matrix wie in Formel 5.16 über:

$$A_F^2 = \frac{1}{2}(\Sigma_{xx} + \Sigma_{yy} + \omega), \quad (5.18)$$

$$B_F^2 = \frac{1}{2}(\Sigma_{xx} + \Sigma_{yy} - \omega), \quad (5.19)$$

$$\tan 2\Theta_F = 2 \frac{\Sigma_{xy}}{\Sigma_{xx} - \Sigma_{yy}} \quad (5.20)$$

berechnet. Die Hilfsgröße ω wird über:

$$\omega^2 = (\Sigma_{xx} - \Sigma_{yy})^2 + 4\Sigma_{xy}^2 \quad (5.21)$$

berechnet. Die Umrechnung der Parameter der Helmertschen Fehlerellipse A_F , B_F und Θ_F in die Parameter einer Konfidenzellipse A_K , B_K und Θ_K mit einer Irrtumswahrscheinlichkeit α erfolgt über die Formeln:

$$A_K = \sqrt{2 \cdot F_{2,f,1-\alpha}} \cdot A_F, \quad (5.22)$$

$$B_K = \sqrt{2 \cdot F_{2,f,1-\alpha}} \cdot B_F, \quad (5.23)$$

$$\Theta_K = \Theta_F. \quad (5.24)$$

Als Ursprung der Konfidenzellipsen wird der jeweilige Mittelpunkt $\bar{\mathbf{x}}_j$ verwendet. Da die „wahre“ Lage des Kontrollpunktes $\tilde{\mathbf{x}}_j$ aus der Simulation bekannt ist, kann der Abstand Δs zwischen $\bar{\mathbf{x}}_j$ und $\tilde{\mathbf{x}}_j$ als weiteres Maß für die Richtigkeit herangezogen werden. Δs wird über:

$$\Delta s = \sqrt{(\bar{x}_{x,j} - \tilde{x}_{x,j})^2 + (\bar{x}_{y,j} - \tilde{x}_{y,j})^2} \quad (5.25)$$

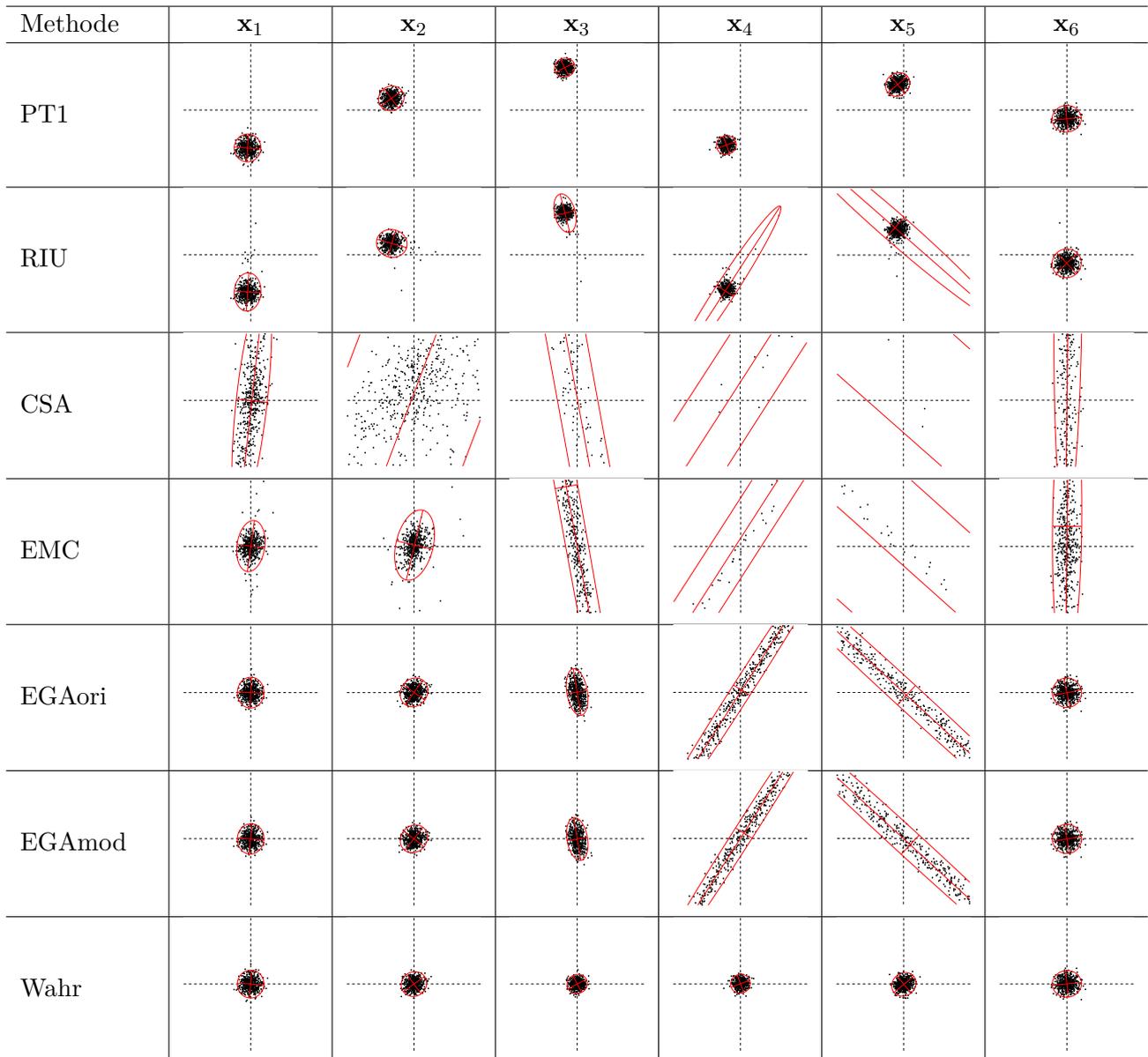
berechnet.

5.4.1 Ergebnisse

In Tabelle 5.12 ist die Lage der geschätzten Kontrollpunkte (schwarze Punkte) in den 500 Simulationsdurchläufen dargestellt. Alle Kontrollpunkte sind im selben Maßstab dargestellt, wobei die Länge einer gestrichelten Linie immer 0.04 beträgt. Der Schnittpunkt der beiden gestrichelten Linien gibt die „wahre“ Lage des Kontrollpunktes $\tilde{\mathbf{x}}_j$ an. Zusätzlich ist die aus $\Sigma_{\text{emp},j}$ berechnete Konfidenzellipse mit einer Irrtumswahrscheinlichkeit von $\alpha = 5\%$ dargestellt. Bei Datensatz 9 werden insgesamt sechs Kontrollpunkte bestimmt. Mit der deterministischen Knotenvektormethode PT1 wird in allen 500 Simulationsdurchläufen ein identischer Knotenvektor erzeugt. Dadurch werden kleine und runde Konfidenzellipsen erzeugt. Da der erzeugte Knotenvektor nicht optimal ist, gibt es einen signifikanten Abstand Δs zwischen $\bar{\mathbf{x}}_j$ und $\tilde{\mathbf{x}}_j$. Die Ergebnisse der deterministischen Knotenvektormethode RIU ähneln den Ergebnissen von PT1, allerdings weicht die geschätzte Lage der Kontrollpunkte in mehreren Simulationsdurchläufen stark von den übrigen Ergebnissen ab. Insbesondere bei den Kontrollpunkten \mathbf{x}_4 und \mathbf{x}_5 führt dies zu einer signifikant erweiterten, länglich gezogenen Konfidenzellipse. Die heuristische Knotenvektormethode CSA führt bei diesem Datensatz zu einer großen Variation der geschätzten Kontrollpunkte. Alle Konfidenzellipsen sind größer

als der dargestellte Ausschnitt. Besonders groß sind die Konfidenzellipsen in Kontrollpunkt \mathbf{x}_2 und \mathbf{x}_5 . Mit Ausnahme von \mathbf{x}_1 ist bei jedem Kontrollpunkt ein signifikanter Abstand Δs zwischen $\bar{\mathbf{x}}_j$ und $\tilde{\mathbf{x}}_j$ festzustellen. Dieser Abstand übersteigt den bei PT1 festgestellten Abstand bei weitem. Ein ähnliches Bild ergibt sich bei der Verwendung der EMC-Methode. Hier sind die Konfidenzellipsen der Kontrollpunkte \mathbf{x}_3 bis \mathbf{x}_6 in einem ähnlichen Maß wie bei der Verwendung des CSA vergrößert. EGAori und EGAmo erzielen wesentlich kleinere Konfidenzellipsen als CSA und EMC. Lediglich bei den Kontrollpunkten \mathbf{x}_4 und \mathbf{x}_5 sowie in geringerem Maße bei \mathbf{x}_3 sind signifikant vergrößerte Konfidenzellipsen festzustellen. Insgesamt sind nur geringe Abstände Δs zwischen $\bar{\mathbf{x}}_j$ und $\tilde{\mathbf{x}}_j$ zu erkennen. Die Lage der Kontrollpunkte \mathbf{x}_4 und \mathbf{x}_5 scheint entlang einer Geraden verschoben zu sein. Daraus resultiert eine schmale und längliche Konfidenzellipse. In der letzten Zeile von Tabelle 5.12 ist die Lage der Kontrollpunkte eingezeichnet, die sich bei der Verwendung des wahren Knotenvektors ergibt. Hier werden kleine und runde Konfidenzellipsen erzielt. Der Abstand Δs ist bei allen Kontrollpunkten minimal. Bei genauem Hinsehen erkennt man, dass die Konfidenzellipsen bei den äußeren Kontrollpunkten größer sind als bei den mittleren Kontrollpunkten. Dies ist darauf zurückzuführen, dass zur Schätzung der äußeren Kontrollpunkte bei diesem Datensatz weniger Beobachtungen beitragen als bei den mittleren Kontrollpunkten.

Tabelle 5.12: Lage der geschätzten Kontrollpunkte (schwarz) und daraus empirisch berechnete Konfidenzellipse (rot, $\alpha = 5\%$) bei Rauschmodell I und Datensatz 9



In Tabelle 5.13 sind die numerischen Werte der bereits in Tabelle 5.12 graphisch dargestellten Konfidenzellipsen angegeben. Interessant ist, dass die großen und kleinen Halbachsen der Konfidenzellipsen bei PT1 identische Werte aufweisen wie bei der Verwendung des wahren Knotenvektors. Da bei beiden Varianten in den 500 Simulationsdurchläufen konstante Knotenvektoren verwendet werden, lässt dies den Schluss zu, dass die Varianz der Kontrollpunkte maßgeblich aus einer Varianz des Knotenvektors resultiert. Zweiter interessanter Aspekt ist, dass die kleine Halbachse B_K bei Verwendung von EGAori und EGAmo bei den Kontrollpunkten x_4 und x_5 in einer ähnlichen Größenordnung liegt wie die kleine Halbachse bei Verwendung des wahren Knotenvektors, obwohl die große Halbachse A_K bei EGAori und EGAmo signifikant größer ist. Dies bedeutet, dass sich die Varianz im Knotenvektor systematisch auf die Varianz der Kontrollpunkte auswirkt. Dies war auch schon in Tabelle 5.12 zu erkennen.

Tabelle 5.13: Parameter der empirisch berechneten Konfidenzellipsen ($\alpha = 5\%$) in Rauschmodell I und Datensatz 9

Methode	\mathbf{x}_1			\mathbf{x}_2			\mathbf{x}_3		
	Δ_s	A_K	B_K	Δ_s	A_K	B_K	Δ_s	A_K	B_K
PT1	0.0114	0.0042	0.0040	0.0076	0.0039	0.0036	0.0133	0.0031	0.0028
RIU	0.0112	0.0057	0.0040	0.0075	0.0046	0.0042	0.0130	0.0059	0.0031
CSA	0.0003	0.0341	0.0046	0.0304	0.9079	0.0215	0.3677	3.1092	0.0060
EMC	0.0002	0.0077	0.0041	0.0005	0.0108	0.0055	0.0182	0.1399	0.0034
EGAori	0.0000	0.0046	0.0040	0.0001	0.0045	0.0039	0.0001	0.0071	0.0031
EGAmo	0.0001	0.0046	0.0040	0.0000	0.0044	0.0038	0.0001	0.0065	0.0031
Wahr	0.0000	0.0042	0.0040	0.0000	0.0039	0.0036	0.0000	0.0031	0.0028
Methode	\mathbf{x}_4			\mathbf{x}_5			\mathbf{x}_6		
	Δ_s	A_K	B_K	Δ_s	A_K	B_K	Δ_s	A_K	B_K
PT1	0.0113	0.0030	0.0028	0.0079	0.0038	0.0034	0.0026	0.0043	0.0040
RIU	0.0119	0.0303	0.0030	0.0087	0.0435	0.0042	0.0025	0.0044	0.0043
CSA	0.8789	2.3316	0.0112	1.0946	2.5821	0.0356	0.0277	0.0637	0.0041
EMC	0.1569	1.2816	0.0066	0.2167	1.7777	0.0210	0.0060	0.0434	0.0043
EGAori	0.0007	0.0579	0.0029	0.0012	0.0801	0.0040	0.0000	0.0044	0.0043
EGAmo	0.0019	0.0554	0.0029	0.0028	0.0770	0.0040	0.0001	0.0044	0.0043
Wahr	0.0001	0.0030	0.0028	0.0001	0.0038	0.0034	0.0000	0.0043	0.0040

Im Folgenden soll die empirisch bestimmte Varianz-Kovarianz-Matrix $\Sigma_{\text{emp},j}$ mit der im GMM der Kontrollpunktschätzung bestimmten Varianz-Kovarianz-Matrix verglichen werden. Der Vergleich zwischen $\Sigma_{\text{emp},j}$ und $\Sigma_{\text{GMM},j}$ erfolgt auf Basis der Konfidenzellipsen. Beide Konfidenzellipsen werden mit einer Irrtumswahrscheinlichkeit von $\alpha = 5\%$ dargestellt.

In Abbildung 5.25 sind die beiden Konfidenzellipsen exemplarisch für Kontrollpunkt \mathbf{x}_1 und \mathbf{x}_5 bei Verwendung der Knotenvektormethode EGAmo dargestellt. Zusätzlich ist die Lage der geschätzten Kontrollpunkte eingezeichnet. Bei Kontrollpunkt \mathbf{x}_1 in Abbildung 5.25a hat die aus $\Sigma_{\text{emp},1}$ bestimmte rote Konfidenzellipse eine vergleichsweise ähnliche Größe wie die aus $\Sigma_{\text{GMM},1}$ bestimmte blaue Konfidenzellipse. Bei Kontrollpunkt \mathbf{x}_5 ist die aus $\Sigma_{\text{emp},5}$ bestimmte rote Konfidenzellipse deutlich lang gestreckter als die aus $\Sigma_{\text{GMM},5}$ bestimmte blaue Konfidenzellipse.

In Tabelle 5.14 sind die numerischen Werte der Parameter beider Konfidenzellipsen bei Verwendung von EGAmo für alle Kontrollpunkte angegeben. Interessant ist hier, dass die Werte für die kleine Halbachse B_K fast identisch sind, obwohl die Werte der großen Halbachse A_K zum Teil sehr stark abweichen. Bei den aus $\Sigma_{\text{GMM},j}$ bestimmten Konfidenzellipsen fällt auf, dass die große und kleine Halbachse bei allen Kontrollpunkten identisch sind. Dies bedeutet, dass x- und y-Komponente der Kontrollpunkte im GMM unkorreliert und gleichgenau geschätzt wurden. Dies ist auf das funktionale Modell der B-Spline Kontrollpunktschätzung (siehe Kapitel 3.4.1) und die als unkorreliert und gleichgenau simulierten sowie angenommenen Beobachtungen zurückzuführen.

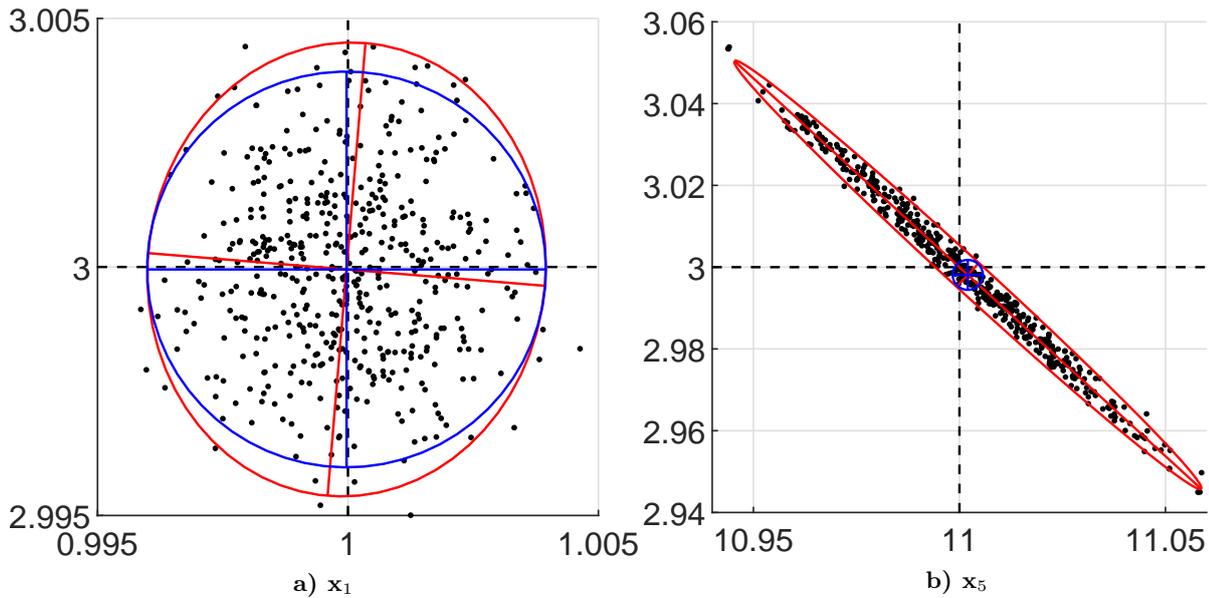


Abbildung 5.25: Lage der geschätzten Kontrollpunkte \mathbf{x}_1 und \mathbf{x}_5 (schwarz) mit empirisch bestimmter Konfidenzellipse (rot, $\alpha = 5\%$) und Konfidenzellipse aus der mittleren Varianz-Kovarianz-Matrix der B-Spline-Kontrollpunktschätzung (blau, $\alpha = 5\%$) bei Rauschmodell I, Datensatz 9 und der Knotenvektormethode EGAmoD

Tabelle 5.14: Eigenschaften der verschiedenen 95 %-Konfidenzellipsen in Rauschmodell I, Datensatz 9 und der Knotenvektormethode EGAmoD

Berechnung	\mathbf{x}_1			\mathbf{x}_2			\mathbf{x}_3		
	A_K [-]	B_K [-]	Θ_K [°]	A_K [-]	B_K [-]	Θ_K [°]	A_K [-]	B_K [-]	Θ_K [°]
emp.	0.0046	0.0040	85.2534	0.0044	0.0038	51.8861	0.0065	0.0031	99.5781
GMM	0.0040	0.0040	0.0000	0.0037	0.0037	0.0000	0.0029	0.0029	0.0000
Berechnung	\mathbf{x}_4			\mathbf{x}_5			\mathbf{x}_6		
	A_K [-]	B_K [-]	Θ_K [°]	A_K [-]	B_K [-]	Θ_K [°]	A_K [-]	B_K [-]	Θ_K [°]
emp.	0.0554	0.0029	57.6124	0.0770	0.0040	137.2653	0.0044	0.0043	99.4259
GMM	0.0029	0.0029	0.0000	0.0037	0.0037	0.0000	0.0040	0.0040	0.0000

Anhand von Abbildung 5.26 soll der Zusammenhang zwischen erhöhter Varianz in den Knoten zu der signifikant vergrößerten großen Halbachse der Konfidenzellipse einiger Kontrollpunkten verdeutlicht werden. Wie in den Abbildungen 5.24e und 5.24f ersichtlich, weist bei Datensatz 9 unter Verwendung der Knotenvektormethoden EGAori und EGAmoD der dritte interne Knoten eine erhöhte Varianz auf. In Abbildung 5.26a sind die B-Spline-Kurve, deren Kontrollpunkte und Knoten sowie die Linearkombinationen aus Kontrollpunkten und Basisfunktionen farblich dargestellt³. Jeder Kontrollpunkt ist der farblich passenden Linearkombination aus Basisfunktion und Kontrollpunkt zugeordnet. Die Knoten legen Start und Ende der einzelnen Basisfunktionen fest. Verschiebt man nun, wie in Abbildung 5.26b zu sehen, den dritten internen Knoten in einer Richtung, so ändert sich die B-Spline-Kurve signifikant im Vergleich zu der ursprünglichen B-Spline-Kurve. Bei Datensatz 9 verändern sich durch den verschobenen Knoten vor allem die beiden Basisfunktionen, die zu den Kontrollpunkten \mathbf{x}_4 und \mathbf{x}_5 zugeordnet sind. Behält man nun den verschobenen Knoten aus

³Die zweidimensionale Darstellung der Linearkombinationen ist geometrisch nicht 100% korrekt, da sie eigentlich pro Koordinatenkomponente in Abhängigkeit vom Ortsparameter u berechnet werden. So ändert sich z.B. die grün markierte Linearkombination durch den verschobenen Knoten in Abbildung 5.26b gar nicht, obwohl dies in der Abbildung so aussieht. Aufgrund der besseren Anschaulichkeit wurden diese Fehler allerdings in Kauf genommen.

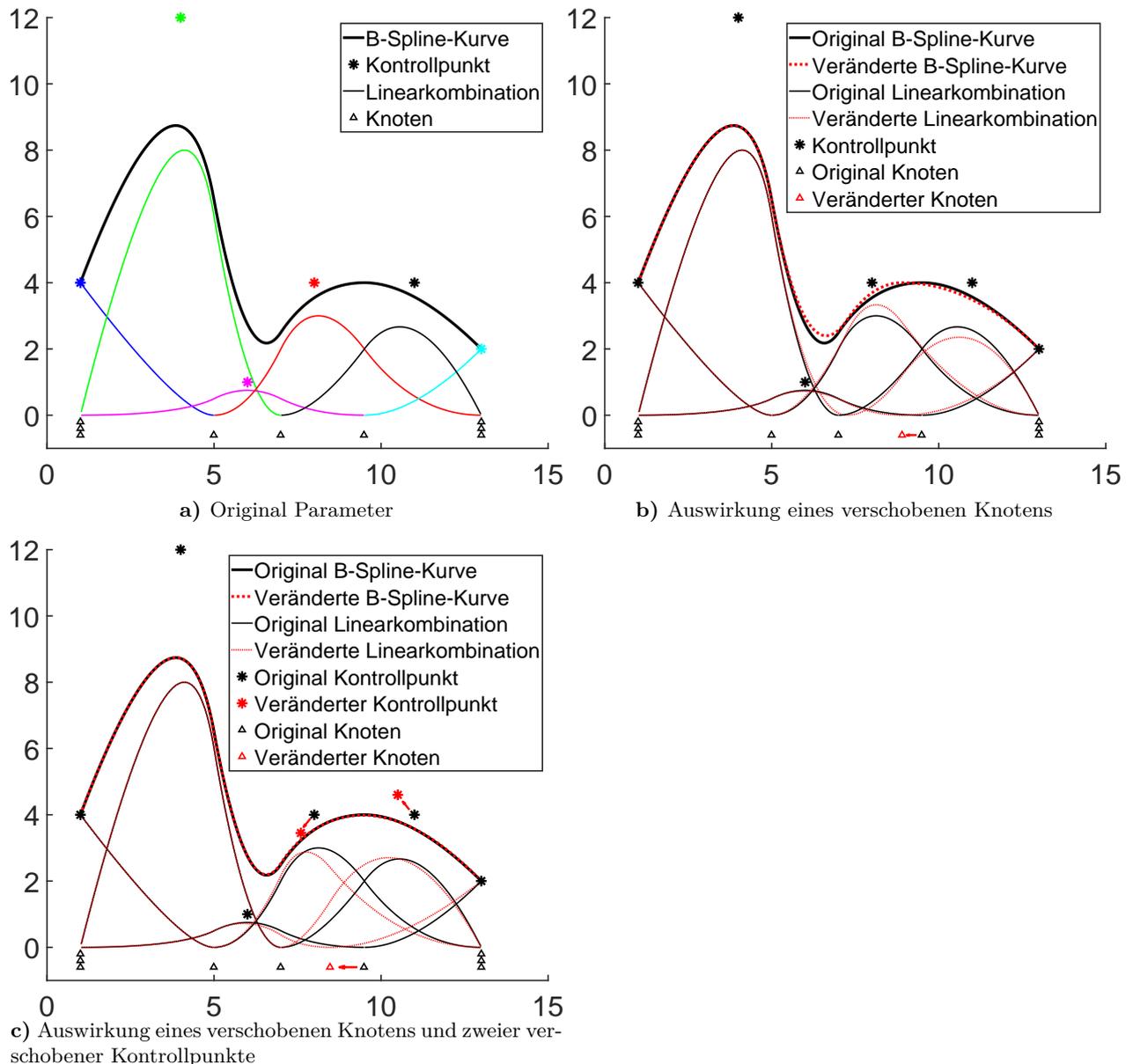


Abbildung 5.26: Analyse der B-Spline-Parameter in Datensatz 9

Abbildung 5.26b bei und verschiebt zusätzlich die beiden Kontrollpunkte x_4 und x_5 , so ergibt sich die in Abbildung 5.26c rot dargestellte B-Spline-Kurve. Diese entspricht beinahe dem in schwarz dargestellten ursprünglichen Verlauf der B-Spline-Kurve. Die Richtungen, in die die Kontrollpunkte x_4 und x_5 verschoben werden müssen, um den verschobenen Knoten „auszugleichen“, entsprechen den Richtungswinkeln der empirisch bestimmten Konfidenzellipsen (siehe Tabelle 5.12).

5.4.2 Zusammenfassung

Die in Kapitel 5.4.1 dargestellten Ergebnisse lassen folgende Schlussfolgerungen zu:

- Konstante Knotenvektoren führen zu kleinen und runden Konfidenzellipsen bei der Kontrollpunktschätzung.
- Nicht optimale Knotenvektoren führen zu einem Abstand zwischen dem wahren Wert des Kontrollpunktes und dem Mittelwert der geschätzten Kontrollpunkte.

- Eine erhöhte Varianz in einem Knoten wirkt sich in einer signifikant vergrößerten großen Halbachse der Konfidenzellipse einiger Kontrollpunkte aus. Der Richtungswinkel der Konfidenzellipse ist maßgeblich vom Datensatz abhängig.
- Die Größe der Konfidenzellipsen der Kontrollpunkte wird auch von der Anzahl der Beobachtungen beeinflusst, die zur Bestimmung des Kontrollpunktes beitragen. Kontrollpunkte, die einer vergleichsweise schmalen Basisfunktion zugeordnet sind, weisen aufgrund weniger beitragender Beobachtungen eine größere Konfidenzellipse auf. Bei dem untersuchten Datensatz waren das die beiden äußeren Kontrollpunkte.

5.5 Abweichende Modellwahl

In der vorhergehenden Untersuchung war die Anzahl der Kontrollpunkte $n + 1$ bekannt oder wurde vorab adäquat bestimmt. In diesem Kapitel soll untersucht werden, inwieweit sich eine geringfügig abweichende Wahl von n auf die kAE-Werte und die Häufigkeitsverteilung der erstellten internen Knoten bei den verschiedenen Methoden auswirkt. Außerdem soll der Einfluss der verschiedenen Methoden auf die Modellwahl untersucht werden. Hierzu erfolgt anhand des bereits in den Kapiteln 5.2 und 5.3.1 verwendeten normalverteilten Datensatzes 4 eine beispielhafte Untersuchung. Der Datensatz 4 wird zusätzlich mit zu wenig ($n = 5$) und zu viel ($n = 7$) Kontrollpunkten approximiert. Die Approximation mit der „korrekten“ Anzahl an Kontrollpunkten ($n = 6$) ist bereits erfolgt. Die erzielten kAE-Werte aus Tabelle 5.8 werden aus Gründen der Vergleichbarkeit im folgenden erneut dargestellt. Bei den heuristischen Methoden wird das Ergebnis nach Iteration 100 dargestellt.

5.5.1 Ergebnisse

In Tabelle 5.15 sind verschiedene Kennziffern der Modellwahl dargestellt. Der erste Block gibt den Prozentsatz der Simulationendurchläufe an, bei dem der Globaltest ($\alpha = 5\%$) mit der jeweiligen Methode zur Knotenvektorstimmung und dem gewählten n bestanden wurde. Wenig überraschend führt ein zu klein gewähltes $n = 5$ bei allen Methoden in allen Simulationendurchläufen zu einem Verwerfen des Globaltests. Mit $n = 6$, also dem „korrekten“ n , werden die in Tabelle 5.4 dargestellten Werte erreicht. Bei den Methoden EGAori und EGAmo wird der Globaltest in 93.2% bzw. 95.0% der Simulationendurchläufe eindeutig am häufigsten bestanden. Mit einem zusätzlichen Kontrollpunkt $n = 7$ steigt wie zu erwarten der Prozentsatz des bestandenen Globaltests bei allen Methoden (mit Ausnahme des PT1) an. Bei den Methoden EMC, EGAori und EGAmo wird in ca. 95% der Simulationendurchläufe der Globaltest bestanden. Unter Verwendung von PT1 wird der Globaltest bei keinem der untersuchten Werte für n bestanden.

Für den zweiten Block wurde für jede Methode zur Knotenvektorstimmung die Modellwahl mit dem AIC durchgeführt (siehe Kapitel 3.1.1.1). Angegeben ist für jedes n der Prozentsatz, mit dem es bei Verwendung der jeweiligen Methode vom AIC als bestes Modell identifiziert wurde. Bei PT1 wurde das Modell $n = 5$ vom AIC in allen Simulationendurchläufen als bestes Modell identifiziert. Bei allen anderen Methoden zur Knotenvektorstimmung wurde das Modell $n = 7$ in den meisten Simulationendurchläufen als optimales Modell identifiziert.

Für den dritten Block wird analog zum zweiten Block die Modellwahl mit dem BIC durchgeführt (siehe Kapitel 3.1.1.2). Auch beim BIC wird bei Verwendung von PT1 das Modell $n = 5$ als bestes Modell identifiziert. Bei den anderen Methoden ist im Vergleich zum AIC eine Verschiebung der Prozentzahlen von Modell $n = 7$ in Richtung des Modells $n = 6$ festzustellen. Bei Verwendung des RIU, EGAori und EGAmo wird das Modell $n = 6$ in der Mehrzahl der Simulationendurchläufe vom BIC als bestes Modell identifiziert. Bemerkenswert ist das Ergebnis bei Verwendung des EGAmo. Hier wird das Modell $n = 6$ in 99.4% der Simulationendurchläufe vom BIC als bestes Modell identifiziert. Wie in Kapitel 3.1.1.2 erwähnt, tendiert das BIC als asymptotisch konsistentes Kriterium dazu, komplexe Modelle stärker zu bestrafen. Die hier dargestellten Unterschiede zwischen AIC und BIC bei der Modellwahl sind also theoretisch zu erwarten gewesen.

Tabelle 5.15: Kennziffern der Modellwahl bei den verschiedenen Methoden zur Knotenvektorwahl berechnet aus den 500 Simulationsdurchläufen.

	Kennziffer	n	PT1	RIU	CSA	EMC	EGAori	EGAmo
Globaltest ($\alpha = 5\%$) bestanden		5	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
		6	0.0%	64.6%	1.2%	79.4%	93.2%	95.0%
		7	0.0%	72.8%	5.4%	95.2%	94.6%	95.4%
Modellwahl nach AIC		5	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%
		6	0.0%	41.4%	26.6%	20.8%	7.0%	34.0%
		7	0.0%	58.6%	73.4%	79.2%	93.0%	66.0%
Modellwahl nach BIC		5	100.0%	0.0%	0.0%	0.0%	0.0%	0.0%
		6	0.0%	62.0%	26.8%	49.0%	51.4%	99.4%
		7	0.0%	38.0%	73.2%	51.0%	48.6%	0.6%

Die erste Reihe in Tabelle 5.16 enthält die Kennziffern des kAE bei einer zu geringen Anzahl an Kontrollpunkten. Hier wird in allen Simulationsdurchläufen und bei allen Methoden der Knotenvektorwahl ein sehr ähnlicher kAE erreicht. Bei den heuristischen Verfahren und RIU liegt dieser signifikant über den mit der korrekten Anzahl an Kontrollpunkten erzielten kAE-Werten (in der zweiten Reihe von Tabelle 5.16). Ausnahme stellt hier die Approximation mit dem Knotenvektor aus PT1 dar, bei der mit einem Kontrollpunkt weniger ein geringerer kAE erzielt wird als mit der korrekten Anzahl an Kontrollpunkten.

In der dritten Reihe von Tabelle 5.16 sind die Kennziffern des kAE mit einer zu großen Anzahl an Kontrollpunkten dargestellt. Auch hier werden bei der Approximation unter Verwendung von PT1 mit dem zusätzlichen Kontrollpunkt höhere kAE-Werte erzielt als bei der Approximation mit der korrekten oder zu geringen Anzahl an Kontrollpunkten. Bei RIU, CSA und EMC sind mit der erhöhten Anzahl an Kontrollpunkten geringere kAE-Werte zu beobachten. Bei EGAori und EGAmo sind mit dem zusätzlichen Kontrollpunkt geringfügig höhere kAE-Werte zu beobachten als bei der korrekten Anzahl an Kontrollpunkten. Insgesamt erzielen EMC-Methode, EGAori und EGAmo vergleichbare kAE-Werte, wobei die kAE-Werte mit der EMC-Methode geringfügig kleiner sind.

Tabelle 5.16: Kennziffern des kAE der verschiedenen Methoden bei abweichender Modellwahl in Datensatz 4

n	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95%)	↑ KI (95%)
5	PT1	45.537	46.776	46.128	46.133	46.152	0.180	45.770	46.496
	RIU	45.537	46.776	46.128	46.133	46.152	0.180	45.770	46.496
	CSA	46.082	46.675	46.380	46.375	46.360	0.105	46.174	46.590
	EMC	45.932	46.673	46.380	46.373	46.349	0.111	46.170	46.599
	EGAori	46.097	46.678	46.380	46.375	46.356	0.106	46.177	46.585
	EGAmo	46.097	46.679	46.380	46.374	46.359	0.105	46.178	46.590
6	PT1	54.276	54.647	54.469	54.469	54.470	0.060	54.350	54.597
	RIU	0.552	5.964	2.283	2.130	1.689	0.954	0.932	4.663
	CSA	1.878	13.912	8.027	8.270	8.873	2.597	2.840	12.577
	EMC	0.528	7.463	1.526	1.310	1.174	0.765	0.787	3.571
	EGAori	0.691	1.914	1.225	1.231	1.245	0.201	0.832	1.614
	EGAmo	0.536	1.796	1.097	1.101	1.081	0.191	0.739	1.479
7	PT1	64.121	64.610	64.400	64.399	64.389	0.089	64.222	64.568
	RIU	0.564	6.911	2.079	1.863	1.693	0.875	0.937	4.383
	CSA	1.074	13.599	5.684	5.562	5.664	2.492	1.770	10.798
	EMC	0.624	1.837	1.180	1.185	1.222	0.198	0.808	1.598
	EGAori	0.607	1.926	1.249	1.255	1.269	0.202	0.857	1.645
	EGAmo	0.604	1.849	1.224	1.225	1.229	0.195	0.850	1.620

Abbildung 5.27 zeigt die Entwicklung der mittleren kAE-Werte der verschiedenen Methoden zur Knotenvektorwahl bei einer zu geringen Anzahl von Kontrollpunkten. EMC-Methode und CSA

starten mit höheren kAE-Werten, fallen dann rasch ab und konvergieren bei einem Wert um 46.4. Überraschend ist die Entwicklung von EGAori und EGAmo. Diese starten aufgrund der Initialisierung durch PT1 bzw. RIU mit geringen kAE-Werten um ca. 46.1. In der zweiten Iteration steigen diese Werte dann aber deutlich an, um in der Folge ebenfalls zu einem Wert von 46.4 zu konvergieren. Diese auf den ersten Blick ungewöhnliche Entwicklung ist durch die Differenz in der Berechnung von Fitnesswert und kAE zu erklären. Die Abbildung 5.28 zeigt die Häufigkeitsver-

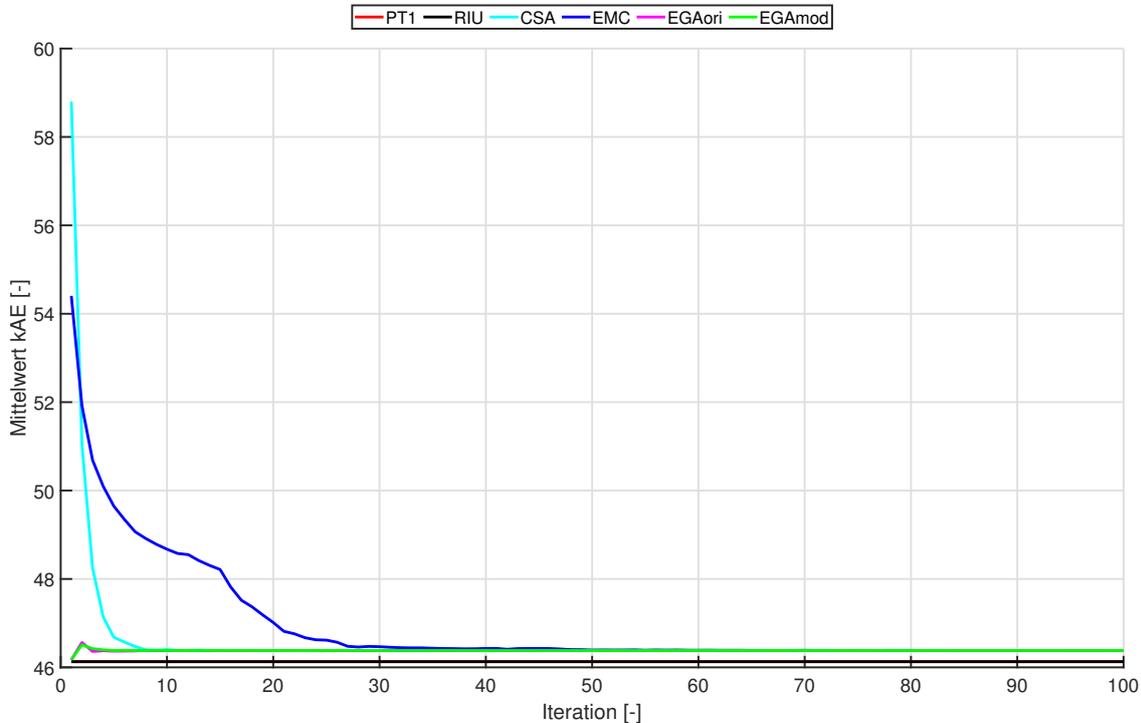


Abbildung 5.27: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4 und $n = 5$

teilung der erstellten internen Knoten. Hier zeigt sich bei allen Methoden ein ähnliches Bild. Die internen Knoten sind nahezu gleichabständig über die Wertespanne verteilt. Die erstellten internen Knoten korrespondieren nicht im geringsten mit den wahren Knoten bei der korrekten Modellwahl.

Abbildung 5.29 zeigt die Entwicklung der mittleren kAE-Werte der verschiedenen Methoden zur Knotenvektorstimmung bei einer zu großen Anzahl von Kontrollpunkten. Der EGAmo startet mit der Initialisierung durch RIU mit einem vergleichsweise kleinen kAE und konvergiert rasch. EMC-Methode und EGAori konvergieren ebenfalls zu diesem Wert, während der CSA zu einem deutlich erhöhten kAE-Wert konvergiert. Die Abbildung 5.30 zeigt die Häufigkeitsverteilung der erstellten internen Knoten bei einer zu großen Anzahl an Kontrollpunkten. Bei PT1 in Abbildung 5.30a werden 5 gleichabständige Knoten erzeugt, von denen 2 zufällig nahe an die wahren Knotenpositionen fallen. Beim RIU in Abbildung 5.30b werden häufiger Knoten an den wahren Knotenpositionen erstellt. Auch an den Positionen wo durch den PT1 interne Knoten generiert wurden, ist eine Häufung zu erkennen. Über den gesamten Wertebereich werden vereinzelt interne Knoten erzeugt. Beim CSA in Abbildung 5.30c ist eine breite Verteilung der internen Knoten in einem Bereich zwischen 0 und 0.75 zu erkennen. An den Positionen der wahren Knoten sind zwei wenig ausgeprägte Maxima zu erkennen. Bei der EMC-Methode in Abbildung 5.30d ist eine starke Konzentration der internen Knoten an den Positionen der wahren Knoten zu erkennen. Nur ganz vereinzelt befinden sich Knoten in anderen Bereichen. Bei EGAori (Abbildung 5.30e) und EGAmo (Abbildung 5.30f) ist ebenfalls eine starke Konzentration der internen Knoten an den Positionen der wahren Knoten erkennbar. Allerdings werden einzelne Knoten über die gesamte Wertespanne verteilt.

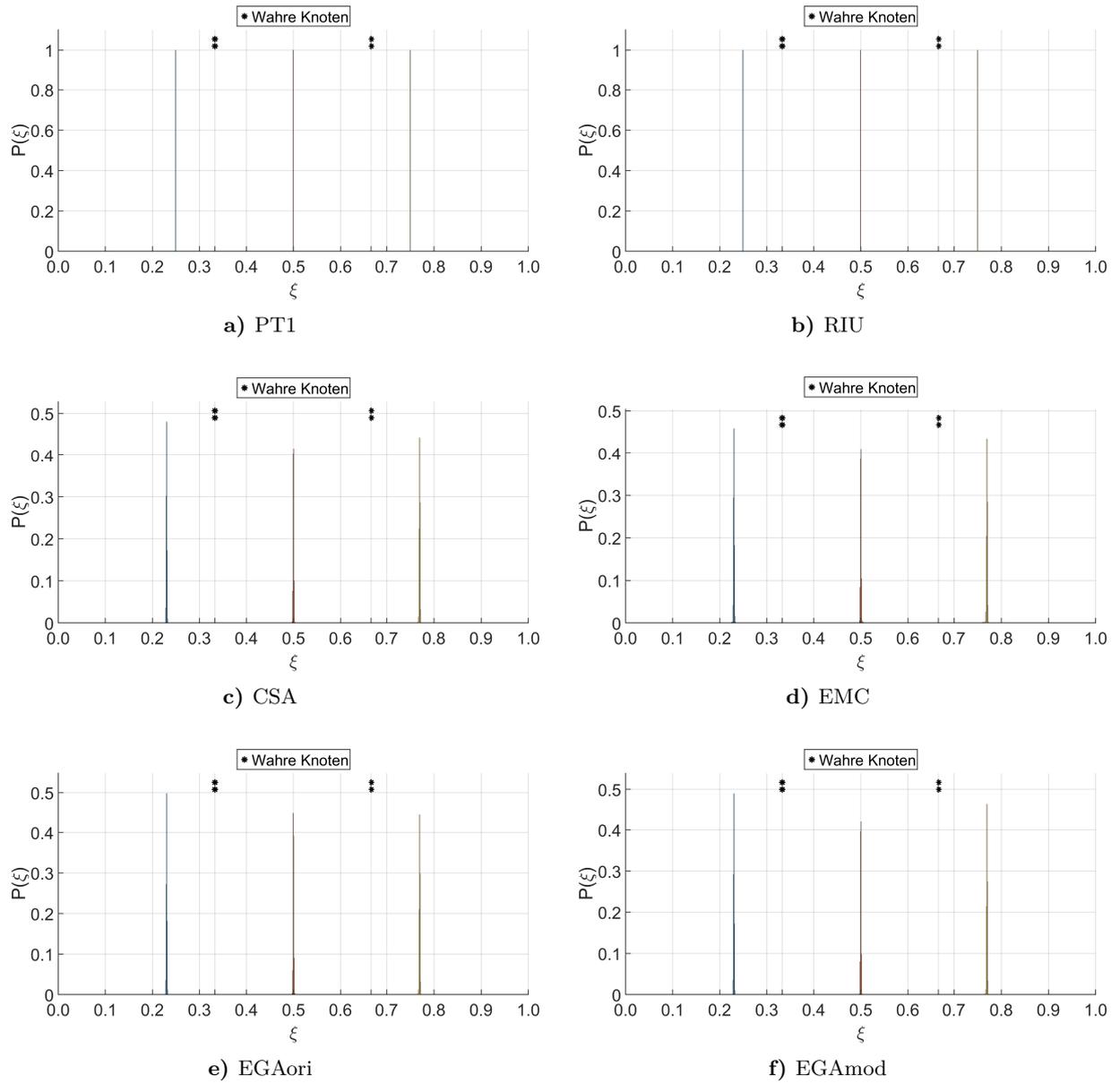


Abbildung 5.28: Knotenverteilung bei Datensatz 4 und $n = 5$

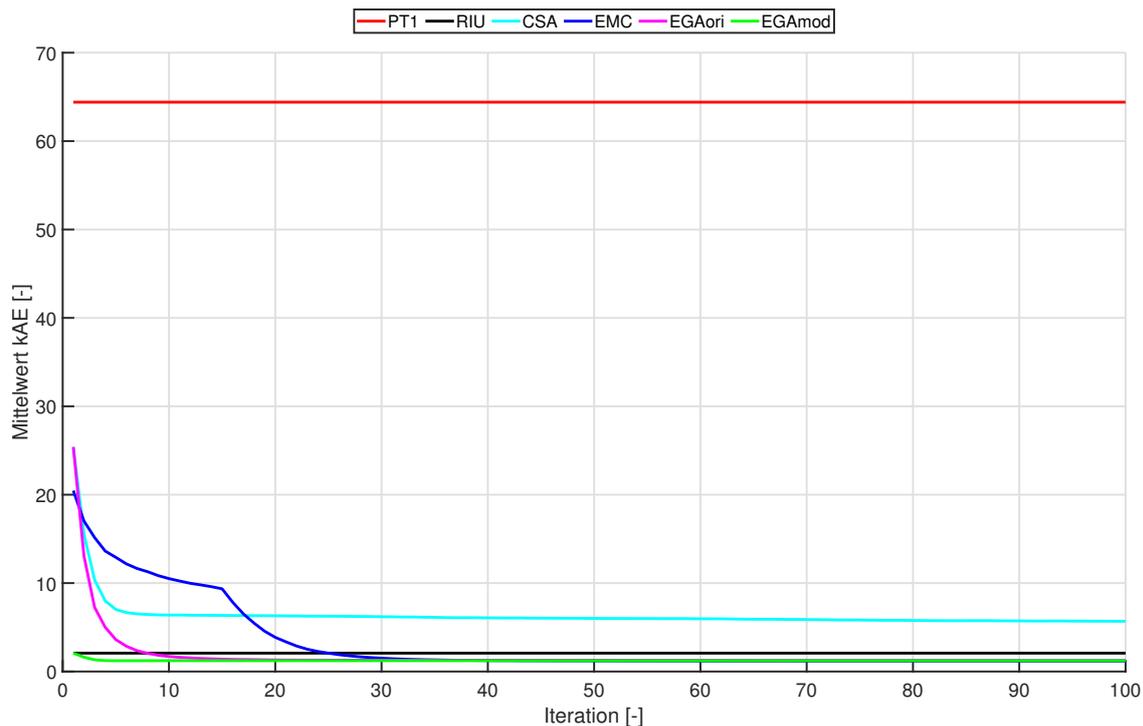


Abbildung 5.29: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4 und $n = 7$

5.5.2 Zusammenfassung

Die in Kapitel 5.5.1 dargestellten Ergebnisse lassen folgende Schlussfolgerungen zu:

- Bei einer zu geringen Anzahl an Kontrollpunkten $n = 5$ wird bei allen Verfahren der Globaltest verworfen.
- Mit der Methode EGAmoD wird sowohl beim korrekten Modell als auch bei dem Modell mit einem zusätzlichen Kontrollpunkt der höchste Prozentsatz bestandener Globaltests erreicht.
- Die Verwendung des AIC führt bei allen Methoden in der Mehrzahl der Simulationsdurchläufe zu einer Identifizierung des komplexeren Modells $n = 7$ als bestes Modell. Bei allen Methoden (ausgenommen PT1) ist der Prozentsatz, bei dem das korrekte Modell als bestes Modell identifiziert wurde, beim AIC kleiner als beim BIC.
- Mit der Methode EGAmoD und dem BIC wird bei diesem Datensatz das korrekte Modell $n = 6$ am häufigsten (in 99.4% der Simulationsdurchläufe) als bestes Modell identifiziert. Bei den anderen Methoden wird das korrekte Modell mit dem BIC weniger häufig erkannt.
- Bei zu geringer Anzahl an Kontrollpunkten erzielen alle Verfahren vergleichbare und hohe kAE-Werte. Die platzierten internen Knoten weichen stark von den Positionen der wahren Knoten ab.
- PT1 liefert bei steigender Anzahl an Kontrollpunkten höhere kAE-Werte. Insbesondere im Hinblick auf die Modellwahl ist dieses Verfahren kritisch zu sehen.
- Bei den Methoden RIU, CSA und EMC sinkt der kAE bei steigender Anzahl an Kontrollpunkten.
- Die Methoden EGAori und EGAmoD weisen bei der korrekten Anzahl an Kontrollpunkten einen geringeren kAE auf als mit einem zusätzlichen Kontrollpunkt mehr. Allerdings lag insbesondere der kAE des EGAmoD bei der korrekten Anzahl an Kontrollpunkten schon sehr nahe

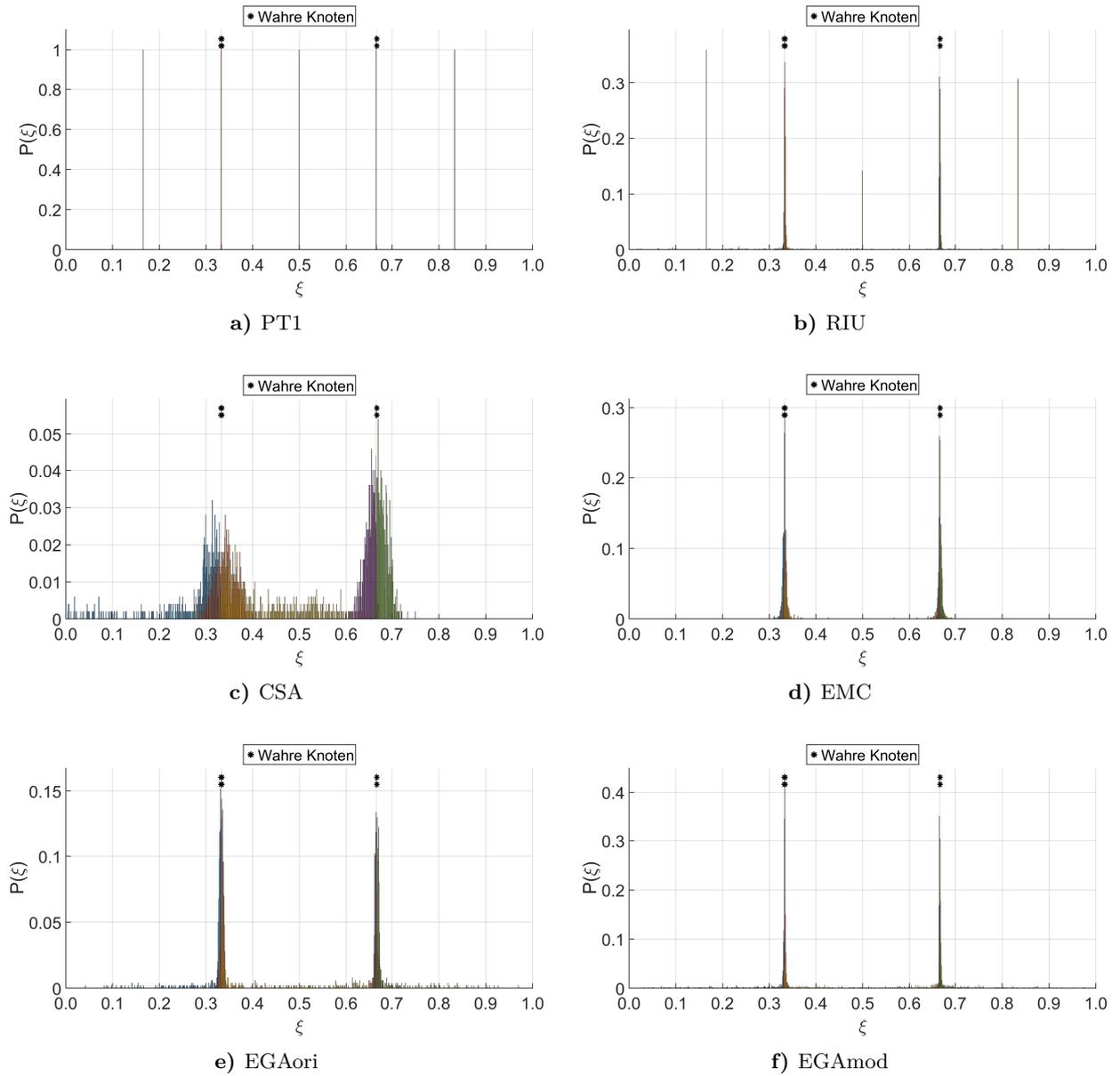


Abbildung 5.30: Knotenverteilung bei Datensatz 4 und $n = 7$

am optimalen Ergebnis. Grundsätzlich ist zu erwarten, dass der kAE ab einer bestimmten Anzahl an Kontrollpunkten mit einer weiter wachsenden Anzahl an Kontrollpunkten ebenfalls größer werden wird, da zunehmend das Messrauschen durch die B-Spline Kurve approximiert wird.

- Wird die korrekte Anzahl an Kontrollpunkten um einen Kontrollpunkt erhöht, ändert sich die Verteilung der internen Knoten im Vergleich zu der korrekten Anzahl an Kontrollpunkten nicht signifikant. Der zusätzliche Knoten wird zum Teil zufällig über die gesamte Wertespanne verteilt. Bei einer weiter zunehmenden Zahl an Kontrollpunkten werden sich die Knoten zunehmend über den gesamten Wertebereich verteilen, während die Konzentration an den wahren Knotenpositionen abnimmt.

5.6 Reale Datensätze - Kreuzvalidierung

In diesem Kapitel wird die Leistungsfähigkeit der verschiedenen Methoden zur Knotenvektorstichwahl bei der Approximation zweier realer Datensätze evaluiert.

5.6.1 Datensätze

Der Datensatz R1 stammt aus der Kranbahnvermessung und stellt die Messung einer Kranbahnschiene dar (siehe Abbildung 5.31a). Eine detaillierte Beschreibung des verwendeten Multisensorsystems ist in Dennig u. a. (2017) zu finden. Der Datensatz wurde durch zwei von schräg oben

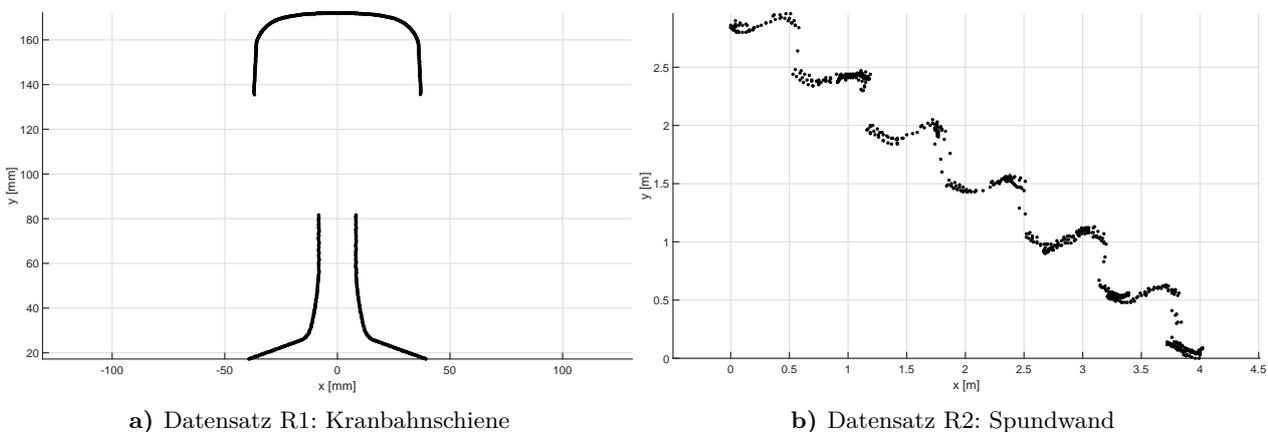


Abbildung 5.31: Die beiden realen Datensätze R1 und R2

auf die Schiene blickende Profils Scanner erfasst. Durch Verdeckungen unterhalb des Schienenkopfes enthält der Datensatz zwei größere Datenlücken. Für die Profils Scanner wurde aus den Herstellerangaben eine Koordinatengenauigkeit von $\sigma_{\text{noise},x} = \sigma_{\text{noise},y} = 0.0667$ mm für die zweidimensionale Punktwolke abgeleitet.

Der Datensatz R2 stammt aus dem Projekt „3D HydroMapper“, welches sich unter anderem mit der automatisierten 3D-Bauwerksaufnahme und Schadenserkenkung unter Wasser befasst. Eine Beschreibung des Projektes und des verwendeten Multisensorsystems ist in Hesse u. a. (2019) zu finden. Der Datensatz resultiert aus der Messung eines hydroakustischen Unterwasserscanners auf eine Spundwand (siehe Abbildung 5.31b). Die Messgenauigkeit dieses prototypischen Systems kann nicht exakt spezifiziert werden, da sie stark von den Umgebungseigenschaften abhängt. Die Koordinatengenauigkeit der zweidimensionalen Punktwolke wird in beiden Koordinatenkomponenten mit $\sigma_{\text{noise},x} = \sigma_{\text{noise},y} = 0.02$ m angenommen.

Tabelle 5.17: Eigenschaften der realen Datensätze und der Modellwahl

Ds.	r	Dim.	m	n	p	$\sigma_{\text{noise},x}, \sigma_{\text{noise},y}$
R1	1280	2	2560	36	3	0.0667 mm
R2	716	2	1432	30	2	0.02 m

Die Parameter der Modellwahl n und p wurden so gewählt, dass sie minimal sind, aber bei einer Approximation der gesamten Punktwolke (unter Verwendung des EGAmoD) der einseitige Globaltest (Kapitel 3.1.2.1) mit einer Irrtumswahrscheinlichkeit von $\alpha = 5\%$ bestanden werden kann.

5.6.2 Kreuzvalidierung

Anders als bei den vorhergehenden Untersuchungen ist bei den realen Datensätzen die „wahre“ Kurvenapproximation unbekannt. Deshalb erfolgt die Validierung der Leistungsfähigkeit der Methoden zur Knotenvektwahl mithilfe der Kreuzvalidierung. Grundidee der Kreuzvalidierung ist es, den vorhandenen Datensatz in zwei Gruppen aufzuteilen. Mit der einen Gruppe, der so genannten Trainingsgruppe, wird ein zu evaluierender Algorithmus trainiert. Mit der zweiten Gruppe, der so genannten Testgruppe, wird der Algorithmus evaluiert (siehe z.B. Arlot und Celisse (2010)). Im hier vorliegenden Fall werden die Daten in der Trainingsgruppe durch die B-Spline-Kurve approximiert. Anschließend wird analysiert, wie groß der Abstand zwischen den Daten der Testgruppe und den geschätzten Punkten auf der B-Spline-Kurve ist. Für weitere Informationen zum Thema Kreuzvalidierung sei an dieser Stelle beispielsweise auf Picard und Cook (1984), Kohavi (1995) und Arlot und Celisse (2010) verwiesen. Im folgenden wird die von Picard und Cook (1984) und Arlot und Celisse (2010) beschriebene Monte-Carlo Kreuzvalidierung angewendet. Hierzu wird die gemessene Punktwolke zufällig in Trainingsgruppe $\mathbf{Q}_{\text{Training}}$ und Testgruppe \mathbf{Q}_{Test} aufgeteilt. $\mathbf{Q}_{\text{Training}}$ werden 75 % und \mathbf{Q}_{Test} die restlichen 25 % der Messdaten zugeordnet. Anschließend wird $\mathbf{Q}_{\text{Training}}$ durch die B-Spline Kurve approximiert und zwischen \mathbf{Q}_{Test} und den geschätzten Punkten auf der B-Spline-Kurve werden die Abstände berechnet. Diese Experiment wird mehrfach, in diesem Fall 500 mal, wiederholt.

Im Falle der B-Spline Approximation sind für die Kreuzvalidierung einige Besonderheiten zu beachten. Die Punktwolke wird erst nach der Parametrisierung aufgeteilt. Somit bezieht sich der Ortsparameter beider Gruppen auf die Gesamtpunktwolke. Der erste und letzte Punkt der Punktwolke wird immer der Trainingsgruppe $\mathbf{Q}_{\text{Training}}$ zugewiesen. Dadurch werden Extrapolationen vermieden. Nachdem der Knotenvektor bestimmt und die Kontrollpunkte geschätzt wurden, werden mit Formel 2.18 an den Ortsparametern der Testgruppe $\bar{\mathbf{u}}_{\text{Test}}$ Kurvenpunkte $\mathbf{C}_{\text{Test}} = \mathbf{C}(\bar{\mathbf{u}}_{\text{Test}})$ auf der approximierten B-Spline Kurve ausgegeben. Zwischen \mathbf{C}_{Test} und \mathbf{Q}_{Test} wird analog zu Formel 5.11 die Differenz-Matrix $\Delta\mathbf{Q}$ berechnet:

$$\Delta\mathbf{Q} = \mathbf{Q}_{\text{Test}} - \mathbf{C}_{\text{Test}}. \quad (5.26)$$

Mit Formel 5.13 erfolgt anschließend die Berechnung des kAE. Der in diesem Kapitel angegebene kAE-Wert wird somit nicht zu einer Soll-Punktwolke sondern zu einer Ist-Punktwolke berechnet.

5.6.3 Ergebnisse

In Tabelle 5.18 sind die Kennziffern des kAE bei der Kreuzvalidierung dargestellt. Insgesamt wurde die Kontrollpunktschätzung mit den drei Schätzern, Methode der kleinsten Quadrate, Huber- und Hampel-Schätzer, durchgeführt. In fett sind jeweils die kleinsten kAE-Werte der jeweiligen Spalte bzw. des jeweiligen Datensatzes markiert. Bei beiden Datensätzen erzielen EGAori und EGAmoD in nahezu allen Kennziffern des kAE die kleinsten Werte. Bei Verwendung von EGAori und EGAmoD

passt die approximierten B-Spline-Kurve besser zu den Daten der Testgruppe als bei den anderen Methoden.

Tabelle 5.18: Kennziffern des kAE für die Kreuzvalidierung der realen Datensätze unter Verwendung verschiedener Schätzer. Die Einheit des Datensatzes R1 ist [mm]. Die Einheit des Datensatzes R2 ist [m].

DS.	Methode	kAE (MdkQ)				kAE (Huber)				kAE (Hampel)			
		Min.	Max.	Mittel	Modus	Min.	Max.	Mittel	Modus	Min.	Max.	Mittel	Modus
R1	PT1	63.714	132.701	93.125	88.724	54.985	131.163	86.583	82.214	60.849	134.673	92.580	88.108
	RIU	30.910	48.295	36.320	35.564	30.871	48.476	36.269	35.535	30.668	56.275	37.348	36.023
	CSA	26.049	79.436	30.331	29.183	26.034	79.490	30.343	29.194	26.119	73.024	30.603	29.370
	EMC	31.012	39.440	34.956	34.744	30.928	39.869	34.944	34.840	30.984	42.359	35.478	35.320
	EGAori	26.039	31.372	28.625	28.559	26.055	31.418	28.636	28.590	26.126	31.651	28.798	28.746
	EGAmo	26.343	32.789	28.749	28.484	26.302	32.789	28.760	28.480	26.283	32.823	28.923	28.630
R2	PT1	4.855	8.736	6.882	6.665	4.697	8.868	6.782	6.563	4.717	10.445	7.223	7.037
	RIU	4.273	7.578	5.679	5.514	4.215	7.448	5.648	5.463	4.245	7.491	5.817	5.616
	CSA	4.401	11.451	5.420	5.394	4.390	11.353	5.390	5.370	4.380	10.810	5.477	5.480
	EMC	4.184	6.435	5.131	5.155	4.142	6.402	5.099	5.144	4.102	6.400	5.150	5.168
	EGAori	3.911	15.021	4.651	4.615	3.897	14.996	4.643	4.580	3.924	15.196	4.678	4.581
	EGAmo	3.832	5.643	4.603	4.538	3.850	5.643	4.596	4.543	3.891	5.708	4.631	4.652

Dies ist ein Indiz dafür, dass mit EGAori und EGAmo nicht nur die Punktwolke, sondern auch das zugrunde liegende Objekt besser approximiert wird als mit den anderen Methoden.

Insgesamt unterschieden sich die Kennziffern der kAE bei den drei verwendeten Schätzern nicht signifikant. Dies ist ein Hinweis, dass in den Datensätzen keine größeren Ausreißer enthalten sind. Eine signifikante Reduktion der kAE-Werte bei Verwendung eines robusten Schätzers im Vergleich zur Methode der kleinsten Quadrate würde hingegen auf das Vorliegen von Ausreißern in der Punktwolke hindeuten.

Im Folgenden werden exemplarisch für die Methode der kleinsten Quadrate die kAE-Werte für die Trainingsdaten dargestellt. Die Kurvenpunkte der Trainingsgruppe $\mathbf{C}_{\text{Training}} = \mathbf{C}(\bar{\mathbf{u}}_{\text{Training}})$ werden an den Ortsparametern der Trainingsgruppe $\bar{\mathbf{u}}_{\text{Training}}$ auf der approximierten B-Spline Kurve ausgegeben. Die Differenz-Matrix $\Delta\mathbf{Q}$ wird über:

$$\Delta\mathbf{Q} = \mathbf{Q}_{\text{Training}} - \mathbf{C}_{\text{Training}} \quad (5.27)$$

berechnet. Mit Formel 5.13 erfolgt anschließend die Berechnung des $\text{kAE}_{\text{Training}}$.

Unter der Annahme, dass die Messdaten normalverteilt, gleichgenau und stochastisch unabhängig⁴ sind, lässt sich auf Basis der Rayleigh-Verteilung ein Erwartungswert für $\text{kAE}_{\text{Training}}$ berechnen (siehe z.B. Kroese u. a. (2011, S.137f.)):

$$E(\text{kAE}_{\text{Training}}) = r \cdot \sigma \cdot \sqrt{\frac{\pi}{2}}. \quad (5.28)$$

In Tabelle 5.20 sind in der linken Spalte die berechneten Werte für $\text{kAE}_{\text{Training}}$ dargestellt. Auch

Tabelle 5.19: Erwartungswerte für $\text{kAE}_{\text{Training}}$

Ds.	$E(\text{kAE}_{\text{Training}})$
R1	80.252 mm
R2	13.461 m

hier erzielen EGAori und EGAmo im Mittel die beste Approximation. Bei Datensatz R1 liegen die Werte für $\text{kAE}_{\text{Training}}$ bei Verwendung von EGAori und EGAmo im Bereich des berechneten

⁴Inwieweit diese Annahme zutreffend ist, wurde im Rahmen dieser Arbeit nicht weiter untersucht.

Erwartungswertes. Bei den anderen Methoden liegen die Werte für kAE_{Training} deutlich über dem berechneten Erwartungswert. Bei Datensatz R2 liegen die kAE_{Training} -Werte bei Verwendung von EGAori und EGAmo unter dem berechneten Erwartungswert. Bei den anderen Methoden liegen die Werte für kAE_{Training} über dem berechneten Erwartungswert.

Zusätzlich ist in Tabelle 5.20 in der rechten Spalte das Verhältnis von kAE_{Training} zu den in Tabelle 5.18 dargestellten kAE -Werten gegeben. Würden die Punkte in der Testgruppe und der Trainingsgruppe gleich genau durch die B-Spline-Kurve approximiert, ergäbe sich ein Verhältnis von 3.0. Dies liegt an der Tatsache, dass der Trainingsgruppe die dreifache Anzahl an Punkten gegenüber der Testgruppe zugeordnet ist. Da im Approximationsprozess aber nur die Punkte der Trainingsgruppe beteiligt sind und insbesondere die Verbesserungen in der Kontrollpunktschätzung nur für diese Punkte minimal werden, ist zu erwarten, dass die Punkte der Trainingsgruppe genauer bzw. besser approximiert werden. Dies bedeutet, dass beim Verhältnis $kAE_{\text{Training}}/kAE$ ein Wert unter 3.0 zu erwarten ist.

Tabelle 5.20: Kennziffern des kAE_{Training} der realen Datensätze unter Verwendung der Methode der kleinsten Quadrate. Die Einheit des Datensatzes R1 ist [mm]. Die Einheit des Datensatzes R2 ist [m]. In der rechten Spalte ist das Verhältnis zwischen kAE_{Training} und kAE dargestellt. Dieses ist einheitenlos.

DS.	Methode	kAE_{Training}				$kAE_{\text{Training}}/kAE$			
		Min.	Max.	Mittel	Modus	Min.	Max.	Mittel	Modus
R1	PT1	219.692	321.358	262.966	256.956	3.45	2.42	2.82	2.90
	RIU	88.965	141.647	103.640	101.552	2.88	2.93	2.85	2.86
	CSA	79.669	111.063	85.913	83.581	3.06	1.40	2.83	2.86
	EMC	90.066	107.412	99.728	99.788	2.90	2.72	2.85	2.87
	EGAori	78.321	84.736	80.862	80.610	3.01	2.70	2.82	2.82
	EGAmo	76.922	84.587	80.986	80.879	2.92	2.58	2.82	2.84
R2	PT1	15.082	21.935	19.043	19.152	3.11	2.51	2.77	2.87
	RIU	13.113	19.341	15.805	15.953	3.07	2.55	2.78	2.89
	CSA	13.334	16.327	14.917	14.844	3.03	1.43	2.75	2.75
	EMC	13.128	15.022	14.042	14.086	3.14	2.33	2.74	2.73
	EGAori	11.185	13.147	12.094	12.092	2.86	0.88	2.60	2.62
	EGAmo	11.129	12.695	12.020	12.137	2.85	2.25	2.61	2.67

Insgesamt zeigen sich bei Datensatz R1 im Mittel bei allen Methoden sehr ähnliche Werte für das Verhältnis $kAE_{\text{Training}}/kAE$. Diese liegen zwischen 2.82 und 2.85. Bei Datensatz R2 weicht das Verhältnis $kAE_{\text{Training}}/kAE$ der Mittelwerte beim EGAori und EGAmo (2.60 bzw. 2.61) geringfügig von dem der anderen Methoden (2.74-2.78) ab. Eine mögliche Erklärung dafür könnte bei diesem Datensatz in der nicht exakt bekannten Messgenauigkeit liegen. Dies könnte bei der Modellwahl (Tabelle 5.17) zur Wahl einer zu großen Anzahl an Kontrollpunkten geführt haben, da andernfalls die Nullhypothese des Globaltests verworfen worden wäre. Dadurch könnte ungewollt das Punktrauschen in der B-Spline-Kurve modelliert worden sein. Da EGAori und EGAmo zu besser an die gegebenen Daten angepassten B-Spline-Kurven führen (siehe z.B. Kapitel 5.3.1.1), ist die Modellierung des Punktrauschens bei diesen beiden Methoden deutlicher sichtbar. Dies könnte auch eine Erklärung dafür sein, dass bei EGAori und EGAmo der berechnete Erwartungswert unterschritten wird.

5.6.4 Zusammenfassung

Die in Kapitel 5.6.3 dargestellten Ergebnisse lassen folgende Schlussfolgerungen zu:

- Von den sechs untersuchten Methoden zur Knotenvektorwahl erzeugen EGAori und EGAmo bei den analysierten realen Datensätzen insgesamt die Knotenvektoren, die nach der Kontrollpunktschätzung zu den geringsten kAE-Werten führen. Ein geringer kAE-Wert ist ein Maß dafür, dass die approximierte B-Spline Kurve gut zu dem zu approximierenden Objekt passt.
- Bei den untersuchten Datensätzen führt die Anwendung robuster Schätzer in der Kontrollpunktschätzung zu keiner signifikanten Änderung in den kAE-Werten. Eine signifikante Reduktion der kAE-Werte bei Verwendung robuster Schätzer würde auf das Vorliegen von Ausreißern in den Daten hindeuten.
- Wird bei der Modellwahl eine zu große Anzahl an Kontrollpunkten gewählt, wird das Punktrauschen in der B-Spline-Kurve mit modelliert. Dies führt dazu, dass im Rahmen der Kreuzvalidierung Punkte der Testgruppe deutlich ungenauer durch die B-Spline-Kurve approximiert werden als Punkte der Trainingsgruppe.

6 Numerische Beurteilung der B-Spline-Approximation bei Ausreißern

Wie in Kapitel 1.1 bereits erwähnt, hat das Auftreten von Ausreißern in einer zu approximierenden Punktwolke einen lokal begrenzten aber dennoch signifikanten Einfluss auf die approximierende B-Spline-Kurve. Gerade bei linien- oder flächenhaften Messverfahren, z.B. dem Laserscanning, wird die hohe Messfrequenz mit einer fehlenden Redundanz bezahlt. Dadurch wird die Identifikation und Elimination von Ausreißern erschwert oder gar verhindert. Die Ursachen für das Auftreten von Ausreißern können dabei, abhängig von den verwendeten Messverfahren, vielfältig sein und sollen im Rahmen dieser Arbeit nicht im Detail untersucht werden. Eine häufige Ursache ist, unabhängig vom verwendeten Messverfahren, eine unzureichende oder fehlende Segmentierung der Punktwolke. Das bedeutet, dass eine zu approximierende Punktwolke neben Punkten, die auf dem zu approximierenden Objekt liegen, auch weitere Punkte enthält, die nicht das Objekt repräsentieren. Dies können Punkte sein, die auf Störobjekten im Messweg liegen oder eigentlich benachbarten Objekten hätten zugeordnet werden müssen. Eine andere Ursache können Abweichungen in Strecken- oder Richtungsmessungen sein, die z.B. durch Reflektionseigenschaften am Objekt indiziert werden.

Im Rahmen dieses Kapitels soll der Mehrwert der Anwendung robuster Schätzer bei der B-Spline-Kurvenapproximation von ausreißerbehafteten Datensätzen evaluiert werden. Wie in Kapitel 5 erfolgt die Evaluation auf Basis unterschiedlicher simulierter (synthetischer) Daten, um die Verfahren und daraus resultierende Schätzergebnisse evaluieren, validieren und analysieren zu können.

6.1 Rauschmodelle

Insgesamt werden die zur Evaluation herangezogenen Datensätze mit vier weiteren Rauschmodellen überlagert. Im Gegensatz zu dem in Kapitel 5.1.3 dargestellten Rauschmodell I wird angenommen, dass die im folgenden beschriebenen Rauschmodelle II bis V Ausreißer enthalten.

Bei Rauschmodell II werden die kompletten Beobachtungen mit der t -Verteilung verrauscht. Aus dem niedrig gewählten Freiheitsgrad $\nu = 3$ resultiert eine Wahrscheinlichkeitsdichtefunktion mit höheren Wahrscheinlichkeiten an den Rändern der Verteilung. Man spricht deshalb auch von einer Verteilung mit schweren Rändern¹. Die t -Verteilung geht in eine Normalverteilung über, wenn $\nu \rightarrow \infty$ geht. Das t -verteilte Rauschen wird mit dem MATLAB-Zufallsgenerator *trnd* der Statistics and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) generiert. Das Rauschen wird über einen Skalierungsfaktor skaliert. Als Skalierungsfaktor wird das bei Rauschmodell I gewählte σ_{noise} verwendet (siehe Kapitel 5.1.3).

In Rauschmodell III werden die Daten mit der Gaußschen Mischverteilung verrauscht. Dabei wird ein Anteil der Daten mit dem üblichen normalverteilten Rauschen $\epsilon_{1\sigma} \sim \mathcal{N}(\mu = 0, \sigma_{\text{noise}}^2)$ verrauscht. Die restlichen Daten werden mit einem erhöhten normalverteilten Rauschen $\epsilon_{5\sigma} \sim \mathcal{N}(\mu = 0, (5\sigma_{\text{noise}})^2)$ verrauscht:

$$\epsilon \sim (1 - r_1) \cdot \mathcal{N}(0, \sigma_{\text{noise}}^2) + r_1 \cdot \mathcal{N}(0, (5\sigma_{\text{noise}})^2) \text{ mit } 0 \leq r_1 \leq 1, \quad (6.1)$$

wobei r_1 den Prozentsatz für den Anteil der Daten mit erhöhtem Rauschen angibt. Das Rauschen der Gaußschen Mischverteilung wird mithilfe der MATLAB-Funktion *gmdistribution* der Statistics

¹Im Englischen als „heavy-tailed“ bezeichnet

and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) generiert. Der Prozentsatz für den Anteil der Daten mit erhöhtem Rauschen r_1 entspricht nicht direkt dem Anteil der tatsächlich auftretenden Ausreißern, da $\epsilon_{1\sigma}$ und $\epsilon_{5\sigma}$ eine Schnittmenge aufweisen. Im Folgenden wird trotzdem bei der Gaußschen Mischverteilung die Bezeichnung „Ausreißer“ für Daten mit erhöhtem Rauschen verwendet. Andererseits bezieht sich der Prozentsatz auf die Anzahl der Beobachtungen m und nicht auf die Anzahl der Punkte r . Im Falle der verwendeten zweidimensionalen Testdatensätzen 4-9 setzt sich der wahre Beobachtungsvektor $\tilde{\mathbf{I}}$ wie in Formel 5.6 zusammen. Bei Rauschmodell III werden entsprechend der gewählten Prozentsätze zufällig Beobachtungen mit dem Rauschen $\epsilon_{1\sigma}$ oder $\epsilon_{5\sigma}$ belegt. Diese Beobachtungen können sowohl x - als auch y -Koordinate betreffen. Dazu folgendes Beispiel:

Im Falle 1000 zweidimensionaler Punkte ergeben sich 2000 Beobachtungen. Wird für 10% der Beobachtungen das Rauschen $\epsilon_{5\sigma}$ festgelegt, werden folglich 200 Beobachtungen mit dem Rauschen $\epsilon_{5\sigma}$ und 1800 Beobachtungen mit dem Rauschen $\epsilon_{1\sigma}$ belegt. Im ungünstigsten Fall werden also 200 Punkte vom Rauschen $\epsilon_{5\sigma}$ beeinflusst, wobei $\epsilon_{5\sigma}$ dann nur auf eine der beiden Koordinatenkomponenten wirkt. Im günstigsten Fall werden nur 100 Punkte von $\epsilon_{5\sigma}$ beeinflusst, wobei $\epsilon_{5\sigma}$ dann auf beide Koordinatenkomponenten wirkt. Bei zweidimensionalen Daten kann der Ausreißeranteil der Punkte r_Q aus dem Ausreißeranteil der Beobachtungen r_1 mit der folgenden Überschlagsformel berechnet werden:

$$r_Q = 2 \cdot r_1 - r_1^2 \text{ mit } 0 \leq r_1 \leq 1. \quad (6.2)$$

Bei zweidimensionalen Daten entspricht die Anzahl der Punkte der halben Anzahl der Beobachtungen. Würden die Ausreißer nur in einer Koordinatenkomponente wirken, würde sich der Ausreißeranteil der Punkte im Vergleich zum Ausreißeranteil der Beobachtungen verdoppeln. Da es aber sein kann, dass sowohl x - als auch y -Komponente eines Punktes ausreißerbehafet sind, muss der erwartete Anteil dieser Fälle (entspricht im zweidimensionalen Fall: r_1^2) in Formel 6.2 abgezogen werden. Die Prozentsätze r_Q und r_1 sind zwischen 0 und 1 normiert: Bei einem Anteil von 10% für $\epsilon_{5\sigma}$ gilt für $r_1 = 0.1$. Durch Einsetzen in Formel 6.2 ergibt sich für $r_Q = 0.19$, also ein Ausreißeranteil der Punkte von 19%.

Bei Rauschmodell IV und V wird ein Teil der Daten mit dem üblichen normalverteilten Rauschen $\epsilon_{1\sigma} \sim \mathcal{N}(0, \sigma_{\text{noise}}^2)$ belegt. Der restliche Teil der Daten wird mit einem einseitigen Rauschen ϵ_{bias} belegt. Das einseitige Rauschen ϵ_{bias} wird nach dem folgenden, in Dorndorf u. a. (2017) verwendeten Schema berechnet.

Zunächst wird ein Vektor ϵ_{χ^2} aus der χ^2 -Verteilung mit dem Freiheitsgrad 5 generiert:

$$\epsilon_{\chi^2} \sim \chi_5^2. \quad (6.3)$$

Hierzu wird der MATLAB-Zufallsgenerator *chi2rnd* der Statistics and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) verwendet. Mit dem niedrig gewählten Freiheitsgrad ergibt sich eine rechtsschiefe Verteilung. Durch Normierung wird ϵ_{χ^2} in den Vektor ϵ_{norm} überführt:

$$\epsilon_{\text{norm}} = \frac{\epsilon_{\chi^2} - \min(\epsilon_{\chi^2})}{\max(\epsilon_{\chi^2}) - \min(\epsilon_{\chi^2})}. \quad (6.4)$$

Der Vektor ϵ_{norm} wird durch Formel 6.4 auf einen Wertebereich zwischen 0 und 1 normiert:

$$\{\epsilon_{\text{norm}} \in \mathbb{R} | 0 \leq \epsilon_{\text{norm}} \leq 1\}. \quad (6.5)$$

Im nächsten Schritt wird ϵ_{norm} so skaliert und translatiert, dass die Werte in einem Wertebereich zwischen x_{unten} und x_{oben} liegen. Die Skalierung und Translation von ϵ_{norm} erfolgt über:

$$\epsilon_{\text{bias}} = \epsilon_{\text{norm}} \cdot (x_{\text{oben}} - x_{\text{unten}}) + x_{\text{unten}}. \quad (6.6)$$

Der Wertebereich des resultierenden Vektors ϵ_{bias} lautet:

$$\{\epsilon_{\text{bias}} \in \mathbb{R} \mid x_{\text{unten}} \leq \epsilon_{\text{bias}} \leq x_{\text{oben}}\}. \quad (6.7)$$

Die Grenzen x_{unten} und x_{oben} werden über:

$$x_{\text{unten}} = F_{\mathcal{N}}^{-1}(p_{\text{unten}} \mid 0, \sigma_{\text{noise}}^2) \quad (6.8)$$

$$x_{\text{oben}} = F_{\mathcal{N}}^{-1}(p_{\text{oben}} \mid 0, \sigma_{\text{noise}}^2) \quad (6.9)$$

und somit über die inverse kumulierte Normalverteilung $F_{\mathcal{N}}^{-1}(p \mid 0, \sigma_{\text{noise}}^2)$ berechnet. Da dort das für die ausreißerfreien Daten verwendete σ_{noise} eingeht, bleibt die Relation zwischen ausreißerfreien und ausreißerbehafteten Daten auch bei wechselnden Werten für σ_{noise} konstant. Die Werte für $p_{\text{unten}} = 0.9985$ und $p_{\text{oben}} = 0.999999$ wurden so gewählt, dass der Großteil der Werte in ϵ_{bias} in einem Bereich zwischen $3\sigma_{\text{noise}}$ und $4\sigma_{\text{noise}}$ der normalverteilten Daten liegt (siehe auch Abbildung 6.1). Zur Berechnung der inversen kumulierten Normalverteilung wird die MATLAB-Funktion *norminv* der Statistics and Machine Learning Toolbox (Version 11.3 und MATLAB R2018a) verwendet. Sowohl bei Rauschmodell IV als auch bei Rauschmodell V wird das Rauschen ϵ_{bias} einseitig angebracht. Die Wahl des Vorzeichens ist zwar zufällig, wirkt dann aber auf den kompletten Datensatz. Die Rauschmodelle IV und V unterscheiden sich durch die Lage der ausreißerbehafteten Beobachtungen. Während bei Rauschmodell IV die ausreißerbehafteten Beobachtungen wie bei Rauschmodell III zufällig gewählt werden, folgen bei Rauschmodell V alle Ausreißer nacheinander und wirken nur in einer Koordinatenkomponente. D.h., es wird zunächst zufällig eine Koordinatenkomponente und der Index einer Anfangsbeobachtung² gewählt. Anschließend werden ausgehend vom Index der Anfangsbeobachtung alle Folgebeobachtungen mit ϵ_{bias} belegt, bis die Gesamtanzahl der ausreißerbehafteten Beobachtungen erreicht wird.

Für Rauschmodell IV gilt wie bei Rauschmodell III, dass der Anteil der ausreißerbehafteten Punkte r_{Q} höher ist als der im Folgenden angegebene Anteil der ausreißerbehafteten Beobachtungen r_1 . Auch hier gilt die Überschlagsformel 6.2 und die zuvor bei Rauschmodell III angeführten Erklärungen und Beispiele.

Für Rauschmodell V und eine zweidimensionale Punktwolke gilt für die Berechnung von r_{Q} :

$$r_{\text{Q}} = 2 \cdot r_1 \text{ mit } 0 \leq r_1 \leq 0.5. \quad (6.10)$$

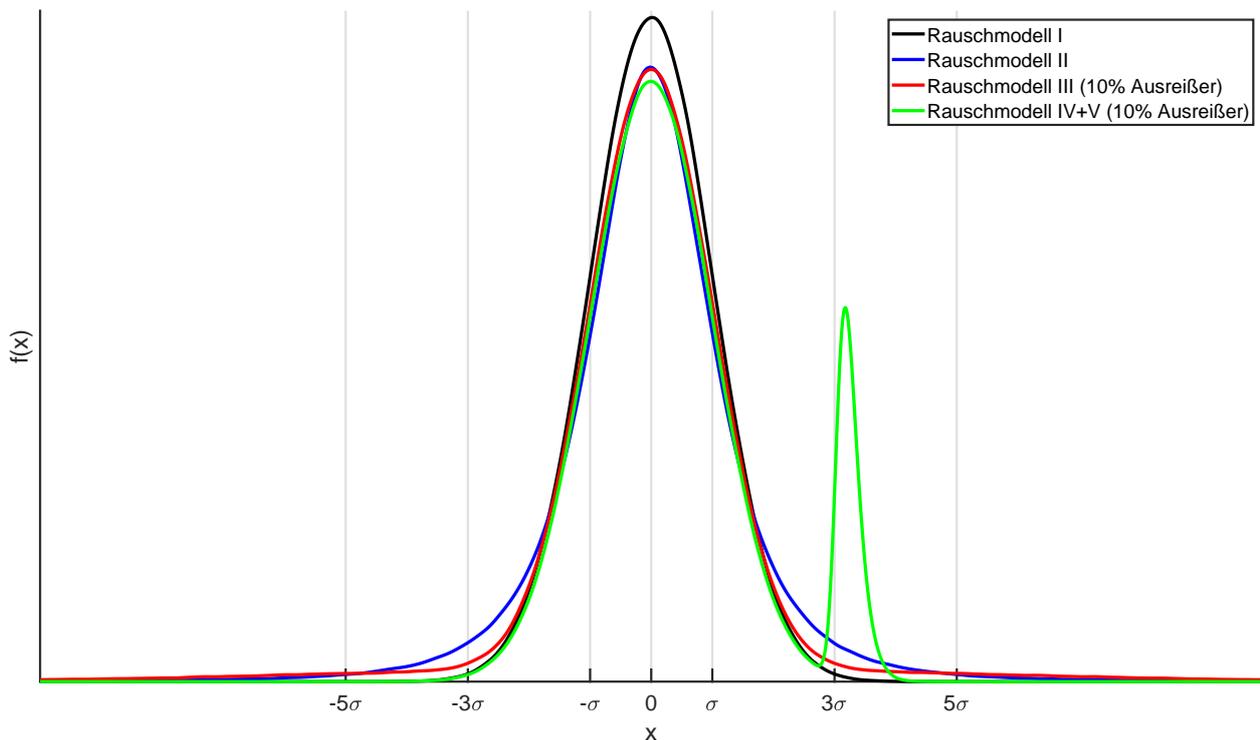
Dies hat zur Folge, dass ab einem Prozentsatz an ausreißerbehafteten Beobachtungen von 50% 100% der Punkte ausreißerbehaftet sind.

Alle beschriebenen und in Kapitel 6.2 und 6.3 untersuchten Rauschmodelle sind in Tabelle 6.1 zusammengefasst.

²Der Index der Anfangsbeobachtung ist so zu wählen, dass alle Ausreißer nacheinander in einer Koordinatenkomponente liegen.

Tabelle 6.1: Verwendete Rauschmodelle (RM)

RM	Verteilung		Ausreißer		
	Messdaten	Ausreißer	Anteil	Richtung	Position
II	$\sigma_{\text{noise}} \cdot t_3$		zufällig	zufällig	zufällig
IIIa			1 %		
IIIb	$\mathcal{N}(0, \sigma_{\text{noise}}^2)$	$\mathcal{N}(0, (5\sigma_{\text{noise}})^2)$	5 %	zufällig	zufällig
IIIc			10 %		
IVa			1 %		
IVb	$\mathcal{N}(0, \sigma_{\text{noise}}^2)$	$\pm \epsilon_{\text{bias}}$	5 %	einseitig	zufällig
IVc			10 %		
Va			1 %		
Vb	$\mathcal{N}(0, \sigma_{\text{noise}}^2)$	$\pm \epsilon_{\text{bias}}$	5 %	einseitig	nacheinander
Vc			10 %		

**Abbildung 6.1:** Wahrscheinlichkeitsdichtefunktion der verschiedenen Rauschmodelle

Analog zu Abschnitt 5.3.1.2 wird in Tabelle 6.2 der kAE-Mittelwert für jeden Datensatz und jedes Rauschmodell dargestellt. Dadurch soll ein besser Eindruck über die zu erwartenden kAE-Werte vermittelt werden. Für Rauschmodell *ii* bis *v* folgen die kAE-Werte nicht der Rayleigh-Verteilung.

Tabelle 6.2: kAE-Mittelwert der rohen Beobachtungen

RM	Datensatz		
	4	5	9
I	12.536	8.404	12.556
II	17.915	11.973	17.924
IIIa	13.141	8.783	13.130
IIIb	15.475	10.335	15.488
IIIc	18.324	12.282	18.341
IVa	13.059	8.726	13.039
IVb	14.897	9.982	14.895
IVc	17.106	11.486	17.118
Va	13.042	8.728	13.042
Vb	14.917	10.009	14.900
Vc	17.223	11.556	17.191

6.2 Ergebnisse Knotenvektormethoden

In diesem Kapitel werden die Kennziffern des kAE der verschiedenen Methoden und Schätzer tabellarisch dargestellt. Die Kennziffern sind aus Gründen der Übersichtlichkeit und Aussagekraft auf Mittelwert und Modus begrenzt. Die Darstellung erfolgt getrennt für jedes Rauschmodell. Zur besseren Vergleichbarkeit werden in der ersten Spalte der jeweiligen Tabelle die Ergebnisse des MdkQ-Schätzers bei der Approximation der normalverteilten Punktwolke in Rauschmodell I aus Kapitel 5.3.1 dargestellt. In der zweiten bis vierten Spalte sind die Ergebnisse der drei Schätzer MdkQ, Huber und Hampel für das jeweilige Rauschmodell dargestellt. Der Umfang der Berechnung wurde dabei auf die Datensätze 4, 5 und 9 beschränkt. Bei den heuristischen Verfahren wird das Ergebnis nach der letzten Iteration 100 dargestellt. Wie in Kapitel 5.3 ist in der untersten grün markierten Zeile für jeden Datensatz das mit dem wahren Knotenvektor erzielte Ergebnis dargestellt. Bei diesem Ergebnis werden die Kontrollpunkte sehr wohl von den verschiedenen Datensätzen beeinflusst, weshalb es zu Differenzen zwischen den Rauschmodellen kommt.

Rauschmodell II Tabelle 6.3 zeigt die Kennziffern des kAE für Rauschmodell II mit t -verteilten Daten. Aufgrund des gering gewählten Freiheitsgrades der t -Verteilung enthalten die so generierten Daten einen erhöhten Ausreißeranteil. Folglich unterscheiden sich auch die Ergebnisse unter Verwendung des MdkQ-Schätzers zwischen Rauschmodell I und II deutlich.

Zunächst werden die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und II verglichen. Bei Datensatz 4 ist in Rauschmodell II mit Ausnahme des PT1 und CSA bei alle Methoden, einschließlich des wahren Knotenvektors, eine signifikante Erhöhung der kAE-Werte zu erkennen. Der mittlere kAE steigt bei den Methoden RIU, EMC, EGAori, EGAmo um ca. 60%. Dies ist bei einem höheren Ausreißeranteil auch zu erwarten. Überraschend ist die starke Reduktion des mittleren kAE beim CSA um über 50% bei Rauschmodell II. Die kAE-Werte des PT1 sind bei beiden Rauschmodellen nahezu identisch. Das Verhältnis zwischen Mittelwert und Modus ändert sich zwischen Rauschmodell I und II bei den Methoden nicht signifikant. Ausnahme bildet hier wiederum der CSA. In Rauschmodell I deutet der im Vergleich zum Mittelwert größere Modus auf eine linksschiefe Verteilung der kAE-Werte hin. Abbildung A.10c unterstützt diese Vermutung. Bei Rauschmodell II hingegen ist der Mittelwert größer als der Modus. Dies deutet auf eine rechtschiefe Verteilung der kAE-Werte hin. Für Datensatz 5 ist in Rauschmodell II bei allen Methoden, einschließlich dem wahren Knotenvektor, ein deutlicher Anstieg der kAE-Werte um über 70% zu

erkennen. Ausnahme bilden die kAE-Werte bei PT1, die in Rauschmodell I und II nahezu identisch sind. Bei Datensatz 5 ist zwischen den verwendeten Rauschmodellen bei allen Methoden keine signifikante Änderung des Verhältnisses zwischen Mittelwert und Modus zu erkennen. Bei Datensatz 9 ergibt sich ein ähnliches Bild wie bei Datensatz 4. Mit Ausnahme des CSA steigen die kAE-Werte aller anderen Methoden in Rauschmodell II um 8 % bis über 70 % an. Besonders die mit EGAori, EGAmo und dem wahren Knotenvektor erzielten mittleren kAE-Werte steigen um über 70 %. Beim CSA hingegen ist eine signifikante Reduktion des mittleren kAE um knapp 45 % zu beobachten. Auch bei diesem Datensatz ändert sich nur beim CSA das Verhältnis zwischen Modus und Mittelwert signifikant. In Rauschmodell I deutete der größere Modus noch auf eine linksschiefe Verteilung der kAE-Werte hin. In Rauschmodell II sind Mittelwert und Modus nahezu identisch. Diese deutet auf eine symmetrische Verteilung der kAE-Werte hin.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell II verglichen. In der dritten Spalte sind die Ergebnisse der Kontrollpunktschätzung mit Hilfe des Huber-Schätzers bei Rauschmodell II dargestellt. Der mittlere kAE reduziert sich bei allen Verfahren im Vergleich zu den Ergebnissen des MdkQ-Schätzers bei Rauschmodell II durchschnittlich um ca. 15 %. Tendenziell reduziert sich der kAE prozentual besonders stark bei den Methoden der Knotenvektwahl, die bereits mit dem MdkQ-Schätzer in Rauschmodell I und II einen kleinen kAE erzielt haben. Hier sind Reduktionen von über 20 % zu beobachten. Das Verhältnis zwischen Modus und Mittelwert ändert sich durch die Verwendung des Huber-Schätzers gegenüber dem MdkQ-Schätzer lediglich beim CSA in Datensatz 4 signifikant. Unter Verwendung des Huber-Schätzers deutet die ähnliche Größenordnung von Mittelwert und Modus auf eine symmetrische Verteilung der kAE-Werte hin. Die vierte Spalte enthält die Ergebnisse der Kontrollpunktschätzung mit dem Hampel-Schätzer. Mit Ausnahme von PT1 bei den Datensätzen 4 und 5 ist bei allen Methoden und Datensätzen eine geringfügige Reduktion des kAE im Vergleich zum Huber-Schätzer zu erkennen. Durchschnittlich reduziert sich der kAE um weitere 5 % im Vergleich zum Huber-Schätzer. Im Maximum erfolgt eine Reduktion um über 7 %. Im Vergleich zu den Ergebnissen des MdkQ-Schätzers in Rauschmodell II erfolgt durchschnittlich eine Reduktion um ca. 18 %. Im Maximum erfolgt eine Reduktion um über 29 %. Das Verhältnis zwischen Modus und Mittelwert entspricht beim Hampel-Schätzer bei allen Datensätzen und Methoden in etwa dem Ergebnis des MdkQ-Schätzers in Rauschmodell II.

Tabelle 6.3: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell II (t-verteilte Daten)

DS.	Methode	RM I		RM II					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.520	54.500	53.385	53.402	55.473	55.413
	RIU	2.283	1.689	3.515	3.131	3.267	2.834	3.213	2.778
	CSA	8.027	8.873	3.850	2.787	3.449	3.315	3.293	2.664
	EMC	1.526	1.174	2.204	1.895	1.842	1.551	1.751	1.514
	EGAori	1.225	1.245	1.989	1.921	1.627	1.514	1.556	1.514
	EGAmo	1.097	1.081	1.874	1.864	1.504	1.474	1.437	1.401
	Wahr	1.018	1.006	1.750	1.606	1.347	1.309	1.276	1.178
5	PT1	110.249	110.240	110.271	110.313	99.395	99.411	102.279	102.714
	RIU	0.859	0.798	1.570	1.472	1.251	1.194	1.164	1.042
	CSA	0.892	0.952	1.510	1.595	1.181	1.081	1.092	1.032
	EMC	0.882	0.803	1.509	1.500	1.184	1.242	1.098	1.030
	EGAori	0.905	0.950	1.529	1.578	1.201	1.252	1.111	1.038
	EGAmo	0.903	0.972	1.528	1.547	1.199	1.235	1.107	1.046
	Wahr	0.790	0.704	1.344	1.328	1.045	0.988	0.975	0.956
9	PT1	3.900	3.878	4.220	4.154	4.031	3.974	3.975	3.951
	RIU	3.859	3.878	4.217	4.152	4.027	3.974	3.971	3.949
	CSA	11.127	12.446	6.148	6.273	5.961	6.236	5.939	6.206
	EMC	4.164	1.817	4.709	2.450	4.475	2.138	4.415	2.059
	EGAori	1.090	1.088	1.854	1.791	1.510	1.477	1.433	1.453
	EGAmo	1.075	1.056	1.847	1.769	1.502	1.462	1.425	1.447
	Wahr	0.954	0.929	1.657	1.501	1.261	1.257	1.171	1.147

Insgesamt liegen bei diesem Rauschmodell die kAE-Werte der verschiedenen Methoden unter dem kAE-Mittelwert der rohen Beobachtungen (siehe Tabelle 6.2). Ausnahme bildet PT1 bei Datensatz 4 und 5. Bei allen anderen Methoden ist auch unter Verwendung des MdkQ-Schätzers eine Steigerung der Richtigkeit gegenüber den rohen Beobachtungen festzustellen.

Rauschmodell III In Tabelle 6.4 sind die Kennziffern des kAE für Rauschmodell IIIa dargestellt. Bei Rauschmodell IIIa wurde das Rauschen aus der Gaußschen Mischverteilung generiert, wobei 1 % der Daten ein erhöhtes Rauschen erhalten. Aufgrund des vergleichsweise geringen Ausreißeranteils unterscheiden sich die Ergebnisse unter Verwendung des MdkQ-Schätzers zwischen Rauschmodell I und IIIa nur geringfügig.

Zunächst werden auch hier die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und IIIa verglichen. Bei Datensatz 4 ist in Rauschmodell IIIa mit Ausnahme von PT1 und CSA ein Anstieg der kAE-Werte von durchschnittlich ca. 14 % zu beobachten. Auch bei Rauschmodell IIIa liegt der mittlere kAE des CSA unter dem mit Rauschmodell I erzielten kAE-Wert (−5 %). Der kAE des PT1 liegt bei Rauschmodell I und IIIa nahezu auf gleichem Niveau. Das Verhältnis zwischen Mittelwert und Modus ändert sich zwischen Rauschmodell I und IIIa bei den Methoden nicht signifikant. Ausnahme bildet hier das Ergebnis des RIU. Während der im Vergleich zum Mittelwert kleinere Modus bei Rauschmodell I auf eine rechtsschiefe Verteilung hindeutet, liegen Mittelwert und Modus bei Rauschmodell IIIa deutlich näher zusammen. Dies deutet auf eine symmetrische Verteilung der kAE-Werte hin. Bei Datensatz 5 ist in Rauschmodell IIIa mit Ausnahme des PT1 bei allen Methoden ein Anstieg der mittleren kAE-Werte von durchschnittlich fast 13 % zu beobachten. Die

kAE-Werte des PT1 ändern sich im Vergleich zu Rauschmodell I nur geringfügig. Bei Datensatz 5 ist bei allen Methoden keine grundlegende Änderung im Verhältnis zwischen Mittelwert und Modus zu erkennen. Bei Datensatz 9 ist in Rauschmodell IIIa bei den Methoden EGAori, EGAmo und dem wahren Knotenvektor ein Anstieg der mittleren kAE-Werte von durchschnittlich über 7 % zu beobachten. Die kAE-Werte für PT1, RIU und EMC-Methode stagnieren bzw. steigen nur geringfügig an, während bei den kAE-Werten des CSA wiederum eine Reduktion der kAE-Werte gegenüber Rauschmodell I zu erkennen ist. Wie bei Datensatz 5 ist auch bei Datensatz 9 bei allen Methoden keine signifikante Veränderung im Verhältnis zwischen Mittelwert und Modus zu erkennen.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell IIIa verglichen. Bei Verwendung des Huber-Schätzers bei der Kontrollpunktschätzung reduziert sich der mittlere kAE bei allen Verfahren im Vergleich zum MdkQ-Schätzer in Rauschmodell IIIa durchschnittlich um über 5 %. Bei Verwendung des Hampel-Schätzers (vierte Spalte von Tabelle 6.4) erhöhen sich die kAE-Werte gegenüber dem Huber-Schätzer um durchschnittlich 3 %. Die kAE-Werte des Hampel-Schätzers liegen durchschnittlich knapp 3 % unter den Ergebnissen, die mit dem MdkQ-Schätzer in Rauschmodell IIIa erzielt wurden. Sowohl beim Huber- als auch beim Hampel-Schätzer ist bei allen Methoden und Datensätzen gegenüber dem MdkQ-Schätzer in Rauschmodell IIIa keine signifikante Veränderung im Verhältnis zwischen Mittelwert und Modus zu erkennen.

Tabelle 6.4: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIa (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 1 % der Messwerte)

DS.	Methode	RM I		RM IIIa					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.482	54.494	53.153	53.154	55.731	55.752
	RIU	2.283	1.689	2.925	2.758	2.873	2.702	2.892	2.679
	CSA	8.027	8.873	7.578	8.022	7.003	7.178	6.831	7.138
	EMC	1.526	1.174	1.642	1.323	1.540	1.259	1.562	1.299
	EGAori	1.225	1.245	1.352	1.346	1.268	1.279	1.316	1.333
	EGAmo	1.097	1.081	1.225	1.225	1.140	1.138	1.193	1.218
	Wahr	1.018	1.006	1.147	1.115	1.057	1.014	1.113	1.114
5	PT1	110.249	110.240	110.249	110.229	99.163	99.160	103.300	103.645
	RIU	0.859	0.798	1.005	0.943	0.932	0.866	0.974	0.861
	CSA	0.892	0.952	1.004	0.952	0.931	0.856	0.967	0.888
	EMC	0.882	0.803	0.991	0.959	0.922	0.895	0.959	0.966
	EGAori	0.905	0.950	1.012	0.979	0.941	0.922	0.977	0.857
	EGAmo	0.903	0.972	1.010	0.979	0.940	0.916	0.976	0.859
	Wahr	0.790	0.704	0.883	0.832	0.818	0.757	0.862	0.780
9	PT1	3.900	3.878	3.935	3.927	3.896	3.877	3.893	3.861
	RIU	3.859	3.878	3.887	3.929	3.847	3.883	3.846	3.863
	CSA	11.127	12.446	10.690	12.083	10.602	11.951	10.720	12.062
	EMC	4.164	1.817	4.205	1.846	4.148	1.792	4.180	1.805
	EGAori	1.090	1.088	1.174	1.144	1.096	1.070	1.140	1.093
	EGAmo	1.075	1.056	1.160	1.128	1.082	1.062	1.127	1.062
	Wahr	0.954	0.929	1.027	0.983	0.940	0.909	0.991	0.919

Die Ergebnisse bei Rauschmodell IIIb werden an dieser Stelle nicht erläutert, da die zu beobachtenden Effekte bei Rauschmodell IIIc deutlicher zu erkennen sind. Für die Ergebnisse bei Rauschmodell

IIIb sei deshalb auf Tabelle A.9 im Anhang verwiesen.

In Tabelle 6.5 sind die Kennziffern des kAE für Rauschmodell IIIc dargestellt. Bei Rauschmodell IIIc wurden das Rauschen mit der Gaußschen Mischverteilung generiert, wobei 10 % der Daten ein erhöhtes Rauschen erhalten. Aufgrund des erhöhten Ausreißeranteils bei Rauschmodell IIIc treten hier deutliche Unterschiede zwischen Rauschmodell I und IIIc auf.

Zunächst werden die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und IIIc verglichen. Bei Datensatz 4 ist bei RIU, EMC, EGAori, EGAmo und dem wahren Knotenvektor in Rauschmodell IIIc gegenüber Rauschmodell I ein Anstieg der kAE-Werte von durchschnittlich fast 70 % zu beobachten. Auch hier ist, im Gegensatz zu den anderen Methoden, beim CSA eine Reduktion der kAE-Werte gegenüber Rauschmodell I zu erkennen. Die Reduktion des mittleren kAE-Wertes beträgt 69 %(!). Die kAE-Werte bei PT1 sind bei Rauschmodell I und IIIc nahezu identisch. Das Verhältnis zwischen Mittelwert und Modus ändert sich nur bei den Methoden RIU und CSA signifikant. Beim RIU deuten Mittelwert und Modus bei Rauschmodell IIIc auf eine symmetrische Verteilung der kAE-Werte hin. In Rauschmodell I deuteten Mittelwert und Modus auf eine rechtsschiefe Verteilung hin. Beim CSA deuten Mittelwert und Modus, wie schon bei Rauschmodell II, auf eine rechtsschiefe Verteilung bei Rauschmodell IIIc hin. In Rauschmodell I deuteten Mittelwert und Modus auf eine linksschiefe Verteilung hin. Bei Datensatz 5 ist in Rauschmodell IIIc mit Ausnahme von PT1 bei allen Methoden ein Anstieg der kAE-Werte von durchschnittlich fast 85 % gegenüber Rauschmodell I zu erkennen. Auch hier ändert sich der kAE des PT1 im Vergleich zu Rauschmodell I nur geringfügig. Mit Ausnahme des PT1 ist bei allen Methoden in Rauschmodell IIIc der Modus geringfügig kleiner als der Mittelwert. Dies deutet auf eine leicht rechtsschiefe Verteilung der kAE-Werte hin. Bei Datensatz 9 ist in Rauschmodell IIIc mit Ausnahme des CSA bei allen Methoden ein Anstieg der kAE-Werte zu beobachten. Besonders stark steigen die kAE-Werte der Methoden EGAori und EGAmo sowie die mit dem wahren Knotenvektor erzielten Werte an. Hier steigt der mittlere kAE-Wert um durchschnittlich über 80 %. Bei den kAE-Werten des CSA ist wiederum eine Reduktion gegenüber Rauschmodell I zu erkennen. Hier beträgt die Reduktion im Mittelwert 61 %(!). Das Verhältnis zwischen Mittelwert und Modus ändert sich nur beim CSA signifikant. Wie schon bei Datensatz 4 beschrieben, deuten Mittelwert und Modus bei Rauschmodell IIIc auf eine rechtsschiefe Verteilung der kAE-Werte hin.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell IIIc verglichen. Bei Verwendung des Huber-Schätzers reduziert sich der mittlere kAE bei allen Verfahren im Vergleich zum MdkQ-Schätzer durchschnittlich um über 20 %. Besonders stark ist die Reduktion bei den Methoden EGAori und EGAmo sowie den kAE-Werten, die mit dem wahren Knotenvektor erzielt wurden. Hier reduzieren sich die mittleren kAE-Werte durchschnittlich um fast 30 %. Die Verwendung des Hampel-Schätzers verringert die kAE-Werte gegenüber dem Huber-Schätzer im Durchschnitt um weitere 2 %. Allerdings ist hier, insbesondere bei Datensatz 5, eine heterogene Entwicklung der kAE-Werte bei den verschiedenen Methoden zu erkennen. Der kAE-Werte des Hampel-Schätzers liegen durchschnittlich über 23 % unter denen, die mit dem MdkQ erzielt wurden. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei allen Methoden beim Einsatz der verschiedenen Schätzer in Rauschmodell IIIc nicht signifikant.

Tabelle 6.5: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIc (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 10 % der Messwerte)

DS.	Methode	RM I		RM IIIc					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.536	54.530	53.327	53.330	55.567	55.340
	RIU	2.283	1.689	3.498	3.325	3.108	2.931	3.054	2.919
	CSA	8.027	8.873	2.522	2.199	1.978	1.642	1.893	1.496
	EMC	1.526	1.174	2.336	2.075	1.782	1.529	1.693	1.501
	EGAori	1.225	1.245	2.125	2.135	1.568	1.578	1.505	1.478
	EGAmo	1.097	1.081	2.015	1.991	1.448	1.457	1.387	1.321
	Wahr	1.018	1.006	1.871	1.883	1.251	1.211	1.185	1.217
5	PT1	110.249	110.240	110.269	110.279	99.341	99.324	102.495	102.397
	RIU	0.859	0.798	1.743	1.535	1.234	1.003	1.146	0.981
	CSA	0.892	0.952	1.611	1.492	1.085	1.089	0.994	0.920
	EMC	0.882	0.803	1.609	1.544	1.097	1.082	1.178	0.944
	EGAori	0.905	0.950	1.629	1.497	1.106	1.104	1.183	0.963
	EGAmo	0.903	0.972	1.629	1.484	1.105	1.090	1.064	0.965
	Wahr	0.790	0.704	1.437	1.348	0.962	0.903	0.902	0.871
9	PT1	3.900	3.878	4.246	4.271	3.974	3.971	3.925	3.939
	RIU	3.859	3.878	4.243	4.272	3.970	3.961	3.921	3.935
	CSA	11.127	12.446	4.286	3.604	3.951	3.432	3.938	3.405
	EMC	4.164	1.817	4.712	2.453	4.369	1.979	4.318	1.922
	EGAori	1.090	1.088	1.959	2.041	1.457	1.459	1.398	1.361
	EGAmo	1.075	1.056	1.949	2.051	1.447	1.383	1.389	1.361
	Wahr	0.954	0.929	1.726	1.574	1.147	1.144	1.076	0.984

Insgesamt liegen bei den Rauschmodellen IIIa bis IIIc die kAE-Werte der verschiedenen Methoden unter dem kAE-Mittelwert der rohen Beobachtungen (siehe Tabelle 6.2). Ausnahme bildet wiederum PT1 bei Datensatz 4 und 5. Bei allen anderen Methoden ist auch unter Verwendung des MdkQ-Schätzers eine Steigerung der Richtigkeit gegenüber den rohen Beobachtungen festzustellen.

Rauschmodell IV Tabelle 6.6 enthält die Kennziffern des kAE für Rauschmodell IVa. Bei Rauschmodell IVa wurden 1 % der Daten mit einem einseitigen Rauschen belegt. Aufgrund des vergleichsweise geringen Ausreißeranteils unterscheiden sich die Ergebnisse unter Verwendung des MdkQ-Schätzers zwischen Rauschmodell I und IVa nur geringfügig.

Zunächst werden auch hier die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und IVa verglichen. Bei Datensatz 4 ist mit Rauschmodell IVa ein Anstieg der mittleren kAE-Werte von durchschnittlich über 10 % zu beobachten. Die mittleren kAE-Werte bei PT1 und CSA sind gegenüber Rauschmodell I nahezu unverändert. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei Rauschmodell IVa beim RIU und CSA signifikant. Bei beiden Methoden deuten Mittelwert und Modus auf eine deutlich symmetrischere Verteilung hin als bei Rauschmodell I. Bei Datensatz 5 ist mit Rauschmodell IVa ein durchschnittlicher Anstieg der mittleren kAE-Werte von über 10 % zu beobachten. Der kAE des PT1 bleibt im Vergleich zu Rauschmodell I nahezu unverändert. Das Verhältnis zwischen Mittelwert und Modus ändert sich zwischen Rauschmodell I und IVa nur geringfügig. Bei Datensatz 9 ist in Rauschmodell IVa bei den Methoden EGAori, EGAmo und dem wahren Knotenvektor ein Anstieg der kAE-Werte von durchschnittlich ca. 12 % zu beobachten. Die kAE-Werte für PT1, RIU, CSA und EMC-Methode ändern sich im Vergleich zu Rauschmodell I nur

geringfügig. Beim CSA und der EMC-Methode ist sogar eine geringfügig Reduktion der mittleren kAE-Werte zu beobachten. Wie bei Datensatz 5 ändert sich das Verhältnis zwischen Mittelwert und Modus von Rauschmodell I zu IVa nur geringfügig.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell IVa verglichen. Bei Verwendung des Huber-Schätzers reduziert sich der mittlere kAE bei allen Verfahren im Vergleich zum MdkQ-Schätzer durchschnittlich um ca. 5%. Bei Verwendung des Hampel-Schätzers erhöhen sich die kAE-Werte gegenüber dem Huber-Schätzer geringfügig um durchschnittlich 1.6%. Die kAE-Werte des Hampel-Schätzers liegen durchschnittlich 3.5% unter denen, die mit dem MdkQ-Schätzer erzielt wurden. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei allen Methoden beim Einsatz der verschiedenen Schätzer in Rauschmodell IVa nicht signifikant. Insgesamt ergibt sich hier also ein sehr ähnliches Bild wie bei Rauschmodell IIIa.

Tabelle 6.6: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVa (1% einseitige Ausreißer, zufällige Position)

DS.	Methode	RM I		RM IVa					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.476	54.473	53.160	53.135	55.734	55.642
	RIU	2.283	1.689	2.941	2.841	2.893	2.777	2.893	2.753
	CSA	8.027	8.873	8.004	8.194	7.401	7.399	7.215	7.131
	EMC	1.526	1.174	1.636	1.356	1.537	1.254	1.529	1.291
	EGAori	1.225	1.245	1.367	1.384	1.287	1.305	1.304	1.232
	EGAmo	1.097	1.081	1.244	1.264	1.162	1.181	1.182	1.144
	Wahr	1.018	1.006	1.171	1.229	1.085	1.035	1.106	1.111
5	PT1	110.249	110.240	110.256	110.267	99.169	99.167	103.233	103.444
	RIU	0.859	0.798	0.995	0.941	0.938	0.915	0.962	0.931
	CSA	0.892	0.952	0.994	0.952	0.941	0.913	0.962	0.891
	EMC	0.882	0.803	0.983	0.939	0.931	0.881	0.952	0.911
	EGAori	0.905	0.950	1.005	0.949	0.952	0.918	0.971	0.937
	EGAmo	0.903	0.972	1.002	0.981	0.950	0.896	0.972	0.949
	Wahr	0.790	0.704	0.884	0.884	0.831	0.798	0.855	0.786
9	PT1	3.900	3.878	3.941	3.937	3.907	3.920	3.896	3.885
	RIU	3.859	3.878	3.909	3.938	3.875	3.918	3.865	3.886
	CSA	11.127	12.446	10.982	12.238	10.894	12.112	11.024	12.300
	EMC	4.164	1.817	4.158	1.886	4.104	1.825	4.130	1.864
	EGAori	1.090	1.088	1.214	1.189	1.139	1.111	1.163	1.161
	EGAmo	1.075	1.056	1.197	1.149	1.121	1.102	1.146	1.122
	Wahr	0.954	0.929	1.083	1.062	0.998	0.942	1.027	1.005

Die Ergebnisse bei Rauschmodell IVb werden an dieser Stelle, wie schon bei Rauschmodell IIIb, nicht erläutert, da die zu beobachtenden Effekte bei Rauschmodell IVc deutlicher zu erkennen sind. Für die Ergebnisse bei Rauschmodell IVb sei deshalb auf Tabelle A.10 im Anhang verwiesen.

In Tabelle 6.7 sind die Kennziffern des kAE für Rauschmodell IVc dargestellt. Bei Rauschmodell IVc wurden 10% der Daten mit einem einseitigen Rauschen belegt. Aufgrund des erhöhten Ausreißeranteils und des einseitigen Verrauschens der Ausreißer bei Rauschmodell IVc treten hier sehr deutliche Unterschiede zu Rauschmodell I auf.

Zunächst werden auch hier die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und IVc verglichen. Bei Datensatz 4 ist in Rauschmodell IVc ein Anstieg der mittleren kAE-Werte von

durchschnittlich über 200 % zu beobachten. PT1 und CSA steigen nur in geringem Maße an. Das Verhältnis zwischen Mittelwert und Modus unterscheidet sich bei allen Methoden zwischen Rauschmodell I und IVc nur geringfügig. Bei Datensatz 5 ist ein durchschnittlicher Anstieg der mittleren kAE-Werte von über 250 % zu beobachten. Die kAE-Werte des PT1 bleiben gegenüber Rauschmodell I nahezu unverändert. Wie schon bei Datensatz 4 unterscheidet sich das Verhältnis zwischen Mittelwert und Modus bei Rauschmodell I und IVc nur geringfügig. Bei Datensatz 9 ist bei allen Methoden ein Anstieg der mittleren kAE-Werte von durchschnittlich über 190 % zu beobachten. Bei den kAE-Werten des CSA ist wiederum eine Reduktion der kAE-Werte gegenüber Rauschmodell I zu erkennen. Hier beträgt die Reduktion im Mittelwert 5 %. Die mittleren kAE-Werte bei EGAori und EGAmo sowie dem wahren Knotenvektor steigen durchschnittlich um ca. 385 %. Wie schon bei den Datensätzen zuvor unterscheidet sich auch bei Datensatz 9 das Verhältnis zwischen Mittelwert und Modus bei Rauschmodell I und IVc nur geringfügig.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell IVc verglichen. Bei Verwendung des Huber-Schätzers reduziert sich der mittlere kAE bei allen Verfahren im Vergleich zum MdkQ-Schätzer durchschnittlich um ca. 16 %. Die Verwendung des Hampel-Schätzers verringert die kAE-Werte gegenüber dem Huber-Schätzer im Durchschnitt um weitere 38 %. Bei PT1 kommt es bei den Datensätzen 4 und 5 zu einem leichten Anstieg. Die kAE-Werte des Hampel-Schätzers liegen durchschnittlich ca. 47 % unter denen, die mit dem MdkQ-Schätzer erzielt wurden. Die mit dem Hampel-Schätzer erzielten kAE-Werte liegen durchschnittlich nur knapp 40 % über den mit dem MdkQ-Schätzer in Rauschmodell I erzielten Werten. Das Verhältnis zwischen Mittelwert und Modus ändert sich lediglich bei Verwendung des Hampel-Schätzers und des CSA in Datensatz 4 signifikant. Mittelwert und Modus deuten an dieser Stelle auf eine rechtsschiefe Verteilung hin.

Tabelle 6.7: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVc (10% einseitige Ausreißer, zufällige Position)

DS.	Methode	RM I		RM IVc					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.914	54.915	53.774	53.755	55.868	55.837
	RIU	2.283	1.689	6.030	5.680	5.174	4.738	3.615	3.092
	CSA	8.027	8.873	8.302	9.009	7.443	8.142	5.891	5.539
	EMC	1.526	1.174	5.220	5.060	4.125	4.017	2.121	1.942
	EGAori	1.225	1.245	5.101	5.013	3.986	3.958	1.940	1.871
	EGAmo	1.097	1.081	5.031	4.945	3.897	3.897	1.824	1.807
	Wahr	1.018	1.006	5.003	4.920	3.855	3.851	1.733	1.658
5	PT1	110.249	110.240	110.346	110.355	99.543	99.544	102.355	102.993
	RIU	0.859	0.798	3.511	3.436	2.754	2.724	1.385	1.331
	CSA	0.892	0.952	3.494	3.490	2.737	2.726	1.367	1.374
	EMC	0.882	0.803	3.497	3.465	2.736	2.708	1.365	1.305
	EGAori	0.905	0.950	3.503	3.484	2.746	2.722	1.382	1.360
	EGAmo	0.903	0.972	3.503	3.484	2.745	2.725	1.381	1.357
	Wahr	0.790	0.704	3.441	3.391	2.664	2.566	1.266	1.244
9	PT1	3.900	3.878	6.253	6.187	5.530	5.406	4.254	4.148
	RIU	3.859	3.878	6.247	6.179	5.523	5.405	4.244	4.139
	CSA	11.127	12.446	10.556	11.251	10.238	10.976	9.326	10.070
	EMC	4.164	1.817	7.017	5.299	6.251	4.229	4.742	2.371
	EGAori	1.090	1.088	5.032	4.989	3.904	3.937	1.831	1.902
	EGAmo	1.075	1.056	5.030	4.980	3.902	3.943	1.825	1.904
	Wahr	0.954	0.929	4.998	4.970	3.849	3.871	1.687	1.783

Insgesamt liegen bei den Rauschmodellen IVa bis IVc die kAE-Werte der verschiedenen Methoden unter dem kAE-Mittelwert der rohen Beobachtungen (siehe Tabelle 6.2). Ausnahme bildet wiederum PT1 bei Datensatz 4 und 5. Bei allen anderen Methoden ist auch unter Verwendung des MdkQ-Schätzers eine Steigerung der Richtigkeit gegenüber den rohen Beobachtungen festzustellen. Allerdings ist bei Rauschmodell IVc eine prozentual deutlich höhere Steigerung der kAE-Werte bei der Approximation unter Verwendung des MdkQ-Schätzers gegenüber den kAE-Werte der rohen Beobachtungen zu beobachten. Insbesondere fällt dies bei Verwendung von EGAori, EGAmo und dem wahren Knotenvektor auf.

Rauschmodell V In Tabelle 6.8 sind die Kennziffern des kAE für Rauschmodell Va dargestellt. Bei Rauschmodell Va wurden 1% der Daten mit einem einseitigen Rauschen belegt. Alle Ausreißer folgen dabei nacheinander. Trotz des vergleichsweise geringen Ausreißeranteils unterscheiden sich die Ergebnisse unter Verwendung des MdkQ-Schätzers zwischen Rauschmodell I und Va deutlich. Zunächst werden die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und Va verglichen. Bei Datensatz 4 ist bei Rauschmodell Va ein Anstieg der kAE-Werte von durchschnittlich ca. 39% zu beobachten. Die kAE-Werte bei PT1 und CSA steigen gegenüber Rauschmodell I nur geringfügig. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei allen Methoden nur geringfügig. Bei Datensatz 5 ist bei Rauschmodell Va ein durchschnittlicher Anstieg der mittleren kAE-Werte von ca. 38% zu beobachten. Die kAE-Werte des PT1 ändern sich im Vergleich zu Rauschmodell I wiederum nur geringfügig. Auch hier kommt es zu keinen signifikanten Änderungen im Verhältnis zwischen Mittelwert und Modus. Bei Datensatz 9 ist in Rauschmodell Va bei allen Methoden ein

Anstieg der kAE-Werte von durchschnittlich ca. 30 % zu beobachten. Besonders stark steigen die kAE-Werte bei den Methoden EGAori und EGAmo sowie dem wahren Knotenvektor. Die mittleren kAE-Werte steigen hier durchschnittlich um über 60 %. Die kAE-Werte für den CSA ändern sich im Vergleich zu Rauschmodell I nur geringfügig. Wie bei den vorherigen Datensätzen kommt es auch bei Datensatz 9 zu keinen signifikanten Änderungen im Verhältnis zwischen Mittelwert und Modus.

Im Folgenden werden die Ergebnisse der verschiedenen Schätzer bei Rauschmodell Va verglichen. Bei Verwendung des Huber-Schätzers reduziert sich der mittlere kAE bei allen Verfahren im Vergleich zum MdkQ-Schätzer durchschnittlich um ca. 9 %. Bei Verwendung des Hampel-Schätzers reduzieren sich die kAE-Werte gegenüber dem Huber-Schätzer zusätzlich um durchschnittlich fast 10 %. Die kAE-Werte des Hampel-Schätzers liegen durchschnittlich ca. 18 % unter denen, die mit dem MdkQ-Schätzer in Rauschmodell Va erzielt wurden. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei allen Methoden beim Einsatz der verschiedenen Schätzer in Rauschmodell Va nicht signifikant.

Tabelle 6.8: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Va (1 % einseitige Ausreißer, nacheinander)

DS.	Methode	RM I		RM Va					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.535	54.560	53.229	53.200	55.789	55.641
	RIU	2.283	1.689	3.376	2.945	3.184	2.747	2.998	2.619
	CSA	8.027	8.873	8.278	8.982	7.603	7.997	7.281	7.825
	EMC	1.526	1.174	2.135	1.854	1.886	1.569	1.633	1.335
	EGAori	1.225	1.245	1.881	1.850	1.653	1.590	1.423	1.337
	EGAmo	1.097	1.081	1.757	1.767	1.526	1.498	1.298	1.219
	Wahr	1.018	1.006	1.693	1.721	1.423	1.457	1.148	1.089
5	PT1	110.249	110.240	110.254	110.270	99.194	99.196	103.300	103.803
	RIU	0.859	0.798	1.314	1.301	1.146	1.105	0.982	0.925
	CSA	0.892	0.952	1.336	1.303	1.181	1.145	1.002	0.966
	EMC	0.882	0.803	1.321	1.247	1.169	1.126	1.011	0.939
	EGAori	0.905	0.950	1.349	1.303	1.198	1.167	1.019	0.962
	EGAmo	0.903	0.972	1.350	1.298	1.200	1.183	1.021	0.960
	Wahr	0.790	0.704	1.170	1.214	1.001	1.038	0.871	0.837
9	PT1	3.900	3.878	4.200	4.213	4.064	4.054	3.930	3.919
	RIU	3.859	3.878	4.200	4.214	4.064	4.054	3.931	3.920
	CSA	11.127	12.446	11.212	12.590	11.090	12.429	11.181	12.674
	EMC	4.164	1.817	4.613	2.322	4.464	2.128	4.341	1.953
	EGAori	1.090	1.088	1.730	1.755	1.528	1.489	1.346	1.246
	EGAmo	1.075	1.056	1.722	1.663	1.520	1.489	1.338	1.257
	Wahr	0.954	0.929	1.575	1.550	1.300	1.298	1.043	0.974

Tabelle 6.9 enthält die Kennziffern des kAE für Rauschmodell Vb. Bei Rauschmodell Vb wurden 5 % der Daten mit einem einseitigen Rauschen belegt. Alle Ausreißer folgen dabei nacheinander. Mit dem erhöhten Ausreißeranteils unterscheiden sich die Ergebnisse unter Verwendung des MdkQ-Schätzers zwischen Rauschmodell I und Vb sehr deutlich.

Zunächst werden die Ergebnisse des MdkQ-Schätzers zwischen Rauschmodell I und Vb verglichen. Bei allen Datensätzen ist in Rauschmodell Vb ein Anstieg der kAE-Werte von durchschnittlich

ca. 233% zu beobachten. Lediglich die kAE-Werte beim PT1 steigen bei Datensatz 4 und 5 im Vergleich zu Rauschmodell I nur geringfügig an. Das Verhältnis zwischen Mittelwert und Modus ändert sich trotz des starken Anstiegs der kAE-Werte nur geringfügig. Lediglich bei den Methoden CSA und EMC kommt es in Datensatz 4 zu einer leichten Annäherung von Mittelwert und Modus, was für eine symmetrische Verteilung der kAE-Werte spricht.

Bei Verwendung des Huber- und Hampel-Schätzers bei der Kontrollpunktschätzung kommt es zu keiner signifikanten Änderung der kAE-Werte gegenüber dem MdkQ-Schätzer. Dieses Rauschmodell führt zu einem Versagen der beiden untersuchten robusten Schätzer. Das Verhältnis zwischen Mittelwert und Modus ändert sich bei Verwendung des Huber-Schätzers beim CSA in Datensatz 4 und 9. In beiden Fällen liegen Mittelwert und Modus in einer ähnlichen Größenordnung und deuten somit eine symmetrische Verteilung der kAE-Werte an. Bei Verwendung des Hampel-Schätzers deuten wachsende Diskrepanzen zwischen Mittelwert und Modus auf asymmetrische Verteilungen der kAE-Werte hin. Insbesondere bei den in Methoden EGAori und EGAmo sowie dem wahren Knotenvektor sind diese Diskrepanzen offensichtlich.

Tabelle 6.9: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Vb (5% einseitige Ausreißer, nacheinander)

DS.	Methode	RM I		RM Vb					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	55.331	55.320	54.132	54.066	57.342	57.129
	RIU	2.283	1.689	6.356	6.237	6.492	6.353	6.918	6.877
	CSA	8.027	8.873	10.450	10.842	10.154	10.005	10.529	10.615
	EMC	1.526	1.174	5.413	5.319	5.541	5.410	6.090	6.252
	EGAori	1.225	1.245	5.223	5.269	5.368	5.340	5.947	6.227
	EGAmo	1.097	1.081	5.124	5.094	5.271	5.260	5.854	6.120
	Wahr	1.018	1.006	5.199	5.244	5.353	5.417	5.839	6.170
5	PT1	110.249	110.240	110.409	110.481	99.489	99.206	104.723	105.071
	RIU	0.859	0.798	3.567	3.465	3.626	3.540	3.561	4.267
	CSA	0.892	0.952	3.611	3.657	3.695	3.630	4.086	4.180
	EMC	0.882	0.803	3.597	3.576	3.685	3.632	4.074	4.232
	EGAori	0.905	0.950	3.615	3.652	3.697	3.658	4.082	4.212
	EGAmo	0.903	0.972	3.612	3.649	3.694	3.649	4.076	4.204
	Wahr	0.790	0.704	3.385	3.333	3.424	3.460	2.892	2.178
9	PT1	3.900	3.878	6.939	7.184	6.982	7.188	6.780	7.541
	RIU	3.859	3.878	6.955	7.116	7.006	7.147	6.893	7.415
	CSA	11.127	12.446	13.032	14.135	13.022	13.195	13.317	13.383
	EMC	4.164	1.817	7.543	5.687	7.621	5.772	7.784	6.390
	EGAori	1.090	1.088	5.274	5.263	5.383	5.326	5.514	6.011
	EGAmo	1.075	1.056	5.269	5.250	5.379	5.317	5.508	6.034
	Wahr	0.954	0.929	5.197	5.239	5.289	5.301	4.843	6.024

Die Ergebnisse für Rauschmodell Vc werden an dieser Stelle nicht erläutert, da die verwendeten robusten Schätzer bereits bei Rauschmodell Vb versagt haben und somit keine weiteren Erkenntnisse aus den Ergebnissen für Rauschmodell Vc gezogen werden können. Für die Ergebnisse bei Rauschmodell Vc sei deshalb auf Tabelle A.11 im Anhang verwiesen.

Insgesamt liegen bei den Rauschmodellen Va bis Vc die kAE-Werte der verschiedenen Methoden unter dem kAE-Mittelwert der rohen Beobachtungen (siehe Tabelle 6.2). Ausnahme bildet wie-

derum PT1 bei Datensatz 4 und 5. Bei allen anderen Methoden ist auch unter Verwendung des MdkQ-Schätzers eine Steigerung der Richtigkeit gegenüber den rohen Beobachtungen festzustellen. Allerdings ist hier bei den Rauschmodellen Va bis Vc eine prozentual deutlich höhere Steigerung der kAE-Werte bei der Approximation gegenüber den kAE-Werte der rohen Beobachtungen zu beobachten. Insbesondere fällt dies bei Verwendung von EGAori, EGAmo und dem wahren Knotenvektor auf.

6.2.1 Zusammenfassung

Folgende Erkenntnisse lassen sich aus Kapitel 6.2 gewinnen:

- Bei den Rauschmodellen II bis V ist ein Effekt auf die Kontrollpunktschätzung nach der Methode der kleinsten Quadrate zu erkennen. Dies ist zu erwarten, da dieser Schätzer schon mit dem ersten Ausreißer versagt. Anders ausgedrückt liegt der Bruchpunkt (siehe hierzu auch Kapitel 6.4) der Methode der kleinsten Quadrate bekanntlich bei 0 %.
- Bei einseitig verrauschten Daten ist der kAE bei Verwendung der Methode der kleinsten Quadrate in der Kontrollpunktschätzung sehr viel größer als bei normalverteilten Daten. Bei einseitigen aber zufällig verteilten Ausreißern (10 %, Rauschmodell IVc) liegt der kAE teilweise mehr als 400 % über den Werten, die bei normalverteilten Daten erzielt wurden. Bei einseitigen nacheinander folgenden Ausreißern (10 % in Rauschmodell Vc) werden sogar kAE-Steigerungen um das 9-fache erreicht. Grundsätzlich sind deutliche Steigerungen bei den einseitig verrauschten Daten zu erwarten. Prozentual liegen dies Steigerungen deutlich über den Steigerungen bei den kAE-Mittelwerten der rohen Beobachtungen.
- Die Methoden der Knotenvektorstimmung zeigen ein unterschiedliches Verhalten bei den Rauschmodellen. Methoden, die bei normalverteilten Daten einen Knotenvektor nah am wahren Knotenvektor erzeugen, erfahren einen prozentual stärkeren Anstieg des kAE, wenn ein anderes Rauschmodell verwendet wird. Methoden, die bei normalverteilten Daten einen weit vom wahren Knotenvektor entfernten Knotenvektor erzeugen, erzielen bei anderen Rauschmodellen zum Teil stagnierende oder geringere kAE-Werte. Besonders ist hier der CSA herauszustellen, bei dem zum Teil die kAE-Werte mit steigender Ausreißerzahl fallen. In diesen Fällen ist auch häufig eine signifikante Änderung der Verteilung der kAE-Werte zu beobachten.
- Mit Ausnahme des PT1 bei Datensatz 4 und 5 führt die Approximation auch unter Verwendung des MdkQ-Schätzers mit allen Methoden zu einer Steigerung der Richtigkeit gegenüber den rohen Beobachtungen.
- Die Verwendung robuster Schätzer führt bei den meisten untersuchten Rauschmodellen zu einer deutlichen Reduktion der kAE-Werten im Vergleich zur Methode der kleinsten Quadrate. Es sind Reduktionen beim kAE von über 60 % zu beobachten. Lediglich in den Rauschmodellen Vb und Vc führt die Verwendung der robusten Schätzer zu keiner nennenswerten Reduktion der kAE-Werte. Bei diesen beiden Rauschmodellen versagen die beiden robusten Schätzer.
- Bei einseitiger Verrauschung oder einem hohen Ausreißeranteil liefert der Hampel-Schätzer gegenüber dem Huber-Schätzer deutlich kleinere kAE-Werte (siehe Tabelle 6.5, 6.7 und 6.8). Bei geringeren Prozentsätzen von Ausreißern liefert der Huber-Schätzer geringfügig kleinere kAE-Werte als der Hampel-Schätzer (siehe Tabelle 6.4 und 6.6). Evtl. ist dies durch den Effizienzverlust des Hampel-Schätzers zu erklären, der durch die größere Robustheit in Kauf genommen wird (siehe Jäger (2005, S.108)).
- Die Auswirkungen der robusten Schätzer gelten generell bei allen untersuchten Datensätzen. Es gibt aber deutliche Unterschiede je nach verwendeter Methode zur Knotenvektorstimmung.

Festzuhalten ist hier, dass unter Verwendung von EGAori oder EGAmo d kAE-Werte erzielt werden, die sich bei den verschiedenen Rauschmodellen und Schätzern ähnlich verhalten wie die kAE-Werte des wahren Knotenvektors. In Kapitel 6.4 wird anhand des wahren Knotenvektors die Belastbarkeit der verschiedenen Schätzer näher untersucht.

6.3 Sensitivitätsanalyse der Kontrollpunktschätzung bei ausreißerbehafteten Datensätzen

In diesem Kapitel wird der Einfluss der ausreißerbehafteten Datensätzen auf die Kontrollpunktschätzung analysiert. Dazu wird eine vergleichbare Darstellungsform wie in Kapitel 5.4 gewählt. Um den Einfluss der ausreißerbehafteten Datensätze vom Einfluss des Knotenvektors zu trennen, werden die unter Verwendung des wahren Knotenvektors erzielten Ergebnisse dargestellt. Auch hier wird der äquidistante Ortsparametervektor $\bar{\mathbf{u}}$ aus der Simulation verwendet. Folglich wird die Parametrisierung nicht durch die Ausreißer beeinflusst.

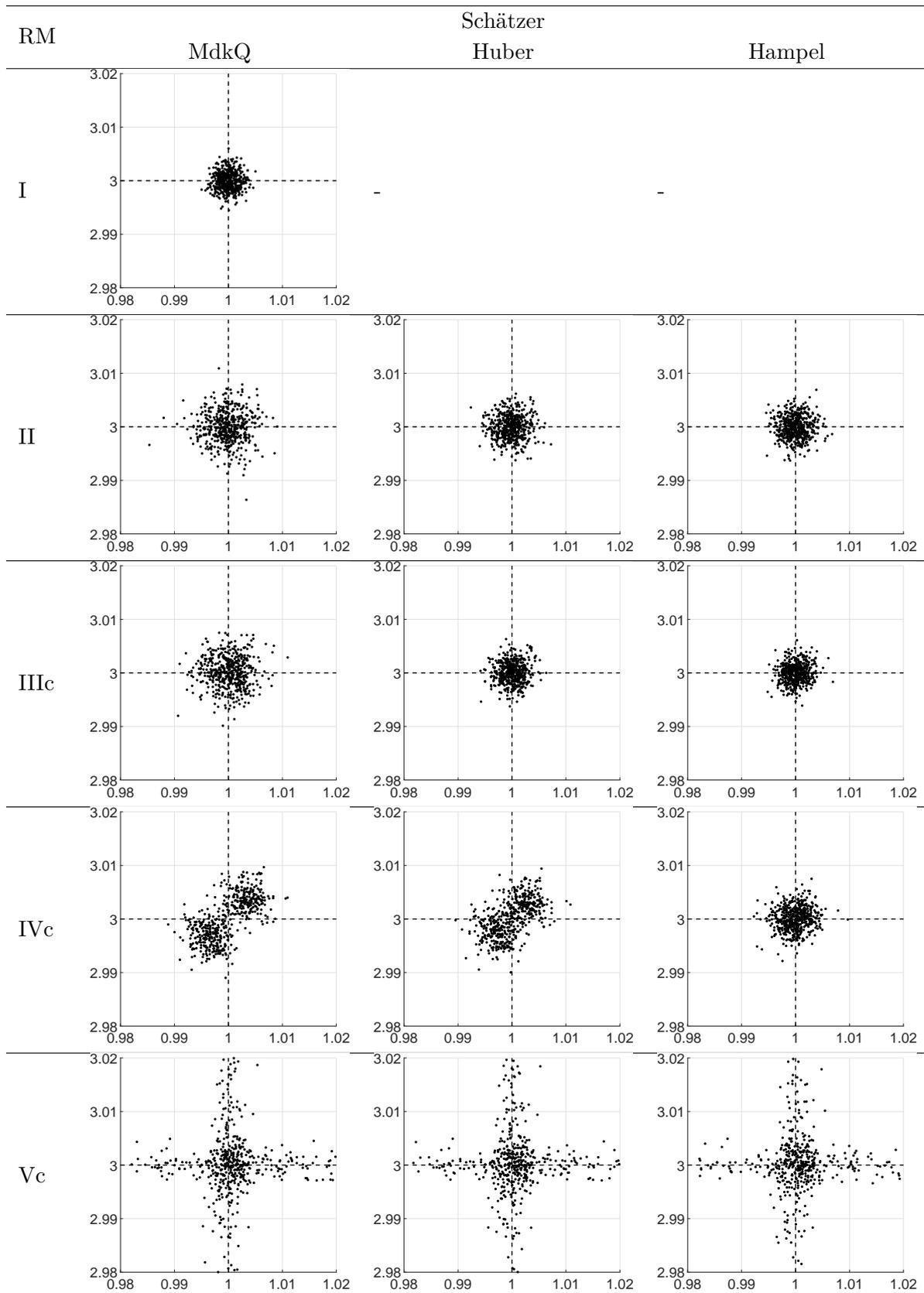
In Tabelle 6.10 ist exemplarisch die Lage des geschätzten Kontrollpunktes \mathbf{x}_1 (schwarz) in Datensatz 9 bei verschiedenen Rauschmodellen und Schätzern dargestellt.

Bei Rauschmodell I wurde lediglich die Methode der kleinsten Quadrate zur Schätzung angewandt. Das in der ersten Zeile von Tabelle 6.10 dargestellte Ergebnis entspricht dem Ergebnis in Tabelle 5.12. Dieses Ergebnis dient fortan als Vergleich.

Bei Rauschmodell II ist bei Verwendung des MdkQ-Schätzers eine deutlich höhere Streuung der geschätzten Kontrollpunktpositionen als bei Rauschmodell I zu erkennen. Diese Streuung wird durch Verwendung des Huber- und Hampel-Schätzers signifikant verringert. Zwischen Huber- und Hampel-Schätzer ist optisch kein signifikanter Unterschied zu erkennen.

Die Ergebnisse bei Rauschmodell IIIc sind rein optisch nicht von den Ergebnissen Rauschmodell II zu unterscheiden.

Tabelle 6.10: Lage des geschätzten Kontrollpunktes \mathbf{x}_1 (schwarz) in Datensatz 9 bei verschiedenen Rauschmodellen und Schätzern unter Verwendung des wahren Knotenvektors



Scheinbar erzeugen beide Rauschmodelle ein Rauschen in ähnlicher Größenordnung. Die Ergebnisse bei Rauschmodell IVc unterscheiden sich deutlich von den vorherigen Ergebnissen. Bei Verwendung des MdkQ-Schätzers ist die Charakteristik des Rauschmodells deutlich zu erkennen. Durch das einseitig positive oder negative Rauschen, angebracht auf beide Koordinatenkomponenten, ergeben sich zwei Schwerpunkte bei $[0.996, 2.996]$ und $[1.004, 3.004]$. Bei Verwendung des Huber-Schätzers verschwimmen die beiden Schwerpunkte. Bei Verwendung des Hampel-Schätzers entspricht die Verteilung des Kontrollpunktes beinahe der bei Rauschmodell I. Die Ergebnisse von Rauschmodell Vc weisen ebenfalls die Charakteristik des Rauschmodells auf. Mit dem einseitig positiv oder negativen Rauschen wird hier nur eine Koordinatenkomponente überlagert. Folglich weist die Verteilung des geschätzten Kontrollpunktes eine kreuzförmige Struktur auf. Zwischen den verschiedenen Schätzern ist bei diesem Datensatz kein signifikanter optischer Unterschied zu erkennen. Wie in Kapitel 6.2 erwähnt, versagen die robusten Schätzer bei diesem Rauschmodell.

6.4 Belastbarkeit der Schätzer

Ein quantitatives Maß für die Robustheit eines Schätzers ist der *Bruchpunkt*. So definiert Caspary (2013, S.18f.) den *Bruchpunkt* wie folgt:

Definition 1. *Der Bruchpunkt ist [...] der größte Prozentsatz an Ausreißern in der Stichprobe, den ein robuster Schätzer verkraften kann, d.h. trotz dessen Existenz er brauchbare Ergebnisse liefert.*

Im Rahmen dieser Arbeit wird nicht der theoretische *Bruchpunkt* der Schätzer untersucht³. Stattdessen sollen die Grenzen der *Belastbarkeit* der drei Schätzer bei realistischen Datensätzen evaluiert werden. Die *Belastbarkeit* soll als empirische Annäherung an den theoretischen *Bruchpunkt* verstanden werden und dient zur Quantifizierung der Robustheit der drei Schätzer. In Anlehnung an die oben genannte Definition des *Bruchpunktes* wird im Rahmen dieser Arbeit als Erreichen der Grenze der *Belastbarkeit* Folgendes definiert:

Definition 2. *Die Grenze der Belastbarkeit des Schätzers ist der Prozentsatz an Ausreißern, bei dem der mittlere kAE des untersuchten Rauschmodells und Schätzers die obere Konfidenzintervallgrenze (95%) des kAE bei der MdkQ-Schätzung im ausreißerfreien Rauschmodell I überschreitet.*

Die *Belastbarkeit* der drei Schätzer MdkQ, Huber und Hampel soll anhand verschiedener simulierter Datensätze bestimmt werden. Dabei wird als Knotenvektor der wahre Knotenvektor (der Simulation) verwendet. Aus diesem Grunde werden nur die in Kapitel 5.1.2 beschriebenen Datensätze 4 bis 9 verwendet. Eine umfangreiche Untersuchung mit allen heuristischen Verfahren ist äußerst rechenintensiv und erlaubt außerdem keine Aussage über die *Belastbarkeit* der Schätzer, sondern nur über die *Belastbarkeit* der Kombination aus Knotenvektormethode und Schätzer. Zudem haben die Ergebnisse in Kapitel 6.2 gezeigt, dass insbesondere die Knotenvektormethoden EGAori und EGAmo vergleichbare Ergebnisse wie der wahre Knotenvektor erzielen.

In diesem Kapitel werden die im Kapitel 6.1 beschriebenen Rauschmodelle III bis V verwendet. Rauschmodell I wurde nicht verwendet, da angenommen wird, dass es keine Ausreißer enthält. Rauschmodell II wurde nicht verwendet, da die Anzahl der Ausreißer bzw. Anzahl der Messdaten mit höherem Rauschen nicht exakt steuerbar ist.

Der Prozentsatz der Ausreißer in den Rauschmodellen III und IV wurde dabei zwischen 0% und

³Es gibt mit Sicherheit Datensätze, bei denen Huber- und Hampel-Schätzer bei der B-Spline-Approximation mit dem ersten Ausreißer versagen. Z.B., wenn der Ausreißer in einer Hebelbeobachtung liegt und deshalb nicht aufgedeckt wird.

30 % in 1 %-Schritten variiert. Für Rauschmodell V wurde der Prozentsatz nur zwischen 0 % und 10 % ebenfalls in 1 %-Schritten variiert. Für jeden Prozentsatz im jeweiligen Rauschmodell wurden so 500 Datensätze generiert und anschließend approximiert. Bei 0 % Ausreißern entspricht das simulierte Rauschen dem normalverteilten Rauschen aus Rauschmodell I.

Die Ergebnisse der drei untersuchten Schätzer MdkQ, Huber und Hampel sind dabei in unterschiedlichen Farben (MdkQ: rot, Huber: blau, Hampel: grün) dargestellt. Die Darstellung des Mittelwertes der kAE-Werte erfolgt linienhaft. Die Konfidenzintervalle der kAE-Werte des jeweiligen Schätzers werden flächenhaft und leicht transparent dargestellt. Zusätzlich sind untere und obere Grenze des Konfidenzintervalls der kAE-Werte des MdkQ-Schätzers bei Rauschmodell I als schwarze Linien eingezeichnet. Nach der oben gegebenen Definition ist die Grenze der Belastbarkeit des Schätzers erreicht, sobald der Mittelwert im ausreißerbehafteten Rauschmodell die obere Konfidenzintervallgrenze des ausreißerfreien Rauschmodells überschreitet.

Die *Belastbarkeit* bei Rauschmodell III sei hier exemplarisch bei Datensatz 4 in Abbildung 6.2 gezeigt. Bei ca. 4 % überschreitet der mittlere kAE-Wert beim MdkQ-Schätzer die obere Konfidenzintervallgrenze des kAE bei Rauschmodell I. Bei ca. 15 % folgt der Huber-Schätzer und bei ca. 27 % folgt der Hampel-Schätzer. Das Konfidenzintervall beim MdkQ-Schätzer wächst bei Rauschmodell

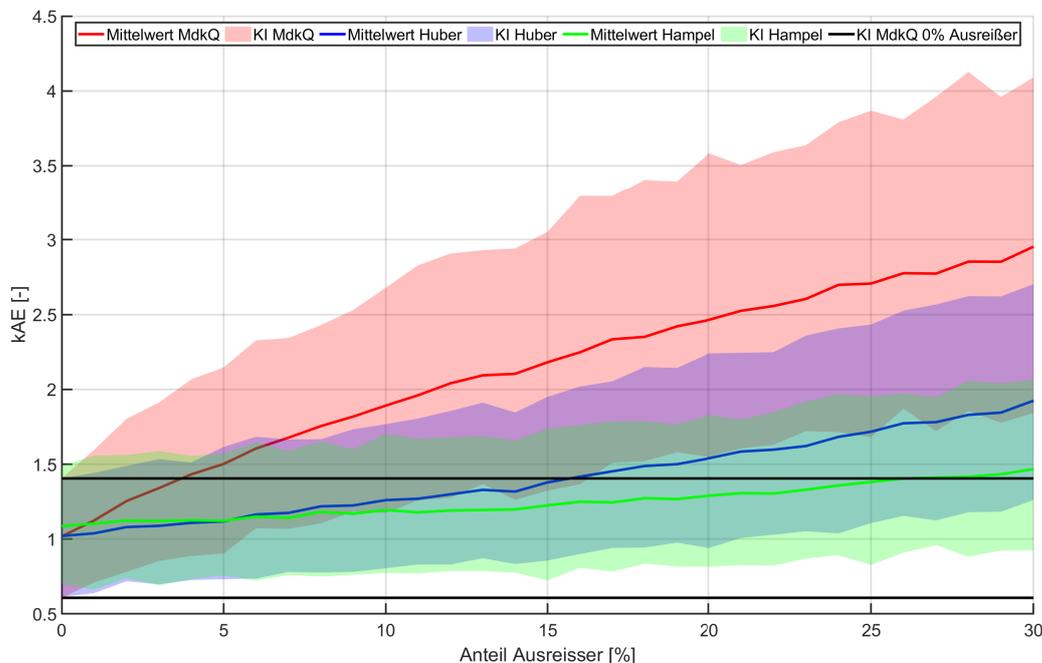


Abbildung 6.2: *Belastbarkeit* Rauschmodell III Datensatz 4

III mit steigender Ausreißeranzahl wesentlich stärker als das Konfidenzintervall der beiden robusten Schätzer. Bei 30 % Ausreißern im Rauschmodell III erzielt der Hampel-Schätzer das schmalste Konfidenzintervall der drei untersuchten Schätzer.

Die Abbildungen der anderen Datensätze ähneln Abbildung 6.2 sehr stark und sind deshalb im Anhang in Abbildung A.43 bis A.47 zu finden.

Die Grenze der *Belastbarkeit* des MdkQ-Schätzers bei Rauschmodell III wird bei den untersuchten Datensätzen zwischen 3 % und 5 % Ausreißern erreicht. Beim Huber-Schätzer wird die Grenze der *Belastbarkeit* bei Rauschmodell III zwischen 15 % und 19 % Ausreißern erreicht. Der Hampel-Schätzer erreicht die Grenze der *Belastbarkeit* bei Rauschmodell III bei einem Ausreißeranteil zwischen 26 % und 30 %.

Die *Belastbarkeit* für Rauschmodell IV wird exemplarisch an Datensatz 5 in Abbildung 6.3 gezeigt.

Beim MdkQ-Schätzer wird die Grenze der *Belastbarkeit* bei ca. 2% und beim Huber-Schätzer bei ca. 3% Ausreißern erreicht. Mit dem Hampel-Schätzer wird die Grenze der *Belastbarkeit* erst bei ca. 8% Ausreißern erreicht. Insgesamt wachsen die kAE-Werte bei allen Schätzern deutlich stärker an als bei Rauschmodell III. Ab ca. 23% Ausreißern erzielen MdkQ-Schätzer und der Huber-Schätzer identische Ergebnisse. Mit steigender Rate an Ausreißern nähern sich die kAE-Werte aller Schätzer an. Die Breite der Konfidenzintervalle bleibt bei allen Schätzern relativ konstant. Lediglich beim

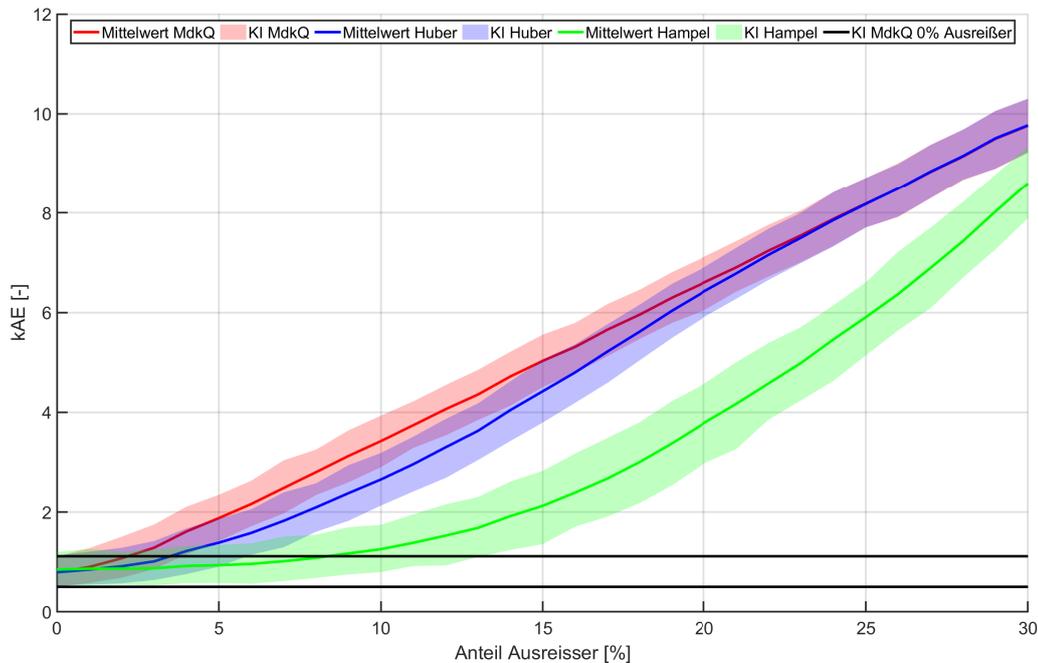


Abbildung 6.3: *Belastbarkeit* Rauschmodell IV Datensatz 5

Hampel-Schätzer ist eine leichte Verbreiterung zwischen 12% und 22% Ausreißern zu beobachten. Die Abbildungen der anderen Datensätze ähneln wiederum Abbildung 6.3 sehr und sind deshalb im Anhang in den Abbildungen A.48 bis A.52 zu finden.

Bei den untersuchten Datensätzen liegt die Grenze der *Belastbarkeit* der Methode des MdkQ-Schätzers bei einem Ausreißeranteil von ca. 2%. Die Grenze der *Belastbarkeit* des Huber-Schätzers liegt mit einem Ausreißeranteil von ca. 3% etwas darüber. Der Hampel-Schätzer ist hingegen deutlich robuster und erreicht die Grenze der *Belastbarkeit* erst bei ca. 7-8% Ausreißern.

Die *Belastbarkeit* für Rauschmodell V wird exemplarisch an Datensatz 9 in Abbildung 6.4 gezeigt. Beim MdkQ-Schätzer wird die Grenze der *Belastbarkeit* bei einem Ausreißeranteil von ca. 0.7% erreicht. Der Huber-Schätzer ist geringfügig robuster und erreicht die Grenze der *Belastbarkeit* bei ca. 1.2% Ausreißern. Mit dem Hampel-Schätzer wird die Grenze der *Belastbarkeit* bei ca. 2.2% Ausreißern erreicht. Insgesamt wachsen die kAE-Werte bei allen Schätzern deutlich stärker an als bei Rauschmodell III oder Rauschmodell IV. Ab ca. 5.5% Ausreißern übersteigt der kAE des Hampel-Schätzers die eng zusammenliegenden kAE-Werte des MdkQ- und des Huber-Schätzers. Interessant ist das breite Konfidenzintervall des Hampel-Schätzers zwischen 1% und 8% Ausreißern. Gerade im Bereich zwischen 2% und 5% Ausreißern werden in vereinzelt Simulationen durchläufen noch sehr niedrige kAE-Werte erzielt. Andererseits liegt auch die obere Intervallgrenze beim Hampel-Schätzer ab ca. 1.7% Ausreißern deutlich über der oberen Intervallgrenze der beiden anderen Schätzer. Diese Schwankungen treten bei MdkQ- und Huber-Schätzer nicht auf. Die Konfidenzintervalle sind dort wesentlich schmaler.

Die Abbildungen der anderen Datensätze sind wiederum im Anhang in den Abbildungen A.53 bis A.57 zu finden, da sie Abbildung 6.4 sehr ähneln.

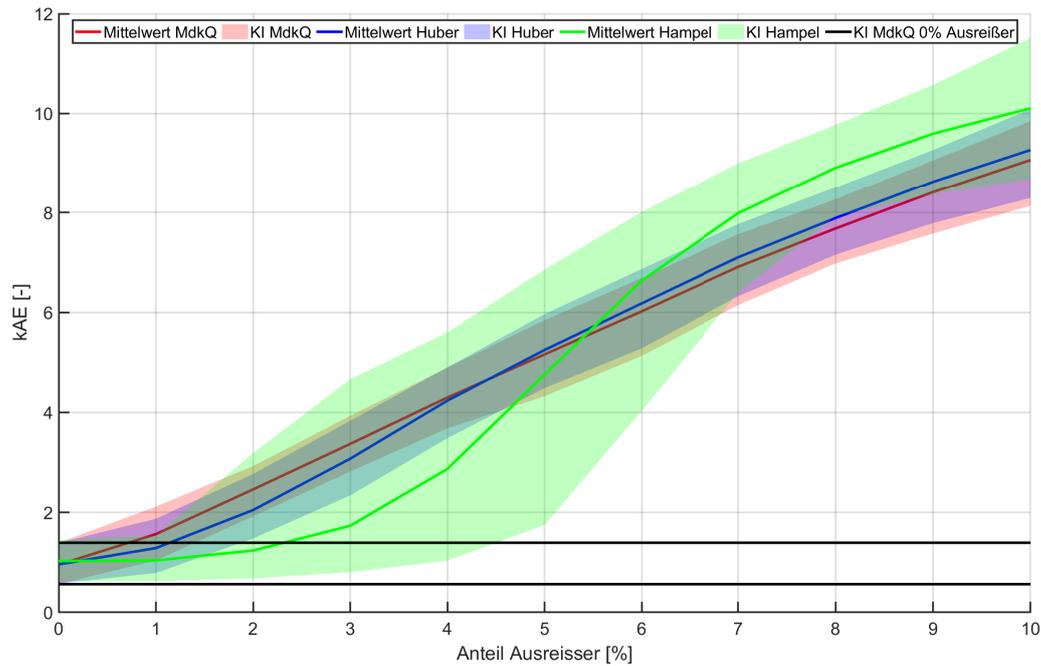


Abbildung 6.4: *Belastbarkeit* Rauschmodell V Datensatz 9

Bei den untersuchten Datensätzen liegt die Grenze der *Belastbarkeit* des MdkQ-Schätzers bei einem Ausreißeranteil von ca. 0.5-0.8%. Die Grenze der *Belastbarkeit* des Huber-Schätzers liegt bei ca. 0.6-1.2% Ausreißern. Auch bei diesem Rauschmodell ist der Hampel-Schätzer robuster und erreicht die Grenze der *Belastbarkeit* erst bei 1.3-2.3% Ausreißern.

6.4.1 Zusammenfassung

Folgende Erkenntnisse lassen sich aus Kapitel 6.4 gewinnen:

- Von den drei untersuchten Schätzern MdkQ, Huber und Hampel weist der Hampel-Schätzer die höchste *Belastbarkeit* auf. Dies ist aufgrund der Konzeption des Hampel-Schätzers mit einer hohen Herabgewichtung von Beobachtungen mit großer Verbesserung zu erwarten.
- Die ermittelte Grenze der *Belastbarkeit* ist kaum von den untersuchten Datensätzen abhängig. D.h. eine Auswirkung der Anzahl der Kontrollpunkte n , der Anzahl der Datenpunkte r oder der Komplexität der zu approximierenden Kurve auf die *Belastbarkeit* des Schätzers ist nicht zu beobachten.
- Die verwendeten Rauschmodelle hingegen haben einen immensen Einfluss auf die *Belastbarkeit* der verschiedenen Schätzer (siehe Tabelle 6.11).

Tabelle 6.11: Durchschnittliche Grenze der *Belastbarkeit* pro Rauschmodell (RM)

RM	<i>Belastbarkeit</i> in % der Beobachtungen			<i>Belastbarkeit</i> in % der Punkte		
	MdkQ	Huber	Hampel	MdkQ	Huber	Hampel
III	4.1	16.3	27.5	8.0	29.9	47.4
IV	1.9	2.9	7.5	3.8	5.7	14.4
V	0.6	1.0	1.9	1.2	2.0	3.8

Die Umrechnung von der *Belastbarkeit* der Beobachtungen zur *Belastbarkeit* der Punkte erfolgt mit den Formeln 6.2 und 6.10. Bei den angegebenen Prozentzahlen für die Grenze der *Belastbarkeit* bei Rauschmodell III ist zu beachten, dass der angegebene Anteil nicht dem Anteil tatsächlicher Ausreißer, sondern dem Anteil Beobachtungen/Punkte mit erhöhtem Rauschen (5σ) entspricht.

- Die gezeigten Ergebnisse wurden mit dem wahren Knotenvektor erzielt. Da dieser in der Regel unbekannt ist, muss in der Realität eine Methode zur Knotenvektorstimmung (siehe z.B. Kapitel 4) angewandt werden. Die Grenzen der *Belastbarkeit* bei Verwendung der verschiedenen Methoden zur Knotenvektorstimmung können von den dargestellten Ergebnissen abweichen. Die Ergebnisse in Kapitel 6.2 legen zumindest nahe, dass die *Belastbarkeit* bei Verwendung der Methoden EGAori oder EGAmo in einer ähnlichen Größenordnung liegt, wie die *Belastbarkeit* bei Verwendung des wahren Knotenvektors.

7 Fazit und Ausblick

Im Rahmen dieser Arbeit wurden drei neue bzw. modifizierte Algorithmen zur Bestimmung des Knotenvektors für die B-Spline-Kurvenapproximation vorgestellt. Außerdem wurde erstmalig die Verwendung robuster Schätzer in der Kontrollpunktschätzung systematisch analysiert und die Grenzen der Belastbarkeit (maximaler Ausreißeranteil) aufgezeigt. Eine systematische Analyse und Beurteilung der Leistungsfähigkeit der verschiedenen Methoden zur Knotenvektorstimmung und der robusten Schätzer bei der Kontrollpunktschätzung kann nur auf Basis simulierter Daten erfolgen. Aus diesem Grund wurden neun verschiedene und zum Teil bereits in der Fachliteratur verwendete Datensätze generiert und mit verschiedenen Rauschmodellen verrauscht. Die Rauschmodelle basieren auf der Normalverteilung, der t -Verteilung, der Gaußschen Mischverteilung sowie zwei Varianten eines einseitig generierten Rauschens. Zusätzlich wurden zwei reale Datensätze in die Analyse einbezogen.

7.1 Fazit

In Kapitel 1.1 wurden zwei Forschungshypothesen formuliert auf die im Folgenden Bezug genommen wird.

Die erste Forschungshypothese besagt, dass es möglich ist *eine Methode zur Knotenvektorstimmung zu entwickeln, bei der sowohl auftretende Datenlücken als auch komplex geformte Objekte keinen signifikanten negativen Einfluss auf den approximierten B-Spline und die numerische Stabilität der B-Spline-Approximation haben*. Mit den in Kapitel 5 dargestellten Ergebnissen kann konstatiert werden, dass diese Forschungshypothese erfüllt ist. Die im Rahmen dieser Arbeit modifizierte Variante des elitären genetischen Algorithmus (EGAmoD) erzeugt Knotenvektoren, die im Ergebnis insgesamt zur besten B-Spline-Kurvenapproximation führt. Die mit dem EGAmoD erzielten Ergebnisse liegen annähernd im Bereich der Ergebnisse, die mit dem wahren Knotenvektor¹ erzielt wurden. Die numerische Stabilität des EGAmoD und auch des ursprünglichen elitären genetischen Algorithmus (EGAori) ist auf die Initialisierung mit einer deterministischen Methode zurückzuführen. Einer der untersuchten Datensätze enthielt eine größere Datenlücke. Bei diesem Datensatz zeigten sich sowohl beim EGAmoD als auch bei den anderen untersuchten Methoden keine numerischen Instabilitäten. Dies ist auf die relativ geringe Anzahl an zu platzierenden Knoten zurückzuführen. Bei der Vorstellung des EGAori zeigten Bureick u. a. (2019), dass Datenlücken mit einer vergleichbaren Größenordnung in Kombination mit einer höheren Anzahl an zu platzierenden Knoten sehr wohl zu numerischen Instabilitäten bei anderen heuristischen Verfahren führen können. Durch die oben beschriebene Initialisierung wird dieses Problem beim EGAori eliminiert. Da der EGAmoD im ungünstigsten Fall die gleiche Initialisierung wie der EGAori verwendet, ist dieses Problem auch beim EGAmoD eliminiert.

Die zweite Forschungshypothese besagt, dass es möglich ist *die Kontrollpunktschätzung so robust zu gestalten, dass der Einfluss von Ausreißern auf den approximierten B-Spline minimal wird*. Diese Forschungshypothese wird mit Blick auf die in Kapitel 6 präsentierten Ergebnisse bis zu einem bestimmten Ausreißeranteil erfüllt. Die Erfüllung dieser Forschungshypothese wird maßgeblich durch den Anteil, die Art und die Konzentration der Ausreißer beeinflusst. Bei in Lage und Richtung zufälligen Ausreißern kann mithilfe robuster Schätzer bis zu einem Ausreißeranteil von ca. 47 % (bezogen auf die Anzahl der Punkte) ein annehmbares Ergebnis in der B-Spline-Kurvenapproximation erzielt werden. Wirken die Ausreißer hingegen einseitig, verringert sich dieser Ausreißeranteil auf ca. 14 %. Sind die Ausreißer zusätzlich noch räumlich konzentriert, verringert sich der Ausreißeranteil auf

¹insofern dieser bekannt war

unter 4%. Die angegebenen Prozentzahlen gelten für den Hampel-Schätzer. Beim Huber-Schätzer werden geringere Prozentzahlen erzielt. Liegt der Anteil der Ausreißer über den angegebenen Prozentzahlen, versagen auch die robusten Schätzer. In diesem Falle ist die zweite Forschungshypothese nicht mehr erfüllt.

Insgesamt kann konstatiert werden, dass mit dem Erfüllen der beiden Forschungshypothesen auch das in Kapitel 1.1 formulierte Ziel *„Die Entwicklung eines Algorithmus zur robusten B-Spline-Approximation einer beliebigen Punktwolke, welcher, trotz auftretender Ausreißer und Datenlücken und unter Berücksichtigung des dem Messprozess zur Erfassung dieser Punktwolke immanenten Unsicherheitsbudgets, die genaue und zuverlässige Approximation eines komplexen Objekt sicherstellt“* erfüllt wurde. Das der Punktwolke immanente Unsicherheitsbudget wird in der Kontrollpunktschätzung berücksichtigt. Einer unsortierten Punktwolke kann mit der geeigneten Wahl der Parametrisierungsmethode begegnet werden. Durch den im Rahmen dieser Arbeit vorgestellten Algorithmus EGAmoD ist nun eine Methode zur Knotenvektorstimmung gegeben, die trotz Datenlücken und komplex geformter Objekte zu einer genauen und zuverlässigen B-Spline-Approximation führt. Werden in zu approximierenden Datensätzen Ausreißer vermutet, so kann der Einfluss dieser Ausreißer bis zu einem gewissen Ausreißeranteil durch die Verwendung des Hampel-Schätzers bei der Kontrollpunktschätzung signifikant reduziert werden.

7.2 Ausblick

In dieser Arbeit wurde nicht untersucht, wie sich eine von der wahren Parametrisierung abweichende Parametrisierung auf die Ergebnisse der B-Spline-Kurvenapproximation auswirkt. Insbesondere der Einfluss einer unzureichenden Parametrisierung auf die Kontrollpunktschätzung sollte in einer systematischen Untersuchung analysiert werden.

Neitzel u. a. (2019) formulieren das funktionale Modell für Splines mit Power Basis in impliziter Form. Die Lösung des Ausgleichsproblems erfolgt in einem iterativen linearisierten Gauß-Helmert Modell. Hierdurch wird insbesondere der Effekt einer unzureichenden Parametrisierung eliminiert. Inwieweit diese Vorgehensweise auch auf B-Spline-Kurven übertragbar ist und welche Vorteile und Herausforderungen sich dadurch ergeben, wurde bisher noch nicht untersucht.

Ebenfalls nicht Teil dieser Arbeit ist eine systematische Untersuchung des Einflusses heteroskedastischer und korrelierter Datensätze auf den beschriebenen Approximationsprozess, insbesondere die Kontrollpunktschätzung. Dazu sei an dieser Stelle auf die Arbeiten Kermarrec u. a. (2019), Zhao u. a. (2019) und Zhao (2019) verwiesen, die sich der Thematik annehmen. In Kombination mit dem beschriebenen optimierten Approximationsprozess sind in der Thematik weitere Erkenntnisse zu erwarten.

Der präsentierte Approximationsprozess kann grundsätzlich auch für die B-Spline-Flächen adaptiert werden. Allerdings treten dort andere Schwierigkeiten und Herausforderungen auf. Zunächst ist die B-Spline-Flächenapproximation aufgrund größerer Datenmengen deutlich rechen- und zeittensiver. Gerade der Einsatz iterativer Methoden bei der Parametrisierung oder heuristischer Methoden bei der Knotenvektorstimmung potenziert diesen Rechenaufwand. Des Weiteren verlagert sich bei B-Spline-Flächen der Einfluss der verschiedenen Stellgrößen bzw. Teilschritte. Bei der B-Spline-Flächenapproximation muss bei der Parametrisierung wesentlich mehr Aufwand betrieben werden als bei der B-Spline-Kurvenapproximation. Hingegen nimmt der Einfluss der Knotenvektoren auf die Approximationsgüte der B-Spline-Flächen gegenüber den B-Spline-Kurven ab.

Die Integration robuster Schätzer in die B-Spline-Flächenapproximation stellt theoretisch keine Schwierigkeit dar. Die praktische Umsetzung wird lediglich durch den bereits erwähnten erhöhten Rechenaufwand bei Flächen bzw. flächenhaften Datensätzen erschwert. Hier wäre es interessant den Einfluss der robusten Schätzer auf die Deformationsanalyse zu evaluieren und zu quantifizieren.

A Anhang

A.1 Simulierte B-spline Kurven

Tabelle A.1: Parameterwerte und Darstellung von Datensatz 4

Parameter	Werte
n	6
p	2
ξ	[0 0 0 0.333 0.333 0.667 0.667 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} -2.0 & -1.0 & 0.0 & 1.0 & 2.0 & 3.0 & 4.0 \\ 0.0 & 1.0 & 2.0 & 1.0 & 0.0 & 1.0 & 2.0 \end{bmatrix}^T$

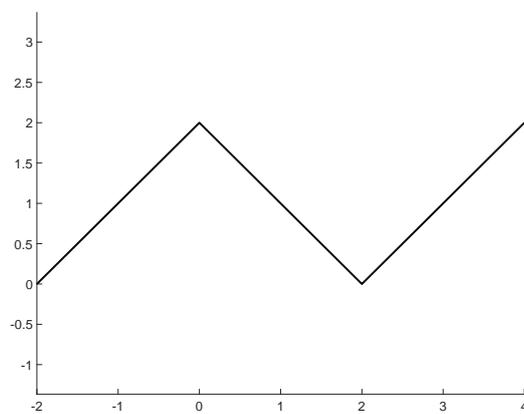
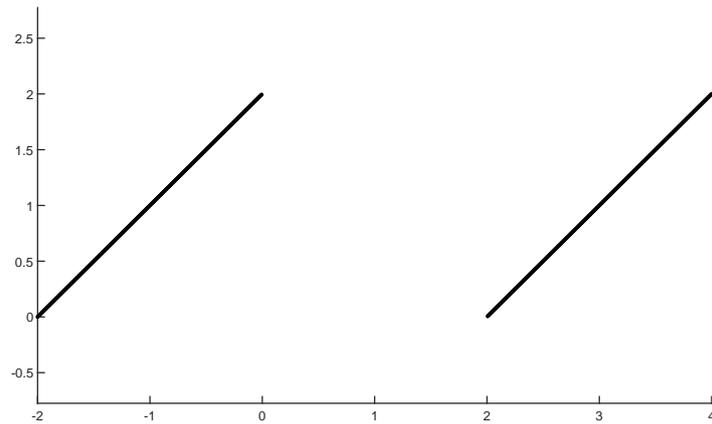


Tabelle A.2: Parameterwerte und Darstellung von Datensatz 5

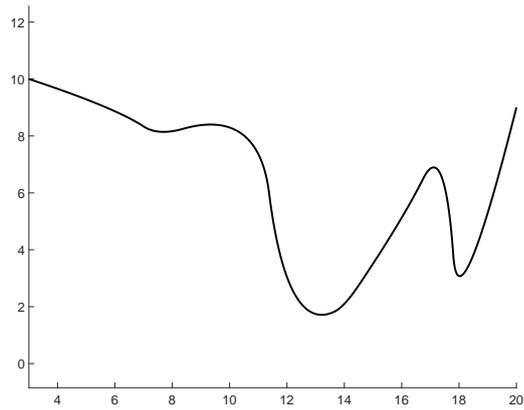
Parameter	Werte
n	6
p	2
ξ	[0 0 0 0.333 0.333 0.667 0.667 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} -2.0 & -1.0 & 0.0 & 1.0 & 2.0 & 3.0 & 4.0 \\ 0.0 & 1.0 & 2.0 & 1.0 & 0.0 & 1.0 & 2.0 \end{bmatrix}^T$



Bemerkung	Entspricht Datensatz 4 ohne Datenpunkte für die gilt: $0.334 < u < 0.666$
-----------	---

Tabelle A.3: Parameterwerte und Darstellung von Datensatz 6

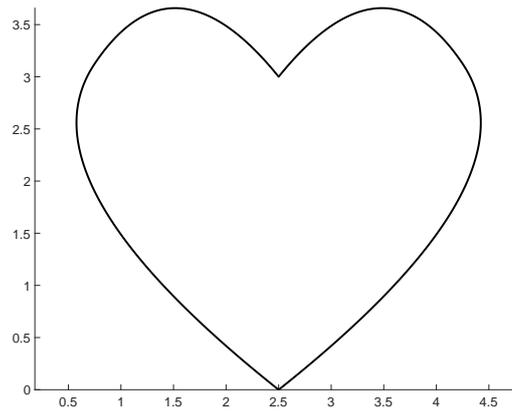
Parameter	Werte
n	9
p	2
ξ	[0 0 0 0.10 0.15 0.30 0.55 0.60 0.75 0.90 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} 3.0 & 6.0 & 7.5 & 11.0 & 12.0 & 14.0 & 16.0 & 17.5 & 18.0 & 20.0 \\ 10.0 & 9.0 & 8.0 & 9.0 & 1.0 & 2.0 & 5.0 & 8.0 & 1.0 & 9.0 \end{bmatrix}^T$



Bemerkung	Datensatz aus Schmitt und Neuner (2015) entnommen
-----------	---

Tabelle A.4: Parameterwerte und Darstellung von Datensatz 7

Parameter	Werte
n	6
p	2
ξ	[0 0 0 0.25 0.50 0.50 0.75 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} 2.50 & 0.00 & 1.50 & 2.50 & 3.50 & 5.00 & 2.50 \\ 0.00 & 2.00 & 4.25 & 3.00 & 4.25 & 2.00 & 0.00 \end{bmatrix}^T$



Bemerkung	Datensatz aus Bureick u. a. (2019) entnommen
-----------	--

Tabelle A.5: Parameterwerte und Darstellung von Datensatz 8

Parameter	Werte
n	9
p	2
ξ	[0 0 0 0.10 0.10 0.30 0.45 0.60 0.80 0.80 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} 1.0 & 1.1 & 1.2 & 1.2 & 1.3 & 1.5 & 1.6 & 1.6 & 1.8 & 2.0 \\ 1.0 & 1.0 & 1.0 & 1.1 & 1.2 & 1.2 & 1.1 & 1.0 & 1.0 & 1.0 \end{bmatrix}^T$

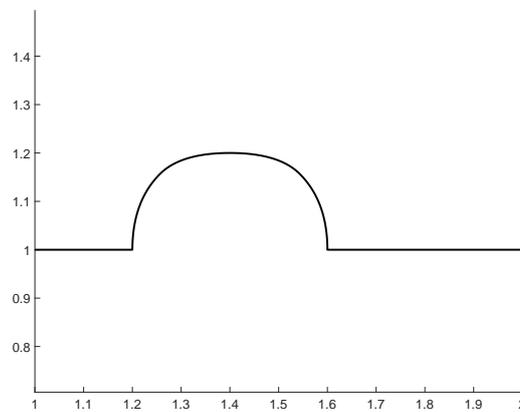
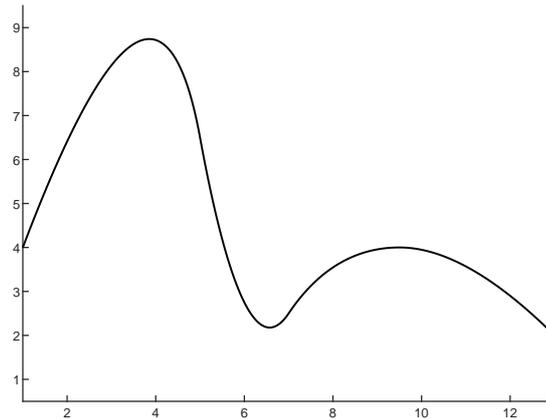


Tabelle A.6: Parameterwerte und Darstellung von Datensatz 9

Parameter	Werte
n	5
p	2
ξ	[0 0 0 0.25 0.50 0.75 1 1 1]
$\begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_y \end{bmatrix}^T$	$\begin{bmatrix} 1.0 & 4.0 & 6.0 & 8.0 & 11.0 & 13.0 \\ 4.0 & 12.0 & 1.0 & 4.0 & 4.0 & 2.0 \end{bmatrix}^T$



Bemerkung	Datensatz aus Bureick u. a. (2019) entnommen
-----------	--

A.2 Numerische Beurteilung und Validierung der Knotenvektorwahl

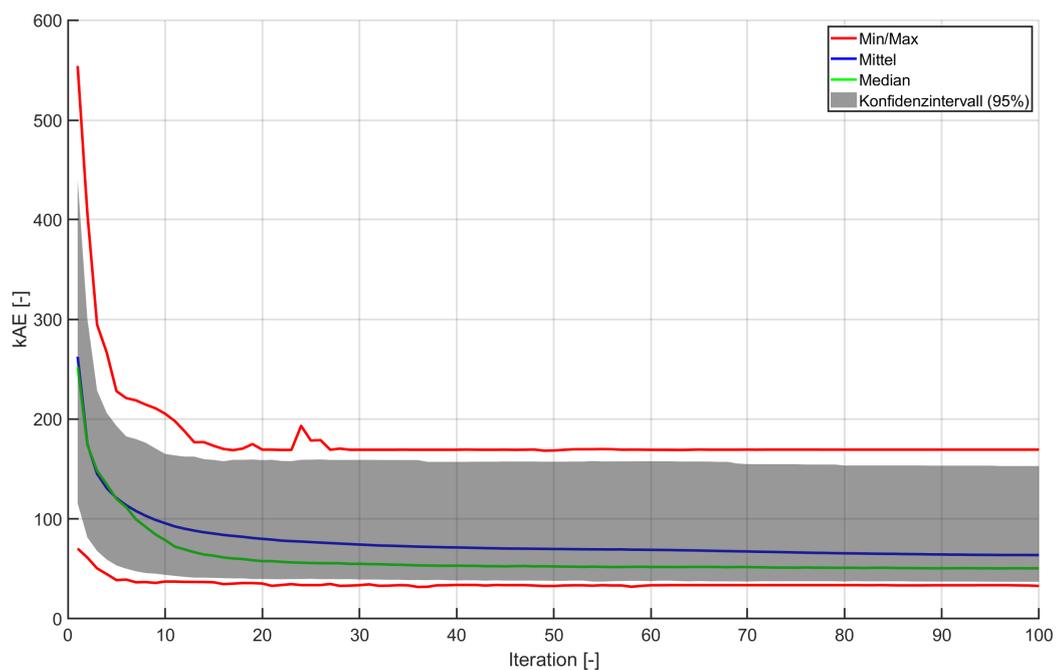


Abbildung A.1: Graphische Darstellung des kAE des CSA bei Datensatz 2

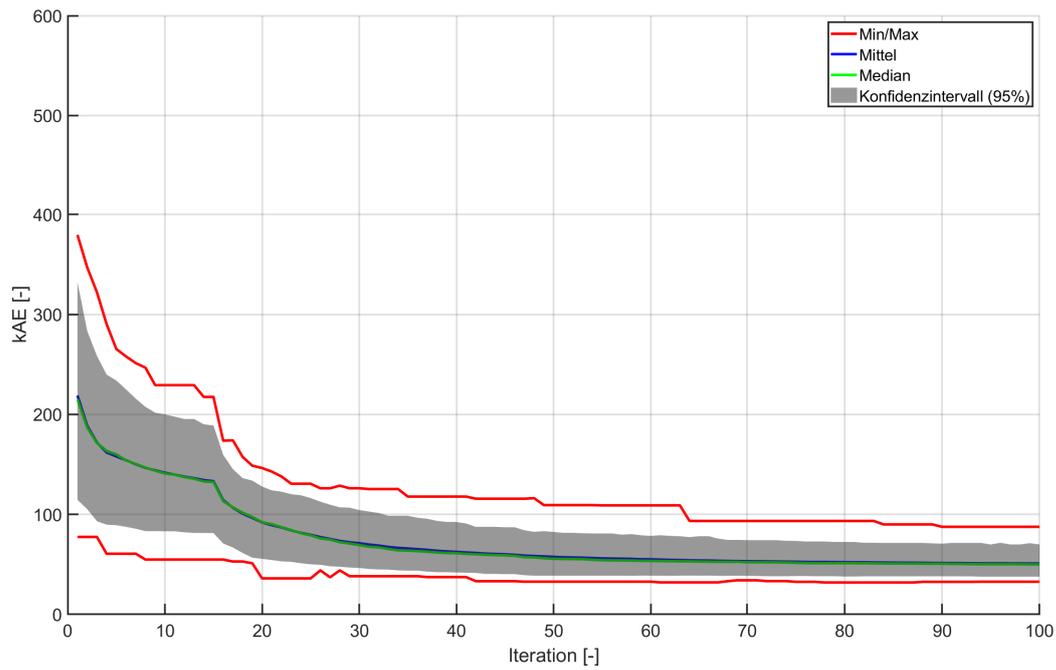


Abbildung A.2: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 2

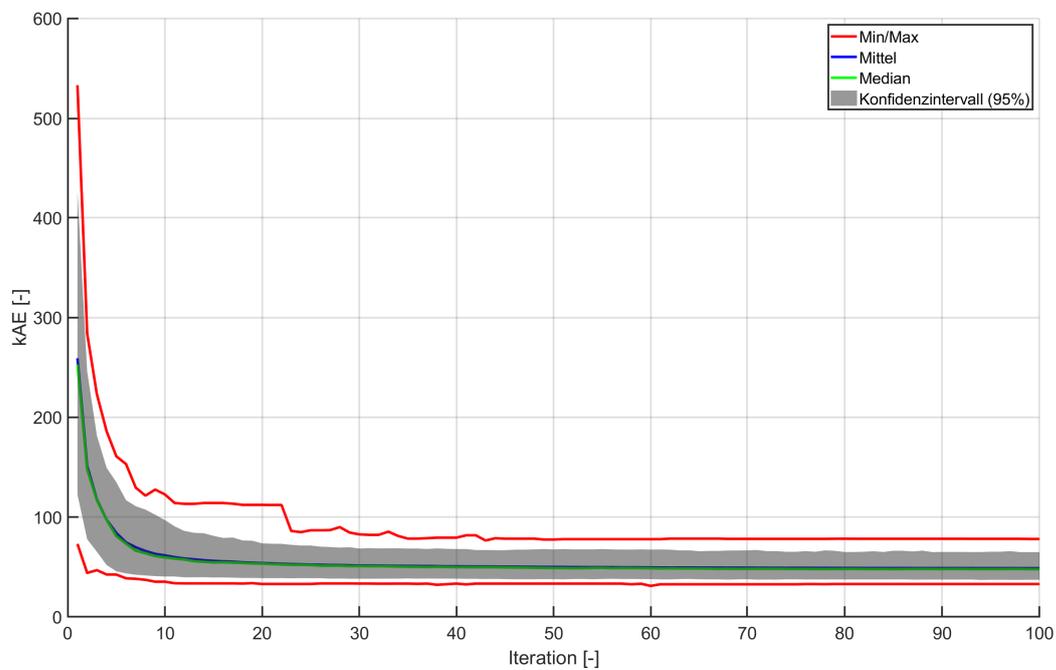


Abbildung A.3: Graphische Darstellung des kAE des EGAori bei Datensatz 2

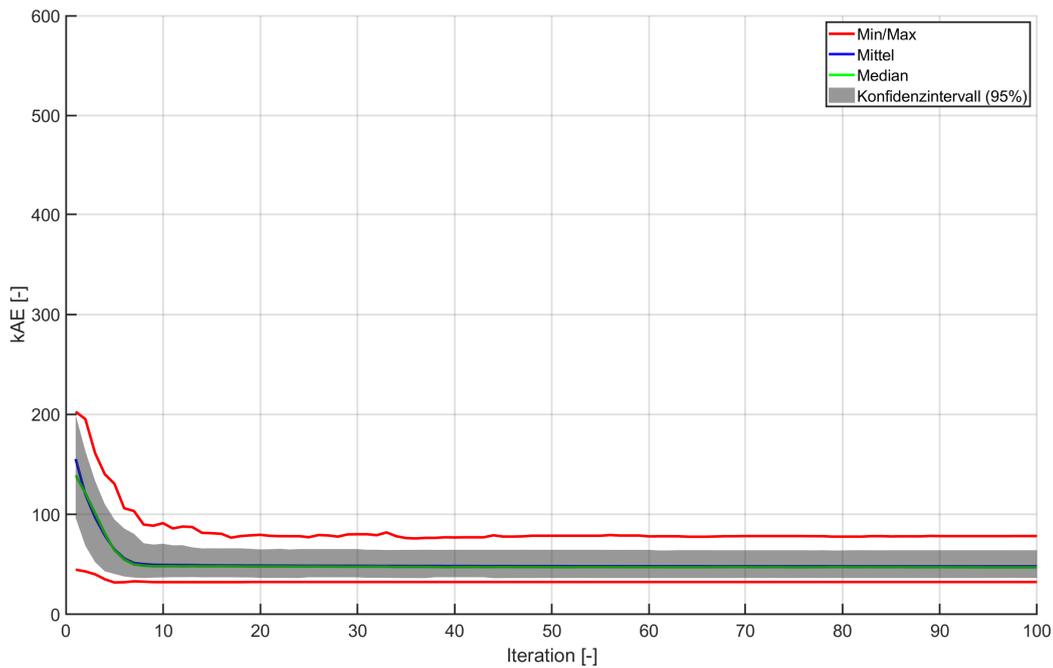


Abbildung A.4: Graphische Darstellung des kAE des EGAmoD bei Datensatz 2

Tabelle A.7: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 7

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	64.284	64.855	64.507	64.508	64.504	0.085	64.339	64.663
-	RIU	8.354	31.956	20.164	20.091	19.280	3.835	13.101	28.171
1	CSA	8.600	99.629	54.571	54.409	48.056	16.195	21.034	87.234
	EMC	12.717	82.740	46.835	46.623	45.904	12.906	21.550	69.973
	EGAori	7.166	70.906	52.824	55.365	63.237	12.153	24.691	64.855
	EGAmoD	8.305	31.956	20.139	20.091	19.512	3.866	12.717	28.171
25	CSA	2.231	17.206	10.511	10.782	10.856	2.872	4.461	15.670
	EMC	2.767	29.066	14.131	13.790	13.569	4.740	5.847	24.145
	EGAori	0.558	4.190	1.918	1.819	1.701	0.670	0.933	3.539
	EGAmoD	0.565	1.824	1.124	1.117	1.104	0.207	0.743	1.528
50	CSA	2.231	17.206	10.487	10.763	10.850	2.886	4.461	15.670
	EMC	0.908	13.622	4.834	4.425	4.032	2.498	1.408	10.599
	EGAori	0.575	2.552	1.499	1.477	1.444	0.340	0.884	2.245
	EGAmoD	0.583	1.793	1.109	1.111	1.107	0.200	0.713	1.485
75	CSA	2.231	17.206	10.483	10.763	10.849	2.876	4.461	15.670
	EMC	0.735	10.986	2.988	2.460	1.509	1.925	0.930	8.201
	EGAori	0.575	2.337	1.376	1.364	1.384	0.283	0.848	1.999
	EGAmoD	0.563	1.798	1.105	1.104	1.097	0.199	0.730	1.487
100	CSA	2.231	17.206	10.483	10.763	10.849	2.876	4.461	15.670
	EMC	0.651	10.986	2.257	1.555	1.275	1.566	0.794	6.574
	EGAori	0.559	2.282	1.310	1.307	1.297	0.245	0.843	1.776
	EGAmoD	0.566	1.786	1.103	1.106	1.095	0.200	0.724	1.484
-	Wahr	0.471	1.532	1.002	0.994	0.961	0.198	0.613	1.405
-	Roh	11.876	13.203	12.539	12.552	12.581	0.207	12.124	12.945

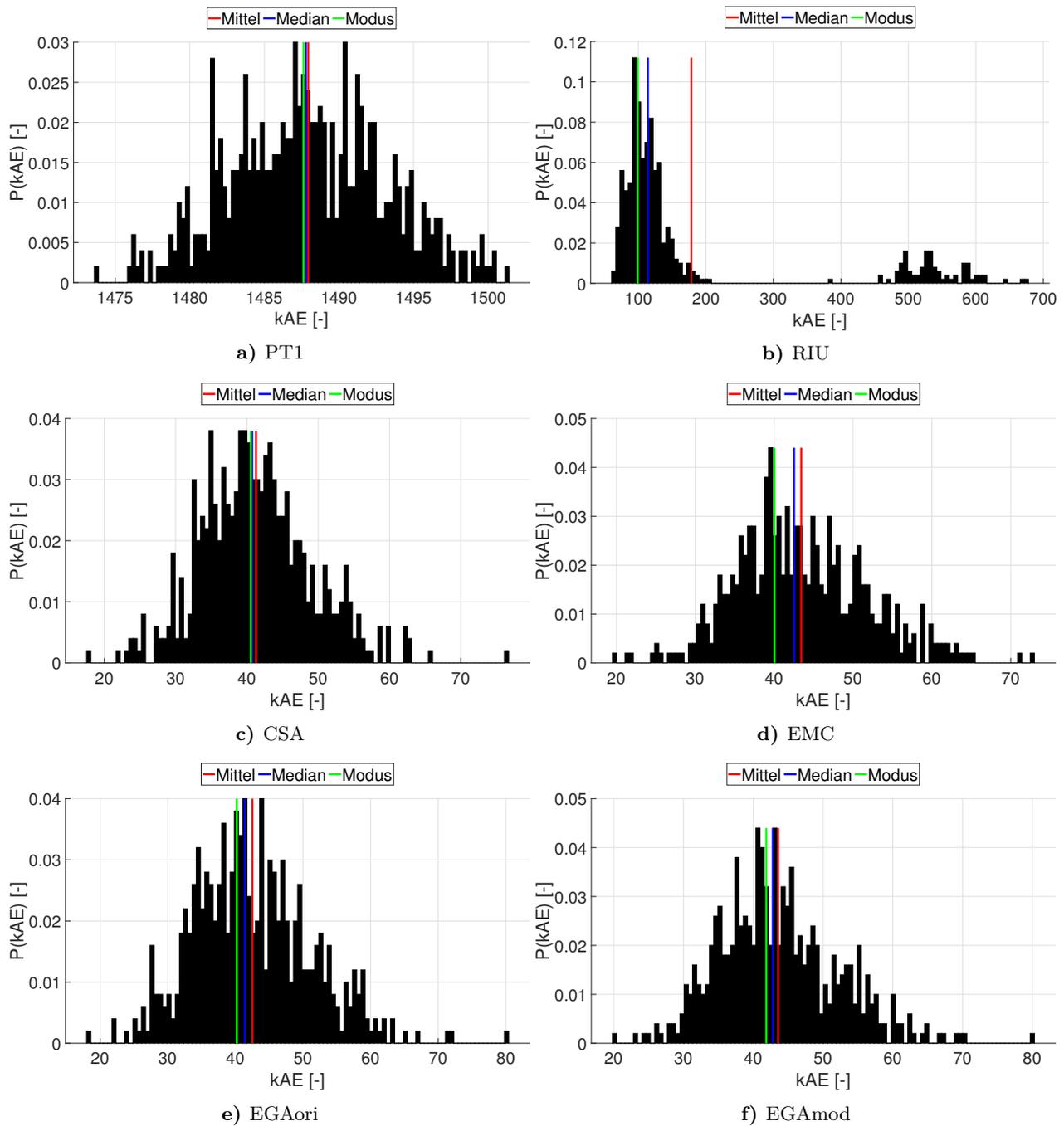


Abbildung A.5: Histogramm der Verteilung der kAE-Werte bei Datensatz 3. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.7 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

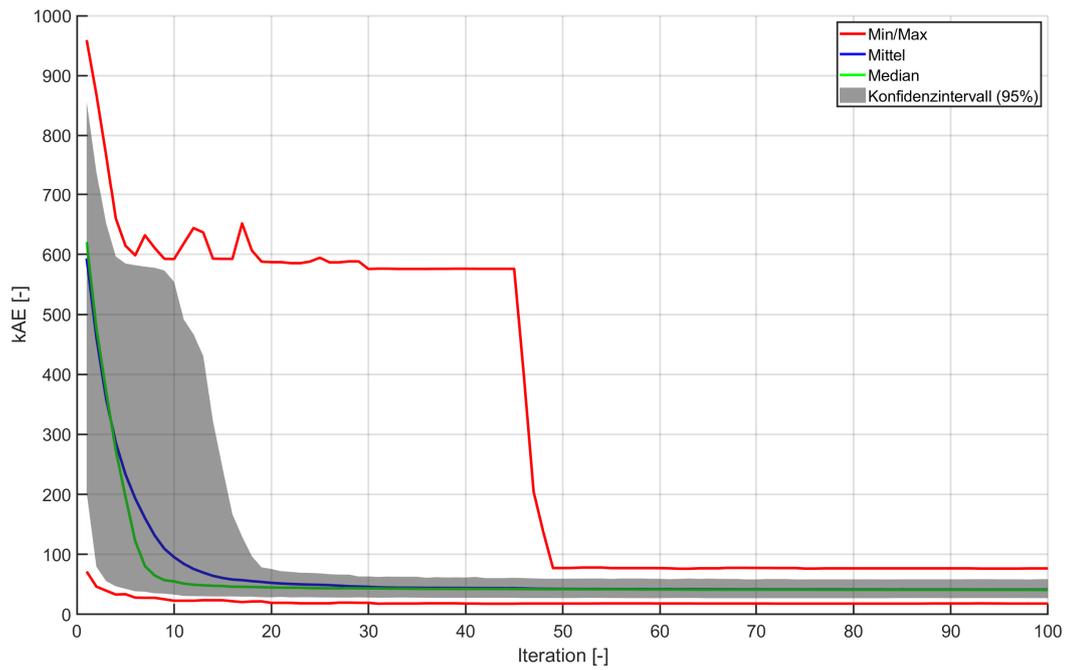


Abbildung A.6: Graphische Darstellung des kAE des CSA bei Datensatz 3

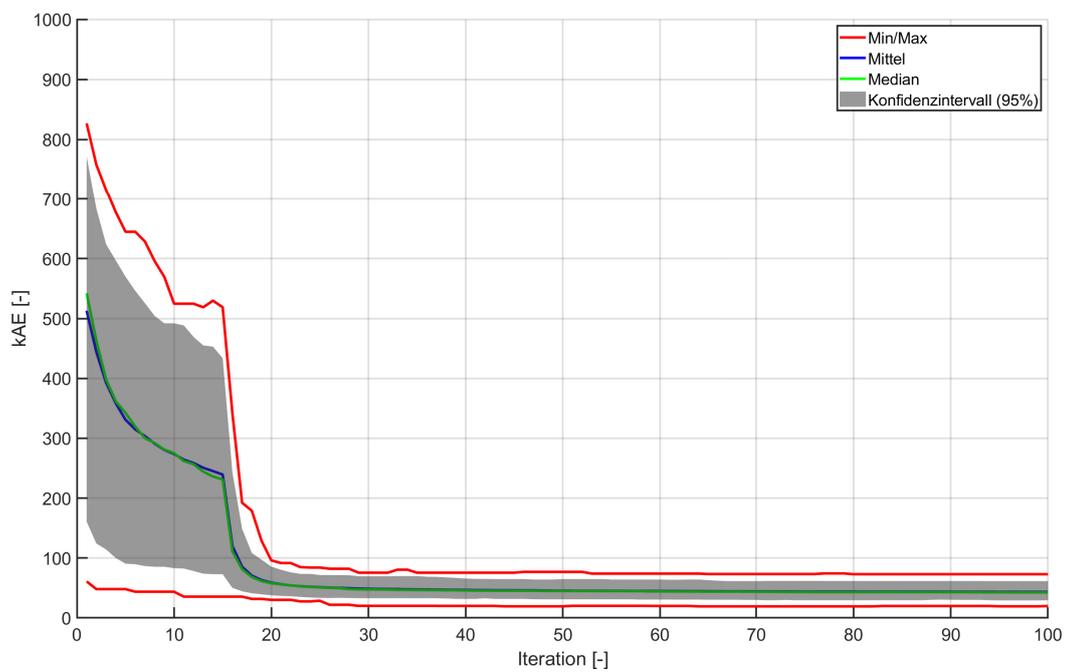


Abbildung A.7: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 3

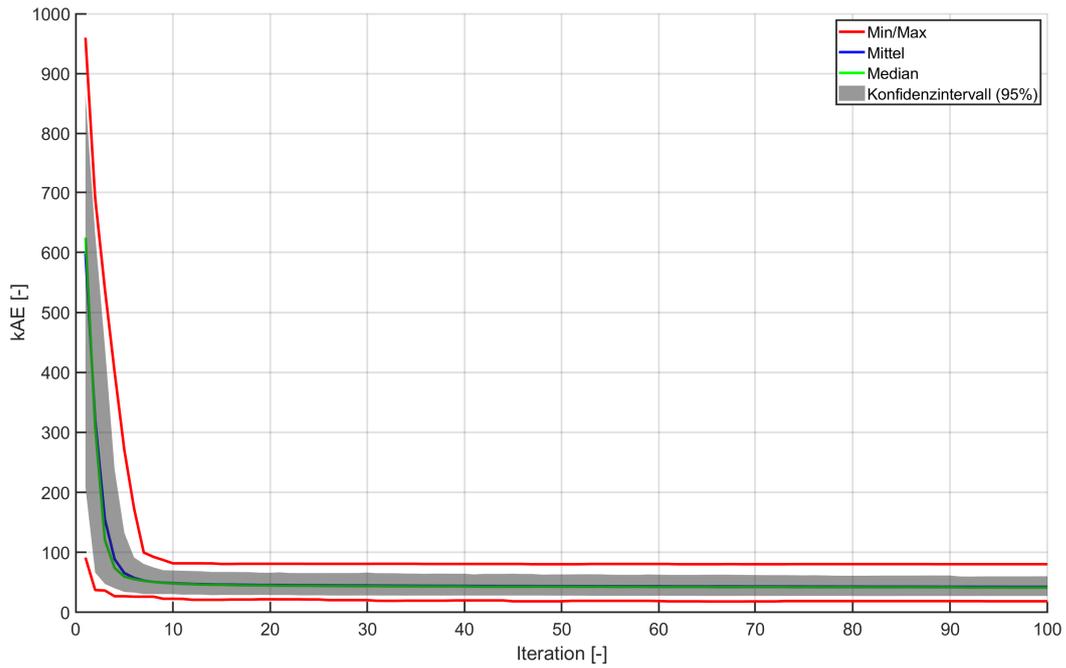


Abbildung A.8: Graphische Darstellung des kAE des EGAori bei Datensatz 3

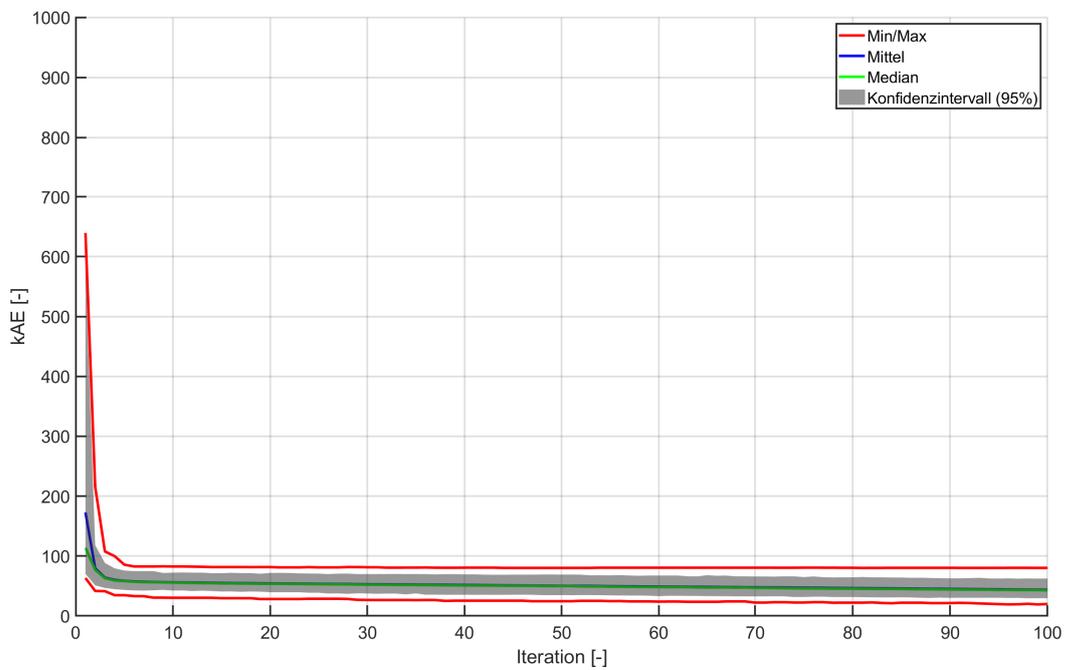


Abbildung A.9: Graphische Darstellung des kAE des EGAmoD bei Datensatz 3

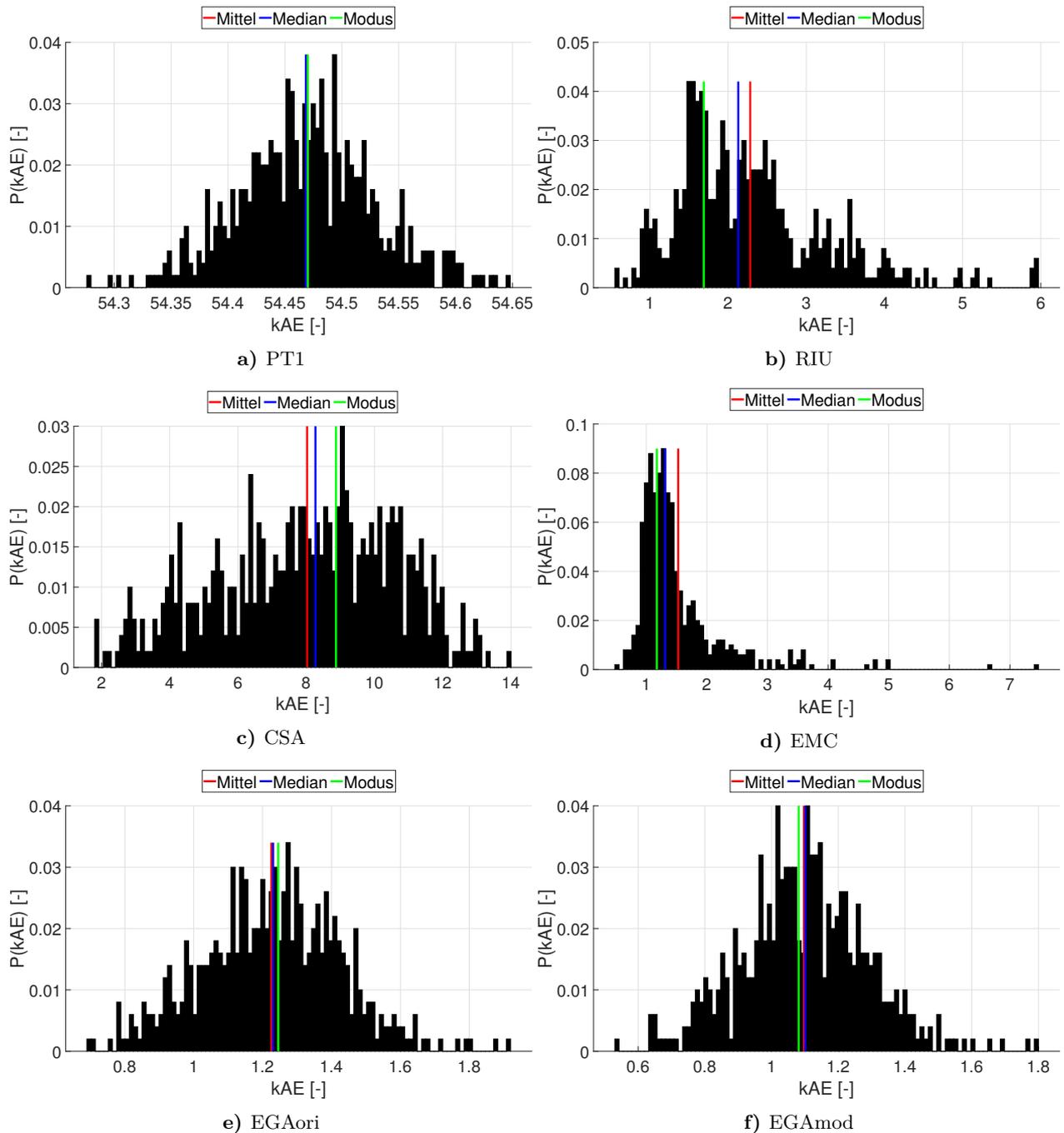


Abbildung A.10: Histogramm der Verteilung der kAE-Werte bei Datensatz 4. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.8 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

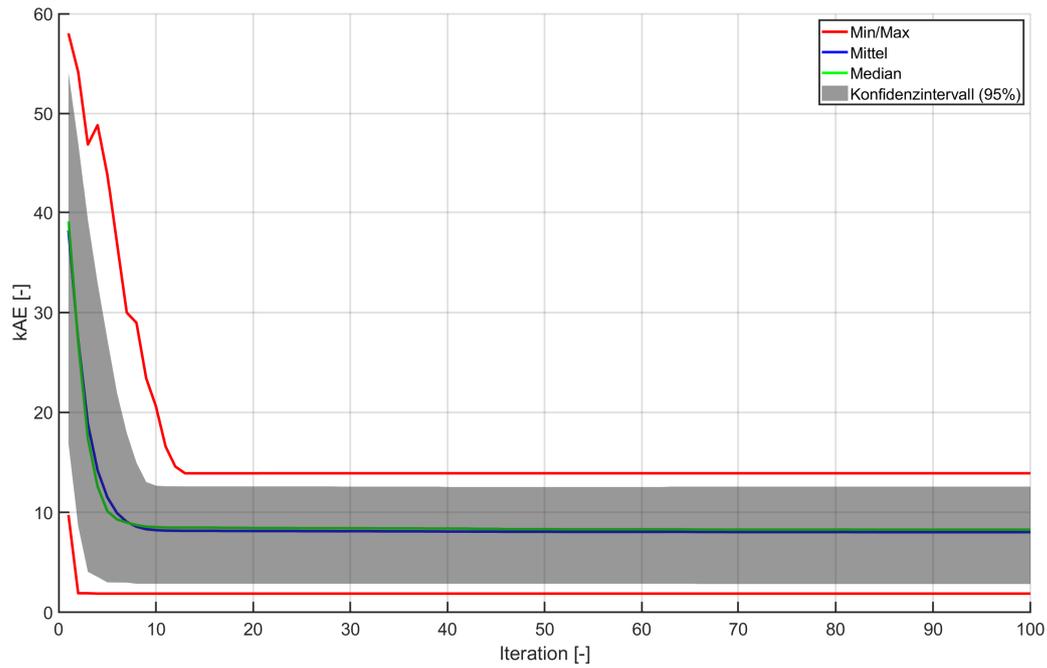


Abbildung A.11: Graphische Darstellung des kAE des CSA bei Datensatz 4

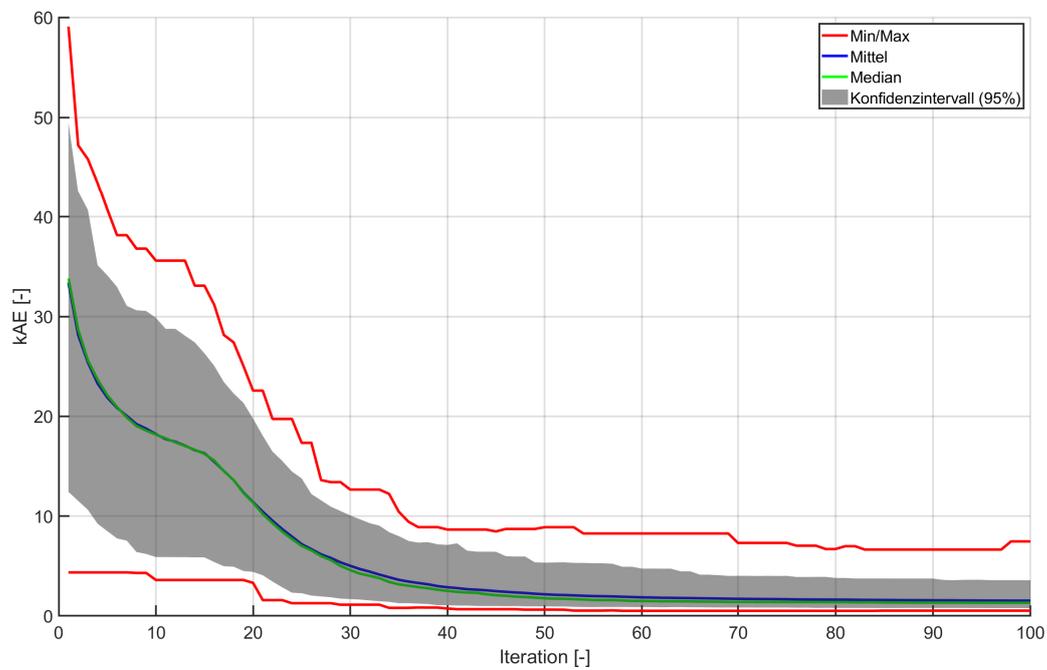


Abbildung A.12: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 4

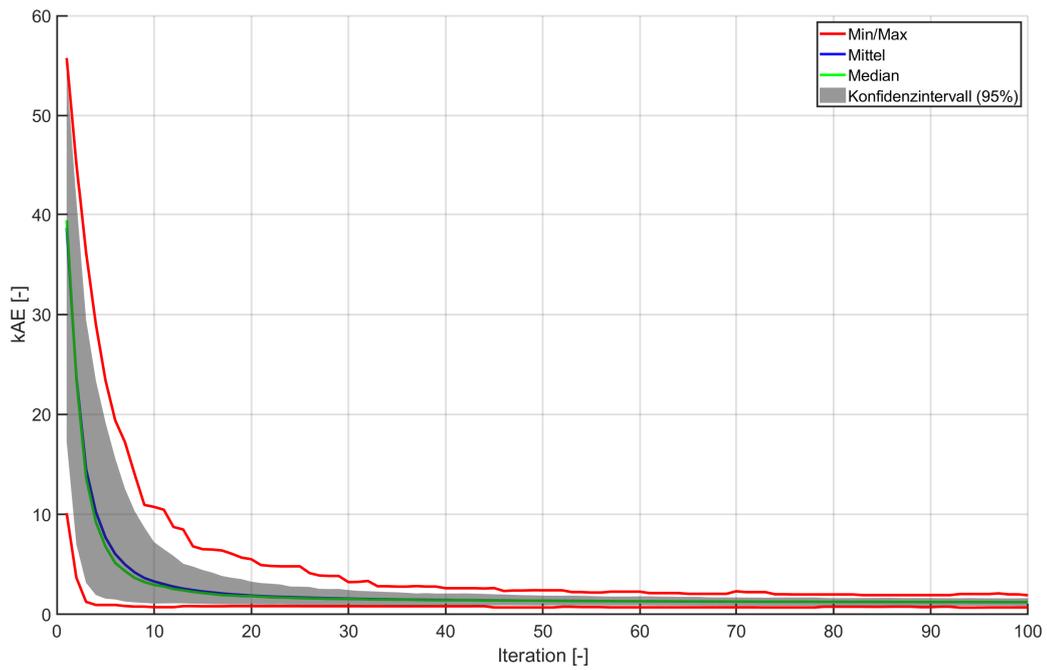


Abbildung A.13: Graphische Darstellung des kAE des EGAori bei Datensatz 4

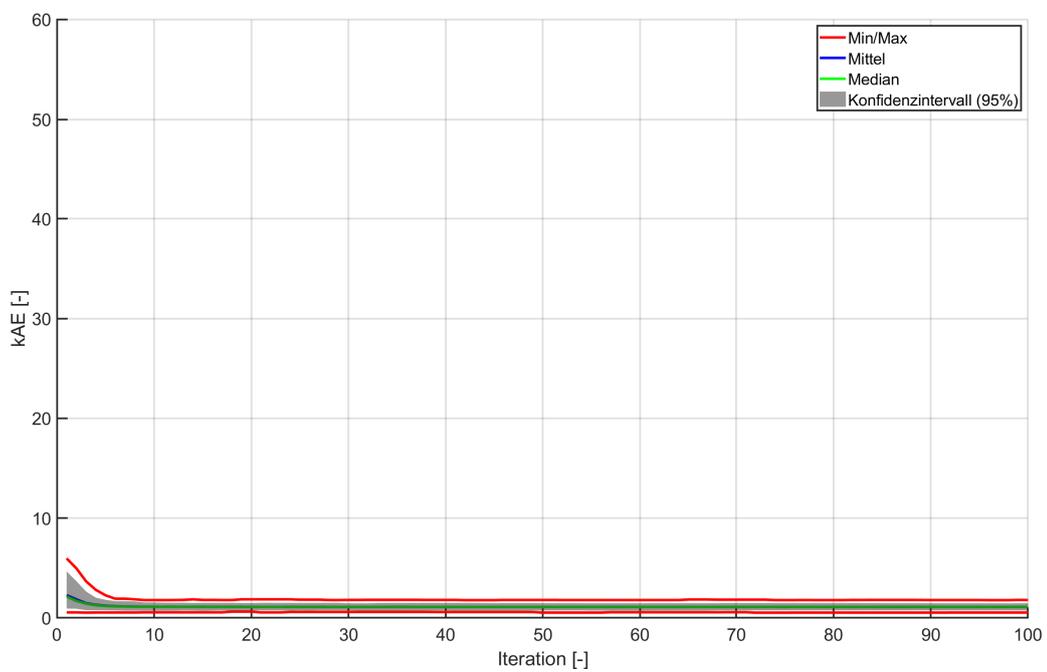


Abbildung A.14: Graphische Darstellung des kAE des EGAmoD bei Datensatz 4

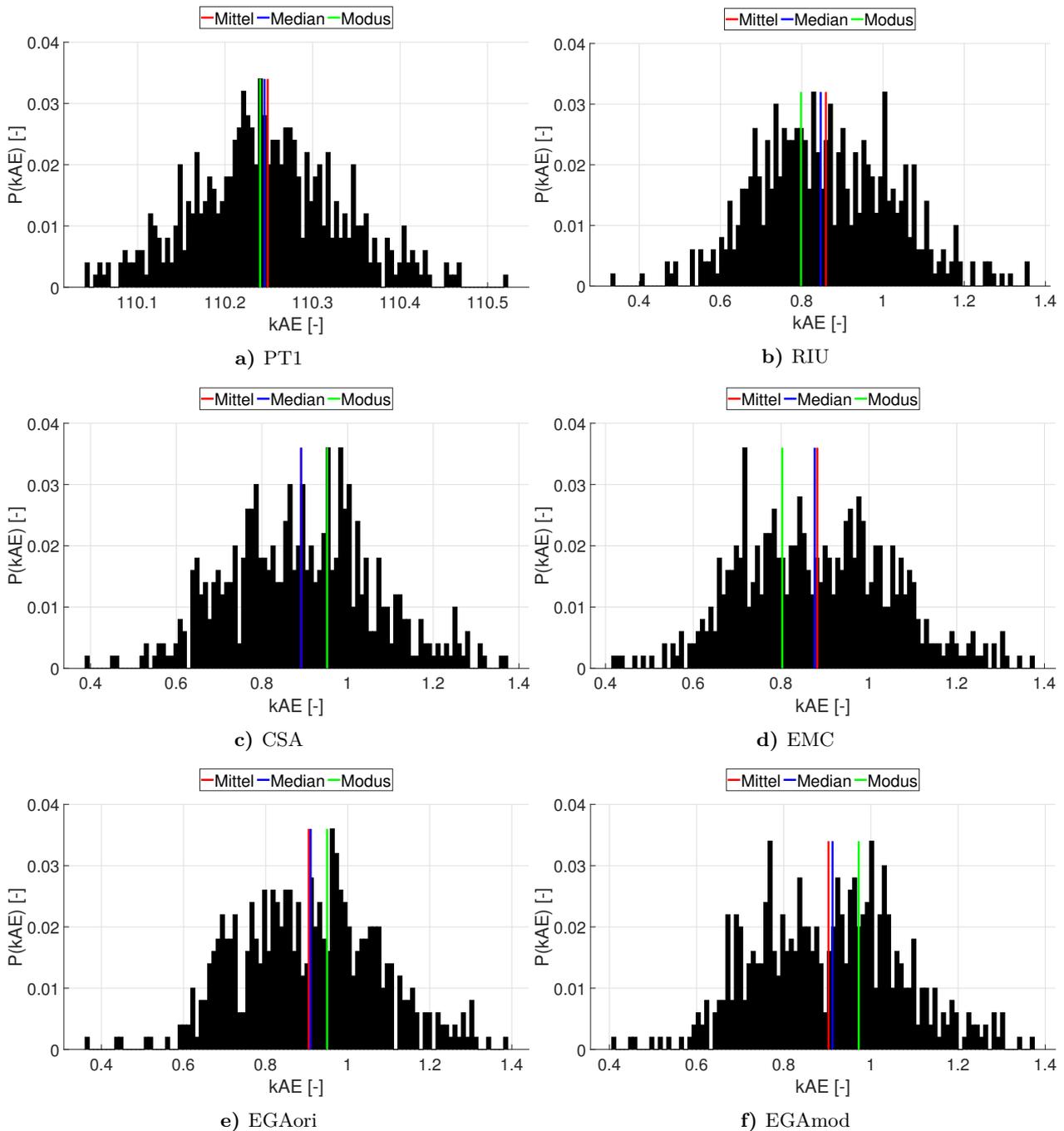


Abbildung A.15: Histogramm der Verteilung der kAE-Werte bei Datensatz 5. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.9 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

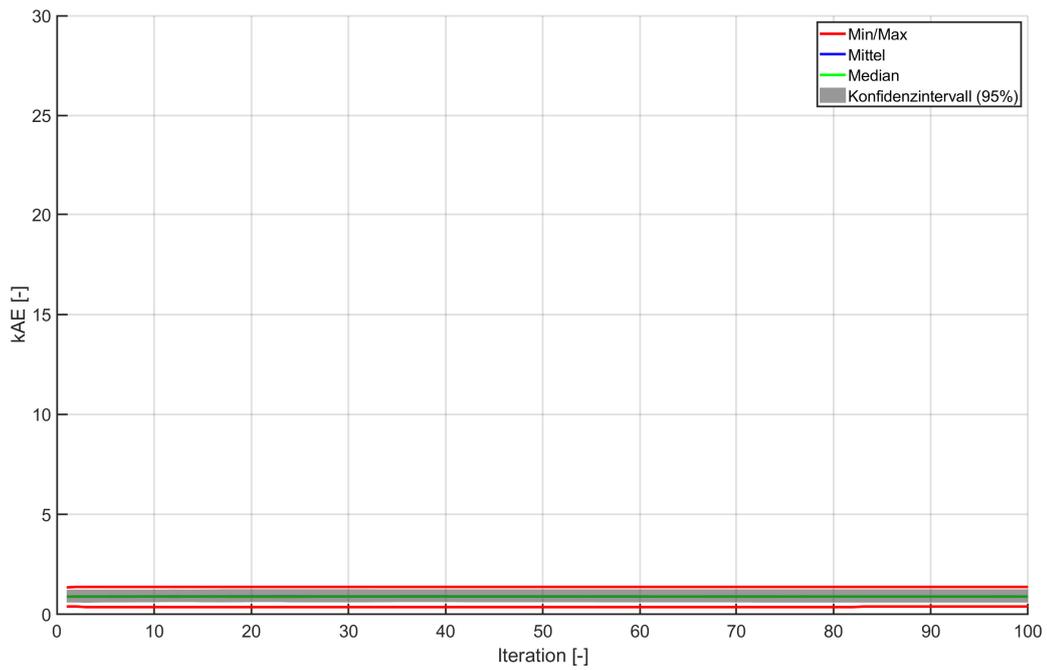


Abbildung A.16: Graphische Darstellung des kAE des CSA bei Datensatz 5

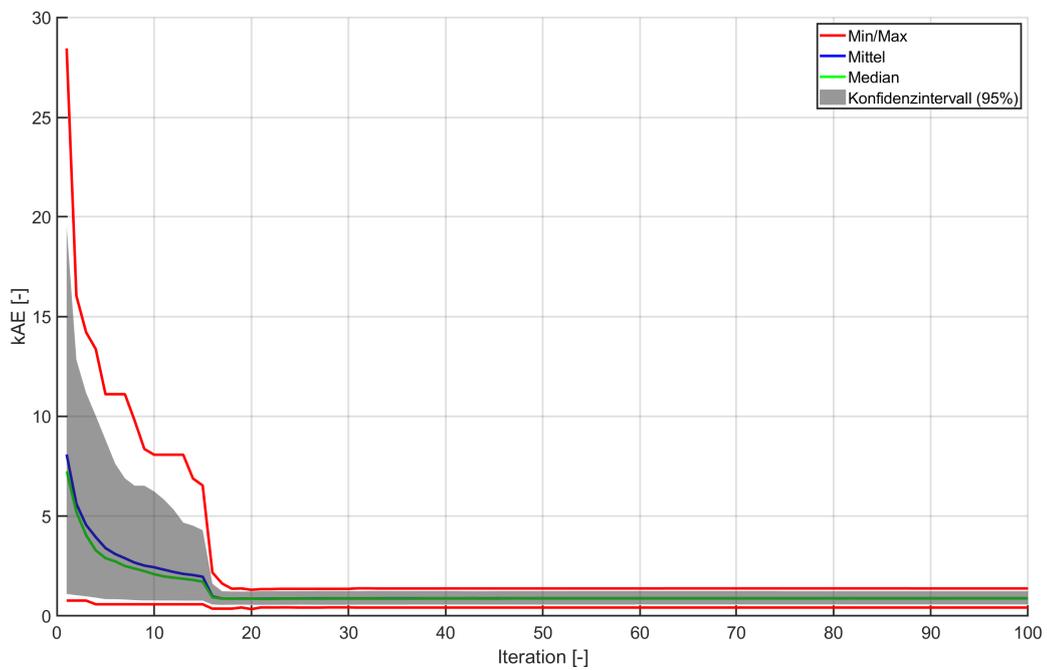


Abbildung A.17: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 5

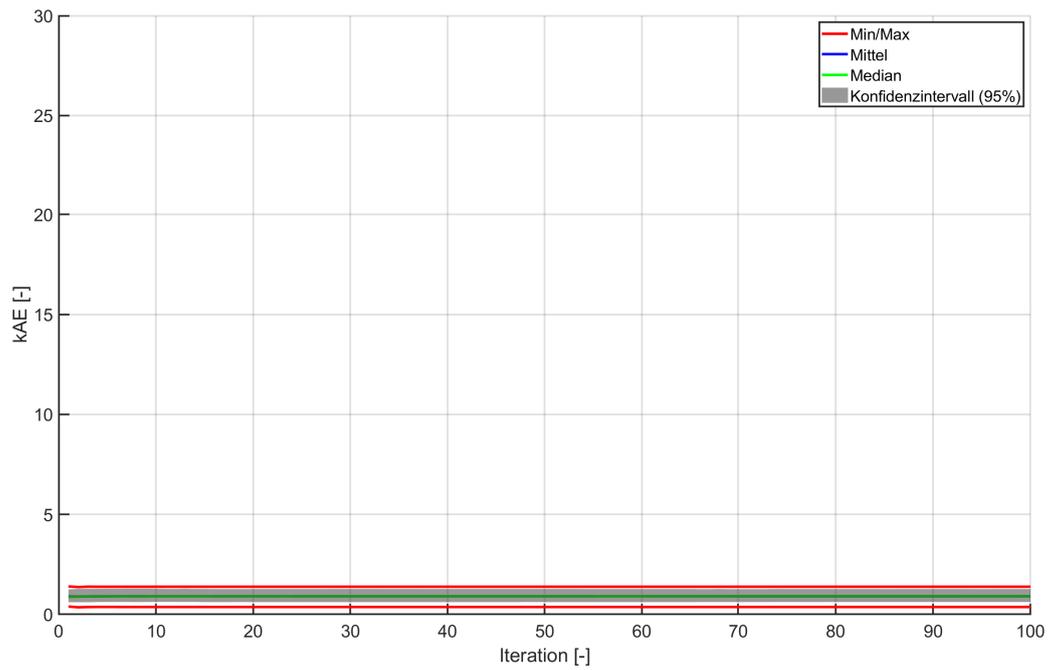


Abbildung A.18: Graphische Darstellung des kAE des EGAori bei Datensatz 5

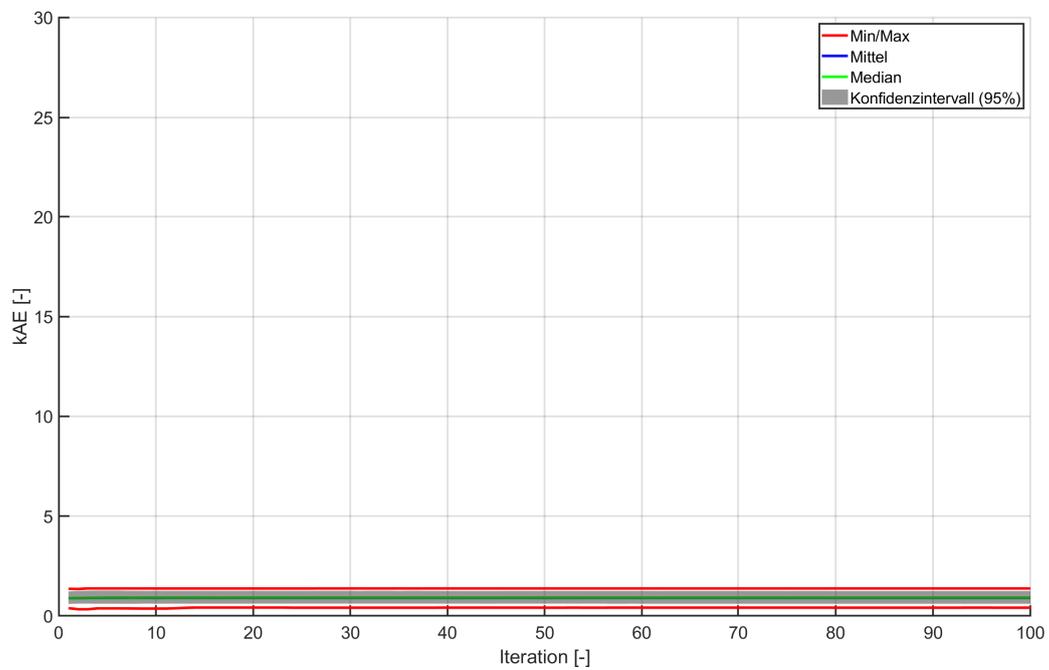


Abbildung A.19: Graphische Darstellung des kAE des EGAmoD bei Datensatz 5

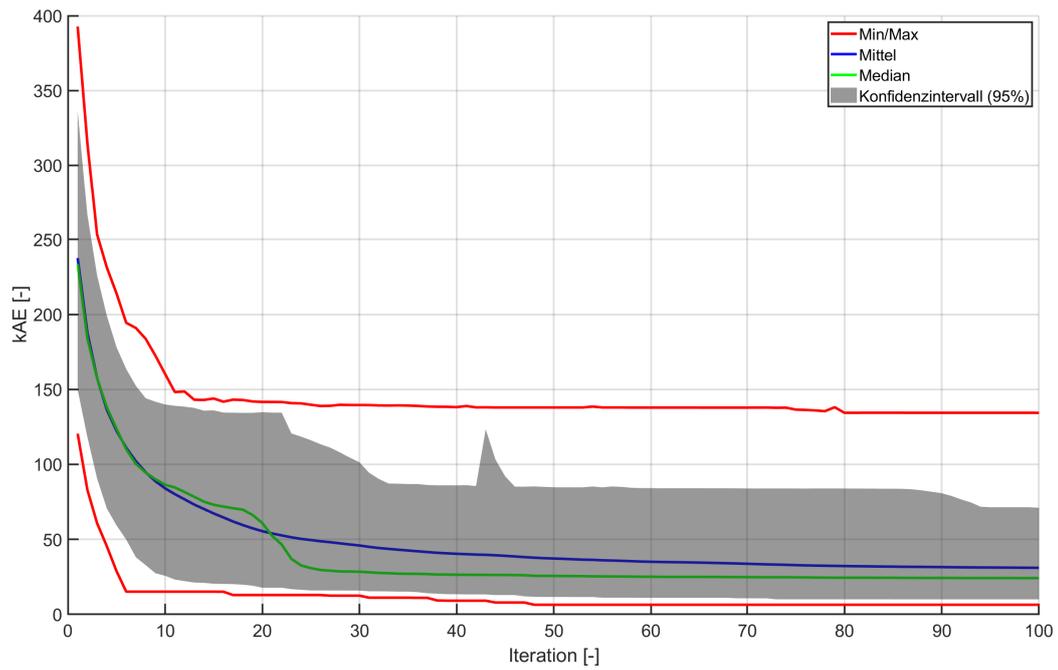


Abbildung A.20: Graphische Darstellung des kAE des CSA bei Datensatz 6

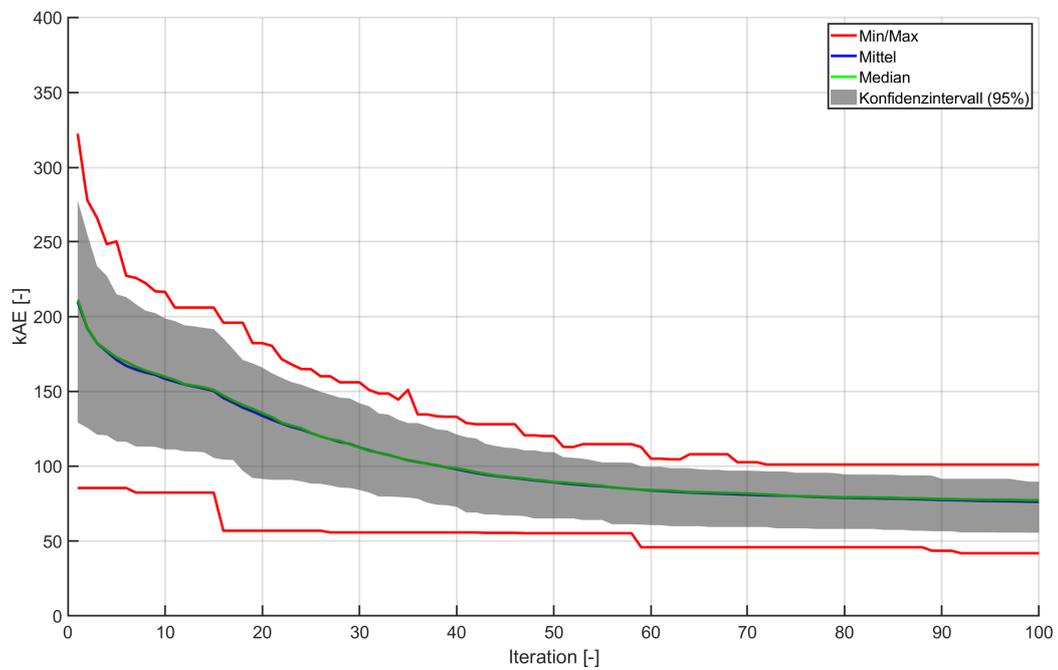


Abbildung A.21: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 6

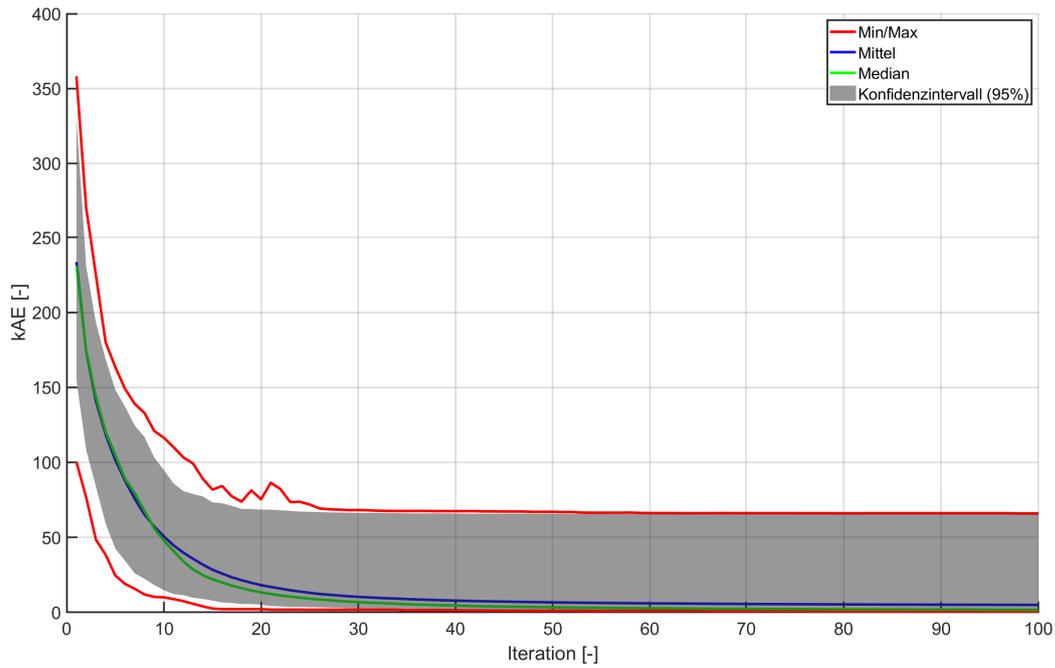


Abbildung A.22: Graphische Darstellung des kAE des EGAori bei Datensatz 6

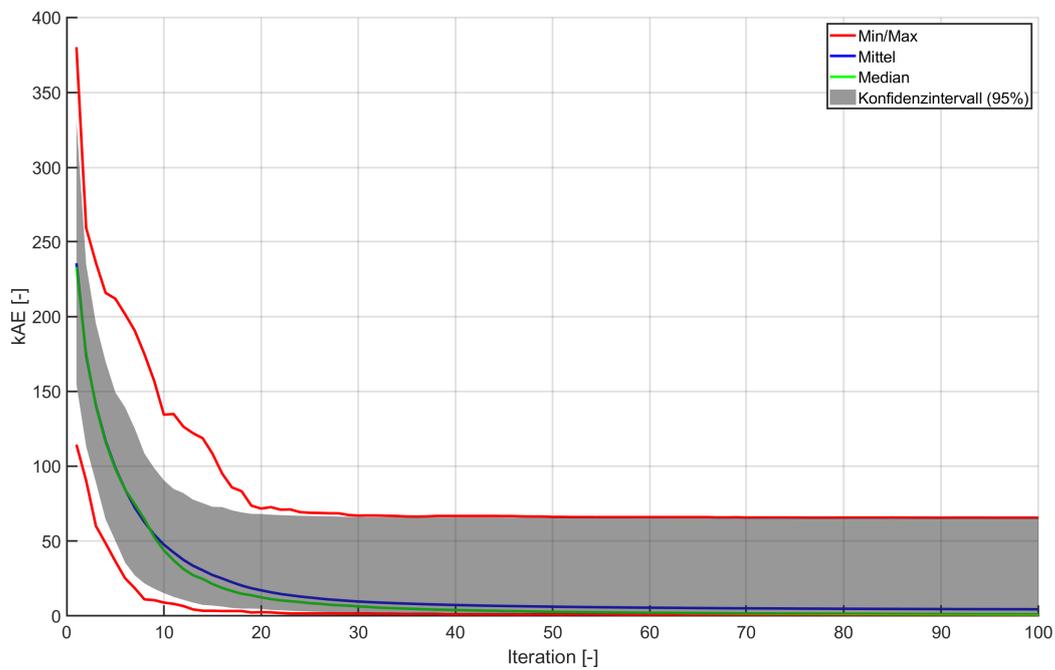


Abbildung A.23: Graphische Darstellung des kAE des EGAmoD bei Datensatz 6

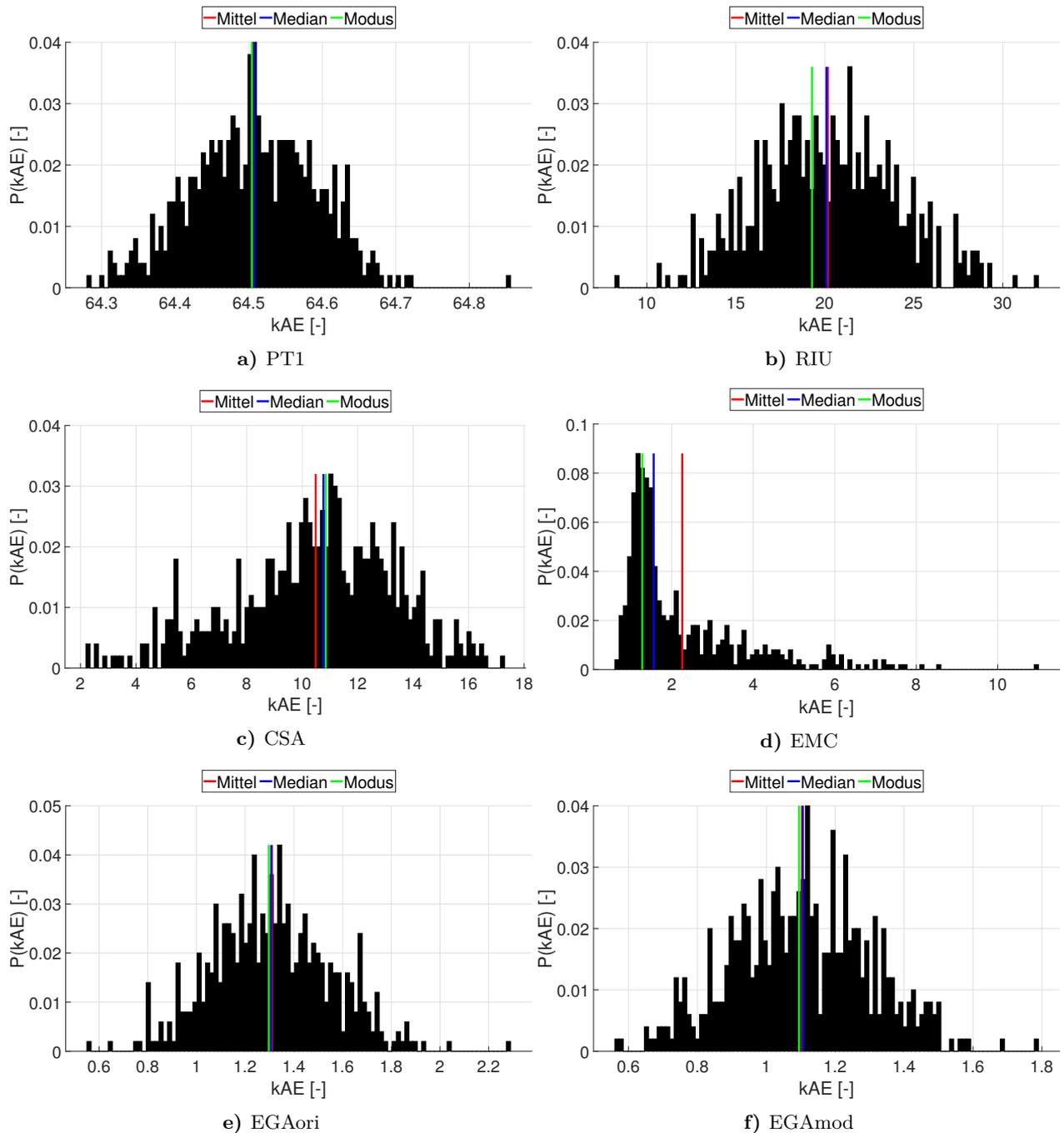


Abbildung A.24: Histogramm der Verteilung der kAE-Werte bei Datensatz 7. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle A.7 angegebenen Wert. Dieser über Kern-dichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

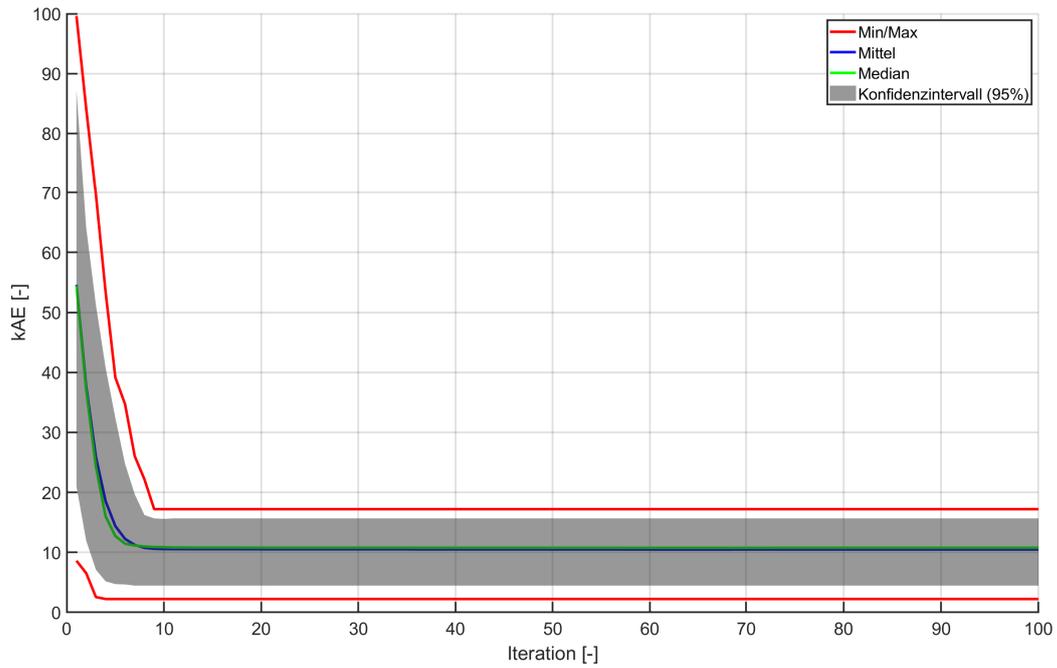


Abbildung A.25: Graphische Darstellung des kAE des CSA bei Datensatz 7

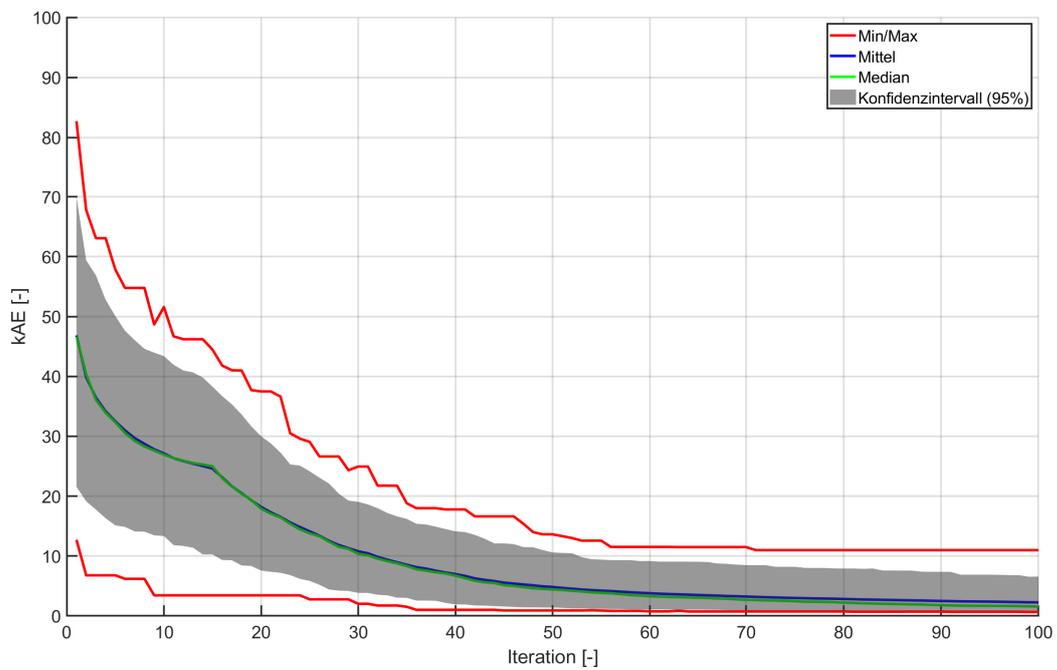


Abbildung A.26: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 7

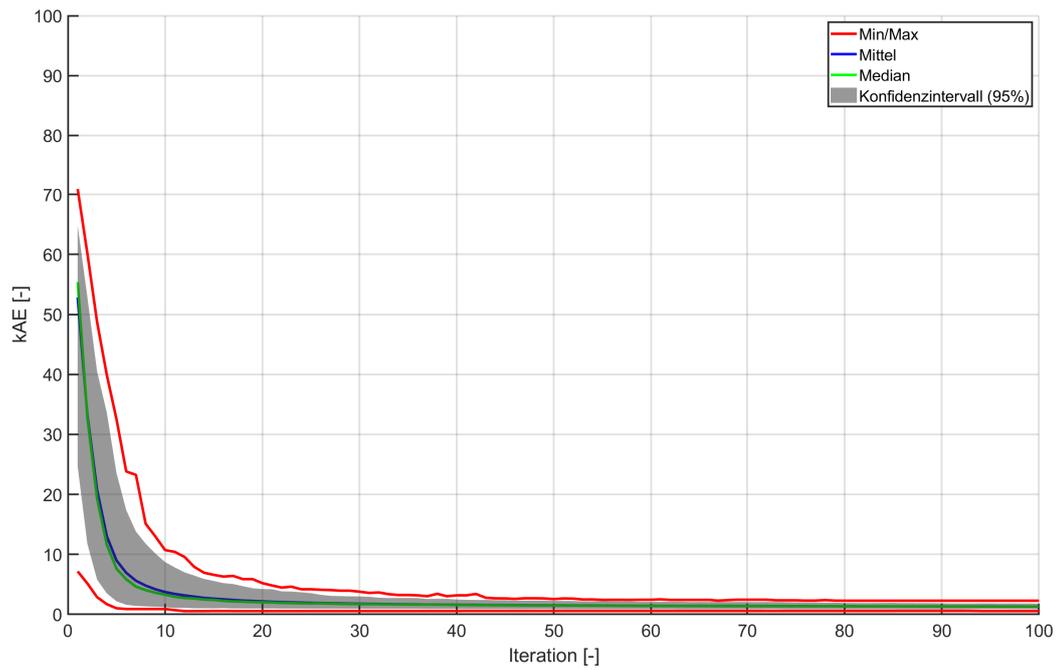


Abbildung A.27: Graphische Darstellung des kAE des EGAori bei Datensatz 7

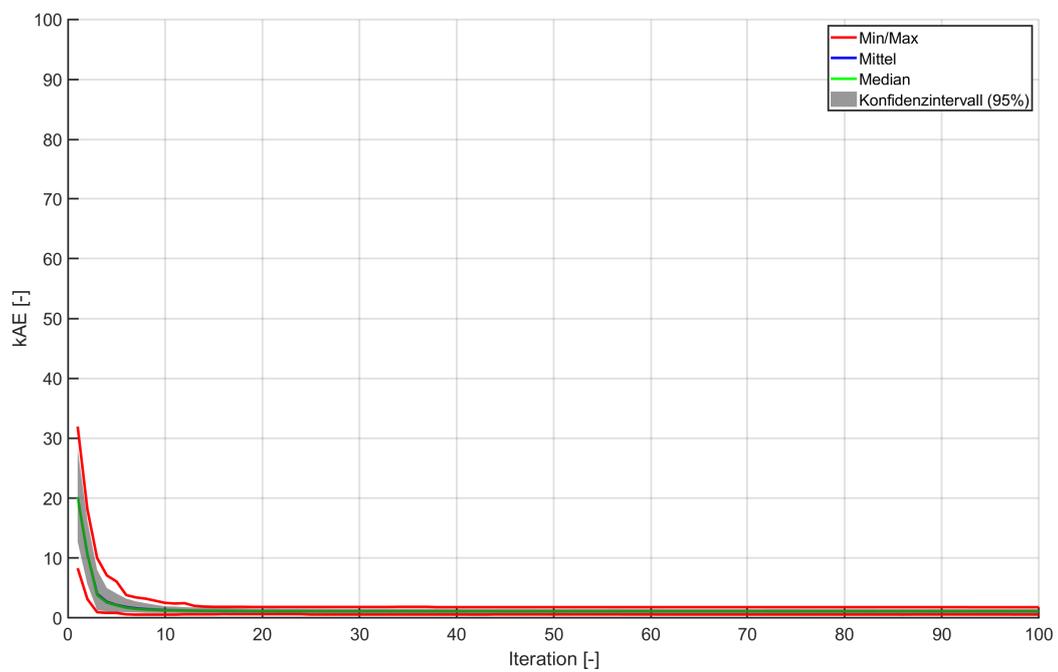


Abbildung A.28: Graphische Darstellung des kAE des EGAm0d bei Datensatz 7

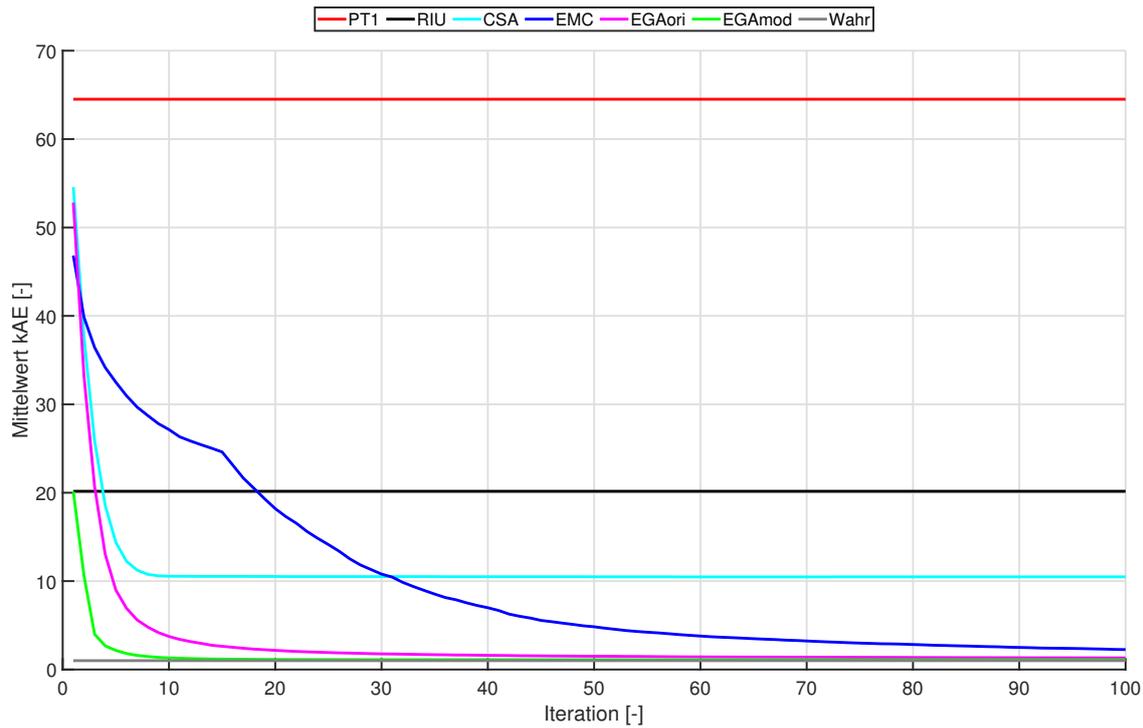


Abbildung A.29: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 7

Tabelle A.8: Kennziffern des kAE der verschiedenen Methoden bei Datensatz 8

Iter.	Methode	Min.	Max.	Mittel	Median	Modus	Std.	↓ KI (95 %)	↑ KI (95 %)
-	PT1	3.857	4.461	4.143	4.142	4.143	0.112	3.920	4.374
-	RIU	1.336	4.686	3.269	3.241	3.109	0.726	1.819	4.374
1	CSA	2.131	5.656	3.473	3.473	3.535	0.609	2.397	4.820
	EMC	1.603	4.813	3.211	3.175	3.078	0.505	2.277	4.222
	EGAori	2.059	4.995	3.476	3.443	3.393	0.528	2.462	4.443
	EGAmo	1.336	4.745	3.028	3.041	3.087	0.598	1.819	4.234
25	CSA	1.567	4.067	2.674	2.665	2.669	0.388	1.974	3.436
	EMC	1.153	2.926	2.157	2.165	2.157	0.271	1.597	2.673
	EGAori	0.780	2.950	1.587	1.492	1.419	0.391	0.996	2.541
	EGAmo	0.850	2.684	1.498	1.438	1.339	0.342	1.006	2.422
50	CSA	1.567	4.067	2.566	2.565	2.590	0.363	1.840	3.264
	EMC	1.065	2.646	1.859	1.865	1.878	0.257	1.364	2.335
	EGAori	0.787	2.947	1.543	1.468	1.401	0.377	1.017	2.488
	EGAmo	0.824	2.658	1.471	1.426	1.354	0.323	0.996	2.325
75	CSA	1.442	3.563	2.487	2.482	2.479	0.350	1.780	3.191
	EMC	1.026	2.577	1.741	1.747	1.735	0.270	1.219	2.248
	EGAori	0.777	2.780	1.519	1.449	1.387	0.357	1.001	2.483
	EGAmo	0.814	2.657	1.462	1.410	1.355	0.316	0.991	2.320
100	CSA	1.442	3.563	2.446	2.441	2.404	0.351	1.772	3.195
	EMC	0.991	2.561	1.694	1.691	1.659	0.272	1.161	2.233
	EGAori	0.793	2.779	1.510	1.448	1.388	0.348	0.998	2.418
	EGAmo	0.801	2.657	1.459	1.410	1.357	0.312	0.989	2.311
-	Wahr	0.618	1.813	1.208	1.209	1.199	0.210	0.799	1.619
-	Roh	11.957	13.208	12.545	12.543	12.528	0.215	12.123	12.992

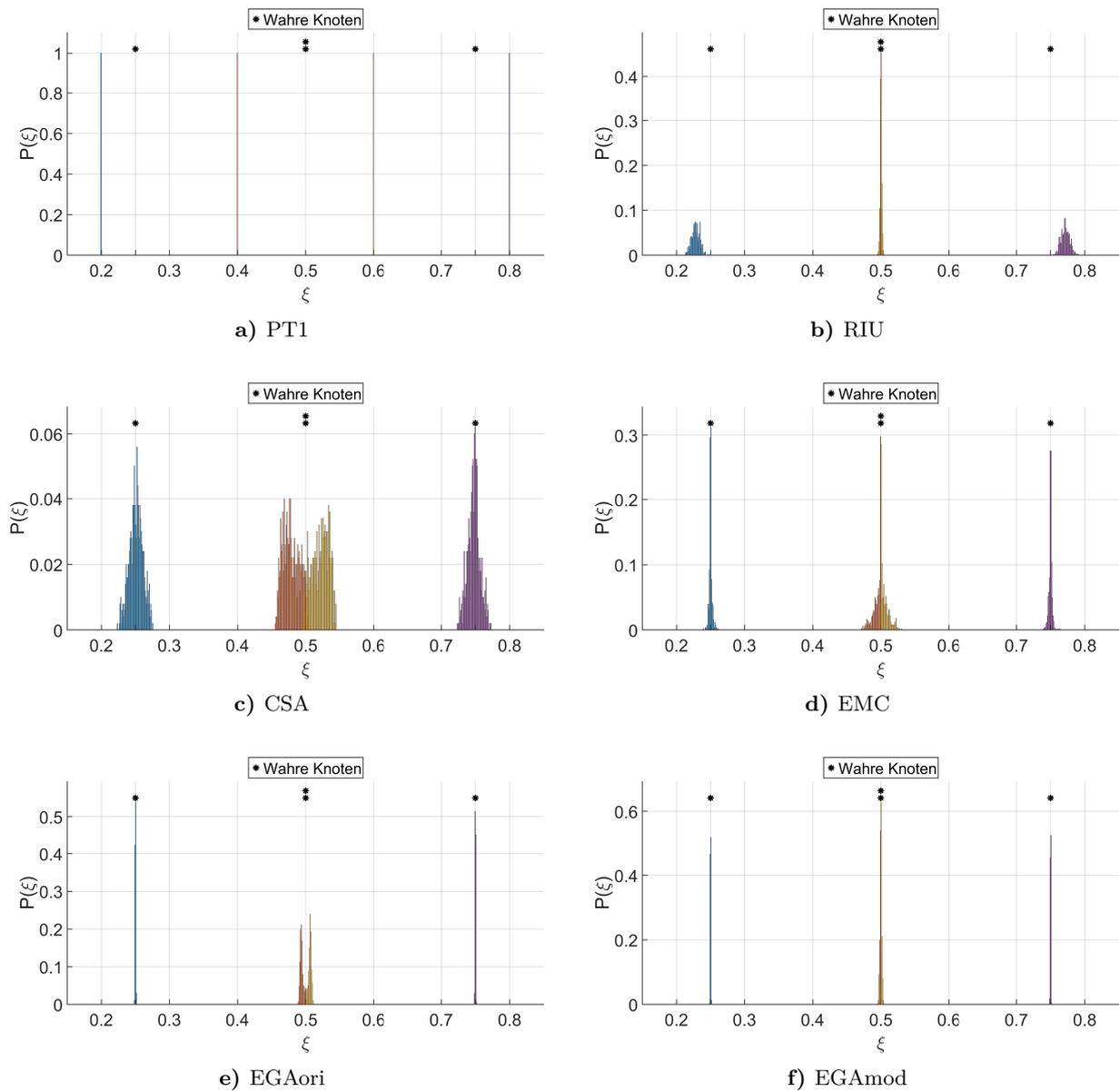


Abbildung A.30: Histogramm der Verteilung der vier internen Knoten bei Datensatz 7. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt. (Zur besseren Darstellung wurde nicht der gesamte Wertebereich zwischen 0 und 1 dargestellt, sondern nur ein Teilausschnitt. In nicht dargestellten Bereichen ist die relative Häufigkeit $P(\xi) = 0$.)

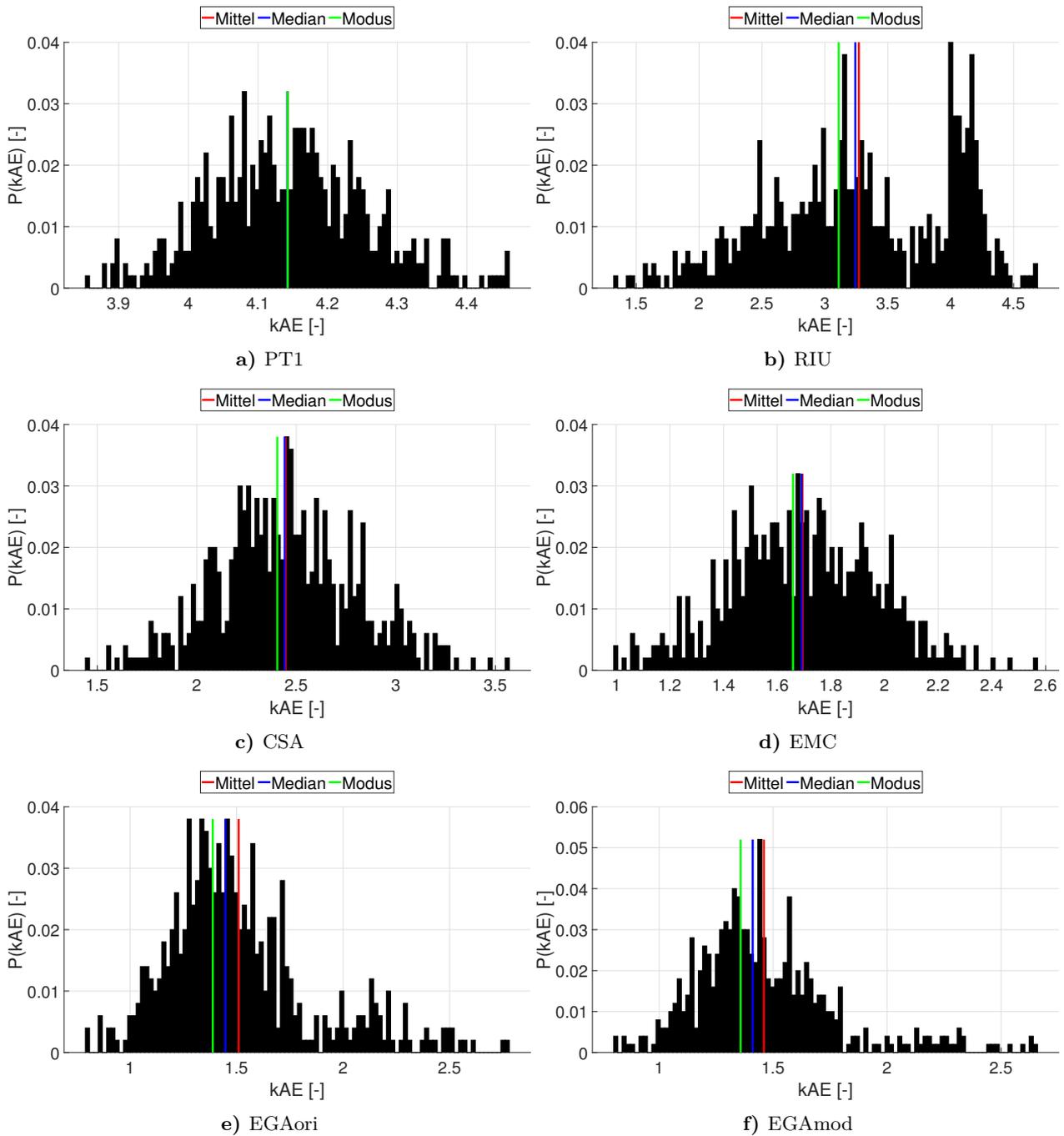


Abbildung A.31: Histogramm der Verteilung der kAE-Werte bei Datensatz 8. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle A.8 angegebenen Wert. Dieser über Kern-dichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

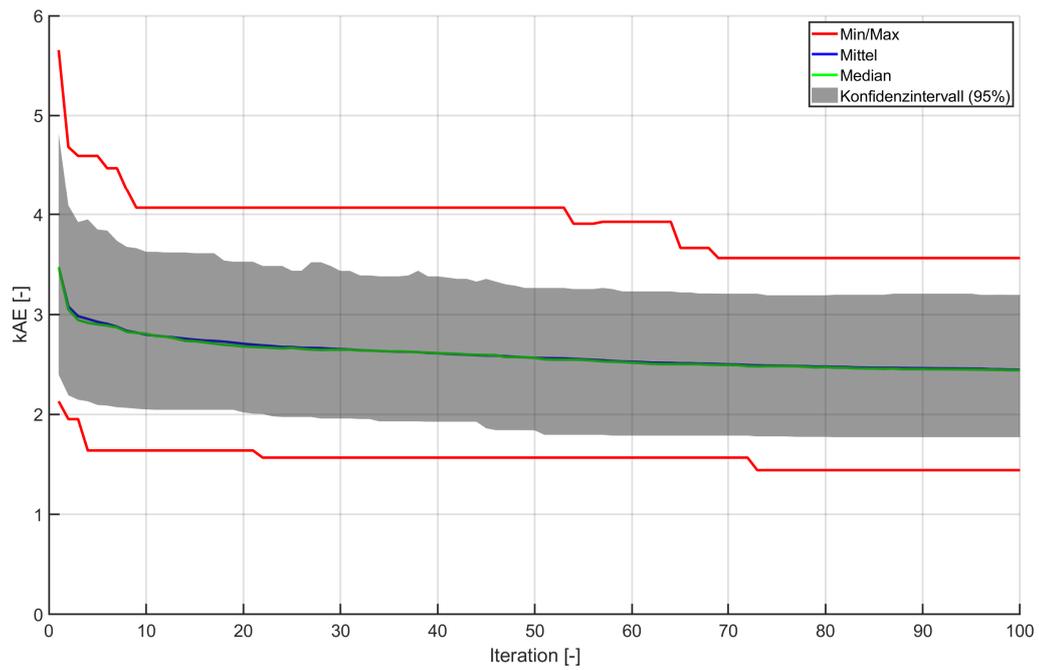


Abbildung A.32: Graphische Darstellung des kAE des CSA bei Datensatz 8

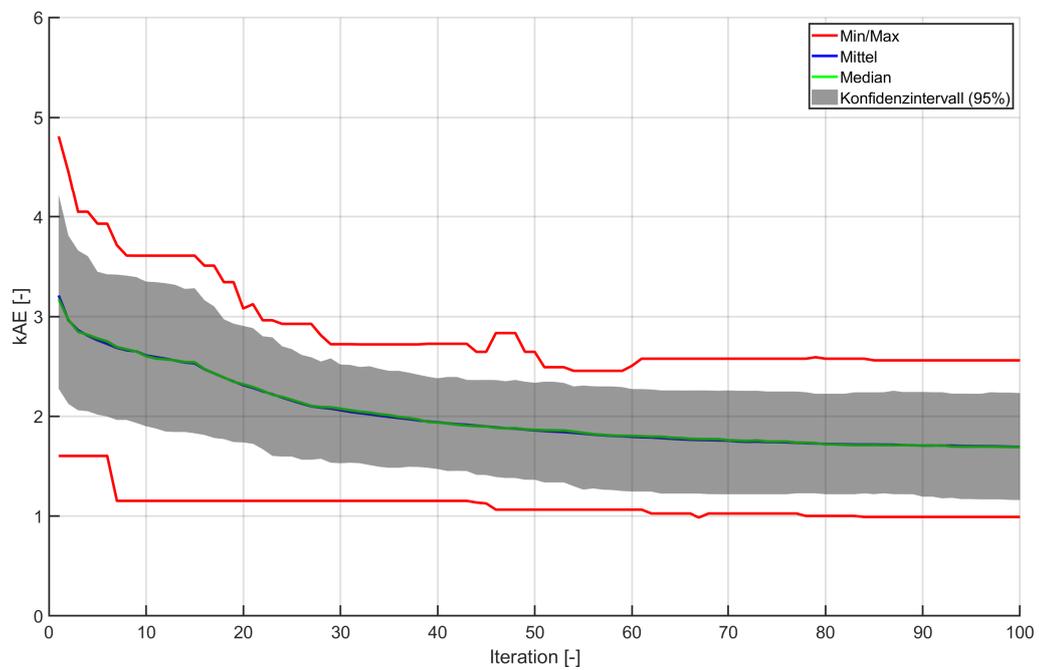


Abbildung A.33: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 8

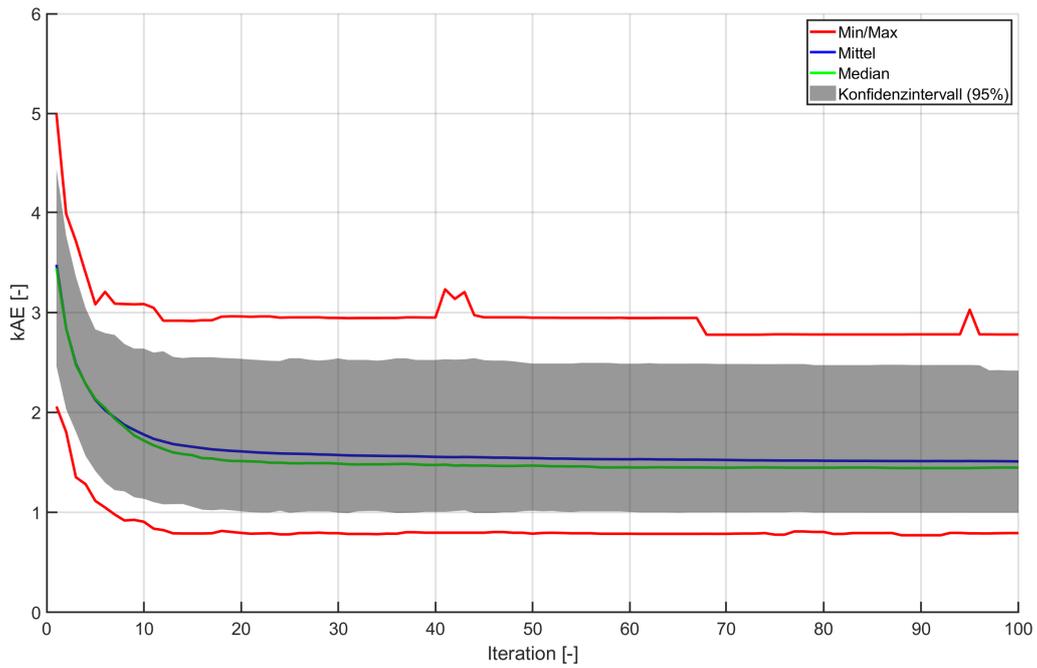


Abbildung A.34: Graphische Darstellung des kAE des EGAori bei Datensatz 8

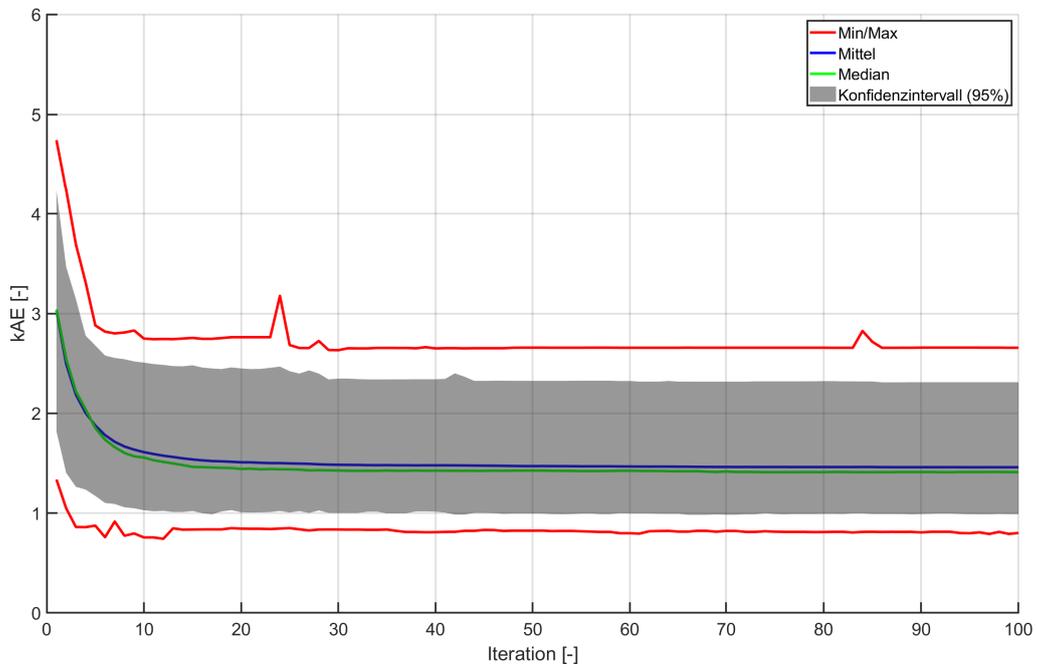


Abbildung A.35: Graphische Darstellung des kAE des EGAmoD bei Datensatz 8

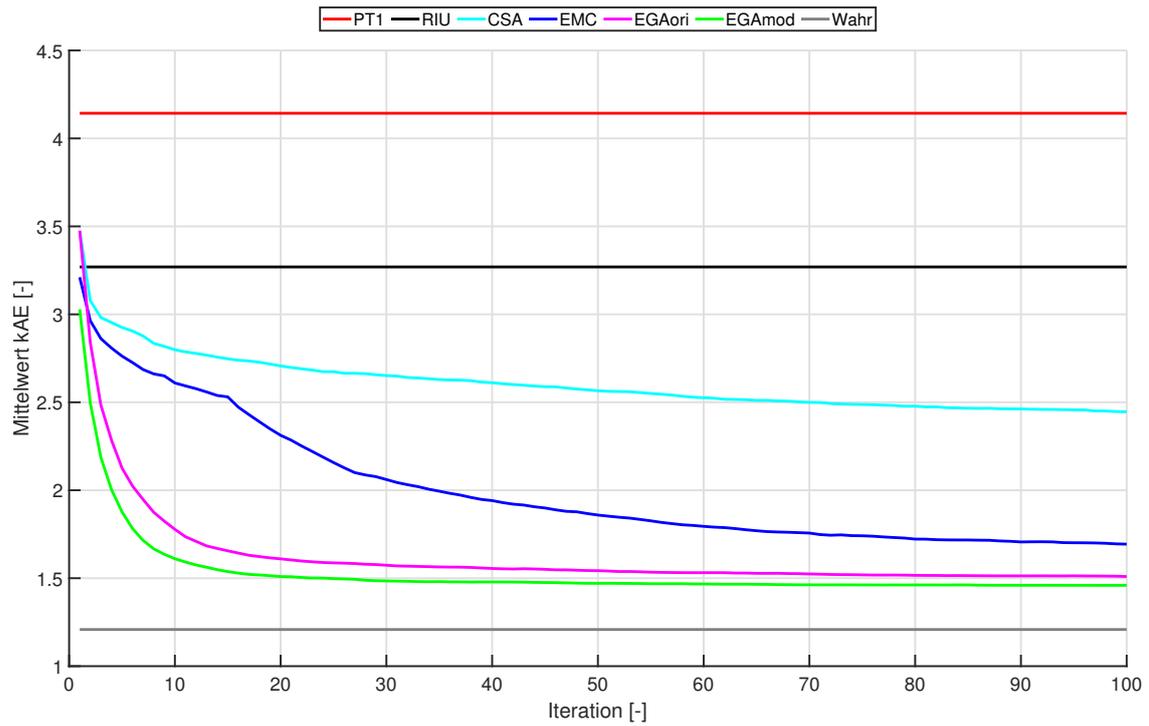


Abbildung A.36: Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 8

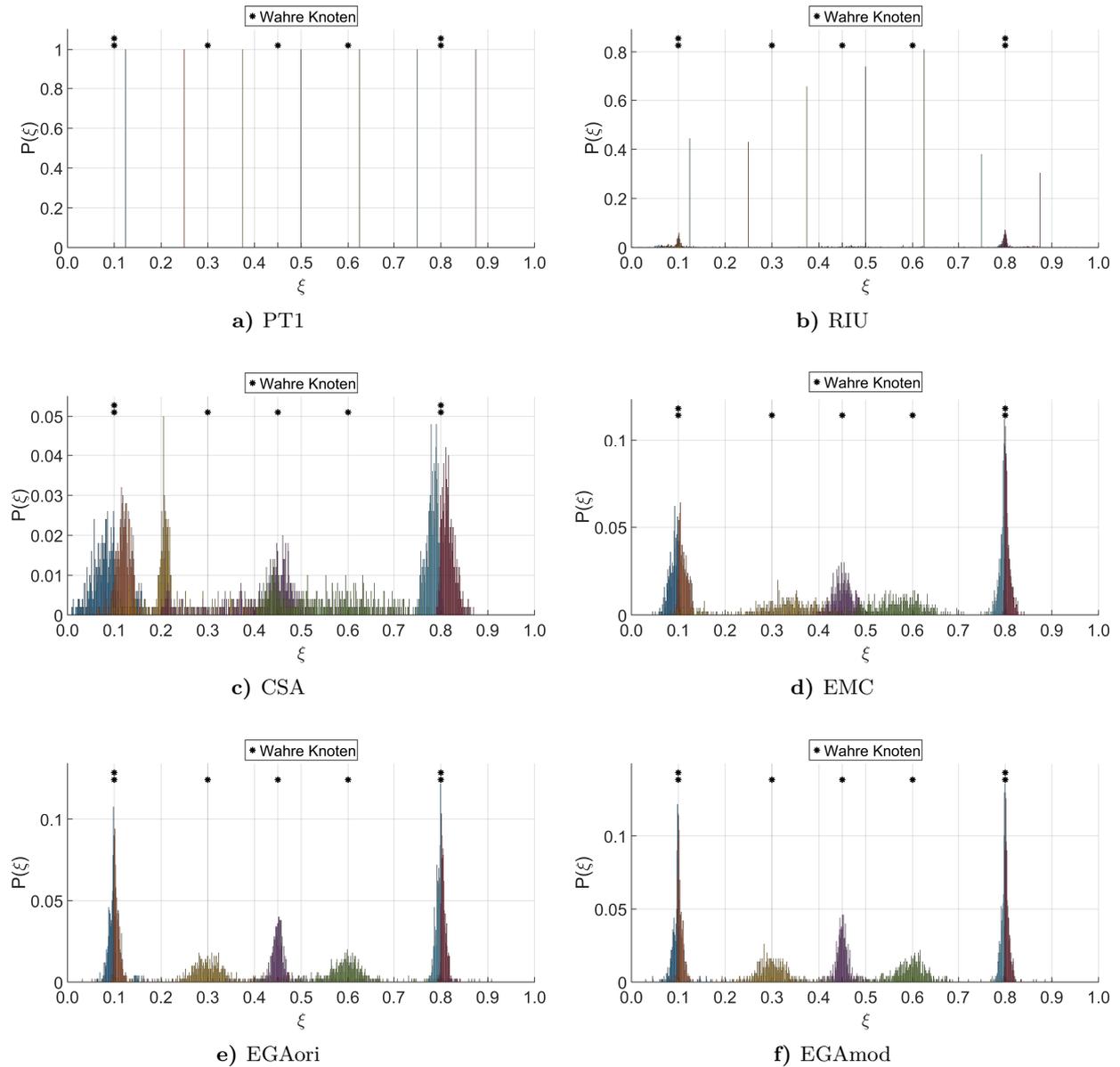


Abbildung A.37: Histogramm der Verteilung der sieben internen Knoten bei Datensatz 8. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.

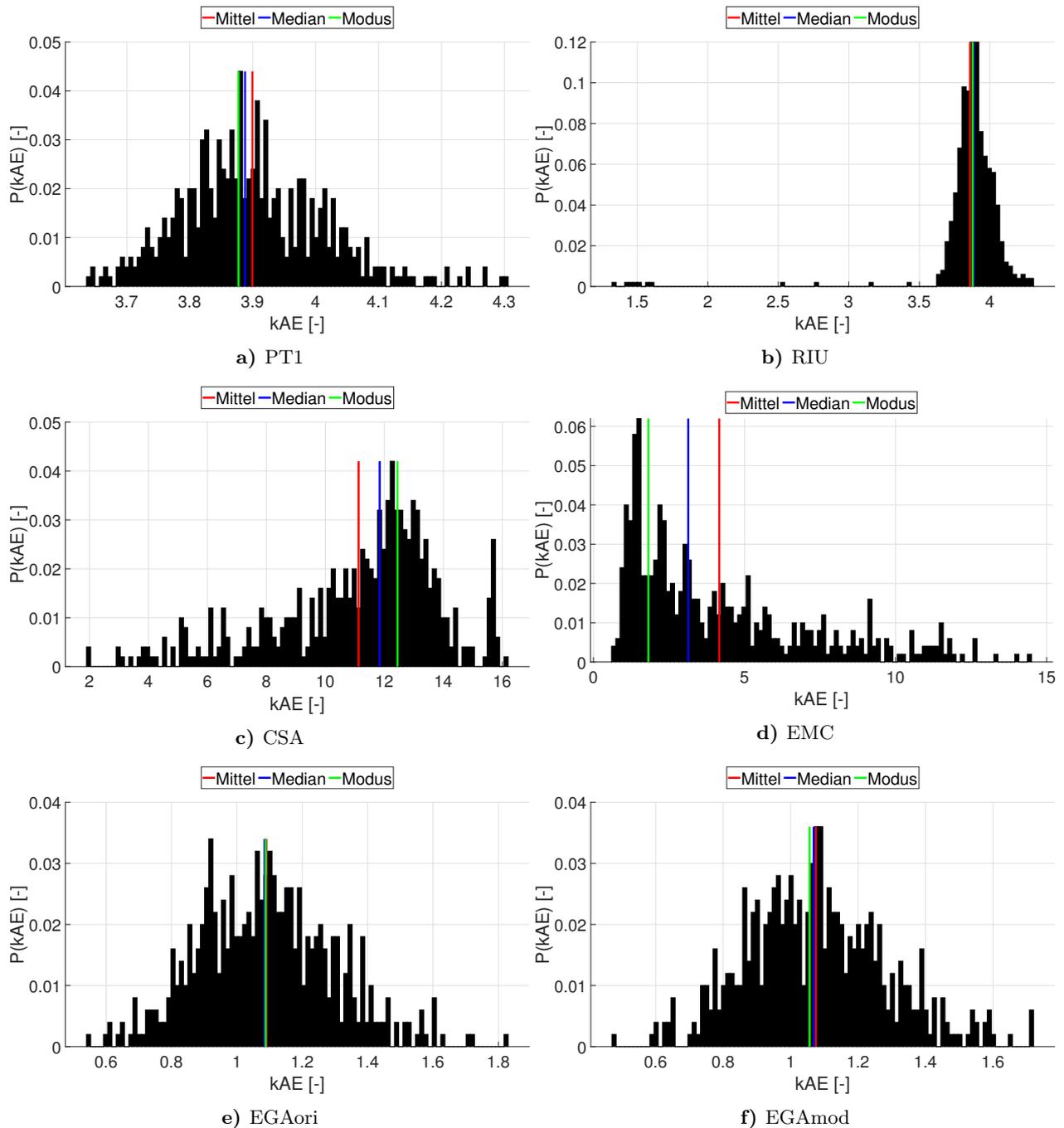


Abbildung A.38: Histogramm der Verteilung der kAE-Werte bei Datensatz 9. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.11 angegebenen Wert. Dieser über Kern-dichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.

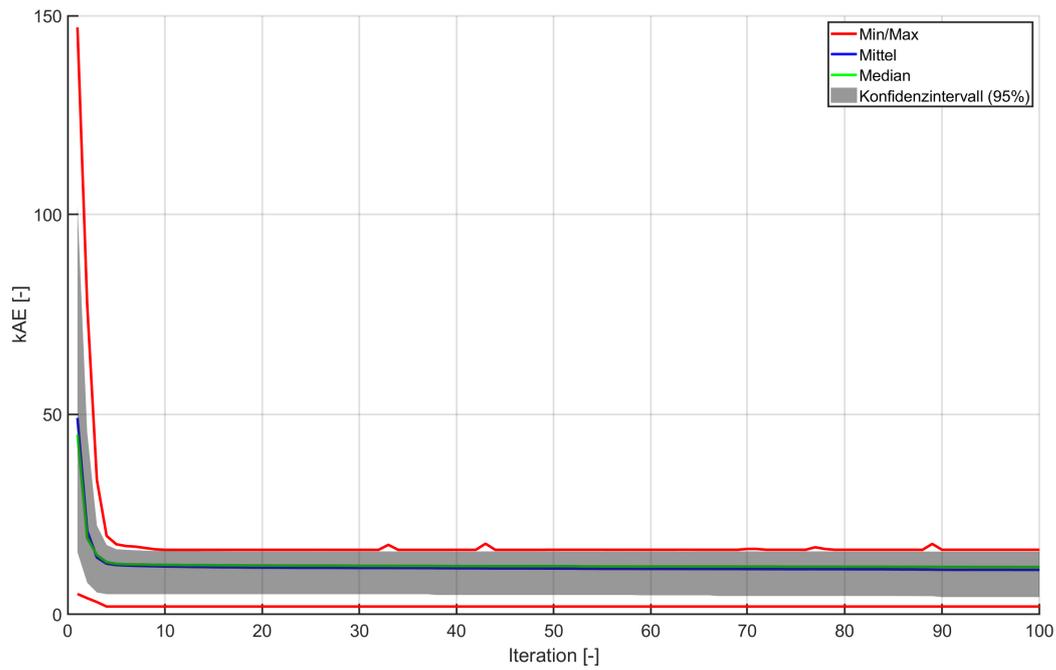


Abbildung A.39: Graphische Darstellung des kAE des CSA bei Datensatz 9

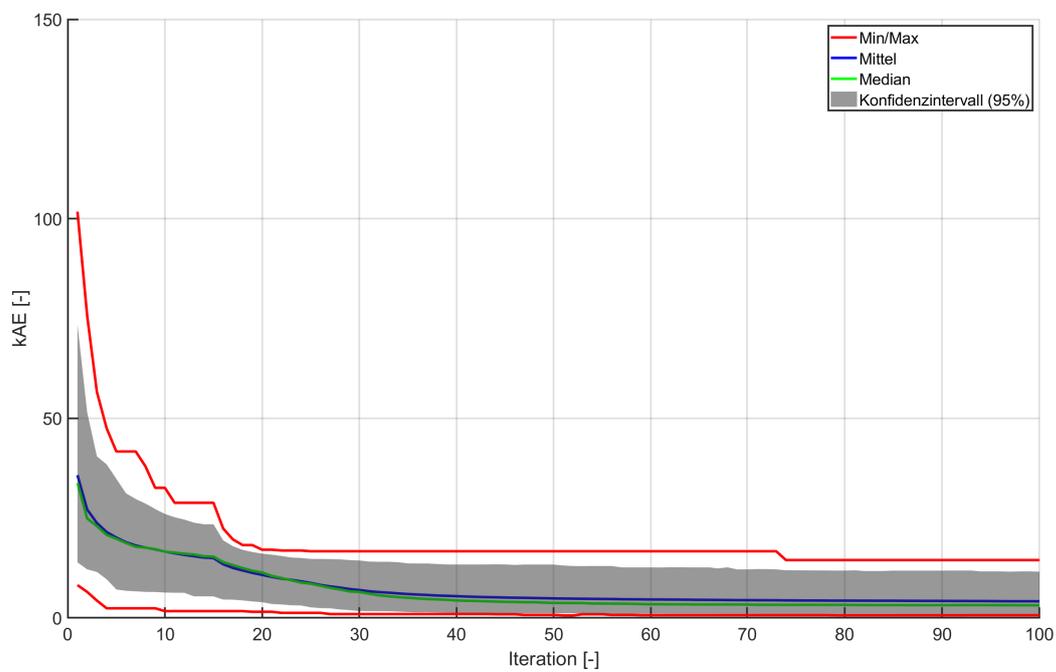


Abbildung A.40: Graphische Darstellung des kAE der EMC-Methode bei Datensatz 9

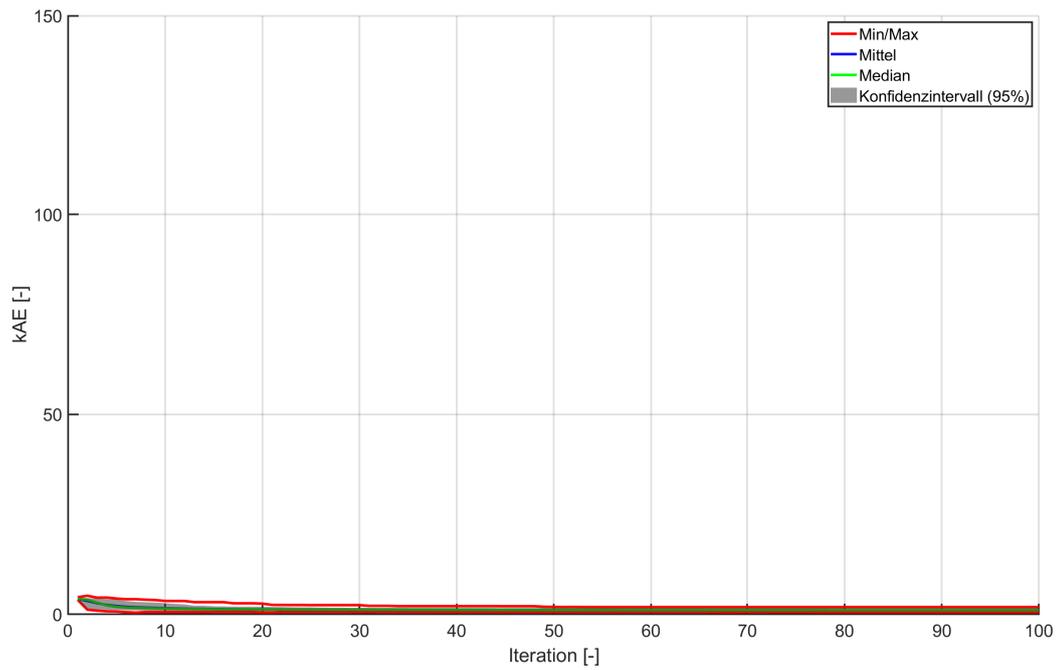


Abbildung A.41: Graphische Darstellung des kAE des EGAori bei Datensatz 9

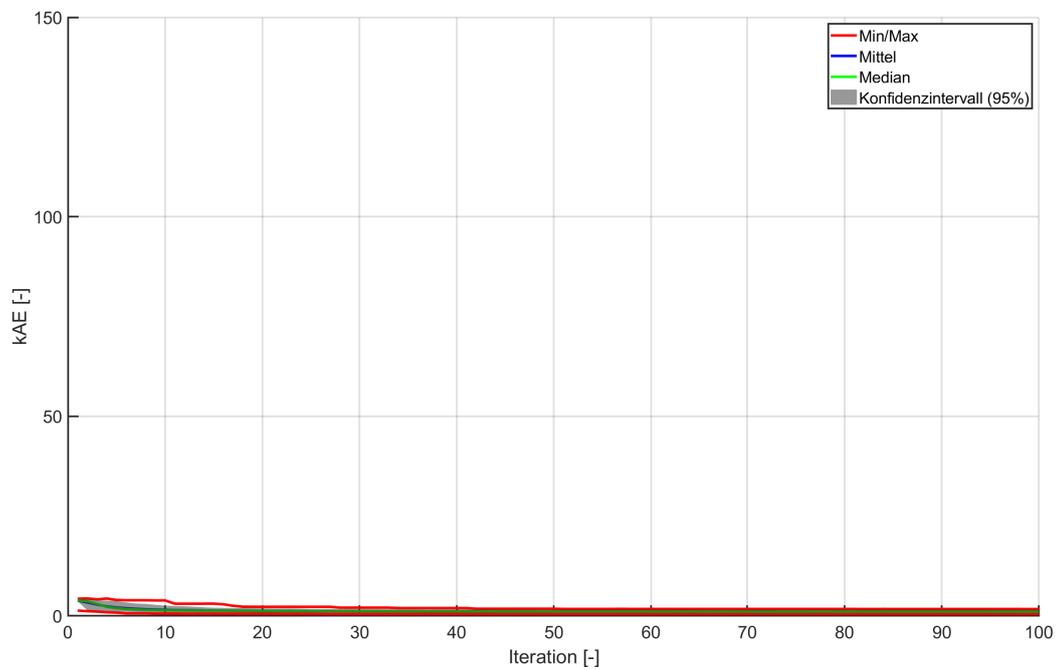


Abbildung A.42: Graphische Darstellung des kAE des EGAm0d bei Datensatz 9

A.3 Numerische Beurteilung der B-Spline-Approximation bei Ausreißern

Tabelle A.9: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIb (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 5 % der Messwerte)

DS.	Methode	RM I		RM IIIb					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.500	54.492	53.227	53.217	55.694	55.611
	RIU	2.283	1.689	3.239	3.140	3.010	2.702	2.997	2.686
	CSA	8.027	8.873	5.589	5.839	5.108	5.329	4.964	5.092
	EMC	1.526	1.174	1.977	1.802	1.630	1.427	1.609	1.415
	EGAori	1.225	1.245	1.733	1.758	1.387	1.361	1.388	1.371
	EGAmo	1.097	1.081	1.621	1.668	1.268	1.262	1.274	1.226
	Wahr	1.018	1.006	1.512	1.551	1.129	1.120	1.136	1.159
5	PT1	110.249	110.240	110.258	110.252	99.231	99.225	102.939	103.283
	RIU	0.859	0.798	1.343	1.180	1.035	0.958	1.028	0.939
	CSA	0.892	0.952	1.298	1.257	0.986	0.918	0.976	0.945
	EMC	0.882	0.803	1.287	1.325	0.982	0.960	1.009	0.920
	EGAori	0.905	0.950	1.309	1.300	0.998	0.939	0.986	0.934
	EGAmo	0.903	0.972	1.306	1.290	0.997	0.921	0.985	0.923
	Wahr	0.790	0.704	1.144	1.197	0.866	0.846	0.871	0.845
9	PT1	3.900	3.878	4.085	4.079	3.934	3.929	3.919	3.900
	RIU	3.859	3.878	4.071	4.080	3.919	3.931	3.905	3.901
	CSA	11.127	12.446	8.450	8.840	8.323	8.687	8.306	8.512
	EMC	4.164	1.817	4.430	2.162	4.234	1.909	4.251	1.921
	EGAori	1.090	1.088	1.567	1.540	1.271	1.271	1.285	1.237
	EGAmo	1.075	1.056	1.558	1.534	1.260	1.283	1.275	1.325
	Wahr	0.954	0.929	1.386	1.398	1.044	1.019	1.058	1.015

Tabelle A.10: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVb (5% einseitige Ausreißer, zufällige Position)

DS.	Methode	RM I		RM IVb					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	54.606	54.591	53.363	53.359	55.741	55.769
	RIU	2.283	1.689	3.985	3.815	3.482	3.296	3.008	2.783
	CSA	8.027	8.873	7.720	8.519	7.003	7.442	6.426	6.782
	EMC	1.526	1.174	3.014	2.878	2.318	2.158	1.641	1.370
	EGAori	1.225	1.245	2.831	2.881	2.122	2.159	1.432	1.486
	EGAmo	1.097	1.081	2.733	2.793	2.008	2.055	1.305	1.367
	Wahr	1.018	1.006	2.690	2.734	1.947	1.985	1.214	1.165
5	PT1	110.249	110.240	110.267	110.254	99.284	99.281	102.742	103.046
	RIU	0.859	0.798	1.983	1.939	1.502	1.490	1.056	0.992
	CSA	0.892	0.952	1.985	1.958	1.507	1.528	1.053	1.023
	EMC	0.882	0.803	1.982	1.912	1.502	1.483	1.052	1.045
	EGAori	0.905	0.950	1.993	1.938	1.517	1.550	1.065	1.051
	EGAmo	0.903	0.972	1.993	1.933	1.517	1.397	1.066	1.062
	Wahr	0.790	0.704	1.909	1.841	1.416	1.361	0.953	0.863
9	PT1	3.900	3.878	4.654	4.619	4.295	4.245	3.962	3.906
	RIU	3.859	3.878	4.646	4.607	4.286	4.242	3.951	3.906
	CSA	11.127	12.446	10.640	11.660	10.446	11.448	10.203	11.095
	EMC	4.164	1.817	5.188	3.146	4.723	2.530	4.267	1.940
	EGAori	1.090	1.088	2.703	2.656	1.984	1.934	1.301	1.298
	EGAmo	1.075	1.056	2.698	2.625	1.977	1.943	1.291	1.307
	Wahr	0.954	0.929	2.648	2.600	1.897	1.844	1.149	1.151

Tabelle A.11: Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Vc (10 % einseitige Ausreißer, nacheinander)

DS.	Methode	RM I		RM Vc					
		MdkQ-Schätzer		MdkQ-Schätzer		Huber-Schätzer		Hampel-Schätzer	
		Mittel	Modus	Mittel	Modus	Mittel	Modus	Mittel	Modus
4	PT1	54.469	54.470	56.574	56.566	55.326	55.228	58.812	58.864
	RIU	2.283	1.689	9.709	9.807	9.842	9.884	10.293	10.154
	CSA	8.027	8.873	13.549	13.227	13.238	12.852	13.569	14.433
	EMC	1.526	1.174	8.816	8.524	8.918	8.628	9.291	8.925
	EGAori	1.225	1.245	8.623	8.526	8.744	8.655	9.135	8.919
	EGAmo	1.097	1.081	8.547	8.405	8.671	8.491	9.067	8.846
	Wahr	1.018	1.006	8.759	8.619	8.904	8.789	9.383	9.234
5	PT1	110.249	110.240	110.679	110.810	99.847	99.425	105.571	104.593
	RIU	0.859	0.798	5.912	5.985	6.031	6.109	6.499	6.123
	CSA	0.892	0.952	5.739	5.730	5.800	5.762	5.976	5.837
	EMC	0.882	0.803	5.735	5.765	5.799	5.768	5.981	5.866
	EGAori	0.905	0.950	5.736	5.779	5.796	5.801	5.964	5.850
	EGAmo	0.903	0.972	5.736	5.777	5.796	5.805	5.964	5.851
	Wahr	0.790	0.704	5.870	5.925	6.026	6.143	6.669	7.041
9	PT1	3.900	3.878	10.394	10.979	10.568	11.158	11.297	11.881
	RIU	3.859	3.878	10.381	10.942	10.548	11.137	11.246	11.866
	CSA	11.127	12.446	15.791	16.370	15.805	16.404	16.373	16.737
	EMC	4.164	1.817	10.755	9.348	10.857	9.635	11.326	10.718
	EGAori	1.090	1.088	8.692	8.592	8.826	8.643	9.309	8.831
	EGAmo	1.075	1.056	8.691	8.613	8.824	8.662	9.306	8.854
	Wahr	0.954	0.929	9.105	9.307	9.315	9.505	10.198	10.351

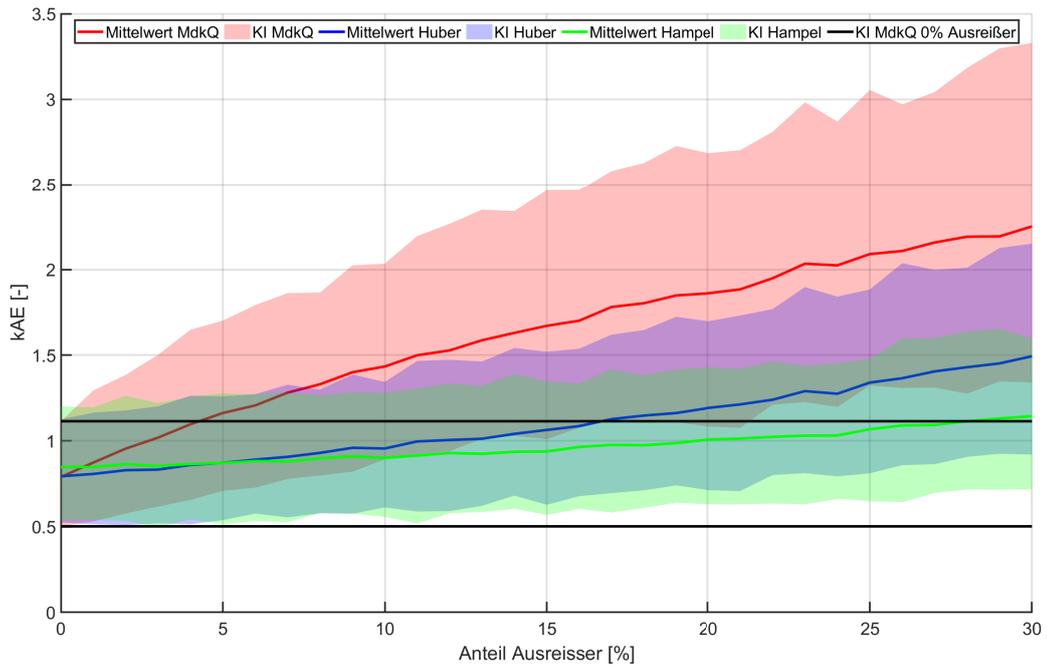


Abbildung A.43: Belastbarkeit Rauschmodell III Datensatz 5

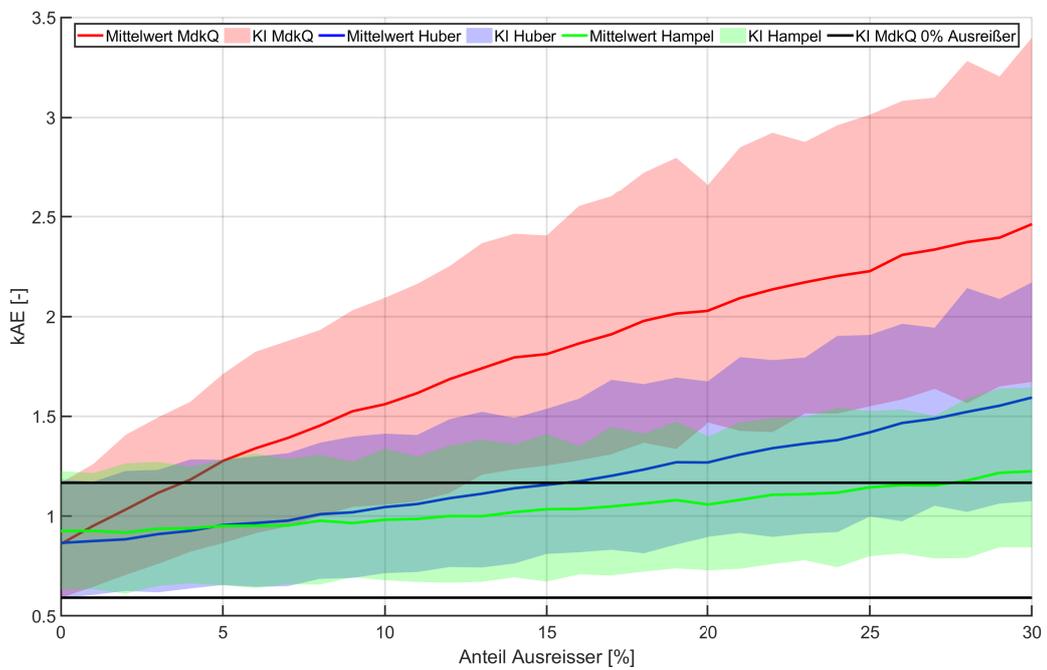


Abbildung A.44: Belastbarkeit Rauschmodell III Datensatz 6

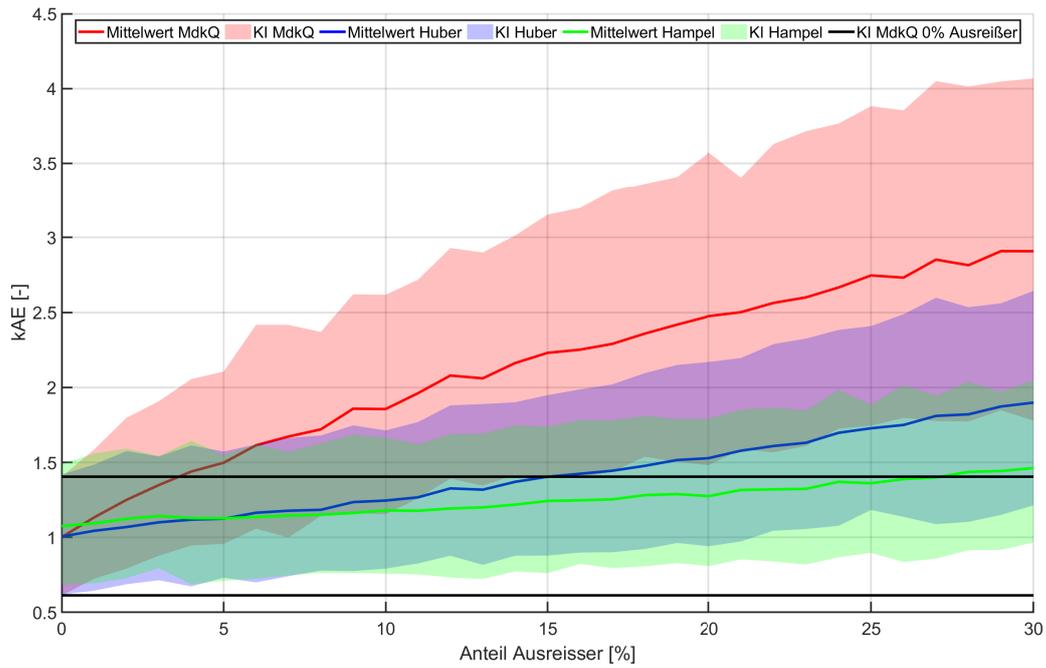


Abbildung A.45: *Belastbarkeit* Rauschmodell III Datensatz 7

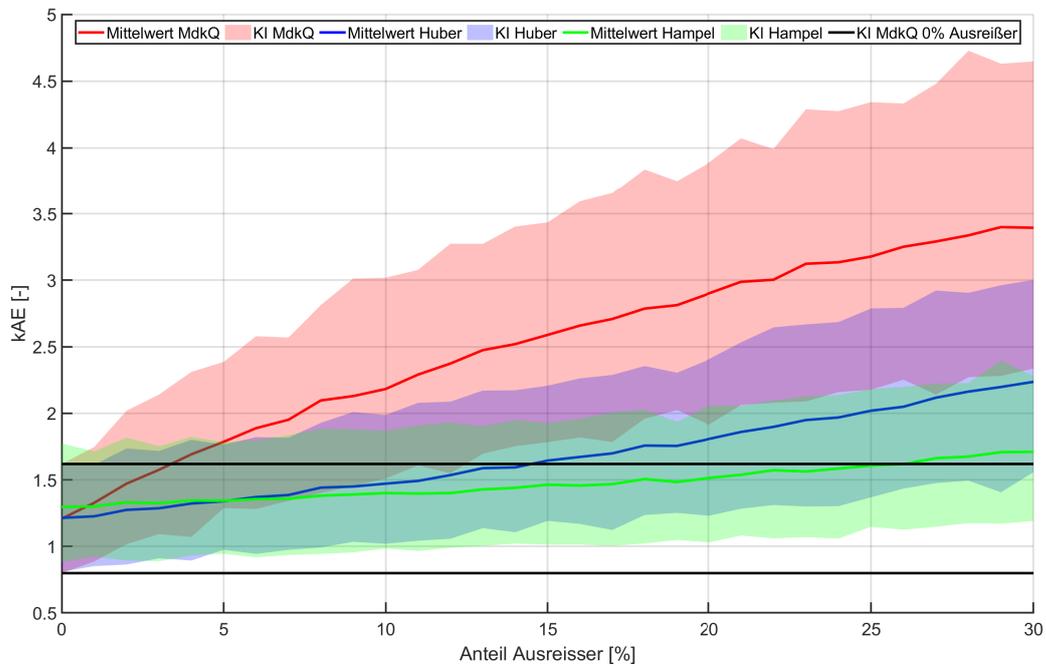


Abbildung A.46: *Belastbarkeit* Rauschmodell III Datensatz 8

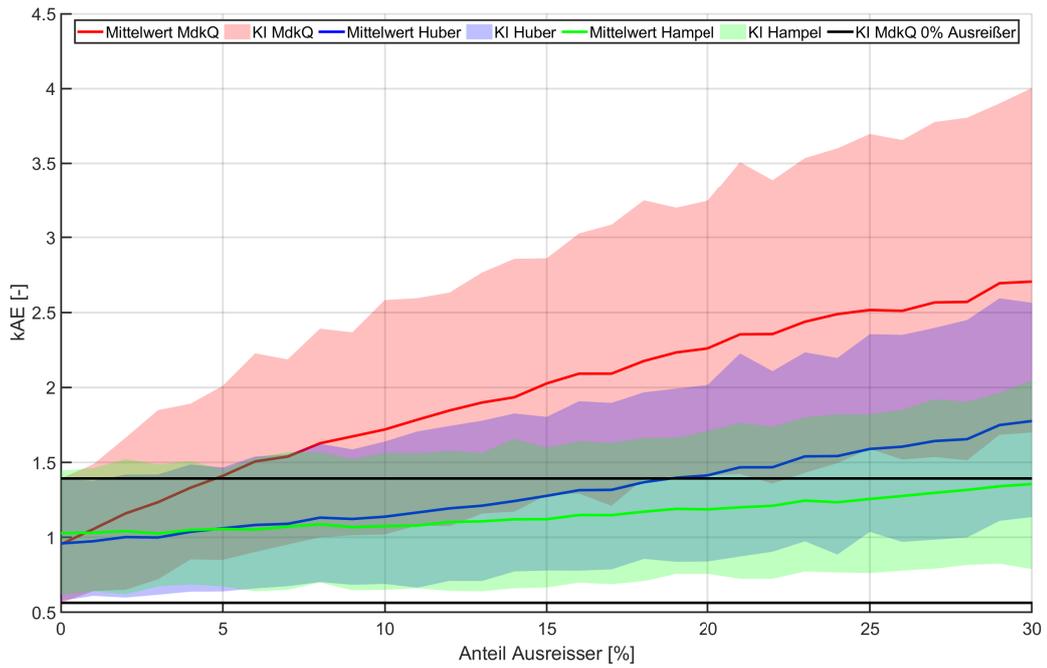


Abbildung A.47: Belastbarkeit Rauschmodell III Datensatz 9

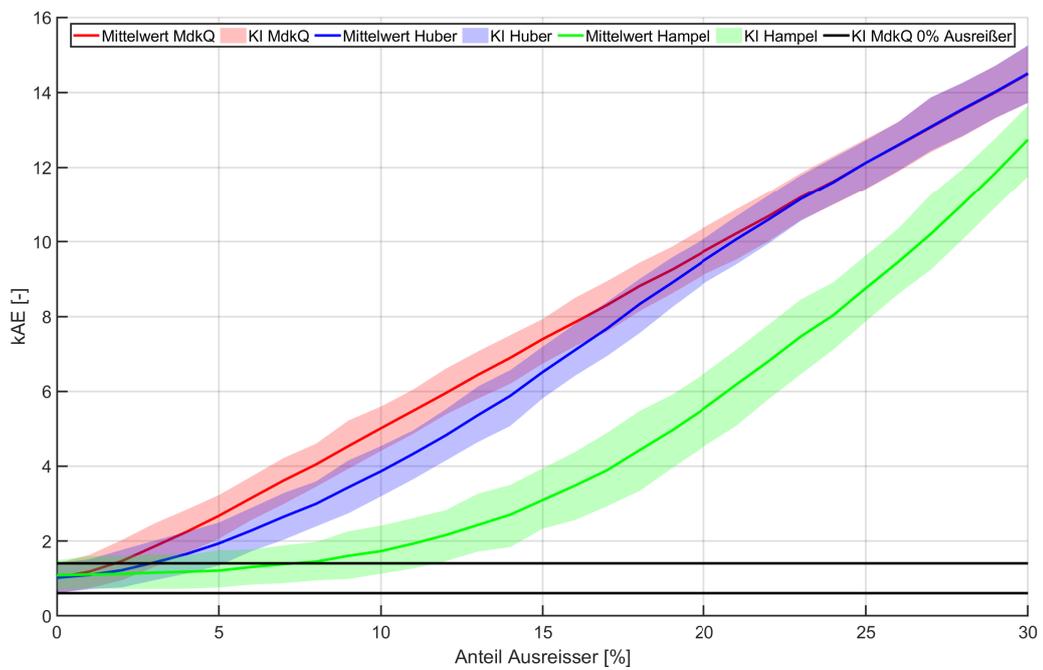


Abbildung A.48: Belastbarkeit Rauschmodell IV Datensatz 4

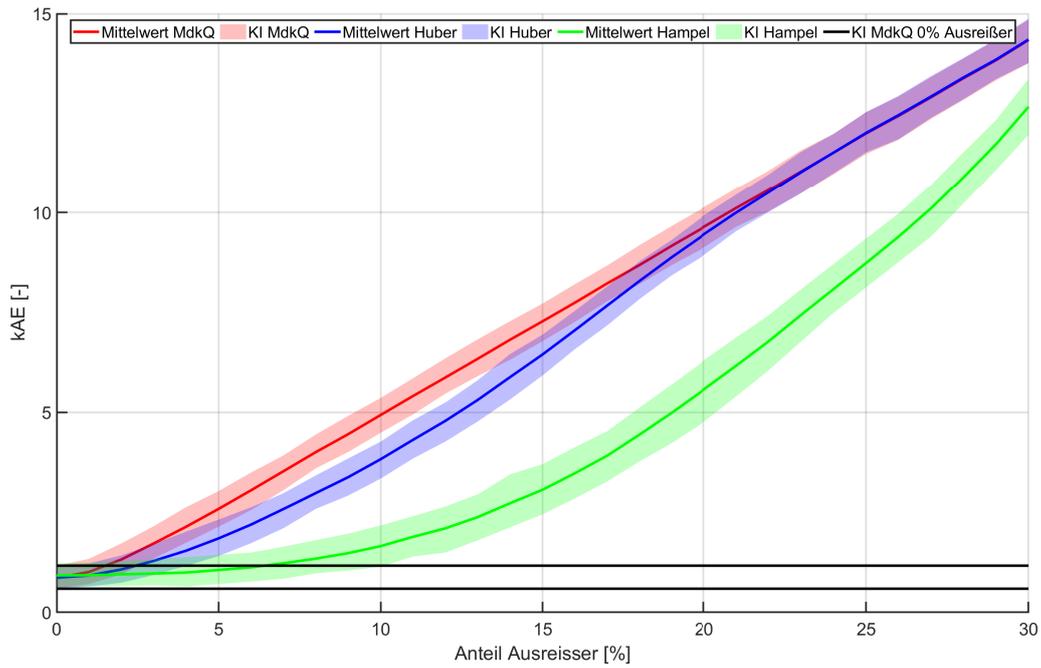


Abbildung A.49: Belastbarkeit Rauschmodell IV Datensatz 6

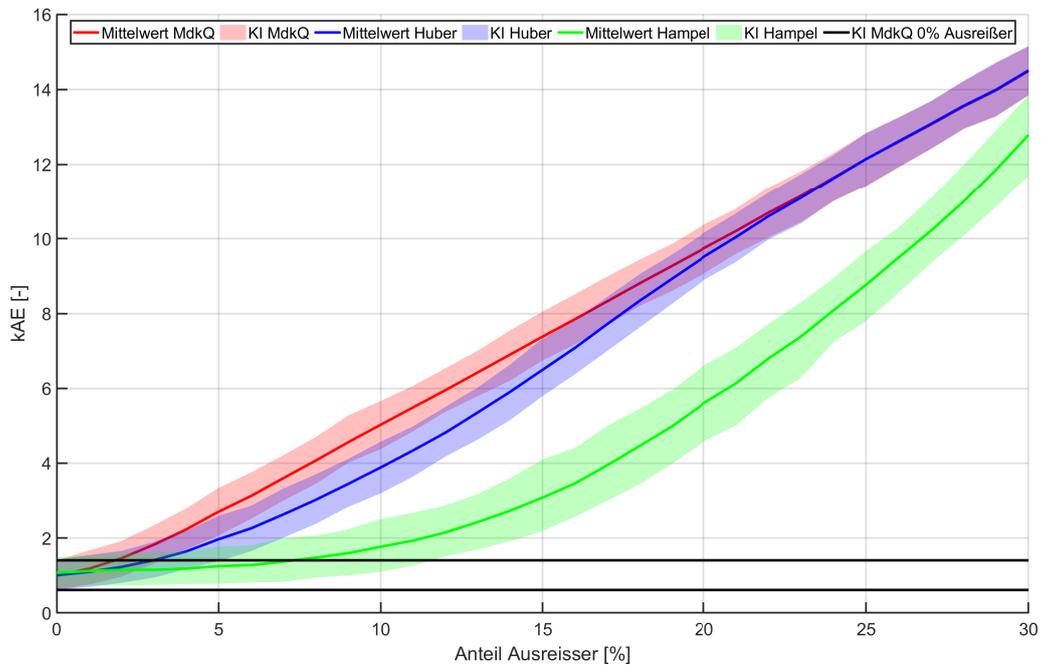
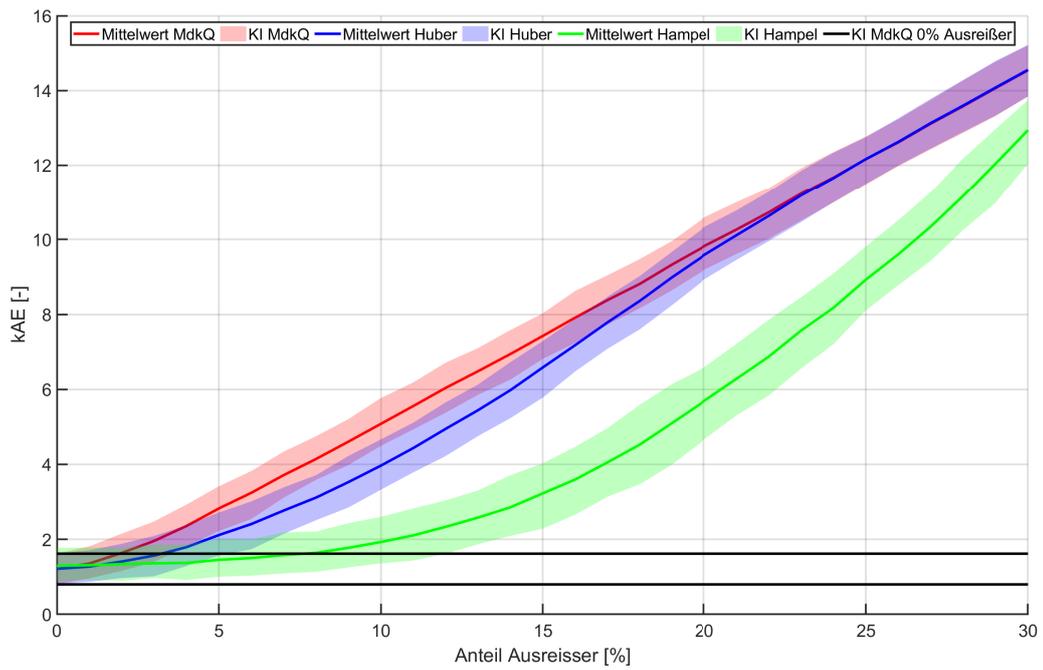
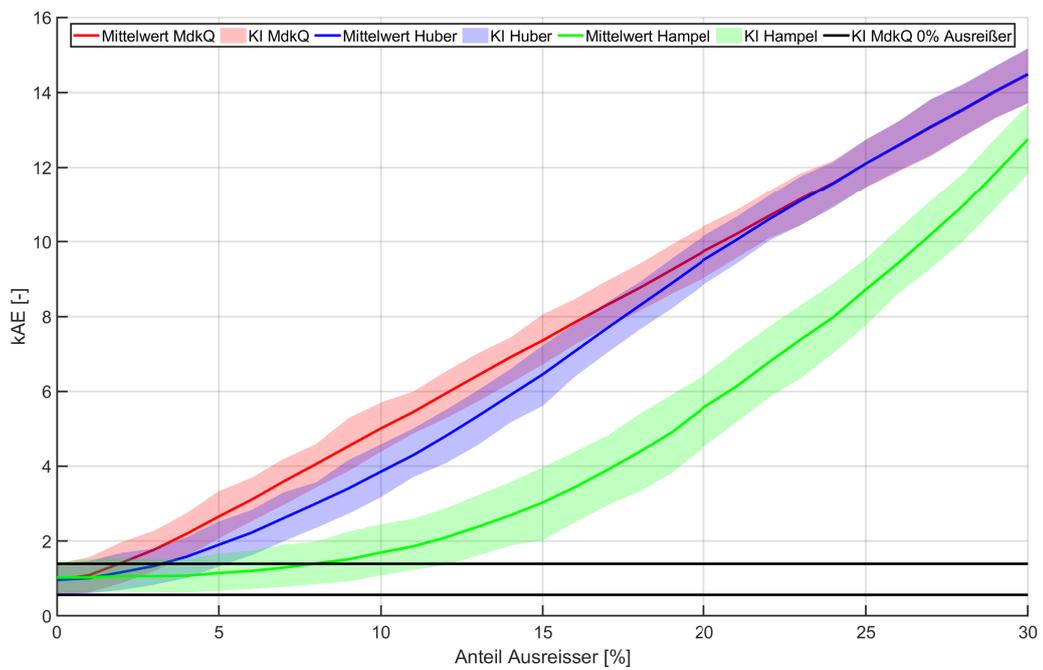


Abbildung A.50: Belastbarkeit Rauschmodell IV Datensatz 7

Abbildung A.51: *Belastbarkeit* Rauschmodell IV Datensatz 8Abbildung A.52: *Belastbarkeit* Rauschmodell IV Datensatz 9

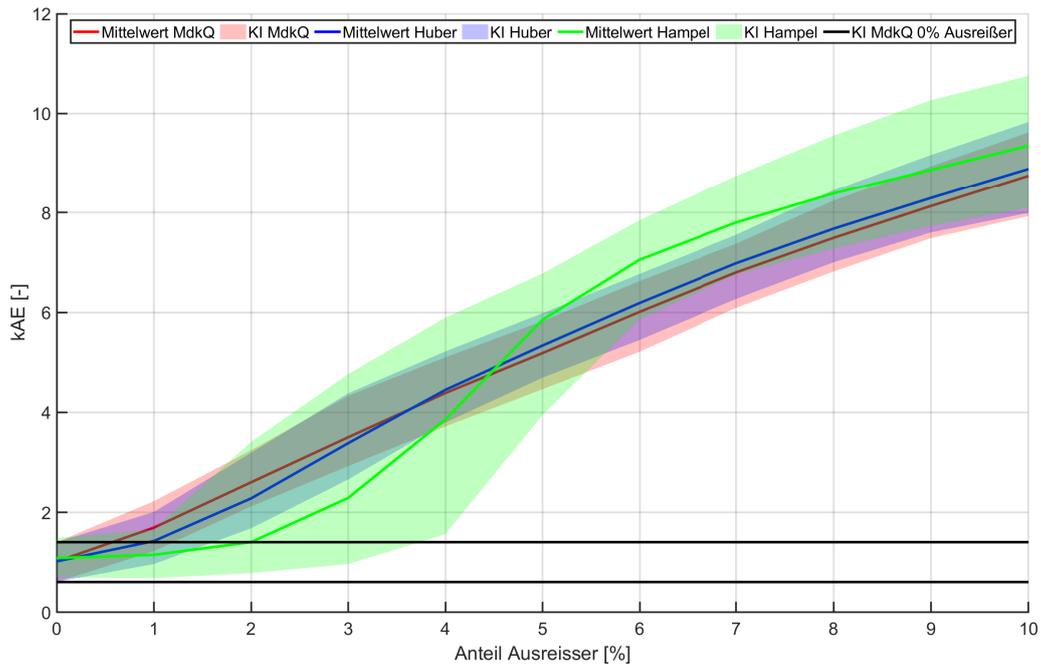


Abbildung A.53: *Belastbarkeit* Rauschmodell V Datensatz 4

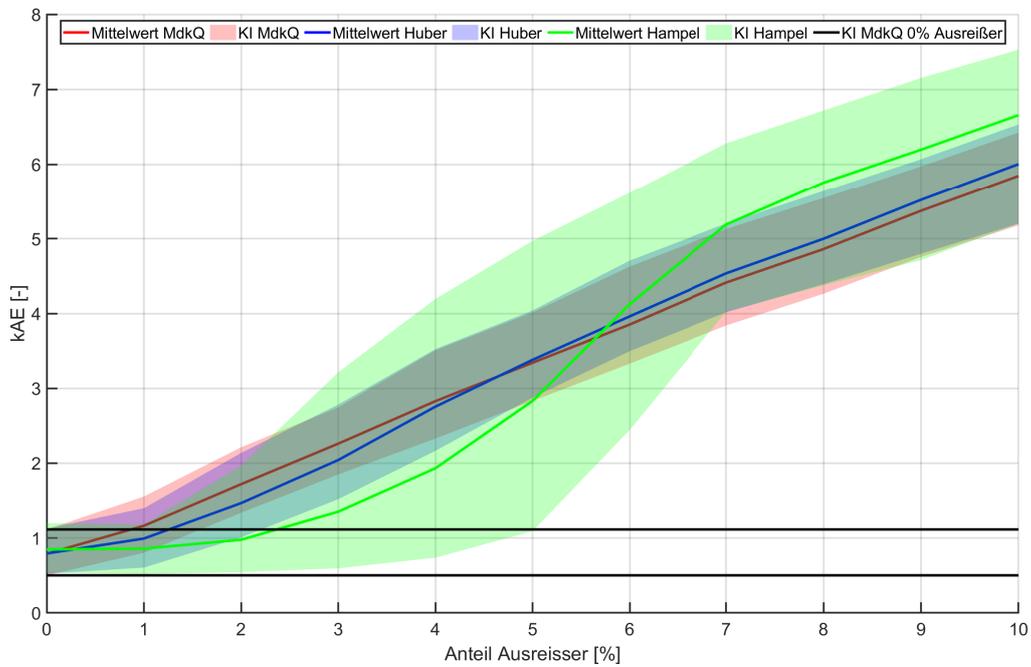


Abbildung A.54: *Belastbarkeit* Rauschmodell V Datensatz 5

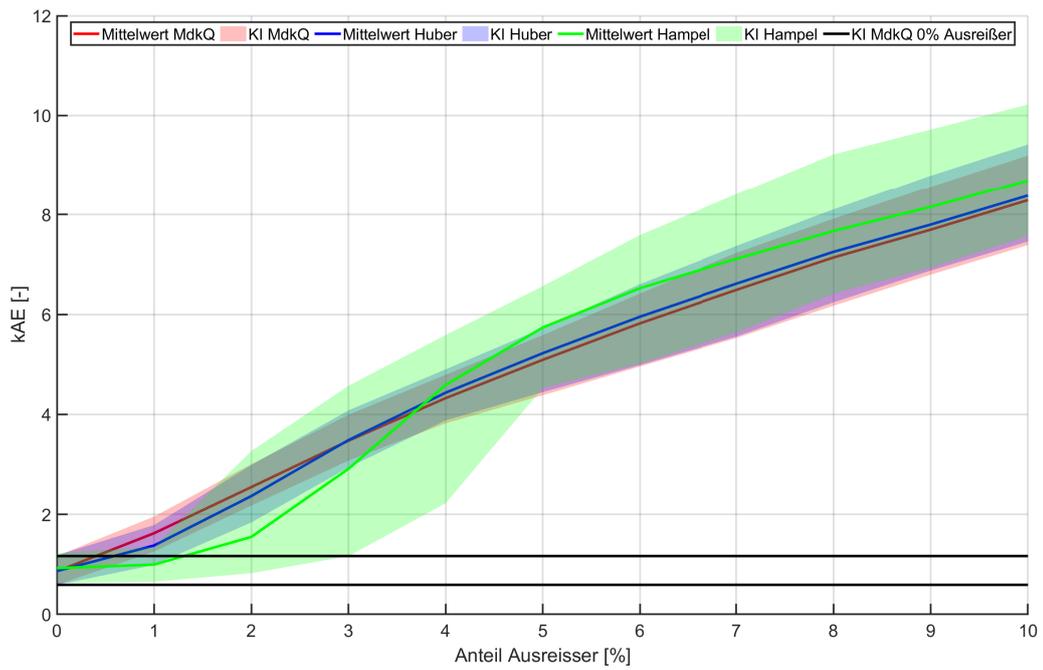


Abbildung A.55: Belastbarkeit Rauschmodell V Datensatz 6

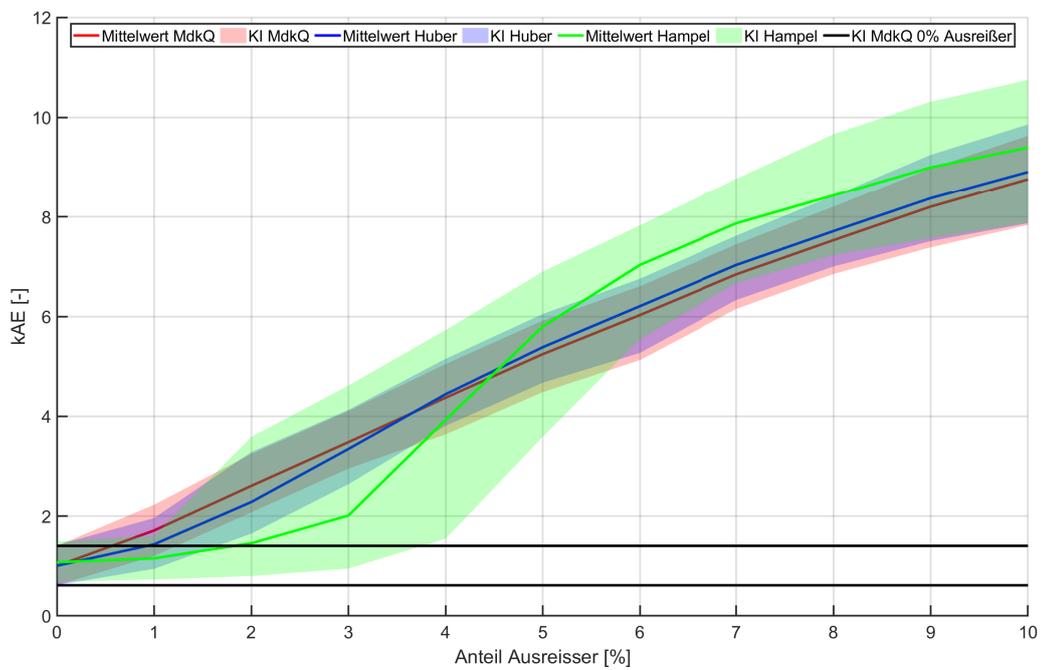


Abbildung A.56: Belastbarkeit Rauschmodell V Datensatz 7

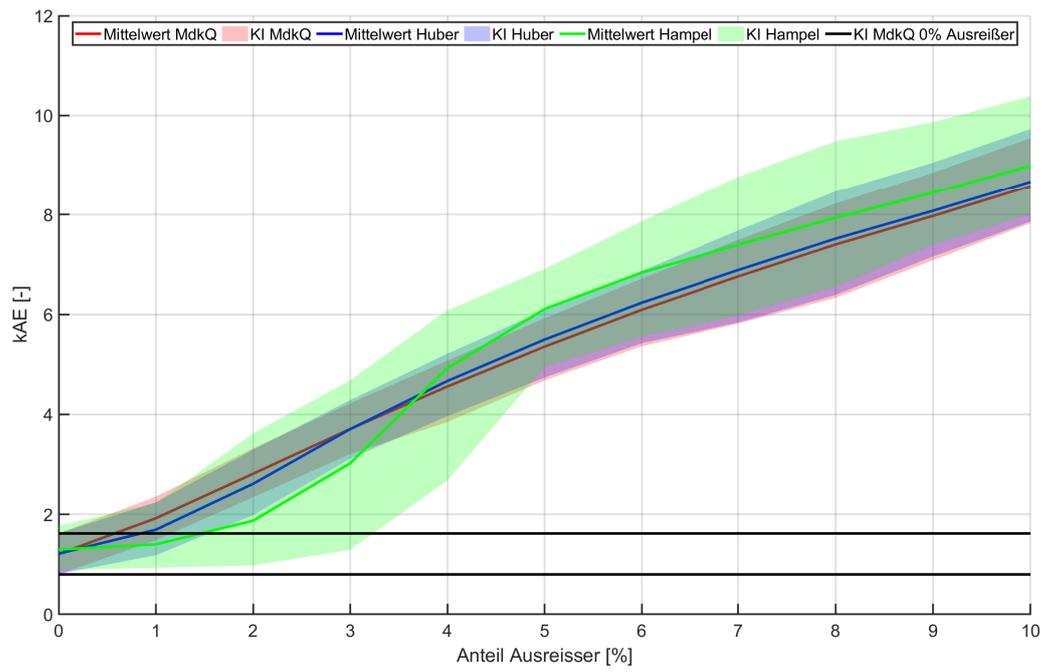


Abbildung A.57: *Belastbarkeit* Rauschmodell V Datensatz 8

Literaturverzeichnis

- [Akaike 1998a] AKAIKE, H.: A Bayesian Analysis of the Minimum AIC Procedure. In: PARZEN, E. (Hrsg.) ; TANABE, K. (Hrsg.) ; KITAGAWA, G. (Hrsg.): *Selected Papers of Hirotugu Akaike*. New York, NY : Springer New York, 1998 (Springer Series in Statistics), S. 275–280. – ISBN 978-1-4612-7248-9
- [Akaike 1998b] AKAIKE, H.: Information theory and an extension of the maximum likelihood principle. In: *Selected Papers of Hirotugu Akaike*. Springer, 1998, S. 199–213
- [Alkhatib u. a. 2018] ALKHATIB, H. ; KARGOLL, B. ; BUREICK, J. ; PAFFENHOLZ, J.-A.: Statistical evaluation of the B-Splines approximation of 3D point clouds. In: *Proceedings of the 2018 FIG-Kongresses, Istanbul, Türkei* (2018), S. 6–11
- [Alkhatib u. a. 2009] ALKHATIB, H. ; NEUMANN, I. ; KUTTERER, H.: Uncertainty modeling of random and systematic errors by means of Monte Carlo and fuzzy techniques. In: *Journal of Applied Geodesy* 3 (2009), Nr. 2, S. 136. – ISSN 1862-9024
- [Arlot und Celisse 2010] ARLOT, S. ; CELISSE, A.: A survey of cross-validation procedures for model selection. In: *Statistics Surveys* 4 (2010), Nr. 0, S. 40–79. – ISSN 1935-7516
- [Autodesk 2011] AUTODESK: *DXF Reference*. 2011. – URL https://images.autodesk.com/adsk/files/autocad_2012_pdf_dxf-reference_enu.pdf. – Zugriffsdatum: 14.01.2019
- [Bureick u. a. 2016a] BUREICK, J. ; ALKHATIB, H. ; NEUMANN, I.: Robust Spatial Approximation of Laser Scanner Point Clouds by Means of Free-form Curve Approaches in Deformation Analysis. In: *Journal of Applied Geodesy* 10 (2016), Nr. 1, S. 27–35. – ISSN 1862-9024
- [Bureick u. a. 2019] BUREICK, J. ; ALKHATIB, H. ; NEUMANN, I.: Fast converging elitist genetic algorithm for knot adjustment in B-spline curve approximation. In: *Journal of Applied Geodesy* 13 (2019), Nr. 4, S. 317–328. – ISSN 1862-9024
- [Bureick u. a. 2016b] BUREICK, J. ; NEUNER, H. B. ; HARMENING, C. ; NEUMANN, I.: Curve and Surface Approximation of 3D Point clouds. In: *Allgemeine Vermessungs-Nachrichten (AVN)* 123 (2016), Nr. 11-12, S. 315–327
- [Burger und Burge 2015] BURGER, W. ; BURGE, M. J.: *Digitale Bildverarbeitung*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2015. – ISBN 978-3-642-04603-2
- [Burnham und Anderson 2004a] BURNHAM, K. P. ; ANDERSON, D. R.: *Model Selection and Multimodel Inference*. New York, NY : Springer New York, 2004. – ISBN 978-0-387-95364-9
- [Burnham und Anderson 2004b] BURNHAM, K. P. ; ANDERSON, D. R.: Multimodel Inference: Understanding AIC and BIC in Model Selection. In: *Sociological Methods & Research* 33 (2004), Nr. 2, S. 261–304. – ISSN 0049-1241
- [Caspary 2013] CASPARY, W.: *Fehlertolerante Auswertung von Messdaten: Daten- und Modellanalyse, robuste Schätzung*. München : Oldenbourg, 2013. – ISBN 978-3-486-72771-5
- [Claeskens und Hjort 2008] CLAESKENS, G. ; HJORT, N. L.: *Model selection and model averaging*. Cambridge and New York : Cambridge University Press, 2008 (Cambridge series in statistical and probabilistic mathematics). – ISBN 9780521852258

- [Clarke 2003] CLARKE, K. A.: Nonparametric Model Discrimination in International Relations. In: *Journal of Conflict Resolution* 47 (2003), Nr. 1, S. 72–93. – ISSN 0022-0027
- [Clarke 2007] CLARKE, K. A.: A Simple Distribution-Free Test for Nonnested Model Selection. In: *Political Analysis* 15 (2007), Nr. 03, S. 347–363. – ISSN 1047-1987
- [Coons 1967] COONS, S. A.: *Surfaces for computer-aided design of space forms: Project MAC-TR 41*. 1967
- [Cox 1972] COX, M. G.: The Numerical Evaluation of B-Splines. In: *IMA Journal of Applied Mathematics* 10 (1972), Nr. 2, S. 134–149. – ISSN 0272-4960
- [Curry 1946] CURRY, H. B.: Review in Recent Mathematical Tables. In: *Mathematical Tables and Other Aids to Computation* 2 (1946), Nr. 16, S. 167–169. – ISSN 08916837
- [de Boor 1972] DE BOOR, C.: On calculating with B-splines. In: *Journal of Approximation Theory* 6 (1972), Nr. 1, S. 50–62. – ISSN 00219045
- [de Boor 2001] DE BOOR, C.: *Applied mathematical sciences*. Bd. 27: *A practical guide to splines*. Rev. ed. New York, N.Y. : Springer, 2001. – ISBN 978-0387953663
- [Dennig u. a. 2017] DENNIG, D. ; BUREICK, J. ; LINK, J. ; DIENER, D. ; HESSE, C. ; NEUMANN, I.: Comprehensive and Highly Accurate Measurements of Crane Runways, Profiles and Fastenings. In: *Sensors (Basel, Switzerland)* 17 (2017), Nr. 5
- [Dierckx 1993] DIERCKX, P.: *Curve and surface fitting with splines*. Repr. Oxford [u.a.] : Clarendon Press, 1993 (Oxford science publications). – ISBN 9780198534402
- [Dorndorf u. a. 2017] DORNDORF, A. ; SOOT, M. ; WEITKAMP, A. ; ALKHATIB, H.: A Heuristic Robust Approach for Real Estate Valuation in Areas with Few Transactions. In: *Proceedings of FIG Working Week* (2017)
- [Eilers und Marx 1996] EILERS, P. H. C. ; MARX, B. D.: Flexible Smoothing with B-splines and Penalties. In: *Statistical Science* 11 (1996), Nr. 2, S. 89–102. – URL <http://www.jstor.org/stable/2246049>. – ISSN 08834237
- [Elfving und Andersson 1987] ELFVING, T. ; ANDERSSON, L.-E.: An algorithm for computing constrained smoothing spline functions. In: *Numerische mathematik* 52 (1987), Nr. 5, S. 583–595. – ISSN 0029-599X
- [Ezhov u. a. 2018] EZHOV, N. ; NEITZEL, F. ; PETROVIC, S.: Spline approximation, Part 1: Basic methodology. In: *Journal of Applied Geodesy* 12 (2018), Nr. 2, S. 139–155. – ISSN 1862-9024
- [Fang und Hung 2013] FANG, J. J. ; HUNG, C. L.: An improved parameterization method for B-spline curve and surface interpolation. In: *Computer-Aided Design* 45 (2013), Nr. 6, S. 1005–1028. – ISSN 00104485
- [Farin 2002a] FARIN, G.: A history of curves and surfaces. In: *Handbook of computer aided geometric design* 1 (2002)
- [Farin 2002b] FARIN, G. E.: *Curves and surfaces for CAGD: A practical guide*. 5th ed. San Francisco, CA and London : Morgan Kaufmann and Academic Press, 2002 (The Morgan Kaufmann series in computer graphics and geometric modeling). – ISBN 1-55860-737-4
- [Ferguson u. a. 1988] FERGUSON, David R. ; FRANK, Paul D. ; JONES, Alan K.: Surface shape control using constrained optimization on the B-spline representation. In: *Computer Aided Geometric Design* 5 (1988), Nr. 2, S. 87–103. – ISSN 01678396

- [Foley und Nielson 1989] FOLEY, T. A. ; NIELSON, G. M.: Knot Selection for Parametric Spline Interpolation: Mathematical Methods in Computer Aided Geometric Design. In: LYCHE, T. (Hrsg.) ; SCHUMAKER, L. L. (Hrsg.): *Mathematical Methods in Computer Aided Geometric Design*. San Diego, CA, USA : Academic Press Professional, Inc, 1989, S. 261–272. – URL <http://dl.acm.org/citation.cfm?id=93803.93823>. – ISBN 0-12-460515-X
- [Gálvez u. a. 2015] GÁLVEZ, A. ; IGLESIAS, A. ; AVILA, A. ; OTERO, C. ; ARIAS, R. ; MANCHADO, C.: Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting. In: *Applied Soft Computing* 26 (2015), S. 90–106. – ISSN 15684946
- [Hampel 1974] HAMPEL, Frank R.: The Influence Curve and Its Role in Robust Estimation. In: *Journal of the American Statistical Association* 69 (1974), Nr. 346, S. 383. – ISSN 01621459
- [Harmening und Neuner 2014] HARMENING, C. ; NEUNER, H. B.: Raumkontinuierliche Modellierung mit Freiformflächen. In: *Schriftenreihe DVW* 78 (2014), S. 105–122
- [Harmening und Neuner 2015] HARMENING, C. ; NEUNER, H. B.: A constraint-based parameterization technique for B-spline surfaces. In: *Journal of Applied Geodesy* 9 (2015), Nr. 3. – ISSN 1862-9024
- [Harmening und Neuner 2016] HARMENING, C. ; NEUNER, H. B.: Choosing the Optimal Number of B-spline Control Points (Part 1: Methodology and Approximation of Curves). In: *Journal of Applied Geodesy* 10 (2016), Nr. 3, S. 139–157. – ISSN 1862-9024
- [Harmening und Neuner 2017] HARMENING, C. ; NEUNER, H. B.: Choosing the optimal number of B-spline control points (Part 2: Approximation of surfaces and applications). In: *Journal of Applied Geodesy* 11 (2017), Nr. 3, S. 43–52. – ISSN 1862-9024
- [Hesse u. a. 2019] HESSE, C. ; HOLSTE, K. ; NEUMANN, I. ; HAKE, F. ; ALKHATIB, H. ; GEIST, M. ; KNAACK, L. ; SCHARR, C.: 3D HydroMapper: Automatisierte 3D-Bauwerksaufnahme und Schadenserkennung unter Wasser für die Bauwerksinspektion und das Building Information Modelling. In: *Hydrographische Nachrichten* (2019), Nr. 113, S. 26–29
- [Hippel 2005] HIPPEL, P. T. v.: Mean, Median, and Skew: Correcting a Textbook Rule. In: *Journal of Statistics Education* 13 (2005), Nr. 2, S. 187
- [Holland und Welsch 1977] HOLLAND, P. W. ; WELSCH, R. E.: Robust regression using iteratively reweighted least-squares. In: *Communications in Statistics - Theory and Methods* 6 (1977), Nr. 9, S. 813–827. – ISSN 0361-0926
- [Hoschek 1988] HOSCHEK, J.: Intrinsic parametrization for approximation. In: *Computer Aided Geometric Design* 5 (1988), Nr. 1, S. 27–31. – ISSN 01678396
- [Hoschek und Lasser 1992] HOSCHEK, J. ; LASSER, D.: *Grundlagen der geometrischen Datenverarbeitung*. 2., neubearbeitete und erweiterte Auflage. Wiesbaden : Vieweg+Teubner Verlag, 1992. – ISBN 978-3-519-12962-2
- [Huber 1964] HUBER, P. J.: Robust Estimation of a Location Parameter. In: *The Annals of Mathematical Statistics* 35 (1964), Nr. 1, S. 73–101. – ISSN 0003-4851
- [Huber 1981] HUBER, P. J.: *Robust statistics*. New York : Wiley, 1981 (Wiley series in probability and mathematical statistics). – ISBN 0-471-41805-6
- [ISO 10303-42 2014–05] ISO 10303-42: *Industrial automation systems and integration - Product data representation and exchange - Part 42: Integrated generic resource: Geometric and topological representation*. 2014-05

- [ISO 16739 2016–10] ISO 16739: *Industry Foundation Classes (IFC) für den Datenaustausch in der Bauindustrie und im Anlagenmanagement (ISO 16739:2013)*. 2016-10
- [Jäger 2005] JÄGER, R.: *Klassische und robuste Ausgleichungsverfahren: Ein Leitfaden für Ausbildung und Praxis von Geodäten und Geoinformatikern*. Heidelberg : Wichmann, 2005. – ISBN 3-87907-370-8
- [Kermarrec u. a. 2019] KERMARREC, G. ; PAFFENHOLZ, J.-A. ; ALKHATIB, H.: How Significant Are Differences Obtained by Neglecting Correlations When Testing for Deformation: A Real Case Study Using Bootstrapping with Terrestrial Laser Scanner Observations Approximated by B-Spline Surfaces. In: *Sensors (Basel, Switzerland)* 19 (2019), Nr. 17
- [Koch 2004] KOCH, K. R.: *Parameterschätzung und Hypothesentests in linearen Modellen*. 4. Auflage. Bonn : Ehemals Ferd. Dümmlers Verlag, 2004
- [Koch 2009] KOCH, K. R.: Fitting Free-Form Surfaces to Laserscan Data by NURBS. In: *Allgemeine Vermessungs-Nachrichten (AVN)* (2009), Nr. 4, S. 134–140
- [Koch und Kargoll 2013] KOCH, K. R. ; KARGOLL, B.: Expectation maximization algorithm for the variance-inflation model by applying the t-distribution. In: *Journal of Applied Geodesy* 7 (2013), Nr. 3. – ISSN 1862-9024
- [Kohavi 1995] KOHAVI, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Ijcai* Bd. 14, 1995, S. 1137–1145
- [Kroese u. a. 2011] KROESE, D. P. ; TAIMRE, T. ; BOTEV, Z. I.: *Handbook of Monte Carlo Methods*. Hoboken, N.J. : John Wiley and Sons, 2011. – ISBN 978-0-470-17793-8
- [Lai und Lu 1996] LAI, J. Y. ; LU, C. Y.: Reverse Engineering of Composite Sculptured Surfaces. In: *Int J Advanced Manufacturing Technology* (1996), Nr. 12, S. 180–189
- [Laurent-Gengoux und Mekhilef 1993] LAURENT-GENGOUX, P. ; MEKHILEF, M.: Optimization of a NURBS representation. In: *Computer-Aided Design* 25 (1993), Nr. 11, S. 699–710. – ISSN 00104485
- [Lee 1989] LEE, E. T.: Choosing nodes in parametric curve interpolation. In: *Computer-Aided Design* 21 (1989), Nr. 6, S. 363–370. – ISSN 00104485
- [Lim 1999] LIM, C. G.: A universal parametrization in B-spline curve and surface interpolation. In: *Computer Aided Geometric Design* 16 (1999), Nr. 5, S. 407–422. – ISSN 01678396
- [Ma und Kruth 1995] MA, W. ; KRUTH, J. P.: Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces. In: *Computer-Aided Design* 27 (1995), Nr. 9, S. 663–675. – ISSN 00104485
- [MathWorks® 2019a] MATHWORKS®: *Documentation: ksdensity - Kernel smoothing function estimate for univariate and bivariate data*. 2019. – URL <https://de.mathworks.com/help/stats/ksdensity.html>. – Zugriffsdatum: 21.05.2019
- [MathWorks® 2019b] MATHWORKS®: *Documentation: mad - Mean or median absolute deviation*. 2019. – URL <https://de.mathworks.com/help/stats/mad.html>. – Zugriffsdatum: 07.05.2019
- [MathWorks® 2019c] MATHWORKS®: *Documentation. medfilt1 - 1D median filtering*. 2019. – URL <https://de.mathworks.com/help/signal/ref/medfilt1.html>. – Zugriffsdatum: 15.05.2019

- [McQuarrie und Tsai 1998] MCQUARRIE, A. D. R. ; TSAI, C. L.: *Regression and time series model selection*. Singapore and River Edge, N.J. : World Scientific, 1998. – ISBN 981-02-3242-X
- [Neitzel u. a. 2019] NEITZEL, F. ; EZHOV, N. ; PETROVIC, S.: Total Least Squares Spline Approximation. In: *Mathematics* 7 (2019), Nr. 5, S. 462
- [Nielson 1987] NIELSON, G. M.: Coordinate free scattered data interpolation. In: *Topics in Multivariate Approximation*. Elsevier, 1987, S. 175–184. – ISBN 9780121745851
- [Nielson und Foley 1989] NIELSON, G. M. ; FOLEY, T. A.: A Survey of Applications of an Affine Invariant Norm. In: LYCHE, T. (Hrsg.) ; SCHUMAKER, L. L. (Hrsg.): *Mathematical Methods in Computer Aided Geometric Design*. San Diego, CA, USA : Academic Press Professional, Inc, 1989, S. 445–467. – ISBN 0-12-460515-X
- [Niemeier 2008] NIEMEIER, W.: *Ausgleichsrechnung: Statistische Auswertemethoden*. 2., überarb. und erw. Aufl. Berlin and New York : De Gruyter, 2008 (De Gruyter Lehrbuch). – ISBN 3110190559
- [O’Sullivan 1986] O’SULLIVAN, F.: A Statistical Perspective on Ill-Posed Inverse Problems. In: *Statistical Science* 1 (1986), Nr. 4, S. 502–518. – URL <http://www.jstor.org/stable/2245801>. – ISSN 08834237
- [O’Sullivan 1988] O’SULLIVAN, F.: Fast Computation of Fully Automated Log-Density and Log-Hazard Estimators. In: *SIAM Journal on Scientific and Statistical Computing* 9 (1988), Nr. 2, S. 363–379. – ISSN 0196-5204
- [Park und Lee 2007] PARK, H. ; LEE, J. H.: B-spline curve fitting based on adaptive curve refinement using dominant points. In: *Computer-Aided Design* 39 (2007), Nr. 6, S. 439–451. – ISSN 00104485
- [Parzen 1962] PARZEN, E.: On Estimation of a Probability Density Function and Mode. In: *The Annals of Mathematical Statistics* 33 (1962), Nr. 3, S. 1065–1076. – ISSN 0003-4851
- [Picard und Cook 1984] PICARD, R. R. ; COOK, R. D.: Cross-Validation of Regression Models. In: *Journal of the American Statistical Association* 79 (1984), Nr. 387, S. 575–583. – ISSN 01621459
- [Piegl und Tiller 1997] PIEGL, L. A. ; TILLER, W.: *The NURBS book*. 2nd ed. Berlin and New York : Springer, 1997 (Monographs in visual communications). – ISBN 3-540-61545-8
- [Piegl und Tiller 2000] PIEGL, L. A. ; TILLER, W.: Surface approximation to scanned data. In: *The Visual Computer* 16 (2000), Nr. 7, S. 386–395. – ISSN 0178-2789
- [Reid u. a. 2001] REID, E. ; MOREA, G. ; DENNETTE, A. H. ; PELTZMAN, A. ; PARKS, C. ; GRUTTKE, W. B.: *The Initial Graphics Exchange Specification (IGES) Version 6.0: Draft Baseline 1/98*. 2001. – URL <https://filemonger.com/specs/igs/devdept.com/version6.pdf>. – Zugriffsdatum: 14.01.2019
- [Reinsch 1967] REINSCH, C. H.: Smoothing by spline functions. In: *Numerische mathematik* 10 (1967), Nr. 3, S. 177–183
- [Rosenblatt 1956] ROSENBLATT, M.: Remarks on Some Nonparametric Estimates of a Density Function. In: *The Annals of Mathematical Statistics* 27 (1956), Nr. 3, S. 832–837. – ISSN 0003-4851
- [Salomon 2006] SALOMON, D.: *Curves and surfaces for computer graphics*. New York : Springer, 2006. – ISBN 978-0-387-24196-8

- [Sarfranz und Raza 2001] SARFRAZ, M. ; RAZA, S. A.: Capturing outline of fonts using genetic algorithm and splines. In: BANISSI, E. (Hrsg.): *Proceedings*. Los Alamitos, Calif. [u.a.] : IEEE Computer Soc., 2001, S. 738–743. – ISBN 0-7695-1195-3
- [Sarkar und Menq 1991] SARKAR, B. ; MENQ, C. H.: Parameter optimization in approximating curves and surfaces to measurement data. In: *Computer Aided Geometric Design* 8 (1991), Nr. 4, S. 267–290. – ISSN 01678396
- [Schmitt und Neuner 2015] SCHMITT, C. ; NEUNER, H. B.: Knot estimation on B-Spline curves. In: *Österreichische Zeitschrift für Vermessung & Geoinformation* 103 (2015), Nr. 2+3, S. 188–197
- [Schoenberg 1946] SCHOENBERG, I. J.: Contributions to the problem of approximation of equidistant data by analytic functions: Part A.—On the problem of smoothing or graduation. A first class of analytic approximation formulae. In: *Quarterly of Applied Mathematics* 4 (1946), Nr. 1, S. 45–99. – URL <http://www.jstor.org/stable/43633538>. – ISSN 0033569X
- [Schoenberg 1964] SCHOENBERG, I. J.: Spline functions and the problem of graduation. In: *Proceedings of the National Academy of Sciences of the United States of America* 52 (1964), Nr. 4, S. 947–950. – ISSN 0027-8424
- [Schwarz 1978] SCHWARZ, G.: Estimating the Dimension of a Model. In: *The Annals of Statistics* 6 (1978), Nr. 2, S. 461–464. – ISSN 0090-5364
- [Simon 2006] SIMON, D.: *Optimal state estimation: Kalman, H [infinity] and nonlinear approaches*. Hoboken, N.J. : Wiley-Interscience, 2006. – ISBN 978-0-471-70858-2
- [Spiriti u. a. 2013] SPIRITI, S. ; EUBANK, R. ; SMITH, P. W. ; YOUNG, D.: Knot selection for least-squares and penalized splines. In: *Journal of Statistical Computation and Simulation* 83 (2013), Nr. 6, S. 1020–1036. – ISSN 0094-9655
- [Vuong 1989] VUONG, Q. H.: Likelihood Ratio Tests for Model Selection and Non-Nested Hypotheses. In: *Econometrica* 57 (1989), Nr. 2, S. 307–333. – ISSN 00129682
- [Xu u. a. 2018] XU, X. ; KARGOLL, B. ; BUREICK, J. ; YANG, H. ; ALKHATIB, H. ; NEUMANN, I.: TLS-based profile model analysis of major composite structures with robust B-spline method. In: *Composite Structures* 184 (2018), S. 814–820. – ISSN 02638223
- [Yoshimoto u. a. 2003] YOSHIMOTO, F. ; HARADA, T. ; YOSHIMOTO, Y.: Data fitting with a spline using a real-coded genetic algorithm. In: *Computer-Aided Design* 35 (2003), Nr. 8, S. 751–760. – ISSN 00104485
- [Yoshimoto u. a. 1999] YOSHIMOTO, F. ; MORIYAMA, M. ; HARADA, T.: Automatic knot placement by a genetic algorithm for data fitting with a spline. In: AIZU-WAKAMATSU: IEEE COMPUTER SOCIETY PRESS (Hrsg.): *Proceedings of the international conference on shape modeling and applications*, 1999, S. 162–169
- [Zhao 2019] ZHAO, X.: *Terrestrial Laser Scanning Data Analysis for Deformation Monitoring*. Hannover, Leibniz Universität Hannover, Dissertation, 2019
- [Zhao u. a. 2018] ZHAO, X. ; KARGOLL, B. ; OMIDALIZARANDI, M. ; XU, X. ; ALKHATIB, H.: Model Selection for Parametric Surfaces Approximating 3D Point Clouds for Deformation Analysis. In: *Remote Sensing* 10 (2018), Nr. 4, S. 634
- [Zhao u. a. 2019] ZHAO, X. ; KERMARREC, G. ; KARGOLL, B. ; ALKHATIB, H. ; NEUMANN, I.: Influence of the simplified stochastic model of TLS measurements on geometry-based deformation analysis. In: *Journal of Applied Geodesy* 13 (2019), Nr. 3, S. 199–214. – ISSN 1862-9024

-
- [Zhao u. a. 2011] ZHAO, X. ; ZHANG, C. ; YANG, B. ; LI, P.: Adaptive knot placement using a GMM-based continuous optimization algorithm in B-spline curve approximation. In: *Computer-Aided Design* 43 (2011), Nr. 6, S. 598–604. – ISSN 00104485

Abbildungsverzeichnis

2.1	Erstellung einer B-Spline-Kurve	13
2.2	Einflussfunktion L2-Norm-Schätzer	19
2.3	Einflussfunktion L1-Norm-Schätzer	20
2.4	Einflussfunktion Huber-Schätzer	21
2.5	Einflussfunktion Hampel-Schätzer	22
3.1	Einflussgrößen der Foley-Methode (adaptiert aus Foley und Nielson (1989))	36
3.2	Projizierter Abweichungsvektor bei der Hoschek-Methode (adaptiert aus Hoschek (1988))	37
4.1	Ablauf Residuenbasierter Iterativer Update Algorithmus	50
4.2	Ablauf Evolutionäre Monte-Carlo Methode	53
4.3	Ablauf elitärer genetischer Algorithmus	58
4.4	Schematischer Ablauf des zufälligen Einzelpunkt-Crossovers (adaptiert aus Bureick u. a. (2019))	60
5.1	Ablauf der Simulationen	64
5.2	Verwendete Datensätze zur Evaluation der B-Spline-Kurven Approximation	65
5.3	Kennziffern einer Stichprobe	70
5.4	Histogramm der Verteilung der kAE-Werte bei Datensatz 1. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.5 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	73
a	PT1	73
b	RIU	73
c	CSA	73
d	EMC	73
e	EGAori	73
f	EGAmod	73
5.5	Graphische Darstellung des kAE des CSA bei Datensatz 1	74
5.6	Graphische Darstellung des kAE der EMC bei Datensatz 1	75
5.7	Graphische Darstellung des kAE des EGAori bei Datensatz 1	75
5.8	Graphische Darstellung des kAE des EGAmod bei Datensatz 1	76
5.9	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 1	76
5.10	Histogramm der Verteilung der acht internen Knoten bei Datensatz 1. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	78
a	PT1	78
b	RIU	78
c	CSA	78
d	EMC	78
e	EGAori	78

f	EGAmo	78
5.11	Histogramm der Verteilung der kAE-Werte bei Datensatz 2. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.6 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	79
a	PT1	79
b	RIU	79
c	CSA	79
d	EMC	79
e	EGAori	79
f	EGAmo	79
5.12	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 2	80
5.13	Histogramm der Verteilung der fünf internen Knoten bei Datensatz 2. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	81
a	PT1	81
b	RIU	81
c	CSA	81
d	EMC	81
e	EGAori	81
f	EGAmo	81
5.14	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 3	83
5.15	Histogramm der Verteilung der vier internen Knoten bei Datensatz 3. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	84
a	PT1	84
b	RIU	84
c	CSA	84
d	EMC	84
e	EGAori	84
f	EGAmo	84
5.16	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4	86
5.17	Histogramm der Verteilung der vier internen Knoten bei Datensatz 4. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt. (Zur besseren Darstellung wurde nicht der gesamte Wertebereich zwischen 0 und 1 dargestellt, sondern nur ein Teilausschnitt. In nicht dargestellten Bereichen ist die relative Häufigkeit $P(\xi) = 0$.)	87
a	PT1	87
b	RIU	87
c	CSA	87
d	EMC	87
e	EGAori	87
f	EGAmo	87

5.18	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 5	89
5.19	Histogramm der Verteilung der vier internen Knoten bei Datensatz 5. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	90
a	PT1	90
b	RIU	90
c	CSA	90
d	EMC	90
e	EGAori	90
f	EGAmo	90
5.20	Histogramm der Verteilung der kAE-Werte bei Datensatz 6. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.10 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	91
a	PT1	91
b	RIU	91
c	CSA	91
d	EMC	91
e	EGAori	91
f	EGAmo	91
5.21	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 6	92
5.22	Histogramm der Verteilung der sieben internen Knoten bei Datensatz 6. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	93
a	PT1	93
b	RIU	93
c	CSA	93
d	EMC	93
e	EGAori	93
f	EGAmo	93
5.23	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 9	95
5.24	Histogramm der Verteilung der drei internen Knoten bei Datensatz 9. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	96
a	PT1	96
b	RIU	96
c	CSA	96
d	EMC	96
e	EGAori	96
f	EGAmo	96

5.25	Lage der geschätzten Kontrollpunkte \mathbf{x}_1 und \mathbf{x}_5 (schwarz) mit empirisch bestimmter Konfidenzellipse (rot, $\alpha = 5\%$) und Konfidenzellipse aus der mittleren Varianz-Kovarianz-Matrix der B-Spline-Kontrollpunktschätzung (blau, $\alpha = 5\%$) bei Rauschmodell I, Datensatz 9 und der Knotenvektormethode EGAmo	102
a	\mathbf{x}_1	102
b	\mathbf{x}_5	102
5.26	Analyse der B-Spline-Parameter in Datensatz 9	103
a	Original Parameter	103
b	Auswirkung eines verschobenen Knotens	103
c	Auswirkung eines verschobenen Knotens und zweier verschobener Kontrollpunkte	103
5.27	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4 und $n = 5$	106
5.28	Knotenverteilung bei Datensatz 4 und $n = 5$	107
a	PT1	107
b	RIU	107
c	CSA	107
d	EMC	107
e	EGAori	107
f	EGAmo	107
5.29	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 4 und $n = 7$	108
5.30	Knotenverteilung bei Datensatz 4 und $n = 7$	109
a	PT1	109
b	RIU	109
c	CSA	109
d	EMC	109
e	EGAori	109
f	EGAmo	109
5.31	Die beiden realen Datensätze R1 und R2	110
a	Datensatz R1: Kranbahnschiene	110
b	Datensatz R2: Spundwand	110
6.1	Wahrscheinlichkeitsdichtefunktion der verschiedenen Rauschmodelle	118
6.2	<i>Belastbarkeit</i> Rauschmodell III Datensatz 4	134
6.3	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 5	135
6.4	<i>Belastbarkeit</i> Rauschmodell V Datensatz 9	136
A.1	Graphische Darstellung des kAE des CSA bei Datensatz 2	145
A.2	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 2	146
A.3	Graphische Darstellung des kAE des EGAori bei Datensatz 2	146
A.4	Graphische Darstellung des kAE des EGAmo bei Datensatz 2	147
A.5	Histogramm der Verteilung der kAE-Werte bei Datensatz 3. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.7 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	148
a	PT1	148
b	RIU	148
c	CSA	148
d	EMC	148
e	EGAori	148

f	EGAmod	148
A.6	Graphische Darstellung des kAE des CSA bei Datensatz 3	149
A.7	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 3	149
A.8	Graphische Darstellung des kAE des EGAori bei Datensatz 3	150
A.9	Graphische Darstellung des kAE des EGAmod bei Datensatz 3	150
A.10	Histogramm der Verteilung der kAE-Werte bei Datensatz 4. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.8 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	151
a	PT1	151
b	RIU	151
c	CSA	151
d	EMC	151
e	EGAori	151
f	EGAmod	151
A.11	Graphische Darstellung des kAE des CSA bei Datensatz 4	152
A.12	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 4	152
A.13	Graphische Darstellung des kAE des EGAori bei Datensatz 4	153
A.14	Graphische Darstellung des kAE des EGAmod bei Datensatz 4	153
A.15	Histogramm der Verteilung der kAE-Werte bei Datensatz 5. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.9 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	154
a	PT1	154
b	RIU	154
c	CSA	154
d	EMC	154
e	EGAori	154
f	EGAmod	154
A.16	Graphische Darstellung des kAE des CSA bei Datensatz 5	155
A.17	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 5	155
A.18	Graphische Darstellung des kAE des EGAori bei Datensatz 5	156
A.19	Graphische Darstellung des kAE des EGAmod bei Datensatz 5	156
A.20	Graphische Darstellung des kAE des CSA bei Datensatz 6	157
A.21	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 6	157
A.22	Graphische Darstellung des kAE des EGAori bei Datensatz 6	158
A.23	Graphische Darstellung des kAE des EGAmod bei Datensatz 6	158
A.24	Histogramm der Verteilung der kAE-Werte bei Datensatz 7. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle A.7 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	159
a	PT1	159
b	RIU	159
c	CSA	159
d	EMC	159
e	EGAori	159
f	EGAmod	159
A.25	Graphische Darstellung des kAE des CSA bei Datensatz 7	160

A.26	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 7	160
A.27	Graphische Darstellung des kAE des EGAori bei Datensatz 7	161
A.28	Graphische Darstellung des kAE des EGAmo bei Datensatz 7	161
A.29	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 7	162
A.30	Histogramm der Verteilung der vier internen Knoten bei Datensatz 7. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt. (Zur besseren Darstellung wurde nicht der gesamte Wertebereich zwischen 0 und 1 dargestellt, sondern nur ein Teilausschnitt. In nicht dargestellten Bereichen ist die relative Häufigkeit $P(\xi) = 0$.)	163
a	PT1	163
b	RIU	163
c	CSA	163
d	EMC	163
e	EGAori	163
f	EGAmo	163
A.31	Histogramm der Verteilung der kAE-Werte bei Datensatz 8. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle A.8 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	164
a	PT1	164
b	RIU	164
c	CSA	164
d	EMC	164
e	EGAori	164
f	EGAmo	164
A.32	Graphische Darstellung des kAE des CSA bei Datensatz 8	165
A.33	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 8	165
A.34	Graphische Darstellung des kAE des EGAori bei Datensatz 8	166
A.35	Graphische Darstellung des kAE des EGAmo bei Datensatz 8	166
A.36	Vergleichende Darstellung des mittleren kAE der verschiedenen Methoden bei Datensatz 8	167
A.37	Histogramm der Verteilung der sieben internen Knoten bei Datensatz 8. Auf der x-Achse ist die Position für einen möglichen Knoten ξ abgetragen. Die y-Achse stellt die relative Häufigkeit $P(\xi)$ für einen Knoten an der Position ξ dar. Die Ergebnisse für die internen Knoten sind in verschiedenen Farben dargestellt.	168
a	PT1	168
b	RIU	168
c	CSA	168
d	EMC	168
e	EGAori	168
f	EGAmo	168
A.38	Histogramm der Verteilung der kAE-Werte bei Datensatz 9. Auf der x-Achse sind die kAE-Werte abgetragen. Die y-Achse stellt die relative Häufigkeit $P(kAE)$ für einen kAE-Wert dar. Der dargestellte Modus entspricht dem in Tabelle 5.11 angegebenen Wert. Dieser über Kerndichteschätzung bestimmte Wert muss nicht zwangsläufig in der häufigsten Klasse des dargestellten Histogramms liegen.	169
a	PT1	169
b	RIU	169

c	CSA	169
d	EMC	169
e	EGAori	169
f	EGAmo	169
A.39	Graphische Darstellung des kAE des CSA bei Datensatz 9	170
A.40	Graphische Darstellung des kAE der EMC-Methode bei Datensatz 9	170
A.41	Graphische Darstellung des kAE des EGAori bei Datensatz 9	171
A.42	Graphische Darstellung des kAE des EGAmo bei Datensatz 9	171
A.43	<i>Belastbarkeit</i> Rauschmodell III Datensatz 5	175
A.44	<i>Belastbarkeit</i> Rauschmodell III Datensatz 6	175
A.45	<i>Belastbarkeit</i> Rauschmodell III Datensatz 7	176
A.46	<i>Belastbarkeit</i> Rauschmodell III Datensatz 8	176
A.47	<i>Belastbarkeit</i> Rauschmodell III Datensatz 9	177
A.48	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 4	177
A.49	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 6	178
A.50	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 7	178
A.51	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 8	179
A.52	<i>Belastbarkeit</i> Rauschmodell IV Datensatz 9	179
A.53	<i>Belastbarkeit</i> Rauschmodell V Datensatz 4	180
A.54	<i>Belastbarkeit</i> Rauschmodell V Datensatz 5	180
A.55	<i>Belastbarkeit</i> Rauschmodell V Datensatz 6	181
A.56	<i>Belastbarkeit</i> Rauschmodell V Datensatz 7	181
A.57	<i>Belastbarkeit</i> Rauschmodell V Datensatz 8	182

Tabellenverzeichnis

2.1	Einfluss der Stellgrößen auf die stückweise polynomialen Funktionen	15
3.1	Zusammenfassung Modellwahl bei B-Splines	32
3.2	Zusammenfassung der Parametrisierung bei B-Splines	38
3.3	Zusammenfassung Knotenvektorwahl bei B-Splines	42
3.4	Zusammenfassung Kontrollpunktschätzung bei B-Splines	45
4.1	Gewählte Parameter bei der EMC-Methode	56
4.2	Gewählte Parameter beim EGA	61
5.1	Eigenschaften der Datensätze und der Modellwahl	66
5.2	Mittlere Iterationsanzahl, bei der der Globaltest bei den verschiedenen Varianten des EGA bestanden wird.	68
5.3	Verglichene Methoden der Knotenvektorwahl	68
5.4	Anteil bestandener Globaltest	71
5.5	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 1	72
5.6	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 2	80
5.7	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 3	82
5.8	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 4	85
5.9	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 5	88
5.10	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 6	92
5.11	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 9	94
5.12	Lage der geschätzten Kontrollpunkte (schwarz) und daraus empirisch berechnete Konfidenzellipse (rot, $\alpha = 5\%$) bei Rauschmodell I und Datensatz 9	100
5.13	Parameter der empirisch berechneten Konfidenzellipsen ($\alpha = 5\%$) in Rauschmodell I und Datensatz 9	101
5.14	Eigenschaften der verschiedenen 95 %-Konfidenzellipsen in Rauschmodell I, Datensatz 9 und der Knotenvektormethode EGAmo d	102
5.15	Kennziffern der Modellwahl bei den verschiedenen Methoden zur Knotenvektorwahl berechnet aus den 500 Simulationsdurchläufen.	105
5.16	Kennziffern des kAE der verschiedenen Methoden bei abweichender Modellwahl in Datensatz 4	105
5.17	Eigenschaften der realen Datensätze und der Modellwahl	111
5.18	Kennziffern des kAE für die Kreuzvalidierung der realen Datensätze unter Verwendung verschiedener Schätzer. Die Einheit des Datensatzes R1 ist [mm]. Die Einheit des Datensatzes R2 ist [m].	112
5.19	Erwartungswerte für kAE_{Training}	112
5.20	Kennziffern des kAE_{Training} der realen Datensätze unter Verwendung der Methode der kleinsten Quadrate. Die Einheit des Datensatzes R1 ist [mm]. Die Einheit des Datensatzes R2 ist [m]. In der rechten Spalte ist das Verhältnis zwischen kAE_{Training} und kAE dargestellt. Dieses ist einheitenlos.	113
6.1	Verwendete Rauschmodelle (RM)	118
6.2	kAE-Mittelwert der rohen Beobachtungen	119
6.3	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell II (t-verteilte Daten)	121

6.4	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIa (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 1 % der Messwerte) . . .	122
6.5	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIc (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 10 % der Messwerte)	124
6.6	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVa (1 % einseitige Ausreißer, zufällige Position)	125
6.7	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVc (10 % einseitige Ausreißer, zufällige Position)	127
6.8	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Va (1 % einseitige Ausreißer, nacheinander)	128
6.9	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Vb (5 % einseitige Ausreißer, nacheinander)	129
6.10	Lage des geschätzten Kontrollpunktes \mathbf{x}_1 (schwarz) in Datensatz 9 bei verschiedenen Rauschmodellen und Schätzern unter Verwendung des wahren Knotenvektors	132
6.11	Durchschnittliche Grenze der <i>Belastbarkeit</i> pro Rauschmodell (RM)	136
A.1	Parameterwerte und Darstellung von Datensatz 4	141
A.2	Parameterwerte und Darstellung von Datensatz 5	142
A.3	Parameterwerte und Darstellung von Datensatz 6	143
A.4	Parameterwerte und Darstellung von Datensatz 7	144
A.5	Parameterwerte und Darstellung von Datensatz 8	144
A.6	Parameterwerte und Darstellung von Datensatz 9	145
A.7	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 7	147
A.8	Kennziffern des kAE der verschiedenen Methoden bei Datensatz 8	162
A.9	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IIIb (Gaußsche Mischverteilung, erhöhtes Rauschen (5σ) bei 5 % der Messwerte) . . .	172
A.10	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell IVb (5 % einseitige Ausreißer, zufällige Position)	173
A.11	Kennziffern des kAE der verschiedenen Methoden und Schätzer in Rauschmodell Vc (10 % einseitige Ausreißer, nacheinander)	174

Algorithmenverzeichnis

1	Iterativ regewichtete kleinste Quadrate	23
2	Ablauf Residuenbasierter Iterativer Update Algorithmus	48
3	Ablauf Evolutionäre Monte-Carlo Methode	52
4	Ablauf elitärer genetischer Algorithmus	57

Lebenslauf

Persönliche Daten

Name: Johannes Bureick
Geburstag: 21.06.1986
Geburtsort: Rheda-Wiedenbrück
Familienstand: ledig

Berufliche Laufbahn

06.2013 bis heute Wissenschaftlicher Mitarbeiter am Geodätischen Institut der Leibniz Universität Hannover

Ausbildung

02.2011 bis 03.2013 Vorbereitungsdienst für den höheren vermessungstechnischen Dienst in Nordrhein-Westfalen (Bezirksregierung Detmold), Abschluss: Vermessungsassessor
10.2008 bis 01.2011 Studium der Geodäsie und Geoinformatik an der Leibniz Universität Hannover, Abschluss: Master of Science
10.2005 bis 01.2009 Studium der Geodäsie und Geoinformatik an der Leibniz Universität Hannover, Abschluss: Bachelor of Science
1996 bis 2005 Einstein-Gymnasium Rheda-Wiedenbrück, Abschluss: Abitur

