



Universität Bremen

Zentrum für Technomathematik

Prinzipienfreie Modellierung von Windenergieanlagen

Bachelorarbeit

vorgelegt von

Sören Geffken

1. Gutachter: Prof. Dr. Christof Büskens, Universität Bremen
2. Gutachter: Dr. Matthias Knauer, Universität Bremen

8. August 2011

Inhaltsverzeichnis

Abbildungsverzeichnis	v
Tabellenverzeichnis	vii
1 Einführung	1
1.1 Einleitung	1
1.2 Zielsetzung der Arbeit	1
1.3 Aufbau der Arbeit	2
1.4 Notation	2
2 Mathematische Grundlagen	3
2.1 QR-Zerlegung und QR-Updateverfahren	3
2.1.1 QR-Zerlegung mit Householder-Spiegelungen	3
2.1.2 QR-Updateverfahren mit Givens-Rotationen	7
2.1.2.1 Givens-Rotationen	7
2.1.2.2 Das Updateverfahren	8
2.2 Prinzipienfreie Modellierung	10
2.2.1 Statisches Modell	10
2.2.1.1 Modellbildung	11
2.2.1.2 Die Vorhersage	12
2.2.2 Dynamisches Modell	13
2.2.3 Das Differenzenmodell	14
2.2.3.1 Algorithmus zur Bestimmung des Parametervektors während der Laufzeit	15
2.2.3.2 Die Vorhersage	15

3	Windenergieanlage DEBRA	16
3.1	Messdaten der DEBRA	17
3.2	Besonderheiten der Messdaten	18
4	Numerische Ergebnisse	19
4.1	Das statische Modell	20
4.1.1	Variation des Ansatzgrades	20
4.1.2	Variation der Eingangsdaten	22
4.1.3	Wahl des Modellbildungszeitraumes	24
4.1.4	Optimierung des Blattneigungswinkels	25
4.2	Das dynamische Modell	28
4.2.1	Variation des Ansatzgrades	28
4.2.2	Variation der Eingangsdaten	28
4.3	Variation des dynamischen Modelles	29
4.3.1	Variation des Ansatzgrades	29
4.3.2	Variation der Eingangsdaten	30
4.4	Vergleich dynamisches Modell und statisches Modell	30
5	Zusammenfassung und Ausblick	31
A	Anhänge zu Tabelle 4.1	32
B	Anhänge zu Tabelle 4.3	40
C	Anhänge zu Tabelle 4.5	46
D	Anhänge zu den Messdaten	52
E	Informationen zum MATLAB-Programm	56
E.1	Statisches Modell	56
E.2	Dynamisches und variiertes dynamisches Modell	57
	Literaturverzeichnis	59

Abbildungsverzeichnis

4.1	Modellbildung mit Polynomen vom Grad 5	21
4.2	Ausschnitt der Vorhersage mit Polynomen vom Grad 3	22
4.3	Vorhersage mit Polynomen vom Grad 3	23
4.4	Vorhersage mit Polynomen vom Grad 3 ohne Windrichtung	24
4.5	Vorhersage mit Polynomen vom Grad 7	25
4.6	Vorhersage mit Polynomen vom Grad 5	26
4.7	Vorhersage mit Polynomen vom Grad 2	27
4.8	Vorhersage mit Polynomen vom Grad 2 bei unterschiedlichen Blatt- neigungen	27
A.1	Modellbildung mit Polynomen vom Grad 1	33
A.2	Vorhersage mit Polynomen vom Grad 1	33
A.3	Modellbildung mit Polynomen vom Grad 2	34
A.4	Vorhersage mit Polynomen vom Grad 2	34
A.5	Modellbildung mit Polynomen vom Grad 3	35
A.6	Vorhersage mit Polynomen vom Grad 3	35
A.7	Modellbildung mit Polynomen vom Grad 4	36
A.8	Vorhersage mit Polynomen vom Grad 4	36
A.9	Modellbildung mit Polynomen vom Grad 5	37
A.10	Vorhersage mit Polynomen vom Grad 5	37
A.11	Modellbildung mit Polynomen vom Grad 6	38
A.12	Vorhersage mit Polynomen vom Grad 6	38
A.13	Modellbildung mit Polynomen vom Grad 7	39
A.14	Vorhersage mit Polynomen vom Grad 7	39

B.1	Modellbildung mit Polynomen vom Grad 1	41
B.2	Vorhersage mit Polynomen vom Grad 1	41
B.3	Modellbildung mit Polynomen vom Grad 2	42
B.4	Vorhersage mit Polynomen vom Grad 2	42
B.5	Modellbildung mit Polynomen vom Grad 3	43
B.6	Vorhersage mit Polynomen vom Grad 3	43
B.7	Modellbildung mit Polynomen vom Grad 4	44
B.8	Vorhersage mit Polynomen vom Grad 4	44
B.9	Modellbildung mit Polynomen vom Grad 5	45
B.10	Vorhersage mit Polynomen vom Grad 5	45
C.1	Modellbildung mit Polynomen vom Grad 1	47
C.2	Vorhersage mit Polynomen vom Grad 1	47
C.3	Modellbildung mit Polynomen vom Grad 2	48
C.4	Vorhersage mit Polynomen vom Grad 2	48
C.5	Modellbildung mit Polynomen vom Grad 3	49
C.6	Vorhersage mit Polynomen vom Grad 3	49
C.7	Modellbildung mit Polynomen vom Grad 4	50
C.8	Vorhersage mit Polynomen vom Grad 4	50
C.9	Modellbildung mit Polynomen vom Grad 5	51
C.10	Vorhersage mit Polynomen vom Grad 5	51
D.1	Plot der Windgeschwindigkeit über alle Messzeitpunkte	53
D.2	Plot des Luftdruckes über alle Messzeitpunkte	53
D.3	Plot der Temperatur über alle Messzeitpunkte	54
D.4	Plot der Windrichtung über alle Messzeitpunkte	54
D.5	Plot der Leistung über alle Messzeitpunkte	55
D.6	Plot der Rotorblattgeschwindigkeit über alle Messzeitpunkte	55

Tabellenverzeichnis

4.1 Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im statischen Modell	21
4.2 Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im statischen Modell. Änderung in Prozentpunkten.	22
4.3 Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im dynamischen Modell	28
4.4 Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im dynamischen Modell. Änderung in Prozentpunkten.	29
4.5 Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im variierten dynamischen Modell	29
4.6 Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im variierten dynamischen Modell. Änderung in Prozentpunkten.	30

Kapitel 1

Einführung

1.1 Einleitung

In dieser Arbeit wird ein Ansatz zur mathematischen Modellierung von Windenergieanlagen vorgestellt. Zur Zeit sind neben der bestehenden Problematik der fossilen Energiequellen die aktuellen Ereignisse in Fukushima ein weiterer Anlass, dass über erneuerbare Energien weltweit diskutiert wird. Die Windenergie ist in Norddeutschland ein besonders interessantes Feld, da vor allem in küstennahen Gebieten der Wind ideale Bedingungen liefert um Energie zu gewinnen. Auch die weitere Entwicklung der Offshore-Windparks an der deutschen Nordseeküste, wie zum Beispiel der Park Alpha Ventus, erhöhen die Attraktivität dieses Themas¹. Gerade die Offshore-Windparks sorgen aber auch für neue Probleme, da die Wartung und Überwachung der Anlagen deutlich schwieriger ausfällt, als bei vergleichbaren Anlagen auf dem Festland. Zur Wartung oder Inspektion der Windräder sind schwierige und vor allem teure Manöver zum Beispiel mit dem Hubschrauber nötig.

In wie weit bietet die Mathematik Möglichkeiten, um die Betreiber der Windparks zu unterstützen und Wartung oder Inspektion zu optimieren? Es werden sehr viele Sensoren und neueste Messtechnik in den Windrädern verbaut, aber wie soll eine solche Datenflut sinnvoll genutzt werden?

1.2 Zielsetzung der Arbeit

Im Laufe dieser Ausarbeitung soll eine datenbasierte mathematische Modellierungsart vorgestellt werden. Diese Art von Modellen wurde bereits erfolgreich zur Modellierung einer Gasturbine in einer Biogasanlage und einer Kraft-Wärme-Kopplungs-Anlage getestet[4, 5]. Das Ziel dieser Arbeit ist zu testen, ob auf diese Weise

¹Weitere Informationen unter <http://www.alpha-ventus.de/>. Seite abgerufen am 06. August 2011

auch Windenergieanlagen modelliert werden können. Sollte eine solche Modellierung möglich sein, so könnten routinemäßige Wartungsintervalle eventuell vergrößert werden. Gleichzeitig könnten aber auch Inspektionen gezielt durchgeführt werden, wenn beispielsweise deutliche Abweichungen zum modellierten Zustand in der Anlage auftreten. Insgesamt sollen diese Modelle das gesamte Condition-Monitoring der Windenergieanlagen unterstützen, erweitern und verbessern.

1.3 Aufbau der Arbeit

Die mathematischen Grundlagen und die Theorie der prinzipienfreien Modellierung werden im folgenden Kapitel erläutert. Das DEBRA-Projekt und die dazugehörigen Messdaten werden im 3. Kapitel dargestellt und im 4. Kapitel werden die numerischen Ergebnisse der prinzipienfreien Modellierung der DEBRA-Daten vorgestellt.

1.4 Notation

Hier sollen kurz in dieser Arbeit gewählte Abkürzungen dargestellt werden:

- $\mathbb{K}^{m \times n}$ sind die $m \times n$ Matrizen aus dem Körper \mathbb{K}
- $GL_n(\mathbb{K})$ ist die Gruppe der invertierbaren $n \times n$ Matrizen aus dem Körper \mathbb{K} .
- $O(n) \subset GL_n(\mathbb{R})$ ist die Untergruppe der orthogonalen Matrizen in \mathbb{R}
- v^T ist der transponierte Vektor von v
- A^T ist die transponierte Matrix von A
- I beziehungsweise I_n ist die Einheitsmatrix der Matrizen des $\mathbb{K}^{n \times n}$
- e_i bezeichnet den i -ten Einheitsvektor des \mathbb{R}^n der Standardbasis
- $\langle v, w \rangle$ ist das Standard-Skalarprodukt der Vektoren v und w
- Bei der Notation von Matrizen und Vektoren stellen eckige Klammern Matrizen und runde Klammern Vektoren dar
- Bei der Darstellung von Algorithmen wird als Schreibweise die Matlab-Syntax für Vektoren und Matrizen verwendet

Kapitel 2

Mathematische Grundlagen

2.1 QR-Zerlegung und QR-Updateverfahren

In diesem Kapitel wird als Beispiel für ein Lösungsverfahren von überbestimmten Gleichungssystemen die QR-Zerlegung mit Hilfe von Householder-Spiegelungen dargestellt. Weiterhin wird das QR-Updateverfahren für sich während der Laufzeit vergrößernde Gleichungssysteme mit Givens-Rotationen erläutert. Diese Verfahren werden bei der Anwendung der prinzipienfreien Modellierung verwendet. Die folgende Theoriebildung orientiert sich an dem Buch von Peter Deuffhard und Andreas Hohmann [1]. Die Argumentation folgt hauptsächlich diesem Buch und wird durch Zusätze aus dem Buch von Martin Hanke-Bourgeois [2] ergänzt.

2.1.1 QR-Zerlegung mit Householder-Spiegelungen

Bei den QR-Verfahren handelt es sich um sogenannte Orthogonalisierungsverfahren. Der Hintergrund für die Entwicklung derartiger Verfahren sind Stabilitätsprobleme bei anderen Lösungsverfahren für lineare Gleichungssysteme. Beispielsweise bei der LR-Zerlegung einer Matrix A zur Lösung eines Problems der Art $Ax = b$ wirken sich die Konditionen der Zerlegungsmatrizen auf die Stabilität des gesamten Verfahrens aus. Die Konditionen dieser Matrizen sind z.B. beim klassischen Gauss-Algorithmus im Allgemeinen nicht beschränkt. Zur Verbesserung der Stabilität haben sich orthogonale Matrizen zur Zerlegung der Matrix A bewährt. Der Vorteil ist dann, dass für die Kondition der Zerlegungsmatrizen Q gilt:

$$\text{cond}(Q) = \|Q\|_2 \|Q^{-1}\|_2 = \|Q\|_2 \|Q^T\|_2 = 1 \cdot 1 = 1 \text{ für } Q \in O(m)$$

Die Zerlegungsmatrizen haben somit keinerlei Auswirkung auf die Stabilität des gesamten Verfahrens[1]. Ein weiterer großer Vorteil besteht darin, dass sich nicht nur klassische Probleme der Gestalt $Ax = b$ lösen lassen, sondern dass auch

überbestimmte Gleichungssysteme der Gestalt $\min_{x \in \mathbb{R}^n} \|b - Ax\|$ mit diesen Verfahren gelöst werden können, wie zum Beispiel bei der linearen Ausgleichsrechnung. Die Idee ist folgendes Resultat aus Deuffhard/Hohmann[1]:

Satz 2.1 Sei $A \in \mathbb{R}^{m \times n}$, $m \geq n$, von maximalem Rang, $b \in \mathbb{R}^m$, und sei $Q \in O(m)$ mit

$$Q^T A = \begin{bmatrix} R \\ 0 \end{bmatrix} \text{ und } Q^T b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$

wobei $b_1 \in \mathbb{R}^n$, $b_2 \in \mathbb{R}^{m-n}$ und $R \in \mathbb{R}^{n \times n}$ eine invertierbare rechte obere Dreiecksmatrix ist. Dann ist $x = R^{-1}b_1$ die Lösung des linearen Ausgleichsproblems $\min_{x \in \mathbb{R}^n} \|b - Ax\|$.

Beweis: Da $Q \in O(m)$, gilt für alle $x \in \mathbb{R}^n$

$$\|b - Ax\|^2 = \|Q^T(b - Ax)\|^2 = \left\| \begin{pmatrix} b_1 - Rx \\ b_2 \end{pmatrix} \right\|^2 = \|b_1 - Rx\|^2 + \|b_2\|^2 \geq \|b_2\|^2$$

Es ist $\text{Rang } A = \text{Rang } R = n$, somit ist R invertierbar. Der erste Summand $\|b_1 - Rx\|^2$ verschwindet daher genau für $x = R^{-1}b_1$. Es bleibt zu beachten, dass das Residuum $r := b - Ax$ im Allgemeinen nicht verschwindet und dass $\|r\| = \|b_2\|$. \square

Ist es also möglich die Matrix A des Problems $\min_{x \in \mathbb{R}^n} \|b - Ax\|$ in eine Orthogonale Matrix Q und eine rechte obere Dreiecksmatrix R zu zerlegen, so ist das Problem gelöst. Zur Konstruktion der Matrizen Q_i werden Householder-Spiegelungen verwendet.

Definition 2.2 Eine Matrix der Gestalt

$$P = I - \frac{2}{v^T v} v v^T \in \mathbb{R}^{n \times n} \text{ mit } v \in \mathbb{R}^n \setminus \{0\}$$

wird Householder-Transformation genannt.

Diese Matrizen beschreiben die Spiegelung an einer zu v senkrecht stehenden Ebene. Es ergeben sich folgende Eigenschaften[1, 2]:

Proposition 2.3

- i. P ist symmetrisch, also $P^T = P$
- ii. P ist orthogonal, also $PP^T = P^T P = I$
- iii. P ist involutorisch, also $P^2 = I$
- iv. $Pv = -v$
- v. $Pw = w$ für alle $w \in v^\perp$

Beweis:

i.) $P = I - \frac{2}{v^T v} vv^T \in \mathbb{R}^{n \times n}$. Hierbei ist I als Einheitsmatrix symmetrisch und vv^T ist ebenfalls symmetrisch. Somit ist auch die Summe symmetrisch.

ii.) & iii.) $PP^T = P^T P$ folgt direkt aus i.

$$\begin{aligned} PP^T &\stackrel{\text{nach i.}}{=} PP = \left(I - \frac{2}{v^T v} vv^T \right) \left(I - \frac{2}{v^T v} vv^T \right) \\ &= I - \frac{4}{v^T v} vv^T + \left(-\frac{4}{(v^T v)^2} vv^T vv^T \right) = I \end{aligned}$$

iv.) $Pv = \left(I - \frac{2}{v^T v} vv^T \right) v = v - \frac{2}{v^T v} vv^T v = -v$

v.) $Pw = \left(I - \frac{2}{v^T v} vv^T \right) w = w$ für $w \in v^\perp$

□

Deuffhard schlägt nun vor v so zu wählen, dass ein Vektor y auf ein Vielfaches des ersten Einheitsvektors e_1 abgebildet wird. Allgemein gilt[1]:

$$\begin{aligned} y \mapsto Py &= I - \left(\frac{2}{v^T v} vv^T \right) y = y - 2 \frac{\langle v, y \rangle}{\langle v, v \rangle} v \\ &\Rightarrow \alpha e_1 = y - 2 \frac{\langle v, y \rangle}{\langle v, v \rangle} v \in \text{span}(e_1) \\ &\Rightarrow |\alpha| = \|y\|_2 \text{ und } v \in \text{span}(y - \alpha e_1) \end{aligned}$$

Um also zu erreichen, dass y auf ein Vielfaches von e_1 abgebildet wird muss gelten:

$$v := y - \alpha e_1 \text{ mit } \alpha = \pm \|y\|_2$$

An dieser Stelle kann die Numerik zu Problemen führen. Bei der Berechnung von $\|y\|_2$ können minimale Rundungsfehler auftreten. Deshalb ist die Subtraktion von $y - \alpha e_1$ problematisch, denn beide Zahlen könnten fast gleichgroß sein. Dies kann bekanntermaßen zu Auslöschung führen. Um diese zu vermeiden, findet sich in Hanke-Bourgeois für α [2]:

$$\alpha = \begin{cases} -\frac{y_1}{|y_1|} \|y\|_2 & \text{für } y_1 \neq 0 \\ -\|y\|_2 & \text{für } y_1 = 0 \end{cases}$$

Hierbei ist y_1 der 1. Eintrag des Vektors y .

Mit Hilfe dieser Konstruktion ist es nun möglich, eine Matrix $A = [A_1, \dots, A_n] \in \mathbb{R}^{m \times n}$ in eine obere Dreiecksmatrix zu transformieren. Um dies durchzuführen, wird Deuffhard/Hohmann folgend die erste Spalte A_1 als Vektor y für das gerade beschriebene Verfahren verwendet. Auf diese Weise ergibt sich nach dem 1. Schritt folgende Zerlegung[1]:

$$Q_1 A = \begin{bmatrix} \alpha_1 & & & & \\ 0 & & & & \\ \vdots & A'_2 & \dots & A'_n & \\ 0 & & & & \end{bmatrix}$$

Dabei gilt $Q_1 A_j = A'_j$ für alle $j = 2, \dots, n$. Das Verfahren wird nun für die Restmatrix $A' \in \mathbb{R}^{m-1 \times n-1}$ wiederholt. Nach dem k -ten Schritt ergibt sich folgende Matrix $A^{(k)}$ mit einer Restmatrix $T^{(k+1)} \in \mathbb{R}^{m-k \times n-k}$:

$$A^{(k)} = \begin{bmatrix} * & \dots & \dots & \dots & * \\ & \ddots & & & \vdots \\ & & * & \dots & * \\ & & 0 & & \\ & & \vdots & T^{(k+1)} & \\ & & 0 & & \end{bmatrix}$$

Es wird nun zu $T^{(k+1)}$ wie oben beschrieben die Matrix \tilde{Q}_{k+1} konstruiert. Somit ergibt sich für die Matrix Q_{k+1} :

$$Q_{k+1} = \begin{bmatrix} I_k & 0 \\ 0 & \tilde{Q}_{k+1} \end{bmatrix}$$

Nach $\min(m-1, n) =: p$ Schritten ergibt sich die gesuchte Zerlegung, da dann gilt:

$$R = Q_p \dots Q_1 A$$

Aus Proposition 2.3.iii ist bekannt, dass $Q_i^2 = I$. Somit folgt das Resultat:

$$A = QR \text{ mit } Q = Q_1 \dots Q_p$$

Nach Satz 2.1 ist es somit nun möglich, ein lineares Ausgleichsproblem $\min_{x \in \mathbb{R}^n} \|b - Ax\|$ zu lösen, indem durch Rückwärtssubstitution $x = R^{-1}b_1$ gelöst wird. Die höhere Stabilität des Verfahrens und die besseren Anwendungsmöglichkeiten als bei einer herkömmlichen LR-Zerlegung erhöhen allerdings auch den benötigten Aufwand des Verfahrens. Nach Deuffhard/Hohmann gilt für den Aufwand dieses Verfahrens[1]:

- $2n^2m$ Multiplikationen, falls $m \gg n$
- $\frac{2}{3}n^3$ Multiplikationen, falls $m \approx n$

2.1.2 QR-Updateverfahren mit Givens-Rotationen

Bei der Anwendung des dynamischen Modelles wird ebenfalls ein lineares Ausgleichsproblem $\min_{x \in \mathbb{R}^n} \|b - Ax\|$ zu lösen sein. Es gibt allerdings unterschiedliche Möglichkeiten, das Modell zu realisieren. Wird es lediglich auf Testdaten angewendet, kann eine normale QR-Zerlegung ausreichen. In der praktischen Nutzung des Modelles ist es allerdings typisch, dass während der Laufzeit neue Messdaten entstehen und das Modell diese direkt berücksichtigen und sich anpassen soll. In diesem Fall wird bei jedem Iterationsschritt eine neue Zeile mit Messdaten an die Matrix A angehängt. Damit nicht jedes Mal die gesamte QR-Zerlegung der neuen Matrix vorgenommen werden muss, ist es möglich, ein sogenanntes Updateverfahren mit Hilfe von Givens-Rotationen zu realisieren. Dieses Verfahren verwendet die bereits vorliegende QR-Zerlegung der vorherigen Matrix, um mit deutlich weniger Aufwand eine QR-Zerlegung der neuen Matrix zu ermitteln.

2.1.2.1 Givens-Rotationen

Es sei daran erinnert, dass eine Rotation um den Winkel ϕ im \mathbb{R}^2 durch die Matrix

$$\Omega = \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix}$$

beschrieben wird. Analog dazu wird nun eine Rotationsmatrix um die k - l -Ebene im n -dimensionalen definiert.

Definition 2.4 *Eine Matrix der Gestalt*

$$\Omega_{kl} = \begin{bmatrix} I & & & \\ & c & s & \\ & & I & \\ & -s & c & \\ & & & I \end{bmatrix} \begin{array}{l} \leftarrow k \\ \\ \leftarrow l \end{array} \in \mathbb{R}^{n \times n} \text{ mit } c^2 + s^2 = 1$$

wird Givens-Rotation genannt. Hierbei sind I jeweils passende Einheitsmatrizen.

Aus dieser Definition folgt[1]:

$$\Omega_{kl}x = y \text{ für } x, y \in \mathbb{R}^n \text{ mit } y_i = (\Omega_{kl}x)_i = \begin{cases} cx_k + sx_l, & \text{falls } i = k \\ -sx_k + cx_l, & \text{falls } i = l \\ x_i, & \text{falls } i \neq k, l \end{cases}$$

Wird nun eine Matrix Ω_{kl} mit einer Matrix A multipliziert, so ergibt dies spaltenweise betrachtet:

$$A = [A_1, \dots, A_n] \Rightarrow \Omega_{kl}A = [\Omega_{kl}A_1, \dots, \Omega_{kl}A_n]$$

Das bedeutet, um eine Matrix A auf obere rechte Dreiecksstruktur zu bringen müssen k, l, c, s geschickt gewählt werden, um die Einträge unterhalb der Diagonalen zu eliminieren. Da die Givens-Rotation nur auf der k - l -Ebene agiert, lässt sich dies auf den 2-Dimensionalen Fall reduzieren. s und c nehmen hierbei die Rollen von Sinus und Kosinus ein. Es folgt:

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{pmatrix} x_k \\ x_l \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}$$

Da die Einträge der Matrix reellwertig sind, soll hier auch der Vektor x reellwertig sein. Weiterhin soll eine Dreiecksstruktur konstruiert werden, darum wird angenommen, dass $x_l \neq 0$. Somit erhält man 2 Gleichungen unter den Nebenbedingungen $s^2 + c^2 = 1$ und $x_k^2 + x_l^2 \neq 0$:

$$\begin{aligned} cx_k + sx_l &= r \\ -sx_k + cx_l &= 0 \end{aligned}$$

In dem Buch von Deuffhard/Hohmann findet sich das Ergebnis

$$r = \pm \sqrt{x_k^2 + x_l^2}, c = \frac{x_k}{r} \text{ und } s = \frac{x_l}{r},$$

um die Gleichungen unter den Nebenbedingungen zu lösen[1]. Somit ist nun eine Rechenvorschrift gegeben um eine Givens-Rotation zu einer Spalte der Matrix A zu konstruieren.

2.1.2.2 Das Updateverfahren

Es sei nun folgendes Minimierungsproblem gegeben:

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$

Die Matrix A wird um eine weitere Zeile mit Messdaten vergrößert. Sei dies der Vektor $a \in \mathbb{R}^n$. Der dazugehörige Messwert der gesuchten Größe sei ebenfalls gegeben und heiße b_{m+1} . Weiterhin ist die QR-Zerlegung der Matrix A bekannt. In

dem Numerikskript von Christof Büskens [3] findet sich folgende Überführung in ein ähnliches Problem[3]:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \left\| \begin{bmatrix} A \\ a^T \end{bmatrix} x - \begin{pmatrix} b \\ b_{m+1} \end{pmatrix} \right\| &= \min_{x \in \mathbb{R}^n} \left\| \begin{bmatrix} Q \begin{bmatrix} R \\ 0 \end{bmatrix} \\ a^T \end{bmatrix} x - \begin{pmatrix} b \\ b_{m+1} \end{pmatrix} \right\| \\ &= \min_{x \in \mathbb{R}^n} \left\| \begin{bmatrix} Q & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \begin{bmatrix} R \\ 0 \\ a^T \end{bmatrix} \\ x \end{bmatrix} - \begin{pmatrix} b \\ b_{m+1} \end{pmatrix} \right\| \end{aligned}$$

Mit Hilfe der bekannten QR-Zerlegung von A folgt nun:

$$\begin{bmatrix} Q & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \begin{bmatrix} R \\ 0 \\ a^T \end{bmatrix} \\ x \end{bmatrix} = \begin{bmatrix} A \\ a^T \end{bmatrix} \Leftrightarrow \begin{bmatrix} \begin{bmatrix} R \\ 0 \\ a^T \end{bmatrix} \\ x \end{bmatrix} = \begin{bmatrix} Q & 0 \\ 0 & 1 \end{bmatrix}^T \begin{bmatrix} A \\ a^T \end{bmatrix}$$

Somit lässt sich das Problem auf die Lösung des folgenden Systems reduzieren:

$$\tilde{R}x := \begin{bmatrix} \begin{bmatrix} R \\ 0 \\ a^T \end{bmatrix} \\ x \end{bmatrix} = \begin{bmatrix} Q & 0 \\ 0 & 1 \end{bmatrix}^T \begin{pmatrix} b \\ b_{m+1} \end{pmatrix} := \begin{pmatrix} \tilde{b} \\ b_{m+1} \end{pmatrix}$$

Mit Hilfe der zuvor dargestellten Theorie über die Givens-Rotationen lässt sich diese Matrix durch Multiplikation mit einer Folge von Givens-Rotationen erneut auf obere rechte Dreiecksstruktur bringen. Vor der Anwendung der Rotationsmatrizen ergibt sich folgende Situation:

$$\begin{bmatrix} \begin{bmatrix} R \\ 0 \\ a^T \end{bmatrix} \\ x \end{bmatrix} = \begin{bmatrix} * & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & 0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & * \\ \vdots & \vdots & \vdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \\ a_1 & a_2 & \dots & a_n \end{bmatrix} = \tilde{R}$$

Hierbei ist $R \in \mathbb{R}^{n \times n}$, $0 \in \mathbb{R}^{m-n \times n}$. Nun muss, mit der Notation aus Definition 2.4, wie folgt vorgegangen werden:

$$\Omega_{n,m+1} \dots \Omega_{1,m+1} \tilde{R}x = \Omega_{n,m+1} \dots \Omega_{1,m+1} \begin{pmatrix} \tilde{b} \\ b_{m+1} \end{pmatrix}$$

Auf diese Weise wurde die Matrix in n Schritten wieder auf Dreiecksstruktur gebracht und das erweiterte Problem ist durch Rückwärtssubstitution lösbar.

2.2 Prinzipienfreie Modellierung

In diesem Kapitel wird der mathematische Hintergrund der prinzipienfreien Modellierung dargestellt. Für diese Art eines datenbasierten mathematischen Modelles stehen zwei unterschiedliche Ansätze zur Verfügung, die in den beiden Abschnitten dieses Kapitels nun dargestellt werden. Die Theoriebildung erfolgt auf Grundlage der Publikationen von Bodo Blume, Christof Büskens und Dennis Wassel [4, 5].

2.2.1 Statisches Modell

Zunächst soll das statische Modell beschrieben werden. Von Interesse ist eine bestimmte Größe im betrachteten System, zum Beispiel bei der Windenergieanlage die produzierte Leistung. Die Annahme dieser Modellvariante besteht darin, dass es möglich ist, aus den vorhandenen Messwerten $x \in \mathbb{R}^k$ an einem Zeitpunkt die gesuchte Größe $y \in \mathbb{R}$ zu bestimmen. $x \in \mathbb{R}^k$ bedeutet in diesem Fall, dass neben y k verschiedene Größen gemessen werden. Weiterhin wird bei den statischen Modellen angenommen, dass der Zusammenhang zwischen den vorhandenen Messwerten und der gesuchten Größe sich während des betrachteten Zeitraumes nicht verändert.

Es liegen Messwerte $x \in \mathbb{R}^k$ vor. Ferner existiert ein Parametervektor $p \in \mathbb{R}^n$ und es gibt eine Funktion $f : \mathbb{R}^k \times \mathbb{R}^n \rightarrow \mathbb{R}$, welche die gesuchte Größe beschreibt. Es gilt somit $f(x, p) = y$. Unter der Annahme, dass f linear in p ist, lässt sich definieren:

$$f(x, p) := c(x)^T p \quad (2.1)$$

Wobei c im Allgemeinen eine nichtlineare Funktion ist.[4]

Zunächst soll nun c weiter präzisiert werden. Als Ansatzfunktion für c werden an dieser Stelle Monome in k Variablen gewählt, sei dies nun $P(x)$. P lässt sich mit dem Multiindex $\alpha \in \mathbb{N}_0^k$ wie folgt darstellen:

$$\text{Mit } |\alpha| = \sum_{i=1}^k \alpha_i \text{ ist } x^\alpha = \prod_{i=1}^k x_i^{\alpha_i} \text{ für } x \in \mathbb{R}^k \text{ und } \alpha \in \mathbb{N}_0^k$$

Wird dies nun um den Parametervektor p ergänzt und der maximale Grad der Monome mit d festgesetzt, folgt[4]:

$$P(x, p) = \sum_{|\alpha| \leq d} p_\alpha x^\alpha$$

Die Auswirkung von d auf die Genauigkeit des Modells wird in den numerischen Experimenten untersucht. Die Anzahl aller Multiindizes α lässt sich mit Hilfe der Kombinatorik bestimmen:

Sei d der betrachtete Grad und es liegen k Variablen vor. Diese Situation ist mit

einem Urnenmodell wie folgt zu beschreiben: In der Urne befinden sich k Kugeln, und jede repräsentiert eine Variable. Es wird d Mal aus dieser Urne eine Kugel gezogen und der entsprechende Eintrag des Multiindex um eins erhöht, dann wird die Kugel zurück in die Urne gelegt. Die Mächtigkeit des Ergebnisraumes Ω wird in diesem Fall beschrieben durch $|\Omega| = \binom{k+d-1}{d}$. Daraus folgt für die Anzahl aller Multiindizes bis zum Grad d [4]:

$$n_d = \sum_{i=0}^d \binom{k+i-1}{i} = \binom{k+d}{d} \quad (2.2)$$

Das Modell wird im Folgenden in zwei Bereiche aufgeteilt. Zunächst findet die Modellbildung zur Bestimmung des Parametervektors p statt und danach wird das Modell angewendet.

2.2.1.1 Modellbildung

Um den Parametervektor p zu bestimmen, werden Messwerte benötigt. Seien Messwerte der Eingangsgrößen $x_1, \dots, x_b \in \mathbb{R}^k$, sowie dazugehörige Messwerte der gesuchten Größe $y_1, \dots, y_b \in \mathbb{R}$ an insgesamt b Zeitpunkten vorgegeben. Wie in (2.1) beschrieben gelte[4]:

$$\begin{aligned} f(x_i, p) &= y_i \text{ für } i = 1, \dots, b \\ \Leftrightarrow c(x_i)^T p &= y_i \text{ für } i = 1, \dots, b \\ \Leftrightarrow P(x_i, p) &= y_i \text{ für } i = 1, \dots, b \end{aligned}$$

Sei im Folgenden $F(x, p) = f(x_i, p)$ für $i = 1, \dots, b$. Dann ist $F(x, p) \in \mathbb{R}^b$. Außerdem sei $y := y_i$ für $i = 1, \dots, b$. Jetzt ist folgendes Minimierungsproblem zur Bestimmung von p zu lösen:

$$\min_{p \in \mathbb{R}^n} \|F(x, p) - y\|_2^2 = \min_{p \in \mathbb{R}^n} \sum_{i=1}^b (f(x_i, p) - y_i)^2$$

Auf Grund der Annahme, dass f linear in p ist, kann dieses Minimierungsproblem mit $b \gg n$ auf die Lösung eines überbestimmten Gleichungssystems reduziert werden. Es sei die Matrix $C(x) \in \mathbb{R}^{b \times n}$ durch die Zeilen $c(x_i)^T$ definiert. Es ist $F(x, p) = C(x)p$, und das Minimierungsproblem lässt sich darstellen als:

$$\min_{p \in \mathbb{R}^n} \sum_{i=1}^b (f(x_i, p) - y_i)^2 = \boxed{\min_{p \in \mathbb{R}^n} \|Cp - y\|_2^2} \quad (2.3)$$

Zur Lösung dieses Minimierungsproblem kann das in Kapitel 2.1.1 erläuterte QR-Verfahren genutzt werden. An dieser Stelle soll die Matrix $C(x)$ noch näher

dargestellt werden. Es wurde bereits erläutert, dass Monome als Ansatzfunktion Verwendung finden. Auf diese Weise ergibt sich die Matrix $C(x)$ wie folgt:

Es liegen wieder die Messwerte $x_i \in \mathbb{R}^k$ für $i = 1, \dots, b$ vor und der maximale Grad der Monome sei festgesetzt mit d . Dann sind außerdem die Multiindizes $\alpha_1, \dots, \alpha_n$ bekannt, wobei nach (2.2) $n = \binom{k+d}{d}$ ist. Die Matrix $C(x)$ ist [4]:

$$C(x) = \begin{pmatrix} x_1^{\alpha_1} & \cdots & x_1^{\alpha_n} \\ \vdots & & \vdots \\ x_b^{\alpha_1} & \cdots & x_b^{\alpha_n} \end{pmatrix} \quad (2.4)$$

Mit dieser Matrix $C(x)$ und dem QR-Verfahren lässt sich der Parametervektor p finden, so dass $P(x_i, p)$ eine gute Approximation an y_i liefert. Die Modellbildung ist mit der Bestimmung von p abgeschlossen.

2.2.1.2 Die Vorhersage

Mit dem bestimmten Parametervektor p können Vorhersagen für beliebige Zeitpunkte gemacht werden. Um dies durchführen zu können, werden zu dem betrachteten Zeitpunkt die Größen der anderen Messwerte benötigt. Sind also Messwerte $x \in \mathbb{R}^k$ zu einem Zeitpunkt bekannt, lässt sich eine Näherung an den zu erwartenden Wert $y \in \mathbb{R}$ wie folgt berechnen:

$$\begin{pmatrix} x^{\alpha_1} & \cdots & x^{\alpha_n} \end{pmatrix} \begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix} = y$$

Auf diese Weise kann zu beliebigen Zeitpunkten unter Kenntnis der anderen Messgrößen die gesuchte Größe ermittelt werden. Das statische Modell liefert in der Regel gute Ergebnisse. Wie bereits eingangs erwähnt allerdings nur unter der Annahme, dass der Zusammenhang zwischen den Daten im betrachteten System sich nicht ändert. Werden aber grundlegende Änderungen am betrachteten System vorgenommen, wie zum Beispiel durch eine Reparatur oder den Austausch von Bauteilen, dann ändern sich die physikalischen Eigenschaften des betrachteten Systems. In diesem Fall würde sich auch der Parametervektor ändern. Da dieser aber nur während der Modellbildung angepasst wird, liefert das Modell ab diesem Zeitpunkt falsche Ergebnisse. Sollte ein solcher Fall eintreten, muss die Modellbildungsphase wiederholt werden. Dieser Sachverhalt wird in den numerischen Auswertungen an den vorliegenden Messdaten noch weiter verdeutlicht.

2.2.2 Dynamisches Modell

Als eine Erweiterung des statischen Modelles soll das dynamische Modell eingeführt werden. In diesem Fall wird eine zeitabhängige Veränderung im System angenommen. Mathematisch wird wie folgt vorgegangen:

Es liegen Messwerte $v_i \in \mathbb{R}^k, i = 1, \dots, M$ vor. Weiterhin sind Eingangswerte $z_i \in \mathbb{R}^m, i = 1, \dots, M$ vorhanden. Theoretisch stellen die Messwerte all jene Größen dar, auf die der Modellierer keinen Einfluss nehmen kann, beispielsweise die produzierte Leistung der Anlage. Die Eingangsgrößen sind diejenigen Werte, die gesteuert werden können. Im vorliegenden Beispiel wäre dies beispielsweise die Rotorblattneigung. In den numerischen Tests wird von dieser Theorie allerdings ein wenig abgewichen, um Vorhersagen für bestimmte Wettersituationen zu machen. Dann wird auch die Windgeschwindigkeit zu einem Eingangswert, da es nicht möglich ist, nur aus den vorliegenden Daten die Windgeschwindigkeit in der Zukunft zu prognostizieren.

Beim dynamischen Modell ist nun die Annahme, dass eine Modellfunktion $v = h(t, z, p)$ existiert, abhängig vom Zeitpunkt t mit Eingangsgrößen $z \in \mathbb{R}^m$ und dem gesuchten Parameter $p \in \mathbb{R}^n$. Im Unterschied zum statischen Modell wird eine unbekannte Differentialgleichung zur Bestimmung von $h(t, z, p)$ angesetzt. Das heißt:

$$\begin{aligned} \dot{h}(t) &= f(h, t, u, p) \\ h(t_0) &= v_0 \\ u : [t_0, t_M] &\rightarrow \mathbb{R}^m, \text{ wobei hier } u(t_i) = q(z_i) \\ p &\in \mathbb{R}^n \\ \Rightarrow g(p) &:= \sum_{i=1}^M \|h(t_i, z_i, p) - v_i\|_2^2 \end{aligned}$$

Somit ergibt sich folgendes Minimierungsproblem:

$$\boxed{\min_{p \in \mathbb{R}^n} g(p)} \quad (2.5)$$

$q(z_i)$ sind hier Ansatzfunktionen für $u(t_i)$, mit denen die Eingangswerte repräsentiert werden. Bei der folgenden Theorie wird $q(z_i)$ lediglich als lineare Funktion angesetzt. Es ist aber möglich, ähnlich wie beim statischen Modell, Multiindizes zu verwenden, um verschiedene Monome miteinander zu kombinieren. Diese Variante zur Verfeinerung der Methode wird in den numerischen Experimente eingesetzt.

Bei dem Minimierungsproblem handelt es sich im Allgemeinen um eine nichtlineare Minimierung. Zur Lösung des Problems ist es nötig, die Differentialgleichungen numerisch zu lösen und dann die Minimierung zum Beispiel mit Hilfe von SQP-Verfahren durchzuführen. Auf diesen Lösungsweg soll an dieser Stelle nicht weiter

eingegangen werden. Stattdessen wird eine Vereinfachung vorgenommen, um das Problem zu linearisieren. Hierzu wird folgendes getan[5]:

$$\dot{h}(t) = f(h, t, u, p) \Rightarrow \dot{h}(t) = A(p)h(t) + B(p)u(t) + C(p) \quad (2.6)$$

Es wird also angenommen, die Differentialgleichung lässt sich auf einen in $h(t)$ linearen Teil mit der Matrix A und einen in $u(t)$ linearen Teil mit der Matrix B und einen konstanten Term C reduzieren. Unter dieser Annahme wird nun ein sogenanntes Differenzenmodell betrachtet.

2.2.3 Das Differenzenmodell

Der Ansatz der sich aus dem linearen Ansatz (2.6) ableiten lässt, sieht wie folgt aus:

$$v_{i+1} = Av_i + Bu_i + C$$

Es sind nun:

$$v_k \in \mathbb{R}^n, u_k \in \mathbb{R}^m, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^n$$

Hieraus lässt sich ein lineares Ausgleichsproblem formulieren:

$$x := (a_{i,j}, b_{i,k}, c_i)^T \in \mathbb{R}^{(n+m+1) \times n}, \quad i, j = 1, \dots, n \quad k = 1, \dots, m$$

$$b := (v_{1,i}, \dots, v_{M,i})^T \in \mathbb{R}^{M \times n}, \quad i = 1, \dots, n$$

$$D := \begin{pmatrix} v_0 & u_0 & 1 \\ \vdots & \vdots & \vdots \\ v_{M-1} & u_{M-1} & 1 \end{pmatrix} \in \mathbb{R}^{M \times (n+m+1)} \quad (2.7)$$

Hieraus ergibt sich folgendes lineares Minimierungsproblem:

$$\boxed{\min_{x \in \mathbb{R}^{((n+m+1) \times n)}} \|Dx - b\|_2^2} \quad (2.8)$$

Die dahinterstehende Idee ist also, dass aus den Zuständen und Eingangsgrößen des vorherigen Zeitpunktes die Zustände des folgenden Zeitpunktes ermittelt werden können. Wird das Minimierungsproblem über einen gewissen Zeitraum, der Notation oben folgend, mit M Zeitpunkten gelöst, so ergibt sich ein Parametervektor x . Mit Hilfe von x ist die Differentialgleichung nun explizit bekannt, da die Matrix A , die Matrix B sowie der konstante Vektor C in x enthalten sind. Mit dem, auf diese Weise bestimmten, Parametervektor x ist es nun möglich, Vorhersagen für spätere Zeitpunkte zu treffen. Somit startet auch das dynamische Modell zunächst mit einer Modellbildungsphase um danach Vorhersagen erstellen zu können.

Dieses Minimierungsproblem lässt sich wie beim statischen Modell mit Hilfe einer QR-Zerlegung effizient lösen. Es gibt zwei Möglichkeiten, das Modell zu implementieren. Zum Einen werden zunächst alle Zeitpunkte in die Matrix D eingetragen um dann die QR-Zerlegung wie beim statischen Modell vorzunehmen. Zum Anderen ist es möglich, die Matrix D erst während der Laufzeit entstehen zu lassen. In diesem Fall wird lediglich für v_0 der exakte Messwert genommen und danach nur noch die ermittelten Werte. Die Matrix D entsteht also erst während der Berechnung, das genaue Vorgehen wird im folgenden Unterabschnitt erklärt.

2.2.3.1 Algorithmus zur Bestimmung des Parametervektors während der Laufzeit

An dieser Stelle soll der Algorithmus zur Bestimmung des Parametervektors dargestellt werden:

Algorithmus

(Schritt 1) Setze $b = v_2$ und erstelle die erste Zeile von D , so dass $D(1, :) := [v(1, :) \ u(1, :) \ 1]$. In diesem Fall ist die QR-Zerlegung trivial. $Q = 1, R = [v(1, :) \ u(1, :) \ 1]$. Bestimme x , so dass $Rx = b$.

(Schritt 2) Setze $v(2, :) = Rx$.

(Schritt 3) Für $i = 2, \dots, M$ setze $b = v_{i+1}$ und erstelle die nächste Zeile von D mit $D(i, :) := [v(i, :) \ u(i, :) \ 1]$. Es wird mittels *qrUpdate* die QR-Zerlegung der neuen Matrix D vorgenommen.

(Schritt 4) Setze $v(i + 1, :) = Rx$.

(Schritt 5) Gehe zu Schritt 3 und inkrementiere i .

2.2.3.2 Die Vorhersage

Nach der Bestimmung des Parametervektors x sollte es nun möglich sein eine Vorhersage zu treffen. Hierzu werden die Messwerte der Eingangsgrößen z_i benötigt und wie bisher auch im Vektor u gespeichert. Außerdem wird der letzte ermittelte Wert von v benötigt. Es ergibt sich nun folgende Rechenvorschrift zur Bestimmung des gesuchten Wertes:

$$\begin{pmatrix} v(M) & u(M) & 1 \end{pmatrix} x = v(M + 1)$$

Auf diese Weise lässt sich die Vorhersage solange berechnen, wie die Messungen der Eingangsgrößen vorliegen.

Kapitel 3

Windenergieanlage DEBRA

Die numerischen Experimente der eben vorgestellten Verfahren werden an den Messdaten des DEBRA-Projektes getestet. Bei DEBRA handelt es sich um ein Vorhaben der Deutschen Forschungs- und Versuchsanstalt für Luft- und Raumfahrt aus dem Jahre 1988. Es ging dabei um „Entwicklung, Bau und Betrieb einer 50/100 kW Windkraftanlage (WKA) in deutsch/brasilianischer Zusammenarbeit“. Es sollte eine Windkraftanlage entwickelt und getestet werden, die einfach zu transportieren und aufzubauen ist und auch für Schwachwindbetrieb geeignet ist.[6]

Es wurden im Rahmen des Projektes zwei Testanlagen aufgestellt. Eine in Brasilien und eine in Deutschland. Die vorliegenden Daten stammen von der deutschen Anlage. Der Standort der Anlage war zwischen Schnittlingen und Stötten in Baden-Württemberg auf der schwäbischen Alb in unmittelbarer Nähe der Wetterstation Stötten.

Das besondere der Anlage ist, dass von den zwei Generatoren je nach Wind entweder nur einer bei schwachem Wind oder beide bei stärkerem Wind eingesetzt werden. Der Betrieb der Anlage ist in zwei Laststufen aufgeteilt, wobei je nach Bedarf zwischen den beiden umhergeschaltet werden kann.

Übersicht über die Laststufen [6]:

- **Lastbetrieb I**

Nennleistung 30 kW, Rotordrehzahl 33 $1/\text{min}$, Schwachwindbetrieb

- **Lastbetrieb II**

Nennleistung 100 kW, Rotordrehzahl 50 $1/\text{min}$, Betrieb bei mittlerem und starkem Wind; Leistungsregelung

Zur Regelung der Funktionsweise der Anlage ist ein Regelungs- und Betriebsführungssystem in der Anlage installiert. Dieses System arbeitet mit einem elektrohydraulischen Blattverstellungssystem. Folgende Aufgaben hat das System zu erfüllen[6]:

- Drehzahlregelung in allen Übergangszuständen beim Beschleunigen und Bremsen des Rotors. Die Drehzahl wird entlang einer istwertgeführten Rampe hochgefahren und entlang einer hardwaremäßig vorgegebenen Rampe verringert.
- Ausregeln der Rotordrehzahl auf die jeweiligen Synchrondrehzahlen für Laststufe I und II
- Netzparallelbetrieb bei Teillast. Hierbei wird ein konstanter Blattwinkel eingestellt, d.h. die Leistungsschwankungen resultieren allein aus den Änderungen der Windgeschwindigkeit.
- Leistungsregelung oberhalb der Nennwindgeschwindigkeit mit Hilfe eines variablen Blattwinkels

Weiterhin ist zur Steuerung der Anlage eine Bremse eingebaut. Diese wird allerdings nur genutzt um Notabschaltungen zusammen mit dem Regelsystem durchzuführen und bei Wartungsarbeiten sicherzustellen, dass keine Rotorbewegung stattfindet.

3.1 Messdaten der DEBRA

Für die numerischen Experimente liegen Daten von der Anlage in Schnittlingen/Stötten vor. Diese Daten bestehen aus:

- Windgeschwindigkeit in m/s
- Lufttemperatur in °C
- Luftdruck in hPa
- Windrichtung aus Anlagensicht in °
- Elektrische Leistung in kW
- Rotorblattgeschwindigkeit in Umdrehungen pro Minute
- Rotorblattneigung in °
- Status Laststufe I
- Status Laststufe II

Die Daten wurden je nach Sensor mit unterschiedlichen Frequenzen von 5ms bis 100ms aufgezeichnet[6], dies ist in den vorliegenden Daten aber nicht mehr erkennbar, da 10 Minuten-Mittelwerte gebildet wurden.

3.2 Besonderheiten der Messdaten

Bei den Daten sind einige problematische Umstände zu beachten. Die Temperaturwerte sind teilweise nicht korrekt, da auf Grund von Nebel, der bei hohen Windgeschwindigkeiten an die Anlage geweht wird, sich eine Eisschicht auf den Rotorblättern bildet und dann die Messgenauigkeit des Temperatursensors geschädigt wird. Deshalb wurde empfohlen, die Messwerte mit negativen Temperaturen zu ignorieren.

Weiterhin wird die Windgeschwindigkeit auf einem in der Nähe befindlichen Mast gemessen. Ist die angegebene Windrichtung zwischen 0° und 45° , so liegt der Mess-Mast im Windschatten der Anlage und die gemessene Windgeschwindigkeit ist fehlerhaft.

Im DEBRA-Bericht [6] wird beschrieben, dass Regen eine Auswirkung auf die Leistungsfähigkeit der Anlage hat. Außerdem sind auch Trockenperioden problematisch, da die zunehmende Verunreinigung der Anlage zum Beispiel durch Insekten oder Ablagerungen von Abgasen nicht durch Regen beseitigt werden. In den vorliegenden Daten sind leider keine Werte über Niederschlag vorhanden, so dass diese Veränderungen zwar Auswirkungen auf die Leistung haben, aber nicht dem tatsächlichen Grund zugeordnet werden können.

Kapitel 4

Numerische Ergebnisse

In diesem Kapitel werden numerische Auswertungen mit den Messdaten der DEBRA-Windenergieanlage durchgeführt. Zunächst sollen die verwendete Rechnerarchitektur und Software aufgelistet werden, um eine objektivere Beurteilung der benötigten Rechenzeiten zu ermöglichen.

- Desktop PC
- CPU: Intel Core 2 Quad @ 2,40 GHZ (Quadcore-Prozessor)
- RAM: 4,00 GB
- Windows 7 Version 6.1 (Build 7600) (64-bit Version)
- MATLAB Version 7.9.0.529 (R2009b)

Programmiert wurde in MATLAB und die erstellten m-Files wurden auch direkt in MATLAB ausgeführt.

In den folgenden numerischen Auswertungen wird die Leistung der Anlage als gesuchte Größe gewählt. Die verwendeten Messdaten der DEBRA werden außerdem bereits vor den eigentlichen numerischen Tests durch die Routine `datenvorb.m` gefiltert. Es werden die bereits in Kapitel 3.2 erwähnten fehlerhaften Messzeitpunkte nach den Kriterien Temperatur unter 0° oder Windrichtung unter 45° aussortiert. Weiterhin wurden die Daten in die beiden aus Kapitel 3.1 bekannten Laststufen aufgeteilt und separat betrachtet. Auf Grund ihrer Proportionalität zur Leistung der Anlage wird zudem die Rotorblattgeschwindigkeit in den numerischen Untersuchungen nicht verwendet.

Bevor die numerischen Auswertungen durchgeführt werden, sollen an dieser Stelle noch einige Hintergrundinformationen gegeben werden. Das MATLAB-Programm erstellt automatisch die zum Test gehörigen Abbildungen. Pro Test sind

dies jeweils zwei Abbildungen. Die erste Abbildung stellt das Modell während der Modellbildungsphase dar. Die zweite Abbildung zeigt das Ergebnis der Vorhersagephase. In beiden Fällen werden zwei Grafiken erstellt. In der oberen Grafik werden in blau die Messwerte der gesuchten Größe eingezeichnet und in rot die vom Modell berechneten Werte. Die untere Grafik stellt den absoluten Fehler, also die Differenz von Modell- und Messwert dar. Für weitere Informationen zur Anwendung des MATLAB-Programmes sei an dieser Stelle auf den Anhang E verwiesen. Dort werden die Dokumentationen der Hauptprogramme, sowie jeweils ein Anwendungsbeispiel abgedruckt.

4.1 Das statische Modell

Im ersten Abschnitt dieses Kapitels soll das statische Modell aus Kapitel 2.2.1 angewendet werden. Zu Beginn sollen die Ergebnisse unter Verwendung unterschiedlicher maximaler Polynomgrade der Ansatzfunktion gezeigt und analysiert werden. Auf Grund der Ähnlichkeit der Tests in beiden Laststufen werden die Experimente mit Laststufe I durchgeführt, da in dieser Laststufe mehr Messwerte zur Verfügung stehen.

4.1.1 Variation des Ansatzgrades

Für diesen Testlauf wurde für die Modellbildung ein Zeitraum von 1008 Messpunkten (Abstand der Messpunkte: 10 Minuten), also genau eine Woche gewählt. Die Vorhersage erstreckt sich über 360 Messpunkte, das entspricht zweieinhalb Tagen. Als Beispiel für die Modellbildungsphase wird in Abbildung 4.1 exemplarisch Grad 5 gezeigt. In der Modellbildungsphase sind die Ergebnisse insgesamt sehr gut und die Abbildungen deshalb auch alle sehr ähnlich. Im Anhang A befinden sich auch die Abbildungen der übrigen Grade. In Tabelle 4.1 werden die Ergebnisse der unterschiedlichen Grade der Ansatzfunktion dargestellt. Der Fehler wurde jeweils relativ zum durchschnittlichen Wert der modellierten Energie im betrachteten Zeitraum in Prozent berechnet.

Die Tabelle zeigt, dass zumindest in der Modellbildungsphase ein höherer Polynomgrad auch einen niedrigeren Fehler erzeugt. Der relative Fehler sinkt allerdings bereits ab Grad 5 nur noch sehr langsam, wohingegen die Berechnungszeit für jeden höheren Grad etwa verdoppelt wird. Insgesamt bleibt festzuhalten, dass ein Ansatzgrad von maximal 5 bereits ausreicht, um gute Ergebnisse zu erzielen.

Im Vorhersagebereich ist ein derartiger Trend nicht zu erkennen. Die besten Ergebnisse erzielen Ansatzpolynome vom Grad 2 und 3. An dieser Stelle könnte die zu Grunde liegende Physik eine Rolle spielen. Die prinzipienfreie Modellierung versucht, eine mathematische Beschreibung des Systems zu finden, ohne die Physik direkt zu

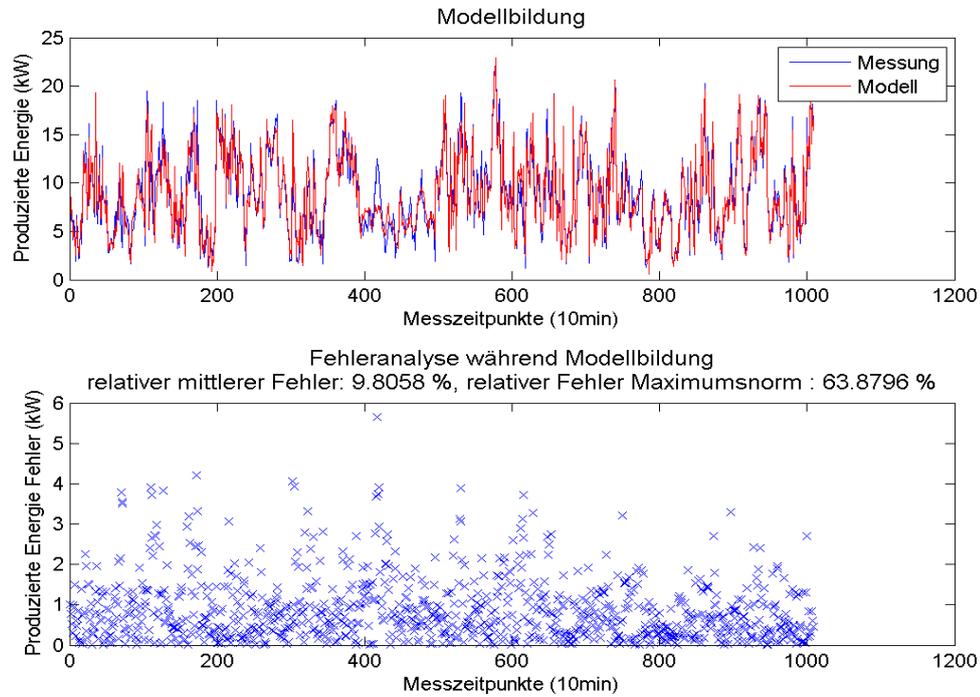


Abbildung 4.1: Modellbildung mit Polynomen vom Grad 5

Grad	Bildung rel. Feh.	Bildung rel. max. Feh.	Zukunft rel. Feh.	Zukunft rel. max. Feh.	Größe C	Ber. C [sec]
1	13,6099	77,6651	13,3492	62,3959	1008×6	0,003292
2	12,6859	77,4238	13,3197	60,7939	1008×21	0,007063
3	11,4944	78,3561	12,8494	68,4048	1008×56	0,020421
4	10,2655	70,0705	15,1252	60,5169	1008×126	0,049919
5	9,8058	63,8796	12,8446	78,7859	1008×252	0,113193
6	9,5594	61,7680	17,3372	96,7008	1008×462	0,229475
7	9,4159	57,2449	14,9510	80,9676	1008×792	0,429192

Tabelle 4.1: Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im statischen Modell

berücksichtigen. Auf diese Weise entstehen aber Fehler, wenn der angesetzte Grad so gewählt ist, dass die korrekte mathematische Formel des Problemes nicht vernünftig dargestellt werden kann.

Tabelle 4.1 ist zu entnehmen, dass die Vorhersage mit Hilfe von Polynomen des Ansatzgrades 3 in Bezug auf den relativen Fehler recht gut ist. Es gibt aber Messzeitpunkte, wie zum Beispiel der in Abbildung 4.2 abgedruckte Bereich 150-200, an

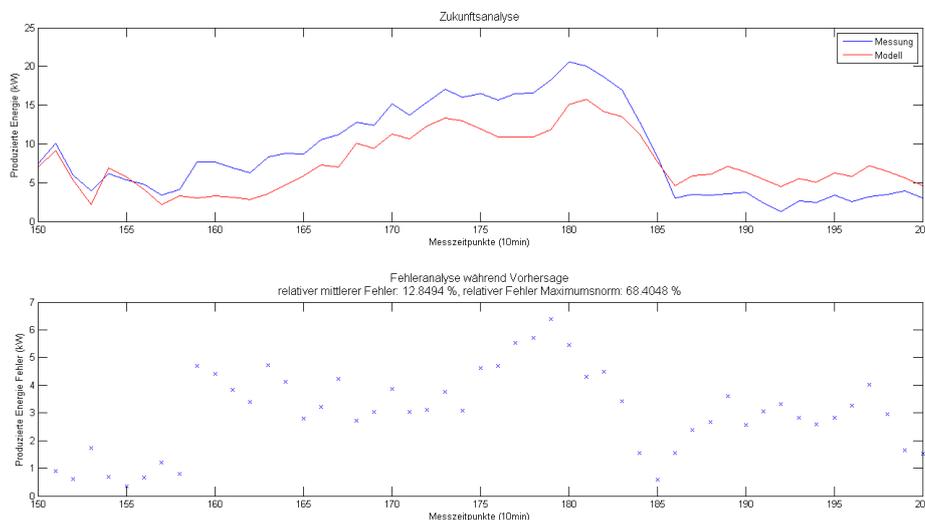


Abbildung 4.2: Ausschnitt der Vorhersage mit Polynomen vom Grad 3

denen das Modell zwar die Tendenzen der Kurve widerspiegelt, aber die tatsächlichen Werte nicht erreicht. Auf diese Weise entstehen im Vorhersagebereich teilweise recht große maximale Fehler.

4.1.2 Variation der Eingangsdaten

Generell ist die Annahme bei der Anwendung der datenbasierten Modellierung, dass mehr Daten auch insgesamt bessere Ergebnisse produzieren. Es stellt sich aber die Frage, welche der verwendeten Größen den stärksten Einfluss auf das Modell hat. Ist bekannt, welche Messwerte das Modell und somit vermutlich auch das System am stärksten beeinflussen, können Schwerpunkte bei der einzusetzenden Messtechnik gesetzt werden um die Modellierung zu unterstützen und zu verbessern. Um diesen Umstand entsprechend zu berücksichtigen, wurden die vorherigen Tests mit Ansatzgrad 2 wiederholt und jede der vorhandenen Messgrößen wurde einmal entfernt. Tabelle 4.2 stellt die resultierenden Fehlerveränderungen während der Modellierung dar. Es ist zu erwarten, dass Windgeschwindigkeit und Luftdruck für die Modellierung am wichtigsten sind. Der Test zeigt, dass beim statischen Modell

Windgeschwindigkeit	Temperatur	Luftdruck	Windrichtung	Blattneigung
24,3412	0,1440	0,3581	0,4372	0,1061

Tabelle 4.2: Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im statischen Modell. Änderung in Prozentpunkten.

wie erwartet die Windgeschwindigkeit der wichtigste Faktor ist. Überraschend ist die geringe Auswirkung des Luftdrucks auf das Ergebnis. Der Einfluss der Blattneigung lässt sich anhand dieses Testes nicht beurteilen, da während des betrachteten Zeitraumes die Blattneigung nahezu unverändert bleibt.

Bei der Vorhersage weiterer Zeiträume fällt auf, dass es immer wieder Fehlerspitzen gibt, die sich über bestimmte Zeiträume entwickeln. Es scheint in diesen Bereichen Probleme in den Messdaten zu geben.

Abbildung 4.3 zeigt die Vorhersage mit Hilfe des statischen Modells, wobei der Zeitraum 1-1000 der Modellbildung diente und der Zeitraum 1001-2000 vorhergesagt wurde. Die angesprochenen Fehlerspitzen finden sich auf dem Bild in der Nähe der Messzeitpunkte 450 und 880 im Vorhersagebereich. Die Grafiken der Messdaten im Anhang D zeigen, dass die Windrichtung im Bereich 1450, also zum Vorhersagezeitpunkt 450 knapp über 45° liegt. Wie in 3.2 erläutert, gibt es Probleme bei einer Windrichtung unter 45° . Dies deutet darauf hin, dass auch im Grenzbereich der Sensor möglicherweise keine korrekten Werte liefert. Um dies zu überprüfen, wird nun derselbe Versuch wie in Abbildung 4.3 wiederholt, aber die Windrichtung wird dieses Mal nicht berücksichtigt.

In Abbildung 4.4 liegt das Modell im Bereich 440 sehr viel näher an der Messkurve. Der relative Fehler wurde ebenfalls um fast 4 Prozentpunkte gesenkt. Durch einen Ausreißer im Bereich 880 ist der maximale Fehler zwar gestiegen, aber insgesamt

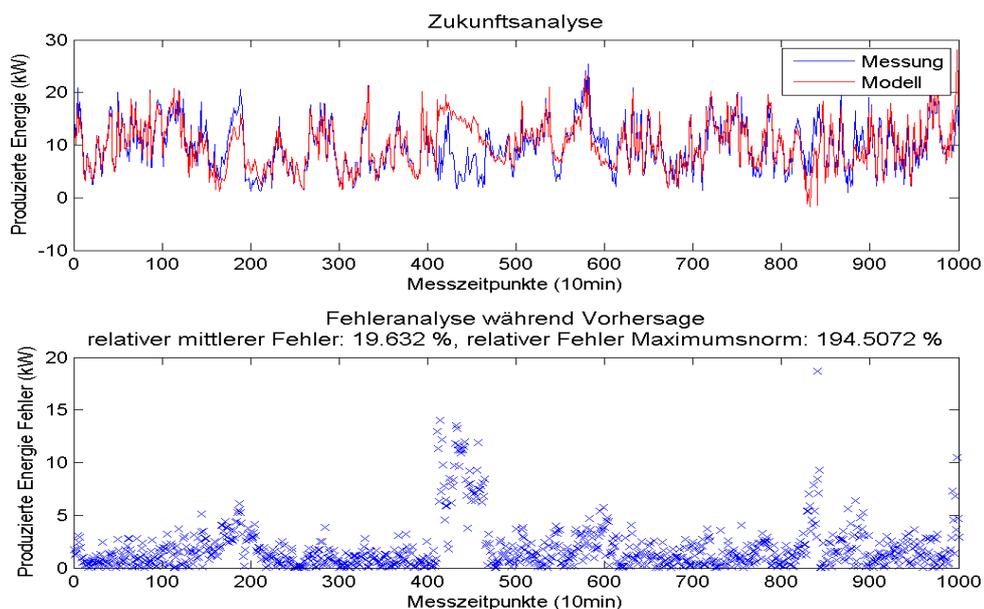


Abbildung 4.3: Vorhersage mit Polynomen vom Grad 3

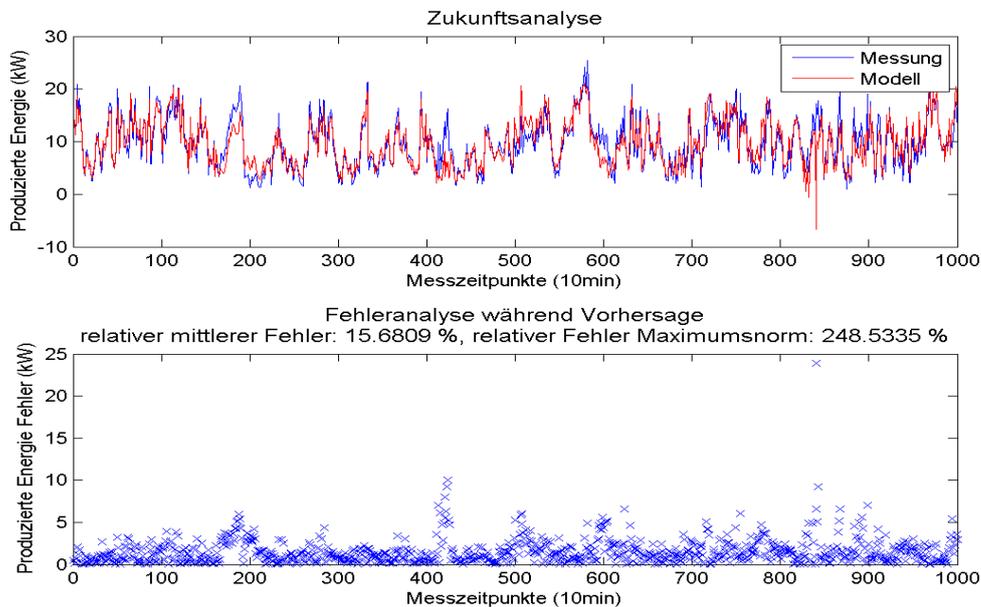


Abbildung 4.4: Vorhersage mit Polynomen vom Grad 3 ohne Windrichtung

ist das Ergebnis deutlich schöner. Dieser Test zeigt, dass es in Einzelfällen durchaus sinnvoll sein kann bestimmte Daten zu ignorieren.

4.1.3 Wahl des Modellbildungszeitraumes

In Kapitel 2.2.1.2 wurde bereits angedeutet, dass Änderungen im System dafür sorgen, dass der alte Parametervektor nicht mehr richtig ist und das Modell ab diesem Zeitpunkt falsche Werte liefert. Hierzu sollen an dieser Stelle zwei Beispiele angeführt werden. Zunächst wird derselbe Fall wie eben ein wenig erweitert. Es werden Polynome bis zum Grad 7 als Ansatzfunktionen verwendet. Der Modellbildungszeitraum umfasst die Zeitpunkte 1-4000 und es wird der Bereich 4001-5000 vorhergesagt.

Abbildung 4.5 zeigt deutlich, dass das Modell lediglich ungefähr für die ersten 40 Zeitpunkte eine brauchbare Vorhersage liefert. Danach haben Modell und Realität keinerlei Gemeinsamkeiten. Der Grund findet sich in den Daten. Der Blattneigungswinkel wird während der gesamten Modellphase quasi gar nicht verändert und etwa zum Zeitpunkt 4040 wird der Neigungswinkel von ungefähr 92° auf etwa $86,5^\circ$ gesenkt. Da das Modell in der Modellbildungsphase aber nicht lernen konnte, wie sich eine Änderung des Blattneigungswinkels auswirkt, sorgt diese Veränderung des Systems für den Zusammenbruch des Modells.

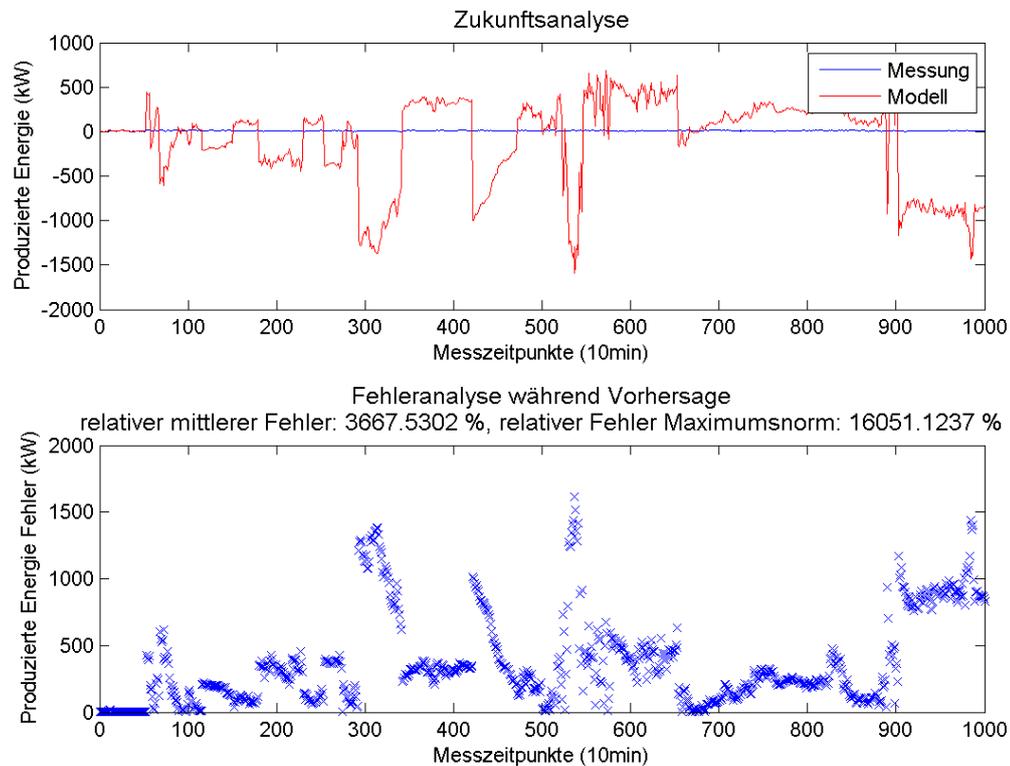


Abbildung 4.5: Vorhersage mit Polynomen vom Grad 7

Es wurden bereits mehrfach die unterschiedlichen Laststufen der Anlage erwähnt. An dieser Stelle soll kurz gezeigt werden, was passiert, wenn die Modellbildung stattfindet während die Anlage in Laststufe I arbeitet und dann versucht wird Voraussagen über die Laststufe II zu treffen. Es werden die ersten 3000 Zeitpunkte der Laststufe I zur Modellbildung verwendet und dann die ersten 500 Zeitpunkte der Laststufe II vorhergesagt mit Ansatzpolynomen vom Grad 5. Abbildung 4.6 zeigt die Ergebnisse.

Auch in diesem Fall ist das Modell nicht in der Lage, eine vernünftige Vorhersage zu treffen. Es ist demnach wichtig die vorhandenen Daten einschätzen zu können und in solchen Fällen eine neue Modellbildung durchzuführen.

4.1.4 Optimierung des Blattneigungswinkels

Nachdem in 4.1.3 einige Problemfälle vorgestellt wurden, soll abschließend gezeigt werden, wie beispielhaft mit dem statischen Modell eine Optimierung der Parameter durchgeführt wird. Das Modell lernt in diesem Fall über 1000 Messpunkte, also etwa

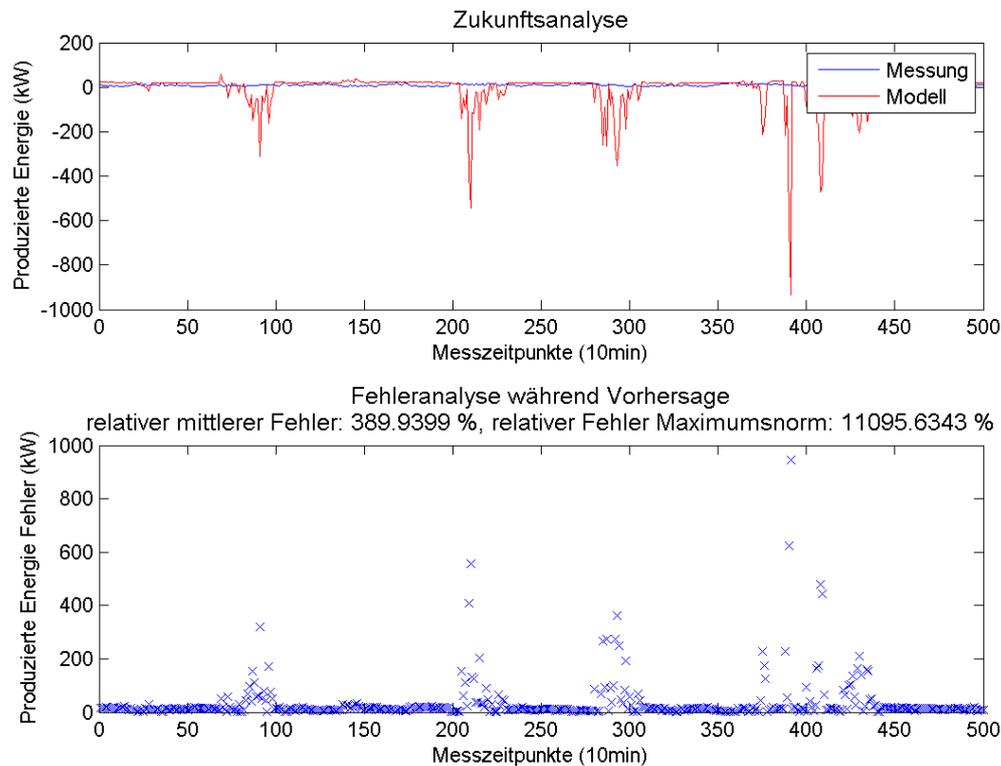


Abbildung 4.6: Vorhersage mit Polynomen vom Grad 5

über eine Woche. Mit dem so ermittelten Parametervektor wird eine Vorhersage für 100 Messpunkte, etwa 17 Stunden gemacht.

Abbildung 4.7 zeigt, dass das Modell die Realität gut abbildet. Es soll untersucht werden, wie sich eine Variation des Blattneigungswinkels auf die produzierte Energie auswirkt. Hierzu wurden verschiedene Blattneigungswinkel über den Vorhersagezeitraum konstant angesetzt. Aus Abbildung 4.8 ist ersichtlich, dass durch die erfolgten Versuche eine theoretisch höhere Leistung für einen Blattneigungswinkel von 89° oder 90° ermittelt wurde. Problematisch für die Beurteilung dieser Lösung ist allerdings, dass während der Modellbildungsphase lediglich die Blattneigungen $86,5^\circ$ und 92° tatsächlich aufgetreten sind. Ein weiteres Problem zeigte sich bei weiteren Testläufen. Wird der verwendete Ansatzgrad der Polynome variiert, so verändert sich das Ergebnis. Damit der Versuch an Aussagekraft gewinnen kann, müssten im Praxisfall auch Testläufe mit den anderen Blattneigungen gemacht werden. Dies würde vermutlich auch für Stabilität bei einer Variation der Ansatzgrade sorgen. Idealerweise würde in der Praxis natürlich ein dauerhafter Betrieb eines automatisierten Optimierungsverfahrens eingesetzt werden, um während der Laufzeit direkt die Blattneigung zu steuern und somit die Leistung der Anlage zu optimieren.

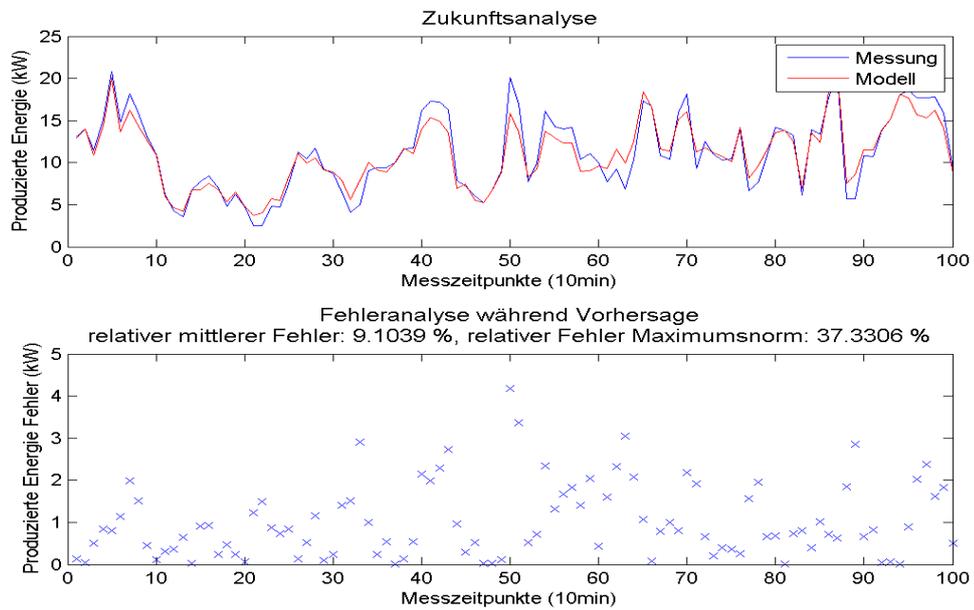


Abbildung 4.7: Vorhersage mit Polynomen vom Grad 2

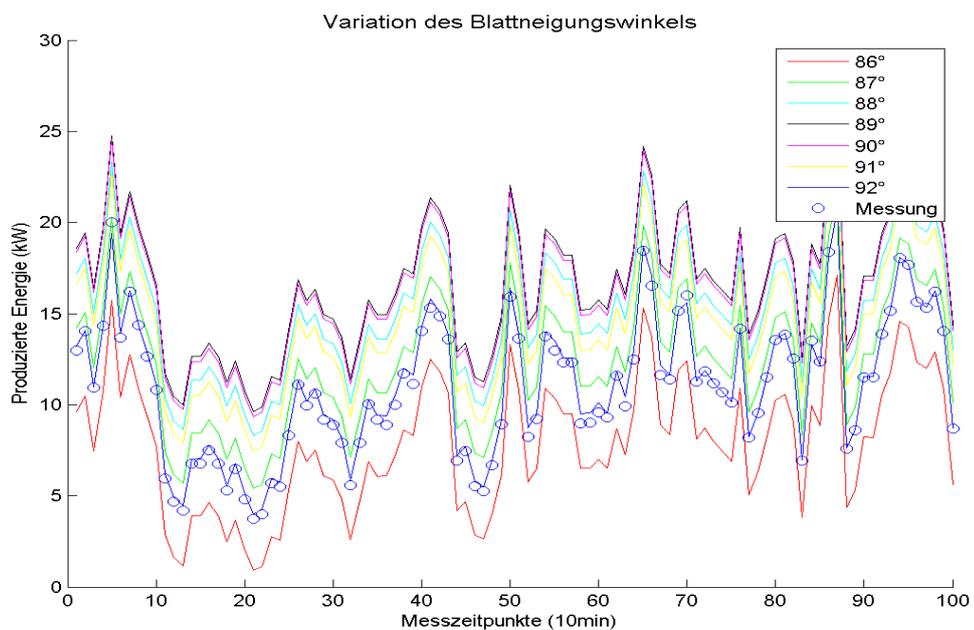


Abbildung 4.8: Vorhersage mit Polynomen vom Grad 2 bei unterschiedlichen Blattneigungen

4.2 Das dynamische Modell

Nach den Tests zum statischen Modell sollen ebenfalls einige Untersuchungen mit dem dynamischen Modell durchgeführt werden.

4.2.1 Variation des Ansatzgrades

Für die Erstellung der Tabelle 4.3 wurde der gleiche Zeitraum wie für Tabelle 4.1 gewählt. Wie beim statischen Modell nimmt auch hier der relative Fehler mit höherem Grad in der Modellbildungsphase ab. Auf den Abbildungen im Anhang B ist erkennbar, dass der maximale Fehler bei Grad 4 und 5 durch einen einzigen Ausreißer so groß ist. Die Vorhersage ist im Vergleich zur Lernphase etwas schlechter, aber beim Betrachten der dazugehörigen Grafik fällt auf, dass das Modell um etwa einen Messpunkt hinter der Realität liegt.

4.2.2 Variation der Eingangsdaten

Wie bereits in Kapitel 4.1.2 sollen in Tabelle 4.4 die Auswirkungen der einzelnen Messwerte auf das Modell bewertet werden. Es wurde unter denselben Rahmenbedingungen wie bei Tabelle 4.2 gearbeitet, Polynomgrad 2, Lernzeitraum wie bei den anderen Tabellen. Der Test zeigt, dass das dynamische Modell ebenfalls am stärksten auf die Windgeschwindigkeit reagiert. Es wird aber deutlich, dass die Auswirkungen der einzelnen Messwerte insgesamt näher beieinander liegen, als beim Test mit dem statischen Modell. Der Test lässt auf Grund seiner Beschaffenheit auch hier wie bereits bei Tabelle 4.2 erwähnt keine Aussagen über die Blattneigung zu, da diese nahezu konstant über den betrachteten Zeitraum ist.

Grad	Bildung rel. Feh.	Bildung rel. max. Feh.	Zukunft rel. Feh.	Zukunft rel. max. Feh.	Ber. x [sec]
1	26,5802	153,8817	27,0163	119,8025	21,321
2	23,9583	148,0858	26,8899	138,1975	26,283
3	20,5395	140,0853	30,8567	120,8254	36,220
4	18,3401	1468,0328	32,6753	252,5711	58,123
5	16,5268	1013,4262	41,1029	382,3257	83,354

Tabelle 4.3: Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im dynamischen Modell

Windgeschwindigkeit	Temperatur	Luftdruck	Windrichtung	Blattneigung
7,2988	1,1263	0,9633	0,7045	0,475

Tabelle 4.4: Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im dynamischen Modell. Änderung in Prozentpunkten.

4.3 Variation des dynamischen Modelles

In Kapitel 2.2.3 wurde das zu Grunde liegende Minimierungsproblem dargestellt. Zur Berechnung der gesuchten Größe v_k werden v_{k-1} und u_{k-1} verwendet. Das bedeutet aber auch, dass zum Beispiel die Windgeschwindigkeit (hier u_{k-1}^1) vor 10 Minuten benutzt wird um die Leistung der Anlage jetzt zu berechnen. Dies stellt auch eine mögliche Erklärung der deutlich geringeren Auswirkung der Windgeschwindigkeit auf das Modell gemäß Tabelle 4.4 dar. Beim statischen Modell wurde die Windgeschwindigkeit zum selben Zeitpunkt verwendet, dies erscheint sinnvoller. Es liegt nahe, eine Abwandlung des dynamischen Modelles zu testen. Hierfür wird die in (2.7) dargestellte Matrix D durch folgende ersetzt:

$$\begin{pmatrix} v_0 & u_1 & 1 \\ \vdots & \vdots & \vdots \\ v_{M-1} & u_M & 1 \end{pmatrix}$$

So entsteht eine Kombination aus dynamischem und statischem Modell.

4.3.1 Variation des Ansatzgrades

Die Testläufe wurden mit diesem variierten Modell wiederholt. Um die Vergleichbarkeit zu gewährleisten, wurde wieder derselbe Zeitraum wie bei den anderen Tabellen gewählt. Tabelle 4.5 demonstriert die gute Qualität des Ergebnisses. Problematisch

Grad	Bildung rel. Feh.	Bildung rel. max. Feh.	Zukunft rel. Feh.	Zukunft rel. max. Feh.	Ber. x [sec]
1	13,4216	106,8417	13,3064	62,317	22,623
2	11,7196	136,1417	13,2643	63,8642	23,932
3	9,2564	78,368	15,007	61,8463	33,273
4	8,0843	451,2727	14,3623	60,7708	49,775
5	6,8173	185,8837	14,7622	59,4688	86,602

Tabelle 4.5: Fehler und Berechnungszeiten bei unterschiedlichem Grad der Ansatzpolynome im variierten dynamischen Modell

an dieser Variation ist, dass nun zur Modellierung der gesuchten Größe an einem Zeitpunkt wieder die Zustandsgrößen am neuen Zeitpunkt bekannt sein müssen. Eine echte Vorhersage ist also nicht möglich, aber es ist mit dieser Variante ebenso wie mit dem statischen Modell möglich, Aussagen zu treffen, wie sich die Leistung der Anlage bei bestimmten äußeren Bedingungen verhalten wird.

4.3.2 Variation der Eingangsdaten

Zur Erstellung von Tabelle 4.6 wurden nun die Auswirkungen der angesetzten Daten auf das variierte dynamische Modell getestet. Die Ergebnisse sind sehr ähnlich zum statischen Modell. Es ergibt sich dieselbe Reihenfolge wie bei Tabelle 4.2 und die Größenordnung ist ebenfalls sehr ähnlich. Dies bestätigt die erwartete Ähnlichkeit des statischen und des variierten dynamischen Modelles.

4.4 Vergleich dynamisches Modell und statisches Modell

Abschließend bleibt die Frage, welche der drei Modellarten sich für die Modellierung am besten eignet. Beim Vergleich der Tabellen 4.1, 4.3 und 4.5 wird deutlich, dass die Fehler beim statischen Modell und beim variierten dynamischen Modell auf einem ähnlich niedrigen Niveau landen. Diese beiden Modellarten sind von der zu Grunde liegenden Idee auch nahezu identisch, beim variierten dynamischen Modell wird aber zusätzlich noch der alte modellierte Wert der gesuchten Größe berücksichtigt. Das statische Modell ist in der Berechnung zwar etwas schneller, aber in der Praxis bietet sich die Implementierung des variierten dynamischen Modelles an, da es bereits mit dem Updateverfahren ausgestattet ist und auf diese Weise direkt neue Messwerte miteinfließen lässt.

Windgeschwindigkeit	Temperatur	Luftdruck	Windrichtung	Blattneigung
18,8524	0,5746	0,8005	0,9987	0,4212

Tabelle 4.6: Auswirkungen der einzelnen Messgrößen auf den relativen Fehler im variierten dynamischen Modell. Änderung in Prozentpunkten.

Kapitel 5

Zusammenfassung und Ausblick

Die Zielsetzung dieser Arbeit bestand darin herauszufinden, ob es möglich ist, mit Hilfe der prinzipienfreien Modellierung die Leistung einer Windenergieanlage darzustellen. Nachdem die Theorie dargestellt wurde, wurden einige numerische Untersuchungen durchgeführt. Es hat sich gezeigt, dass es definitiv möglich ist, die Leistung der Anlage zu modellieren. Teilweise waren die Fehlerspannen zwar recht groß, aber der allgemeine Trend oder die Höhen und Tiefen der modellierten Kurve wurden trotzdem gut erfasst. Beide Varianten der prinzipienfreien Modellierung sind bei der Modellierung sinnvolle Ansätze. Es wäre noch interessant gewesen zu überprüfen, wie sich weitere Daten auf das Modell ausgewirkt hätten. So wurde bereits erwähnt, dass zum Beispiel Niederschlagsdaten Einfluss auf das Modell gehabt hätten und ihre Verwendung vermutlich das Ergebnis beeinflusst hätten. Weiterhin wäre es mit Sicherheit für das dynamische Modell von Vorteil gewesen, wenn die Daten vorher nicht so stark gefiltert worden wären, da das Differenzenmodell auf kontinuierlichen Daten basiert. Trotzdem sind die erzielten Ergebnisse sehr zufriedenstellend und legen deutlich dar, dass diese Art der Modellierung bei Windenergieanlagen erfolgreich sein kann.

In weiterführenden Arbeiten könnte versucht werden, eine Möglichkeit zu finden, die Anlage zu optimieren. Hierzu wäre es vermutlich sinnvoll, Daten zu benutzen, bei denen weitere Größen vorhanden sind, auf die der Anwender Einfluss nehmen kann. Der in Kapitel 4.1.4 gezeigte Praxisfall sollte ein kleiner Schritt in diese Richtung sein. Weiterhin sollte versucht werden, noch andere Anlagen, bei denen vermutlich auch andere Größen gemessen werden, zu modellieren. Da diese Modellierungsart datenbasiert ist, hängt der Erfolg der Modellierung von der Datenqualität ab, deshalb wäre es interessant zu sehen, wie sich andere Größen auf die Qualität und die Aussagefähigkeit des Modelles auswirken.

Anhang A

Anhänge zu Tabelle 4.1

Die Abbildungen A.1 bis A.14 zeigen die Auswertungen, die zur Erstellung der Tabelle 4.1 benutzt wurden.

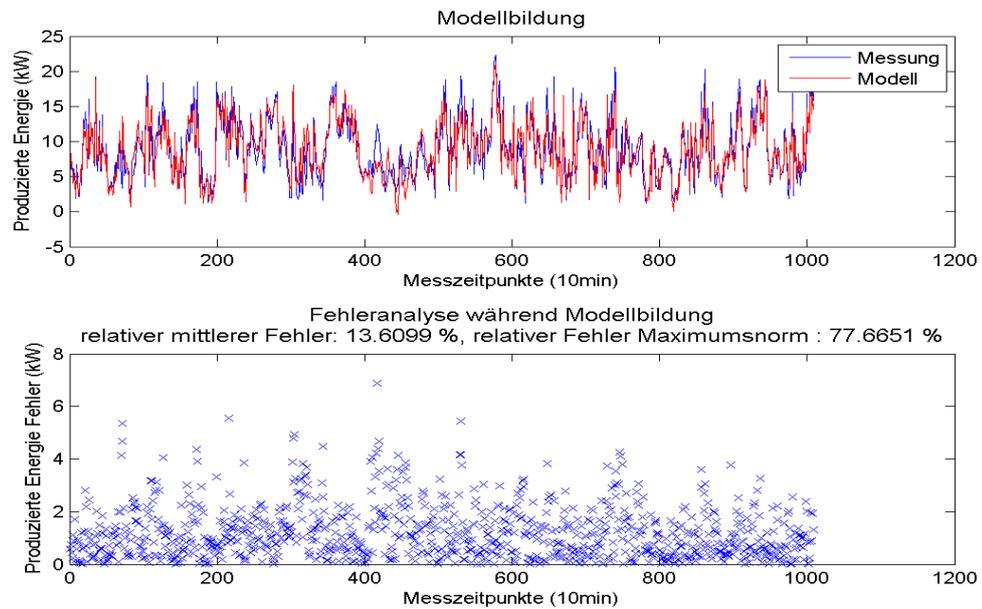


Abbildung A.1: Modellbildung mit Polynomen vom Grad 1

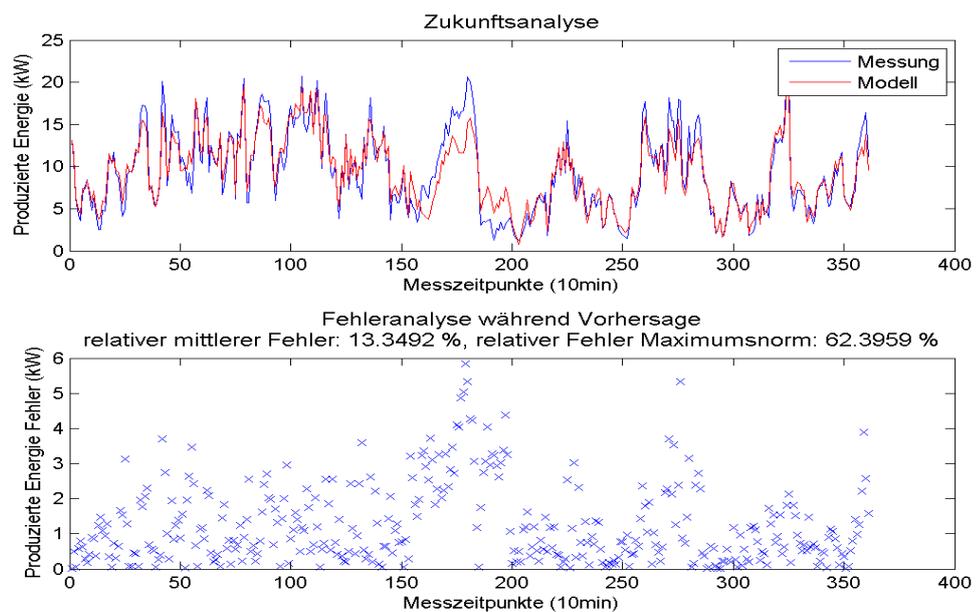


Abbildung A.2: Vorhersage mit Polynomen vom Grad 1

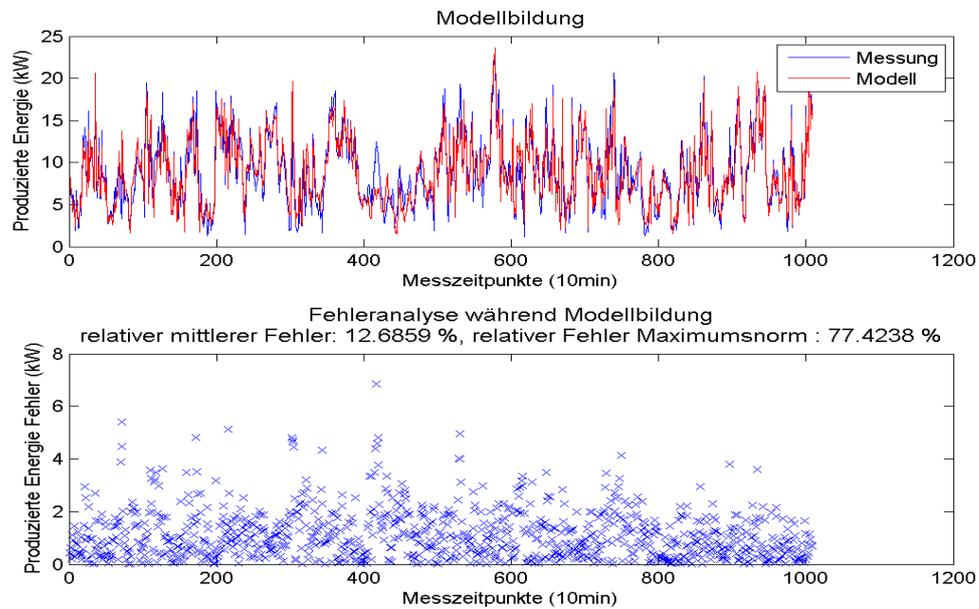


Abbildung A.3: Modellbildung mit Polynomen vom Grad 2

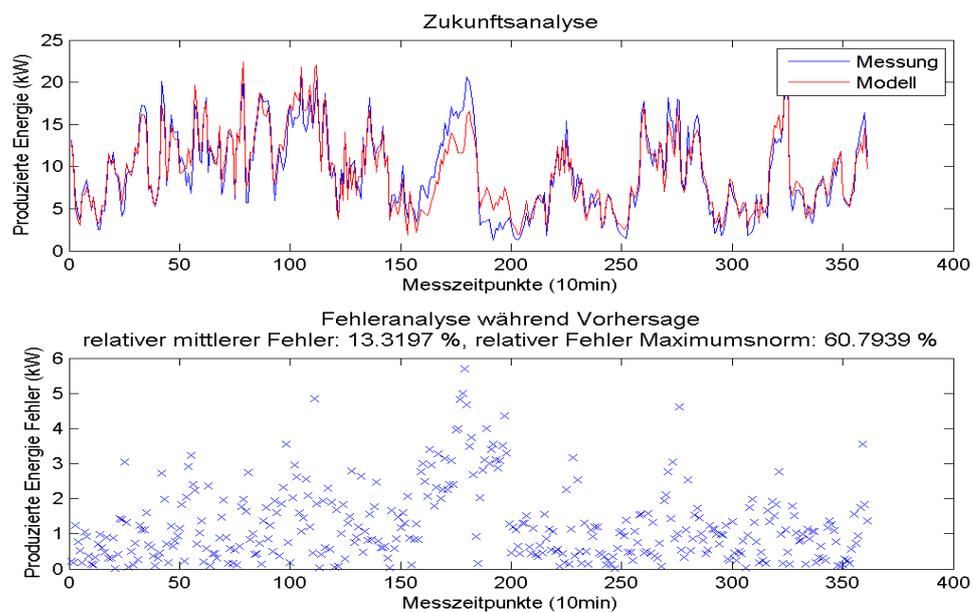


Abbildung A.4: Vorhersage mit Polynomen vom Grad 2

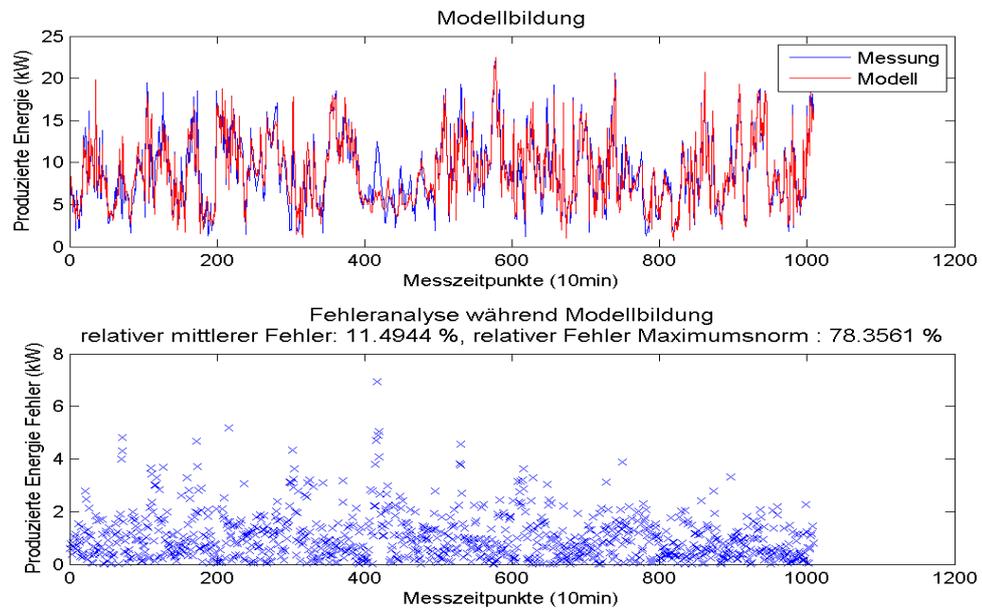


Abbildung A.5: Modellbildung mit Polynomen vom Grad 3

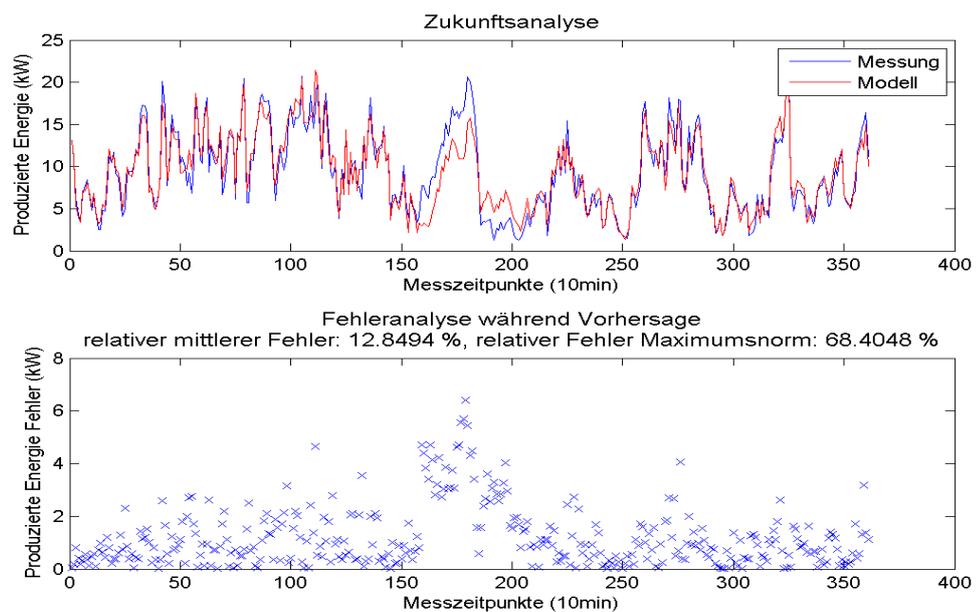


Abbildung A.6: Vorhersage mit Polynomen vom Grad 3

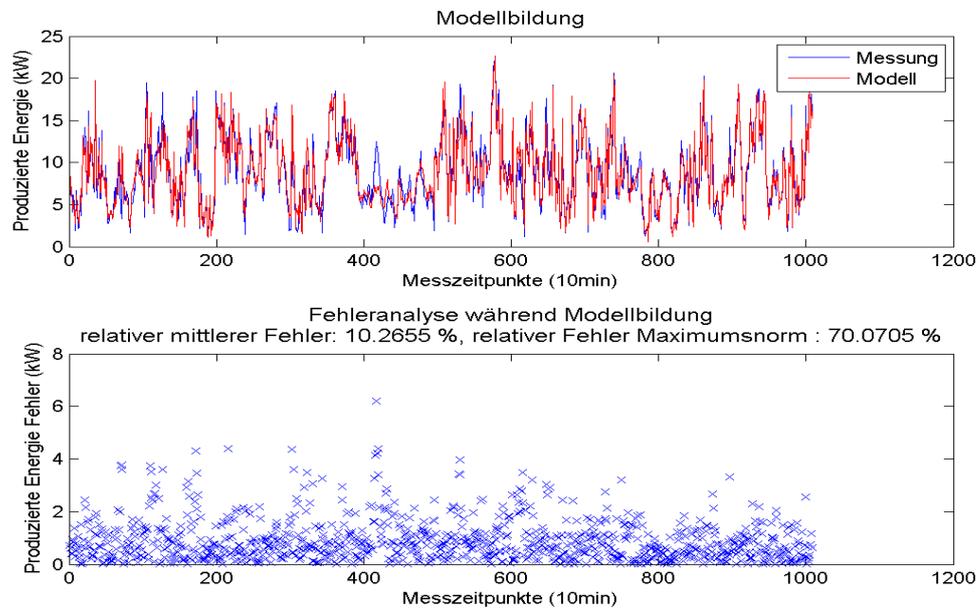


Abbildung A.7: Modellbildung mit Polynomen vom Grad 4

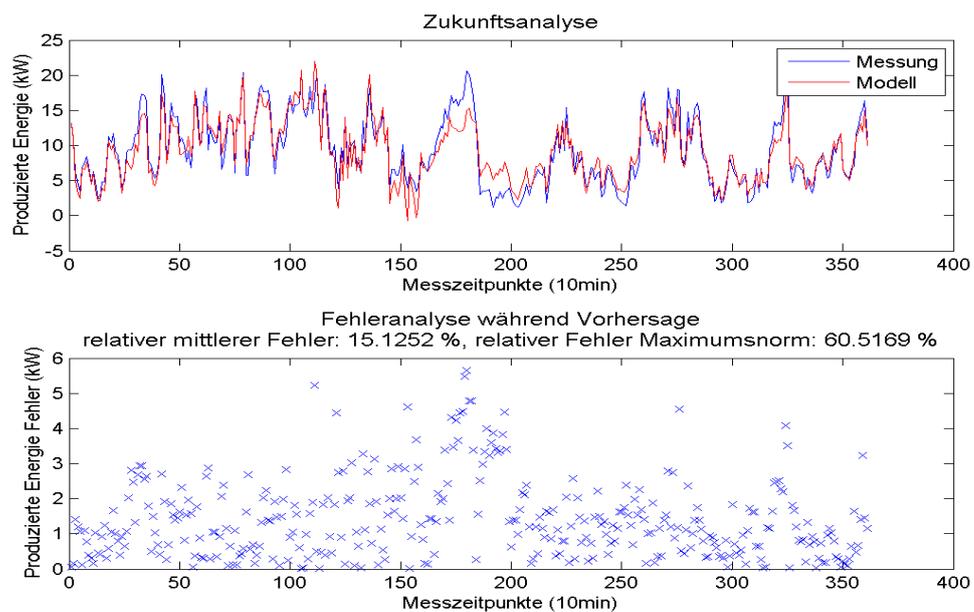


Abbildung A.8: Vorhersage mit Polynomen vom Grad 4

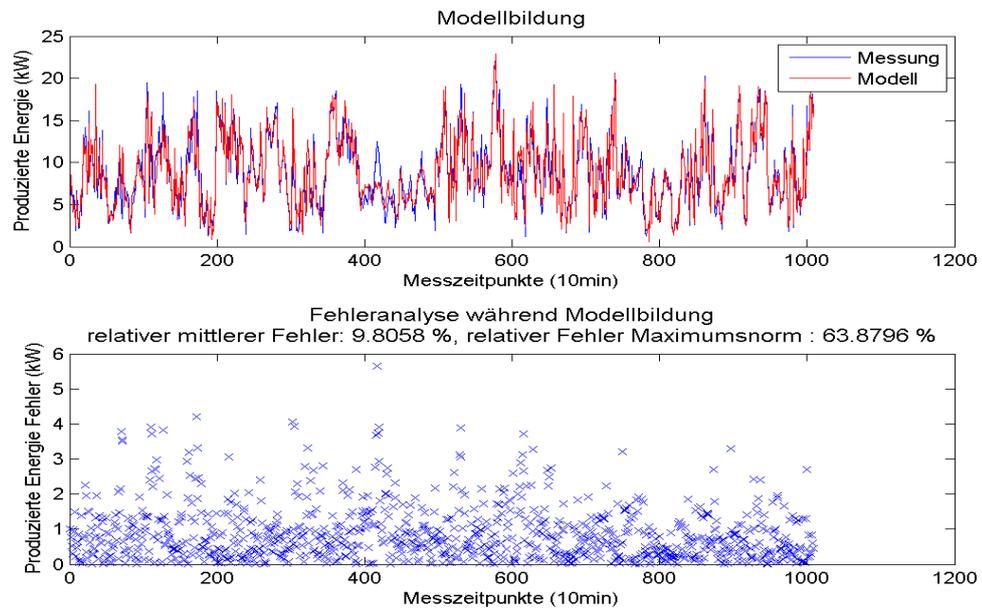


Abbildung A.9: Modellbildung mit Polynomen vom Grad 5

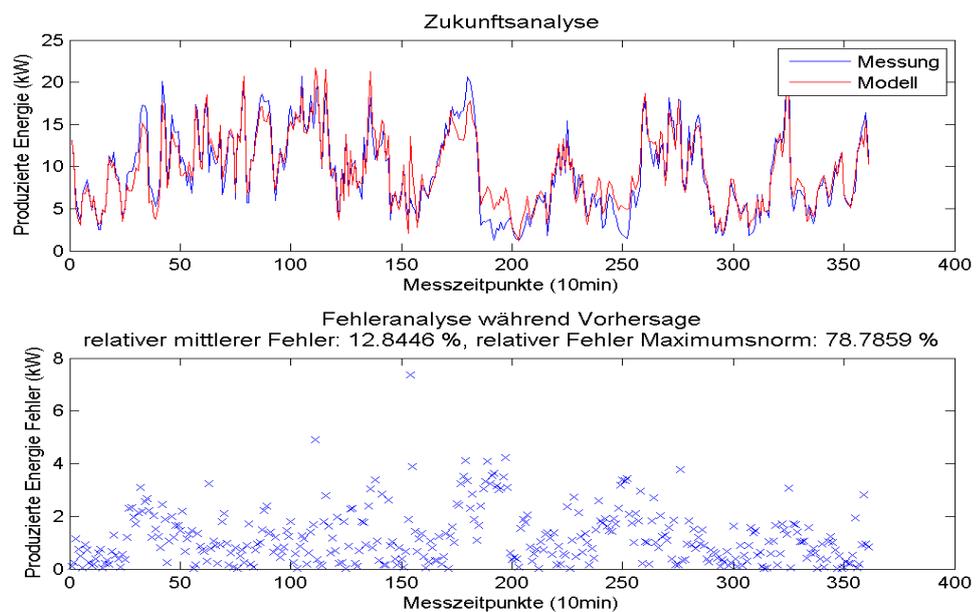


Abbildung A.10: Vorhersage mit Polynomen vom Grad 5

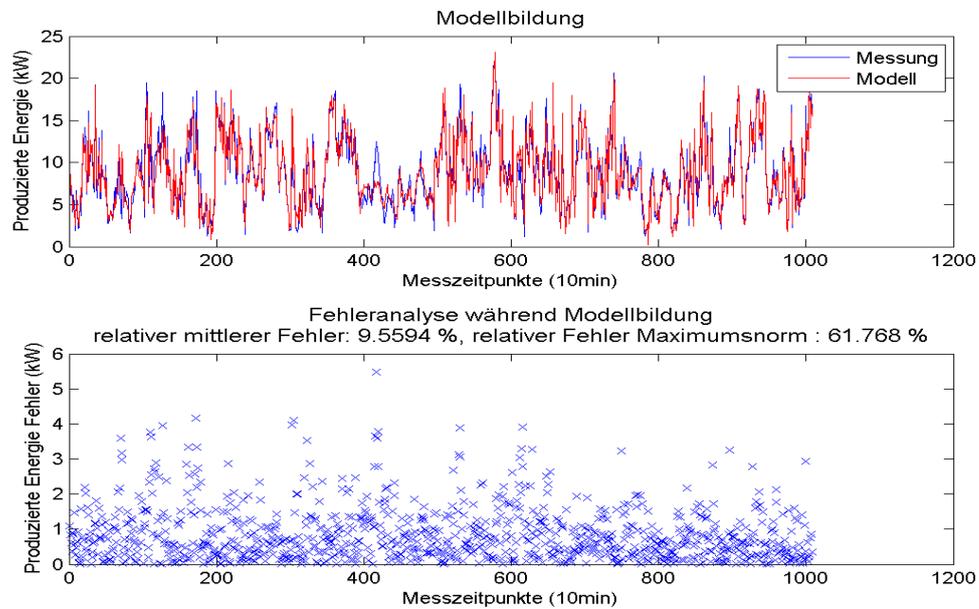


Abbildung A.11: Modellbildung mit Polynomen vom Grad 6

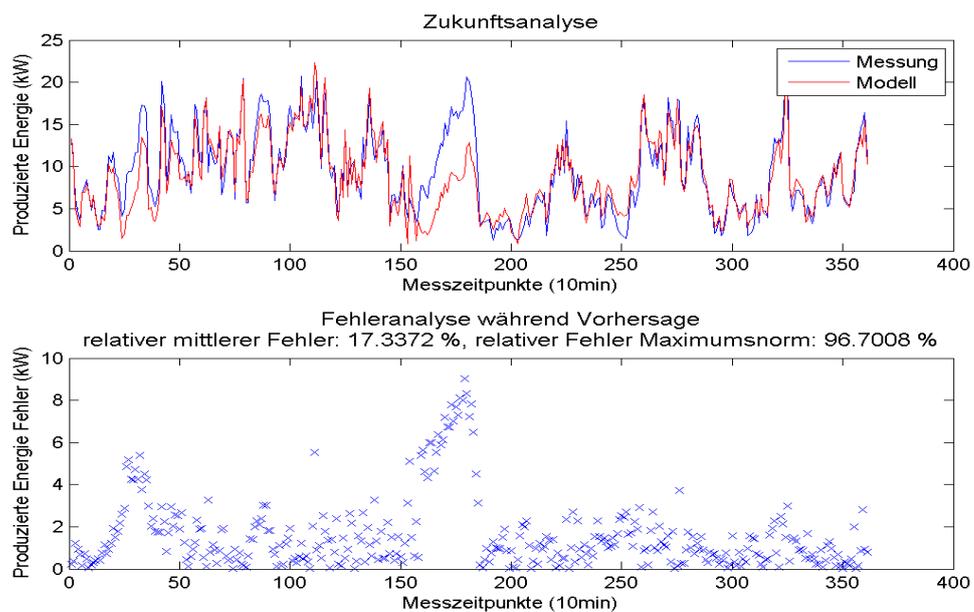


Abbildung A.12: Vorhersage mit Polynomen vom Grad 6

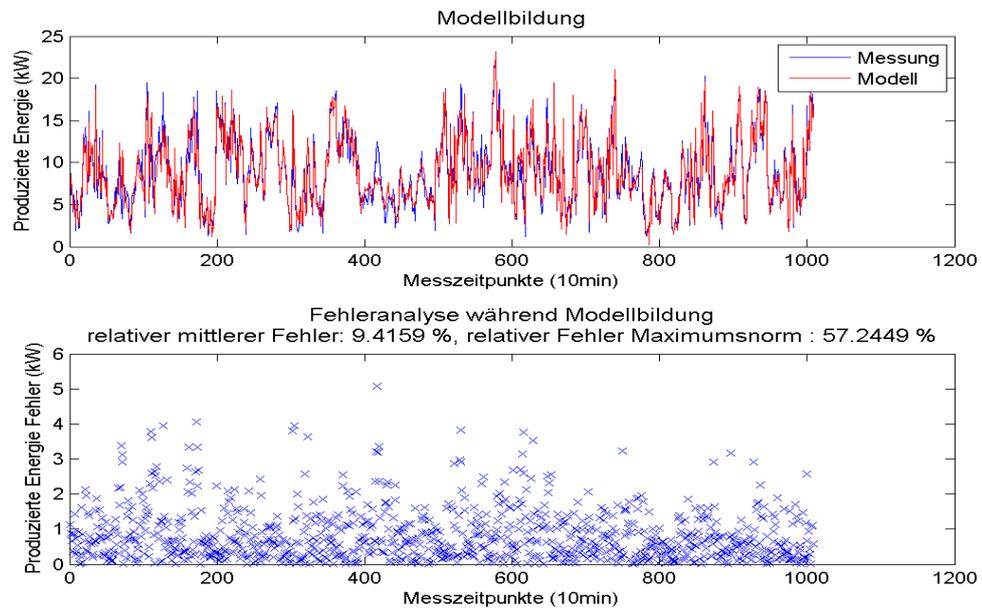


Abbildung A.13: Modellbildung mit Polynomen vom Grad 7

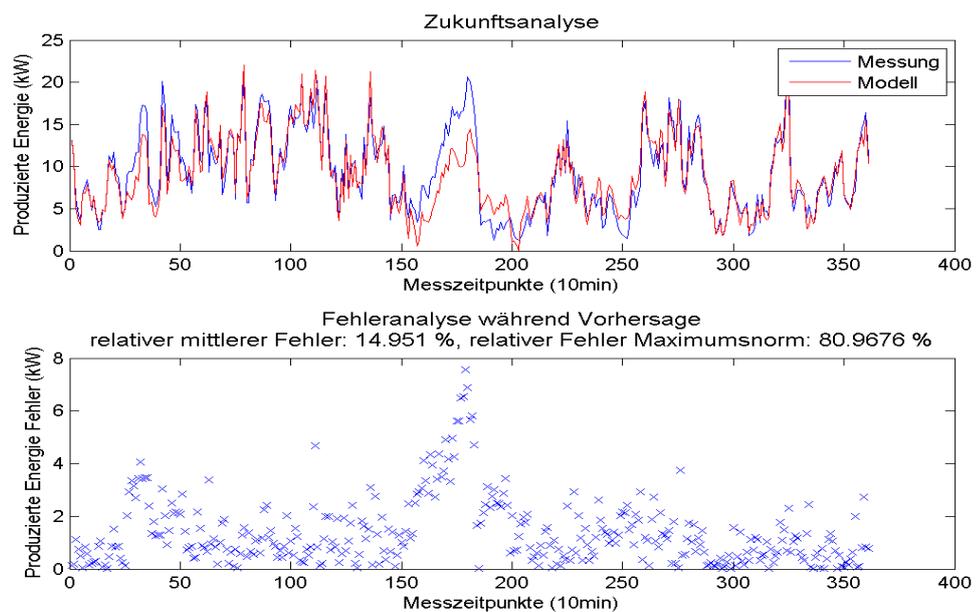


Abbildung A.14: Vorhersage mit Polynomen vom Grad 7

Anhang B

Anhänge zu Tabelle 4.3

Die Abbildungen B.1 bis B.10 zeigen die Auswertungen, die zur Erstellung der Tabelle 4.3 benutzt wurden.

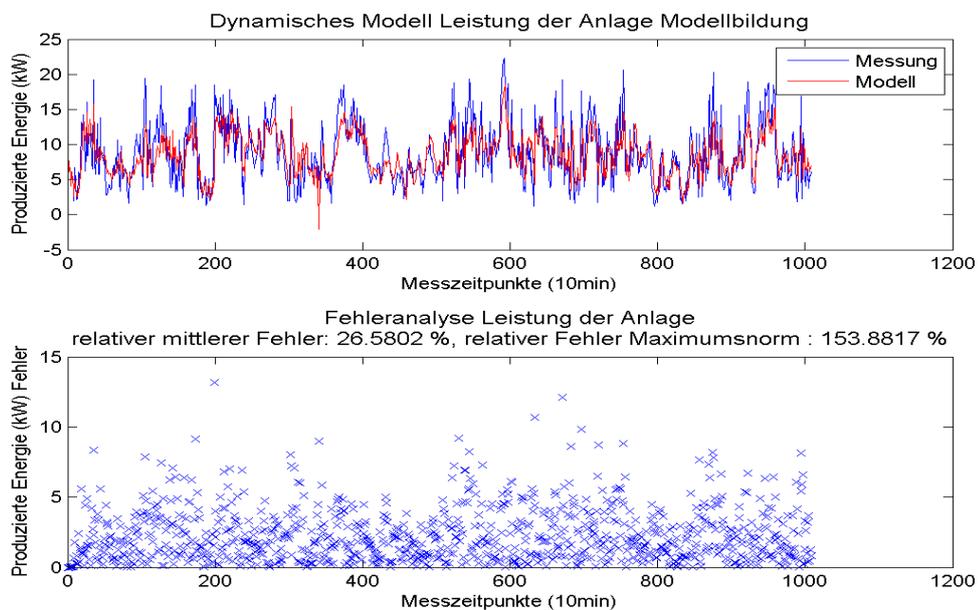


Abbildung B.1: Modellbildung mit Polynomen vom Grad 1

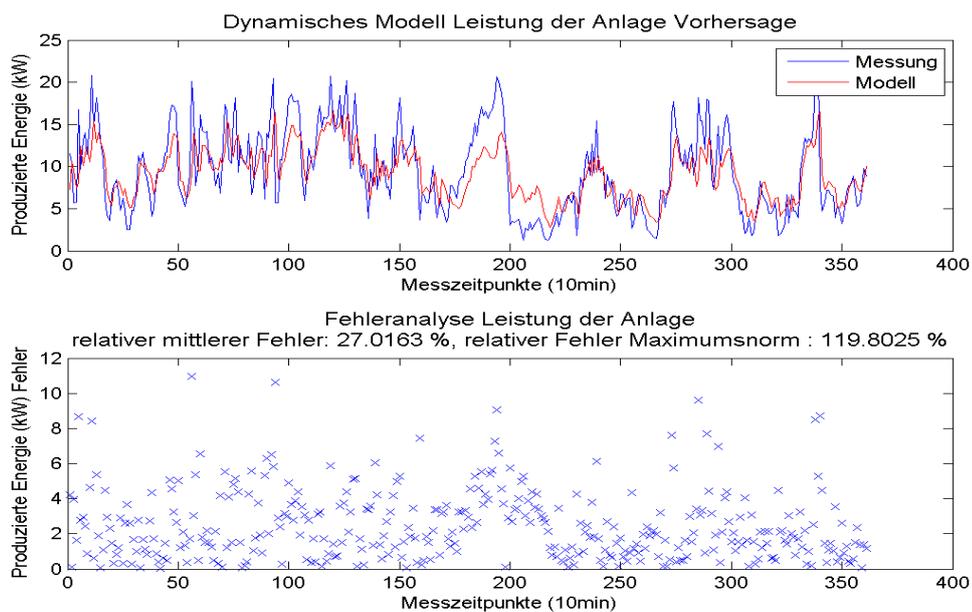


Abbildung B.2: Vorhersage mit Polynomen vom Grad 1

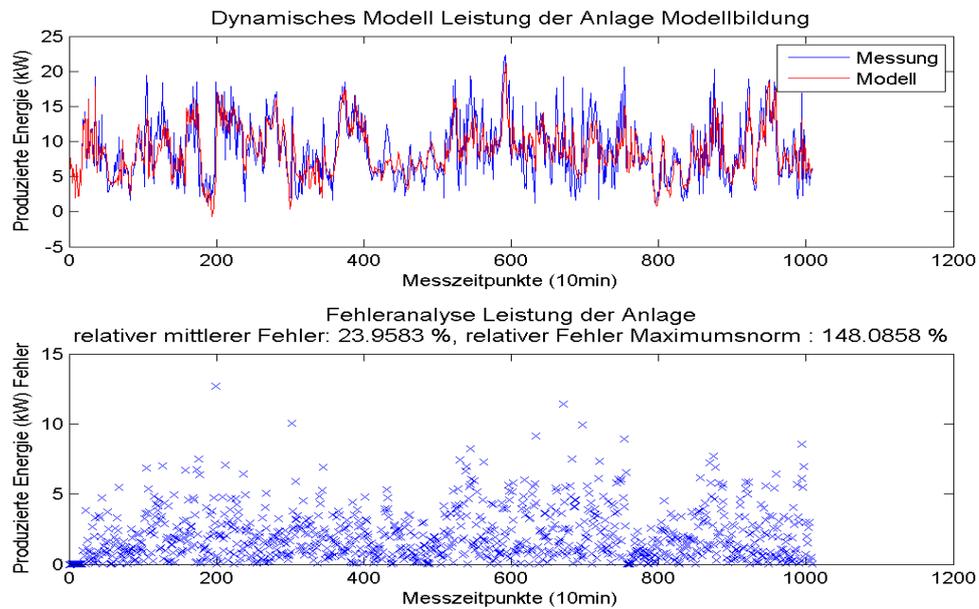


Abbildung B.3: Modellbildung mit Polynomen vom Grad 2

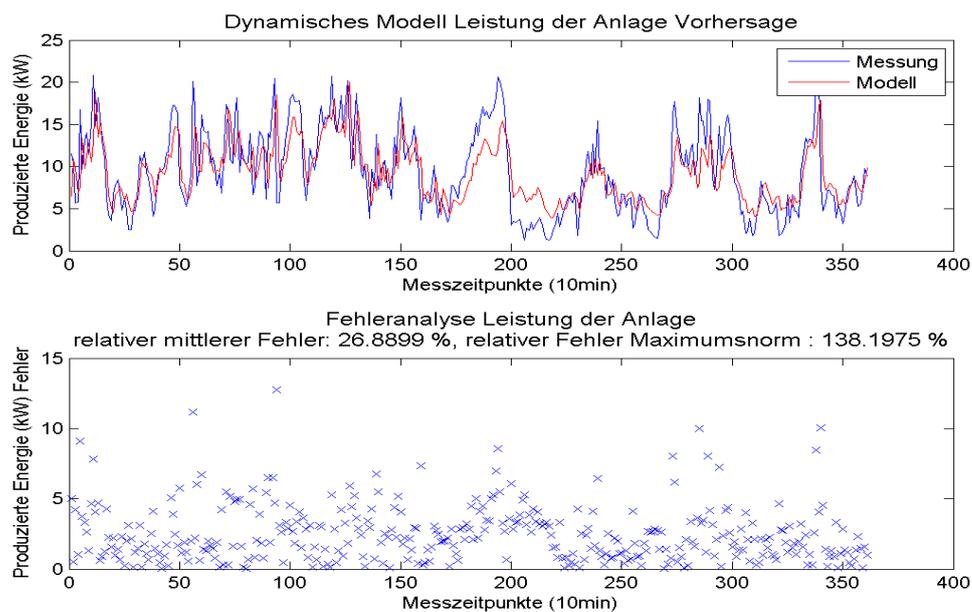


Abbildung B.4: Vorhersage mit Polynomen vom Grad 2

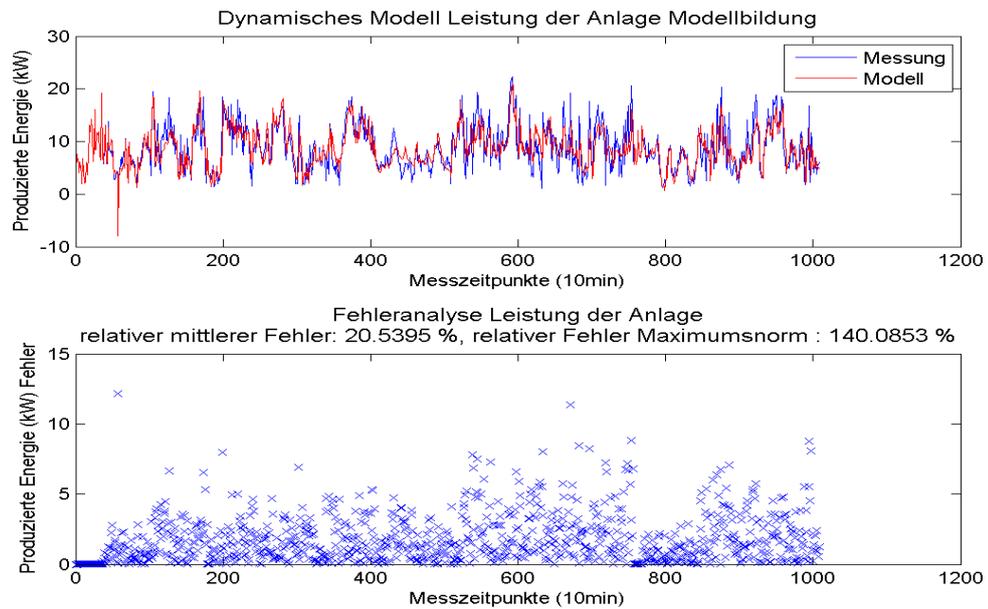


Abbildung B.5: Modellbildung mit Polynomen vom Grad 3

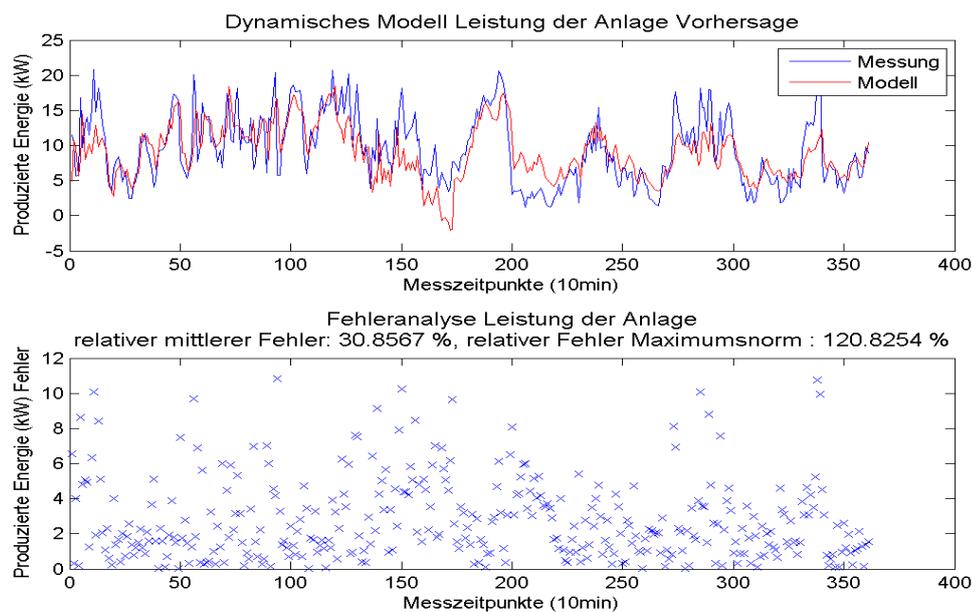


Abbildung B.6: Vorhersage mit Polynomen vom Grad 3

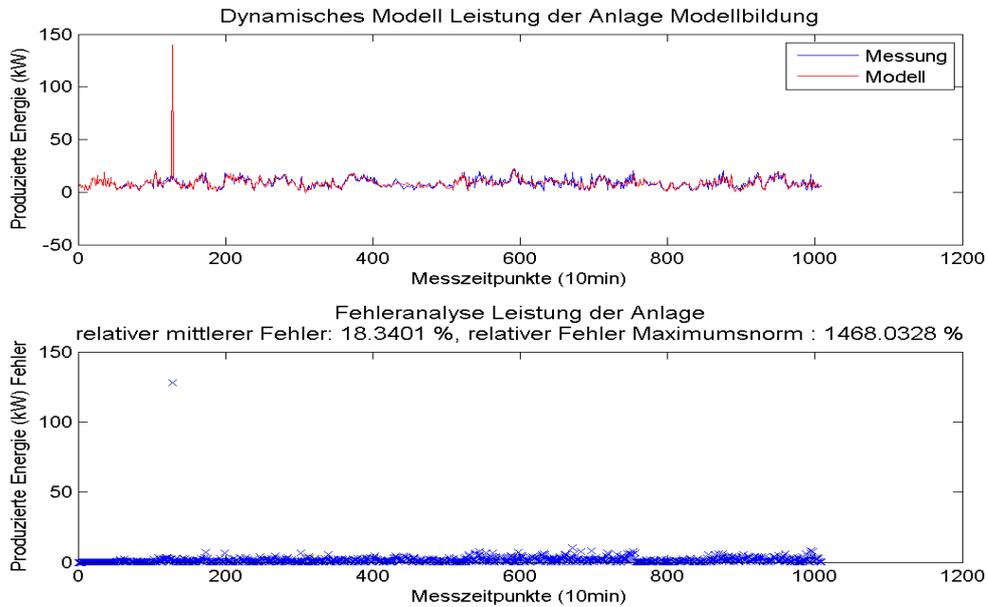


Abbildung B.7: Modellbildung mit Polynomen vom Grad 4

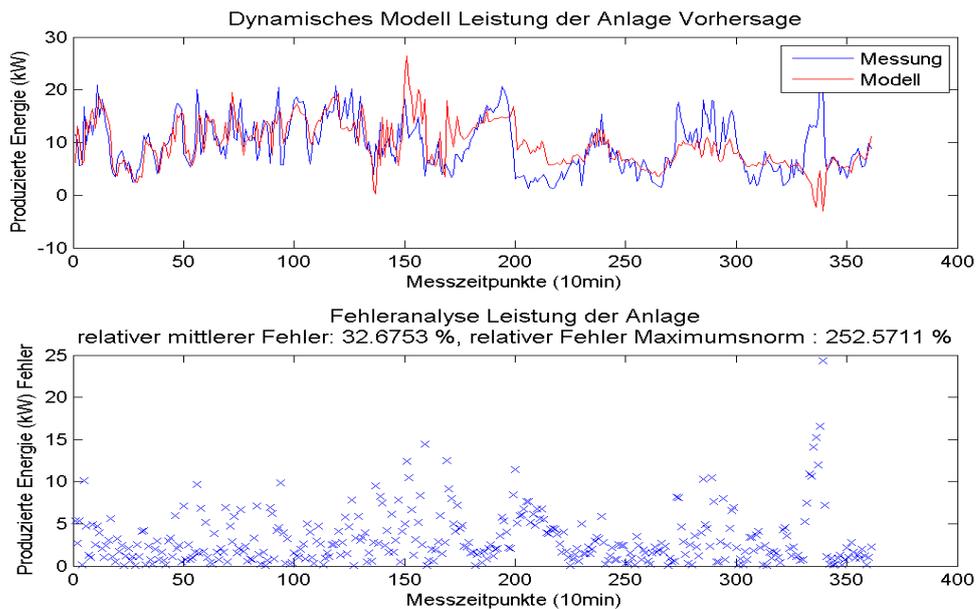


Abbildung B.8: Vorhersage mit Polynomen vom Grad 4

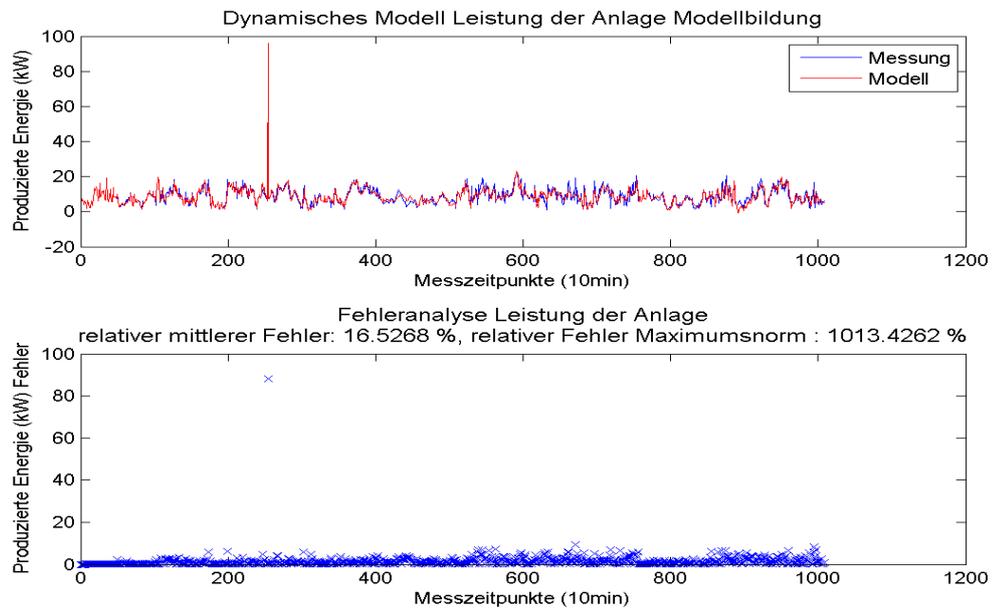


Abbildung B.9: Modellbildung mit Polynomen vom Grad 5

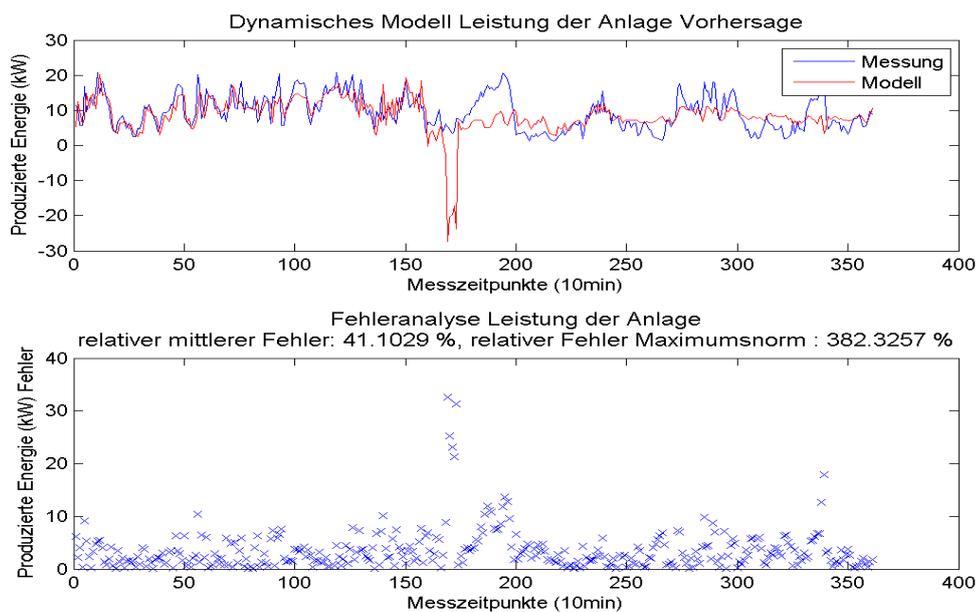


Abbildung B.10: Vorhersage mit Polynomen vom Grad 5

Anhang C

Anhänge zu Tabelle 4.5

Die Abbildungen C.1 bis C.10 zeigen die Auswertungen, die zur Erstellung der Tabelle 4.5 benutzt wurden.

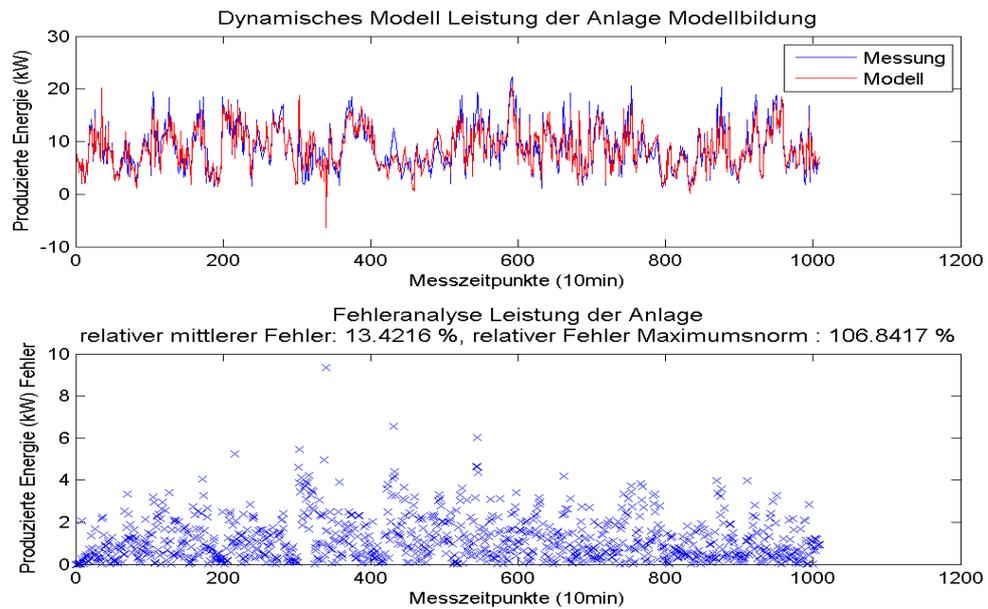


Abbildung C.1: Modellbildung mit Polynomen vom Grad 1

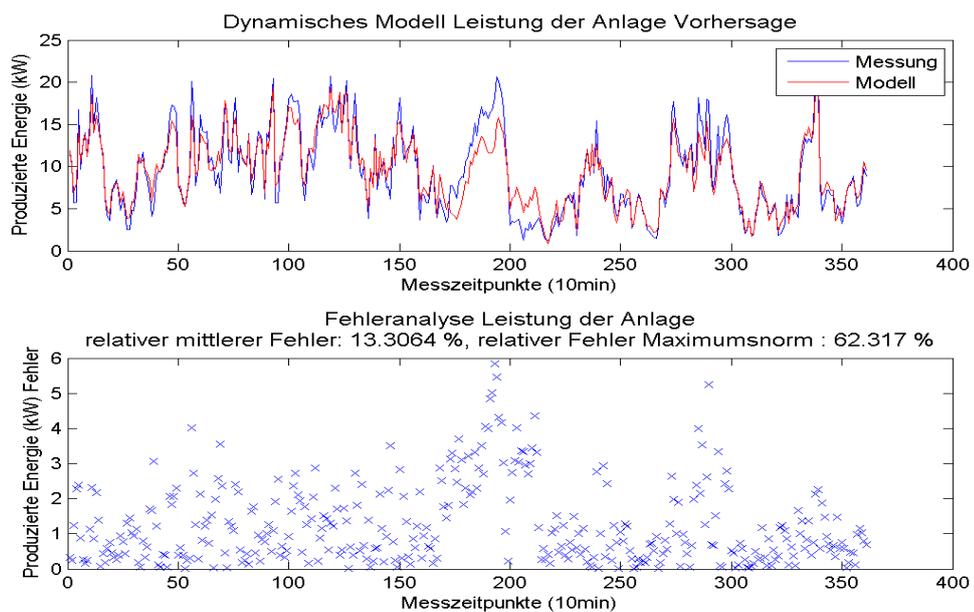


Abbildung C.2: Vorhersage mit Polynomen vom Grad 1

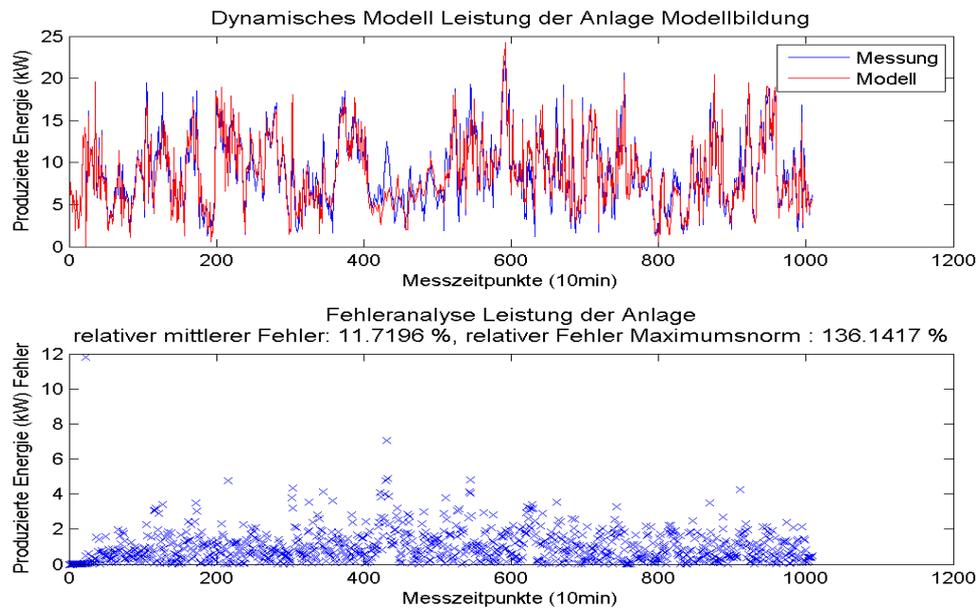


Abbildung C.3: Modellbildung mit Polynomen vom Grad 2

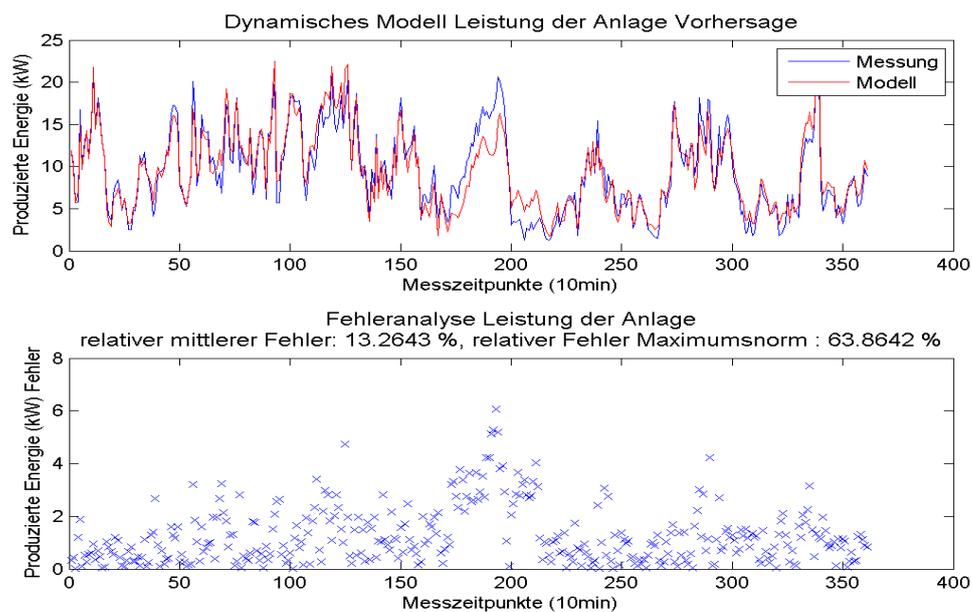


Abbildung C.4: Vorhersage mit Polynomen vom Grad 2

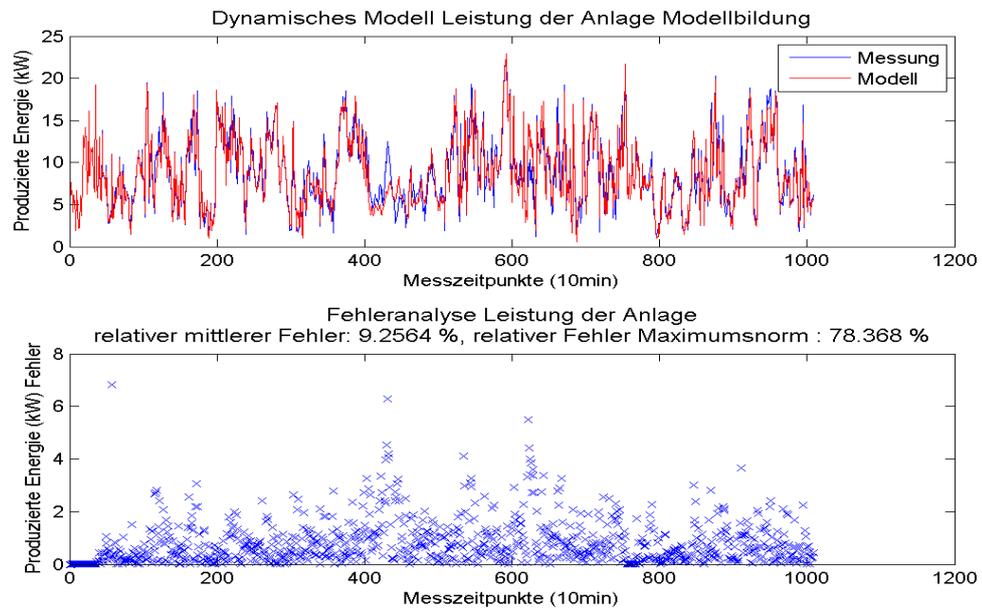


Abbildung C.5: Modellbildung mit Polynomen vom Grad 3

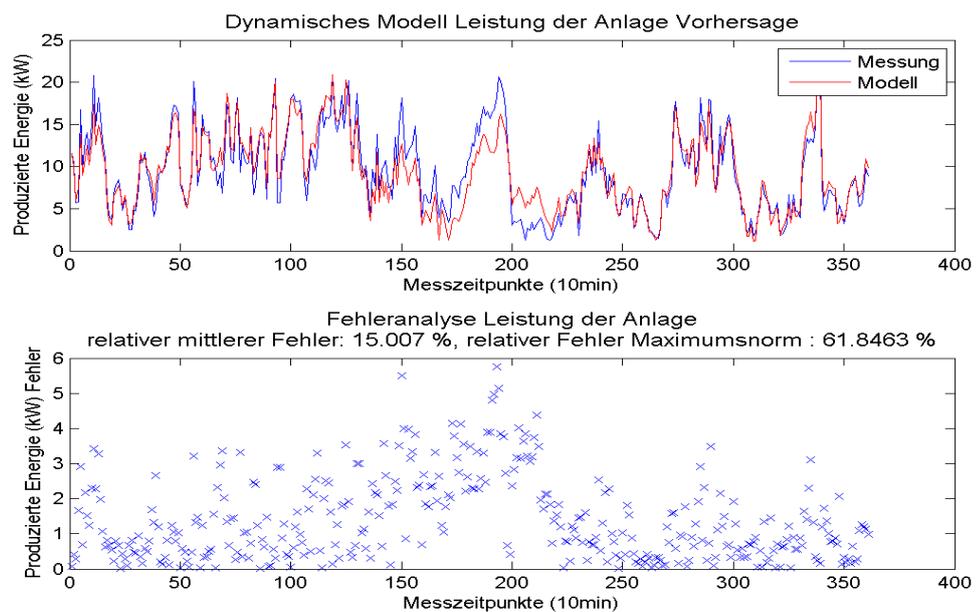


Abbildung C.6: Vorhersage mit Polynomen vom Grad 3

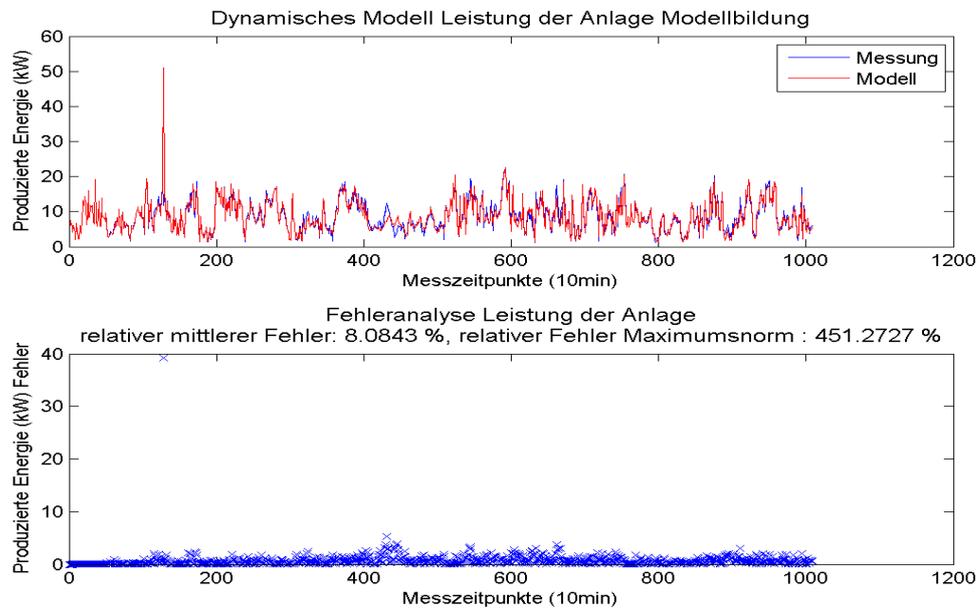


Abbildung C.7: Modellbildung mit Polynomen vom Grad 4

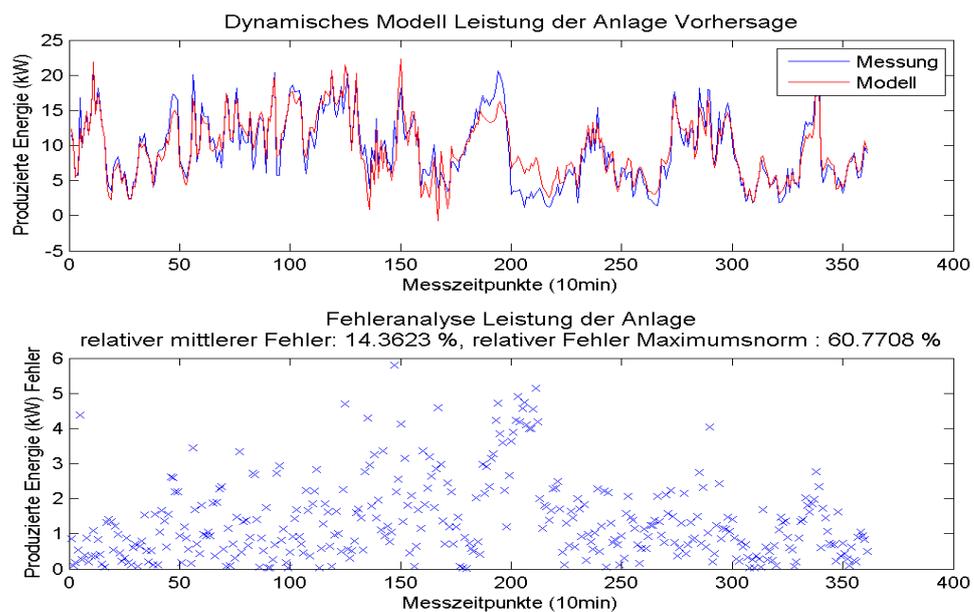


Abbildung C.8: Vorhersage mit Polynomen vom Grad 4

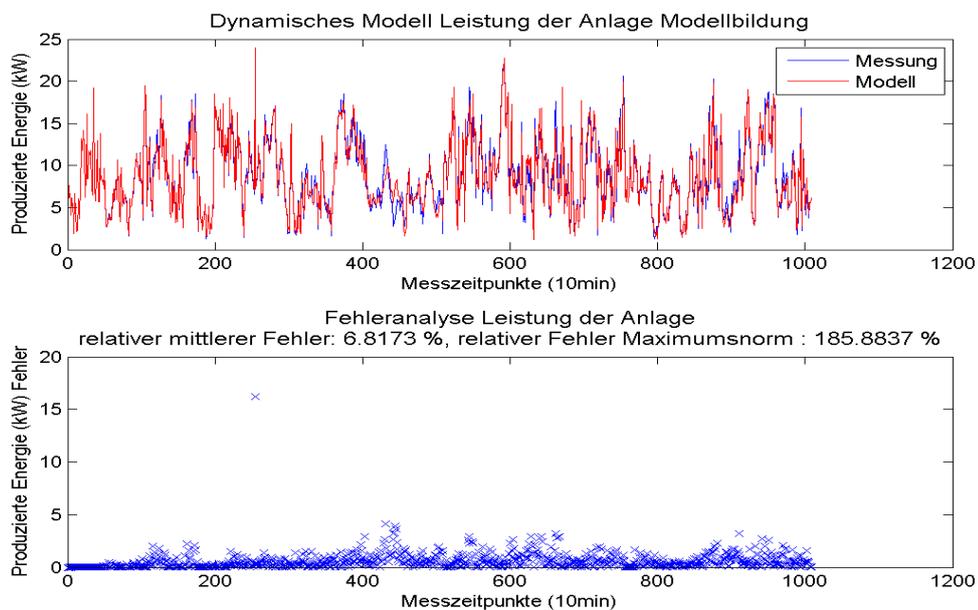


Abbildung C.9: Modellbildung mit Polynomen vom Grad 5

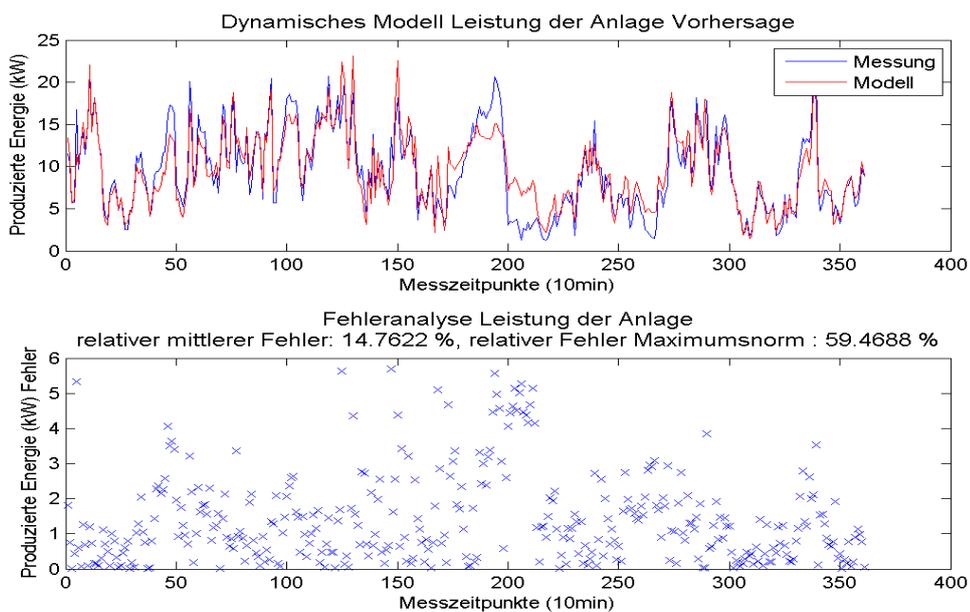


Abbildung C.10: Vorhersage mit Polynomen vom Grad 5

Anhang D

Anhänge zu den Messdaten

An dieser Stelle sind nun die Graphen der einzelnen gemessenen Größen abgedruckt.

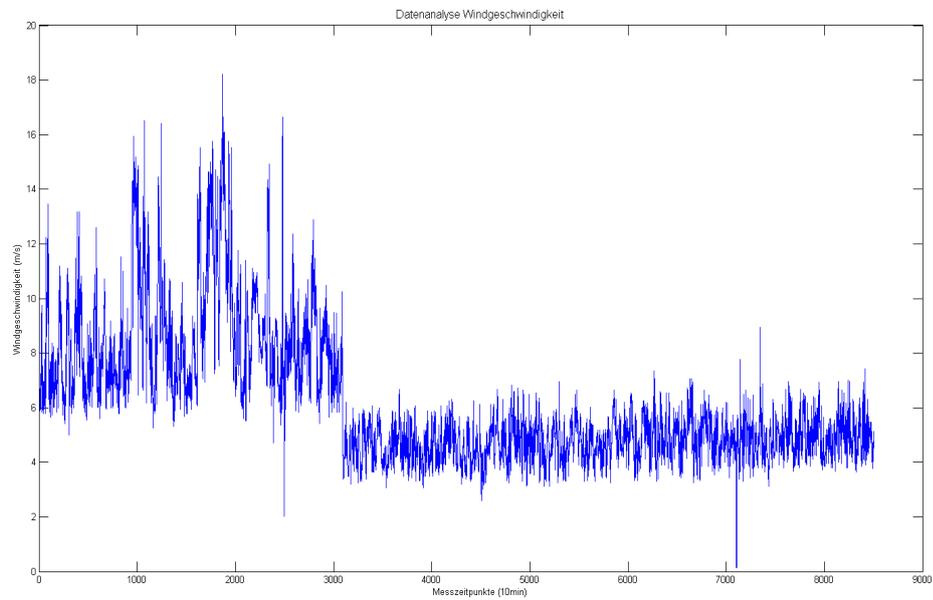


Abbildung D.1: Plot der Windgeschwindigkeit über alle Messzeitpunkte

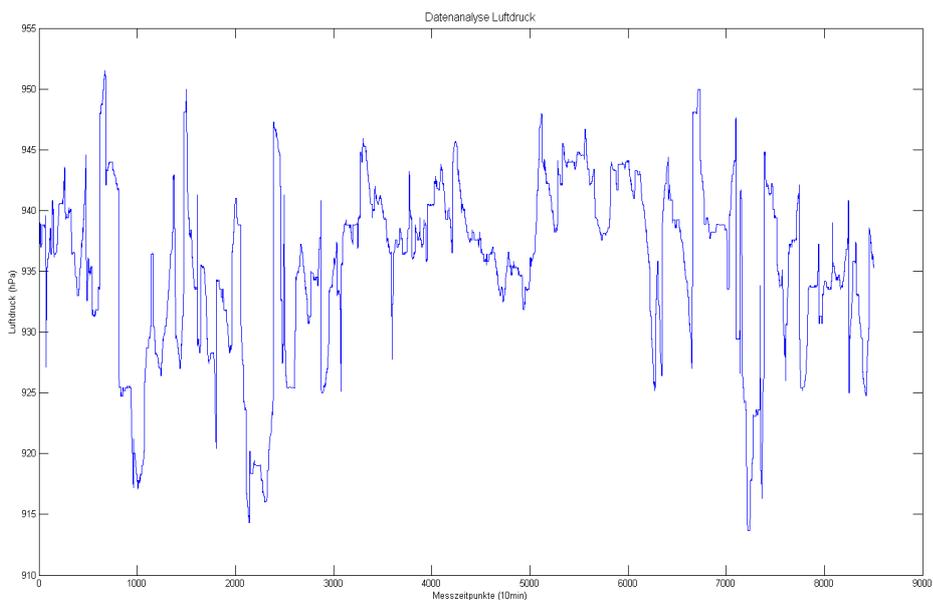


Abbildung D.2: Plot des Luftdruckes über alle Messzeitpunkte

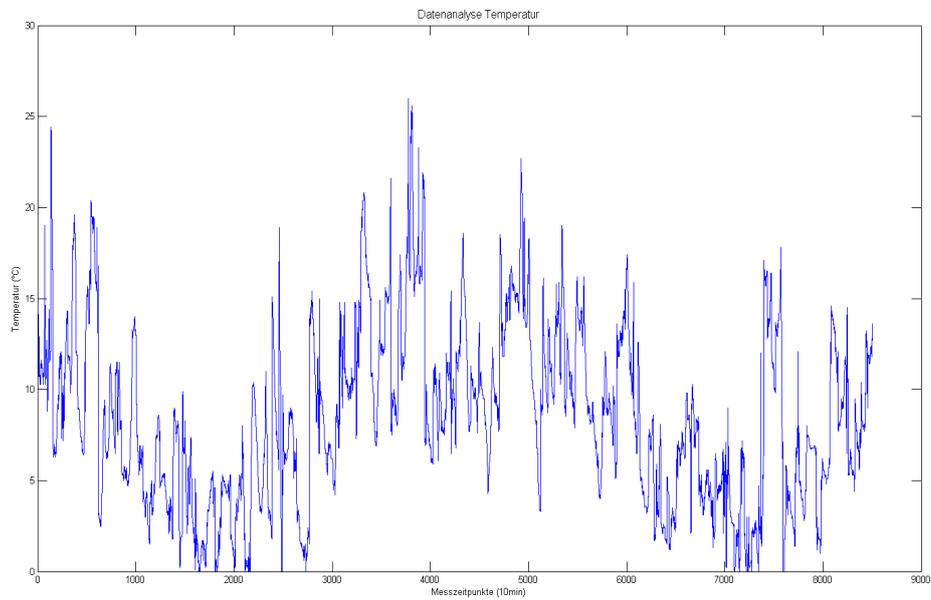


Abbildung D.3: Plot der Temperatur über alle Messzeitpunkte

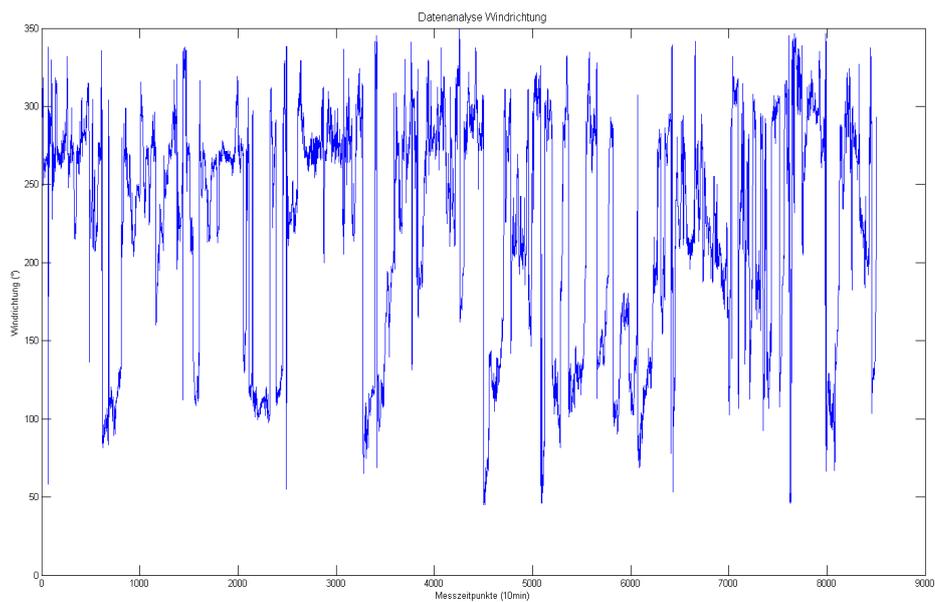


Abbildung D.4: Plot der Windrichtung über alle Messzeitpunkte

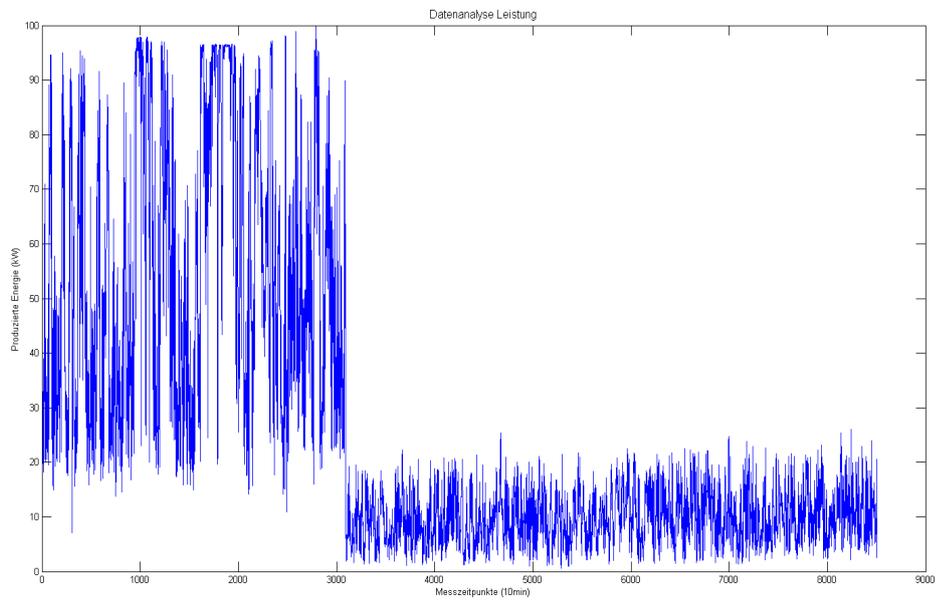


Abbildung D.5: Plot der Leistung über alle Messzeitpunkte

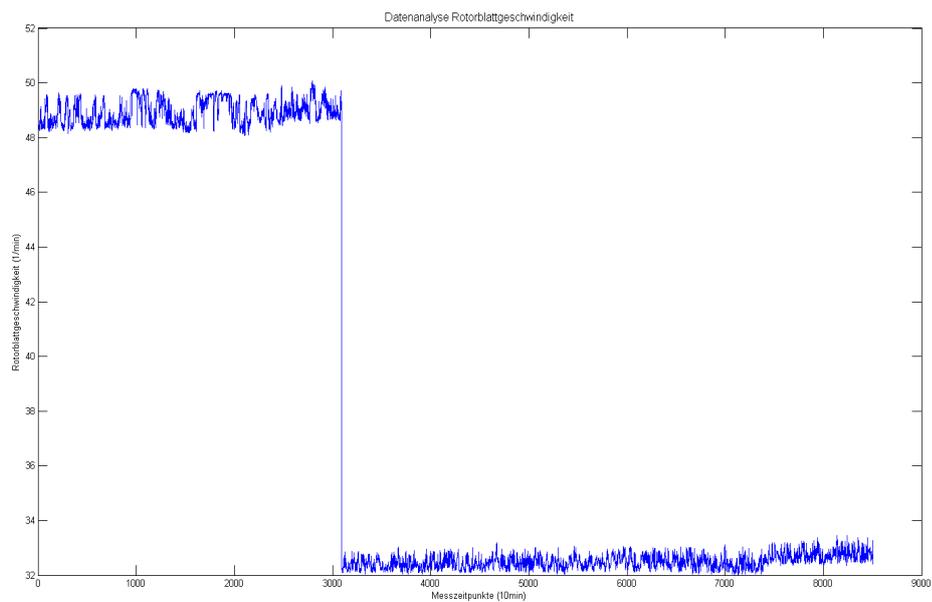


Abbildung D.6: Plot der Rotorblattgeschwindigkeit über alle Messzeitpunkte

Anhang E

Informationen zum MATLAB-Programm

In diesem Abschnitt soll kurz erläutert werden, wie numerische Untersuchungen mit Hilfe des MATLAB-Programmes durchgeführt werden können. Für alle Untersuchungen gilt, dass die Datei mit den Messwerten, im vorliegenden Fall `daten.txt`, sich im selben Ordner befinden muss, wie die auszuführende `.m`-Datei.

E.1 Statisches Modell

Das statische Modell wird mit Hilfe der Methode `statmod.m` durchgeführt. Hier die Dokumentation der Datei:

```
function statmod(k,d,mStart,mEnd,pStart,pEnd)
%statmod führt das statische Modell aus
%Input:
%k - Spaltenindex der gesuchten Größe (Energie k = 5)
%d - Maximaler Grad der Ansatzpolynome
%mStart - Startindex für Modellbildung
%mEnd - Endindex für Modellbildung
%pStart - Startindex für Vorhersage
%pEnd - Endindex für Vorhersage
```

Die Spaltenindizes entsprechen der Reihenfolge der Daten in Kapitel 3.1. Um beispielsweise Abbildung 4.1 zu erstellen lautet der Aufruf: `statmod(5,5,1,1008,1009,1369)`.

Sollen die Eingangsdaten beim statischen Modell variiert werden, so muss dies manuell in der Datei `datenvorb.m`, wie in Zeile 25 angedeutet, vorgenommen werden:

```
%L(:,5) = [];
```

Es muss also die zu ignorierende Spalte in der Matrix L entfernt werden. Hierbei ist zu beachten, dass innerhalb dieser Methode bereits die Spalte mit den Messwerten der zu bestimmenden Größe entfernt wurde und sich deshalb die Indizes der Spalten verschoben haben.

Um Optimierungstests wie in Kapitel 4.1.4 durchzuführen muss die Methode `teststatmod.m` verwendet werden.

```
function teststatmod(k,d,mStart,mEnd,pStart,pEnd)
%teststatmod führt Testoptimierungen mit Hilfe des statischen
%Modelles aus
%Input:
%k - Spaltenindex der gesuchten Größe (Energie k = 5)
%d - Maximaler Grad der Ansatzpolynome
%mStart - Startindex für Modellbildung
%mEnd - Endindex für Modellbildung
%pStart - Startindex für Vorhersage
%pEnd - Endindex für Vorhersage
```

Der Aufruf erfolgt analog zu `statmod.m`

E.2 Dynamisches und variiertes dynamisches Modell

Das dynamische und das variierte dynamische Modell werden mit Hilfe der Methode `dynmod.m` durchgeführt. Hier die Dokumentation der Datei:

```
function dynmod(eW,zG,M,G,d,boolvar)
%dynmod wendet das dynamische Modell auf die einzulesenden Daten an
%Input:
%eW - Zeilenvektor der die Spaltenindices der Eingangswerte enthält
%zG - Zeilenvektor der die Spaltenindices der Zustandsgrößen enthält
%M - Endindex der Modellbildung
%G - Endindex der Vorhersage
%d - Maximaler Grad der Ansatzpolynome
%boolvar - Boolean Wert der zwischen dynamischem und variiert
%dynamischem Modell wählt. 0 - normal, 1 - variiert
```

Die Spaltenindizes entsprechen der Reihenfolge der Daten in Kapitel 3.1. Um beispielsweise Abbildung B.1 zu erstellen lautet der Aufruf: `dynmod([5],[1 2 3 4`

7],1008,1369,1,0).

Analog dazu lautet der Aufruf für Abbildung C.1: `dynmod([5],[1 2 3 4 7],1008,1369,1,1)`.

Sollen die Eingangsdaten bei den dynamischen Modellen variiert werden, so müssen die eingegebenen Vektoren verändert werden. Um beispielsweise Abbildung B.1 ohne den Luftdruck zu erstellen lautet der Aufruf: `dynmod([5],[1 2 4 7],1008,1369,1,0)`.

Literaturverzeichnis

- [1] Peter Deuffhard, Andreas Hohmann „Numerische Mathematik 1. Eine algorithmisch orientierte Einführung. 4. Auflage“. Walter De Gruyter GmbH & Co. KG, Berlin, 2008.
- [2] Martin Hanke-Bourgeois „Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens“ 3. aktualisierte Auflage Vieweg und Teubner, Wiesbaden, 2009
- [3] Christof Büskens „Numerische Mathematik 1 Numerische Mathematik 2“ Vorlesungsskript Wintersemester 2010/11 Sommersemester 11 unkorrigierte Fassung Bremen, 2007
- [4] Bodo Blume, Christof Büskens und Dennis Wassel „Messdatengestützte Modellierung und Simulation einer Gasturbine“ ASIM 2009 - 20th Symposium Simulation Techniques, September 2009, Cottbus, Germany. ASIM-Mitteilung AMB 124, ASIM 2009 - Extended Abstracts, A. Gnauck, B. Luther (Eds.), pp. 231-234, Shaker, 2009.
- [5] Bodo Blume und Christof Büskens „Datenbasierte Modellierung und Simulation von Biogasanlagen mit statischen und dynamischen Modellen“ ASIM 2009 - 20th Symposium Simulation Techniques, September 2009, Cottbus, Germany. ASIM-Mitteilung AMB 124, ASIM 2009 - Extended Abstracts, A. Gnauck, B. Luther (Eds.), pp. 110-113, Shaker, 2009.
- [6] H. Böhhnisch, H. Hald, Ch. Kensche, A. Kußmann, J. P. Molly und H. Seifert „Entwicklung, Bau und Betrieb einer 30/100-kW-Windkraftanlage für die Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt.“

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst und ausschließlich die angegebenen Quellen und Hilfsmittel verwendet habe.

Bremen, den 8. August 2011