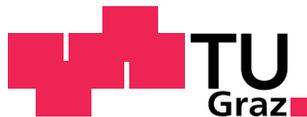


Masterarbeit

Modellierung und Simulation einer kurzfristigen Leistungsprognose ausgewählter dezentraler Stromerzeugungsanlagen

Aichholzer Thomas, BSc

Institut für Technische Informatik
Technische Universität Graz
Vorstand: O. Univ.-Prof. Dipl.-Ing. Dr. techn. Reinhold Weiß



Begutachter: Ass. Prof. Dipl.-Ing. Dr. techn. Christian Steger
Betreuer (TU Graz): Dipl.-Ing. Dr. techn. Philipp Maria Glatz, Bakk.techn.
Betreuer (KELAG Netz GmbH): Dipl.-Ing. Dr. techn. Reinhard Iskra

Graz, im September 2011

Kurzfassung

Auf Grund der steigenden Anzahl dezentraler Anlagen im letzten Jahrzehnt, wie Photovoltaik und Wasserkraft, und den dadurch entstehenden Problemen in Stromnetzen, wird es für Verteilernetzbetreiber immer wichtiger eine Methode zu besitzen, die die Leistung einer dezentralen Anlage in naher Zukunft automatisiert und ohne Einbringung von Expertenwissen prognostiziert.

Diese Arbeit beschäftigt sich mit der Modellierung einer kurzfristigen Leistungsprognose von dezentralen Anlagen, wie Photovoltaik und Wasserkraft. Auf Grund der gegebenen Datensituation der Verteilernetzbetreiberfirma KELAG Netz GmbH in Kärnten/Österreich, werden einerseits verschiedene Modelle auf Basis von maschinellen Lernverfahren für die Wasserkraftwerke untersucht und andererseits ein physikalisches Modell für Photovoltaikanlagen auf die Eignung der kurzfristigen Leistungsprognose geprüft. Die Simulationen dazu werden in Matlab durchgeführt. Bei den Lernverfahren werden *feedforward neuronale Netze*, *rekurrente neuronale Netze* wie z.B. *Elman Netze* und *Echo State Netze* implementiert, und verschiedene Inputkonfigurationen und Konfigurationen der neuronalen Netze simuliert. Für die Modellierung von Wasserkraftwerken gibt es zu den meisten Kraftwerken genügend lange Vergangenheitszeitreihen der Leistung, der Temperatur und des Niederschlags. Somit kann ein regressionsbasierter Modellansatz für Kraftwerke dieser Art gewählt werden. Von den einzelnen Photovoltaikanlagen gibt es keine Messdaten über den zeitlichen Verlauf der Leistung. Es liegen Messdaten einer großen Photovoltaikanlage vor, die im Besitz der Verteilernetzbetreiberfirma ist. Bei nahezu allen anderen Photovoltaikanlagen die am Netz hängen, liegen nur die Werte über die Spitzenleistung, den Standort sowie die Jahressummenwerte der Energieumwandlung vor. Für das Modell der Photovoltaikanlagen wird daher ein physikalisches Modell auf die Referenzanlage getestet und auf die ungemessenen Anlagen angewendet. Die Ergebnisse zeigen, dass die Modelle für Verteilernetzbetreiber eine gute Grundlage für die Schließung der momentanen Lücke in der Prognose von Wasserkraftwerken und Photovoltaikanlagen bieten. Es werden Modelle für die kurzfristige Leistungsprognose für die dezentralen Anlagentypen Photovoltaik und Wasserkraft untersucht, da im Süden und Südosten von Österreich vermehrt diese Art von kleinen dezentralen Erzeugungsanlagen anzutreffen ist.

Diese Arbeit beschreibt ein mögliches Werkzeug zur Behebung von Problemfällen im Stromverteilernetz, wie: (i) Leitungsüberlast durch Abschalten anderer Leitungen und (ii) der Unterdrückung der Stromkreisunterbrechung im Fehlerfall auf einer Leitung, die durch die dezentrale Stromerzeugung hervorgerufen werden.

Schlüsselwörter: kurzfristige Leistungsprognose; dezentrale Erzeugungsanlagen; Photovoltaik; Wasserkraft; maschinelle Lernverfahren; Regression; physikalisches Modell

Abstract

Due to the increasing number of decentralized generation systems in the last decade like photovoltaic and stream-water power plants and the resulting problems in the electricity network, it is getting more and more important for the distribution network operators to have a method or utility which forecasts the power generation of decentralized generation systems in the near future automatically and without expertise.

This work deals with the modelling of a short-term power generation forecast of decentralized systems based on photovoltaic and stream-water power. Based on the given data situation of the distribution network operator company KELAG Netz GmbH in Carinthia/Austria, on the one hand different models based on machine learning methods are studied for the stream-water power plants, on the other hand a physical model for photovoltaic systems is checked to get information about the applicability of these methods for short-term power generation forecasts. All simulations are made in Matlab. For machine learning methods, *feedforward neural networks*, *recurrent neural networks (Elman networks)* and *echo state networks* are implemented and simulated with different input and network configurations. For the modeling of stream-water power plants, there are sufficiently long time series of the past power generation, temperature and precipitation. Thus, a regression model based approach for power plants of this type can be chosen. There are no time-series of the past amount of energy conversion or other necessary measurements of photovoltaic power systems, except from one big photovoltaic system, which is owned by the distribution network operator company and may serve as a reference system. For nearly all other photovoltaic systems which are connected to the electricity network of the distribution network company, there is only information about the peak output of the system and the value of the total annual energy conversion of each photovoltaic system. Therefore a physical model is tested on the reference photovoltaic system and applied on the unmeasured assets. The results show that the models are providing a good basis for the distribution network operators to bridge the gap at short-term forecasting of the power generation of photovoltaic and stream-water power plants. Only models for the forecast of decentralized photovoltaic plants and stream-water power plants are examined, because in the south and southeast of Austria there are almost these kinds of decentralized generation systems.

This work is presenting a possible utility to solve problems at energy distribution networks, like: (i) powerline overload by switching off other powerlines and (ii) suppression of the disconnection of circuits in the case of an error at powerlines, which are evoked by decentralized powergeneration.

Keywords: short-term generation forecast; decentralized generation systems; photovoltaic; waterpower; machine learning; regression; physical model

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am

.....
(Unterschrift)

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....
date

.....
(signature)

Danksagung

Diese Masterarbeit wurde im Studienjahr 2010/11 am Institut für Technische Informatik an der Technischen Universität Graz durchgeführt.

Ich möchte mich an dieser Stelle bei all jenen bedanken, die mich im Laufe meines Studiums finanziell, fachlich als auch seelisch unterstützt haben. Besonderer Dank ergeht dabei an meine Betreuer am Institut und an die Firma KELAG Netz GmbH, die mir die Möglichkeit zur Durchführung dieser Arbeit geboten hat. Hierbei möchte ich speziell meinen Dank an Herrn Ass. Prof. Dipl.-Ing. Dr. techn. Christian Steger, Herrn Dipl.-Ing. Dr. techn. Philipp Maria Glatz, Bakk.techn, Herrn Dipl.-Ing. Dr. Reinhard Iskra sowie Herrn Dipl.-Ing. Obernosterer Gerald richten.

Ich möchte aber auch meinen speziellen Dank an meine Eltern und Familie richten, die mir stets Vertrauen entgegengebracht und mich finanziell unterstützt haben. Dank geht auch an meine StudienkollegInnen und Freunde, die mir den passenden Rahmen sowie Ausgleich zur Bewältigung meines Studiums geboten haben.

Widmen möchte ich diese Arbeit meiner großartigen Frau, die mir im Studium stets großes Verständnis entgegengebracht hat, mich immer wieder motiviert und mir stets zur Seite steht. Auch ihr möchte ich großen Dank aussprechen.

Graz, im September 2011

Aichholzer Thomas

Inhaltsverzeichnis

1	Einleitung	15
1.1	Motivation	16
1.2	Zielsetzung	16
1.3	Ansiedlung der Arbeit im System des Verteilernetzbetreibers	16
1.4	Einsatzmöglichkeit der Modellierung	17
1.5	Gliederung der Arbeit	19
2	Untersuchungen zur vorhandenen Datensituation	21
2.1	Datenanalyse für die physikalische Modellierung von Photovoltaikanlagen	22
2.2	Datenanalyse für Regressionsmodelle für kleine Wasser-Laufkraftanlagen	22
2.2.1	Datenbereinigung	30
2.2.2	Methodeneingrenzung	31
3	Verwandte Forschungsthemen und Projekte	32
3.1	Physikalische Modelle	33
3.1.1	Quaschnig et al. [Qua96]	33
3.1.2	Blank et al. [Bla06]	34
3.2	Regressionsbasierte Modelle	34
3.2.1	Nicklass et al. [Nic10]	35
3.2.2	Basurko et al. [ABIBM06]	36
3.2.3	Stadlober et al. [SHP08]	38
3.2.4	Kales et al. [Kal03]	39
3.2.5	Coulibaly et al. [CAB01]	39
3.2.6	Sacchi et al. [SOP ⁺ 07]	41
3.2.7	Zhao et al. [ZXDH]	42
4	Grundlagen zu den Modellierungsansätzen	45
4.1	Grundlagen zum physikalischen Ansatz	45
4.1.1	Sonnenstandsberechnung nach dem SUNAE - Algorithmus	45
4.1.2	Extraterrestrische Solarstrahlung	45
4.1.3	Berechnung der diffusen Solarstrahlung	46
4.1.4	Solarstrahlung auf geneigte Flächen	47
4.1.5	Temperatureinfluss auf Solarpaneele	49
4.1.6	Wechselrichter Wirkungsgrad	50
4.2	Grundlagen zum regressionsbasierten Ansatz	52
4.2.1	Multilayer Perzeptron basierte neuronale Netze	52

4.2.2	Elman rekurrente neuronale Netze	59
4.2.3	Echo State Netze	64
5	Untersuchungen regressionsbasierter Modelle	71
5.1	Globales Fehlermaß für unterschiedliche Modelle	72
5.2	Multilayer Perzeptron basierte neuronale Netze	74
5.2.1	Modellierungskonzept	74
5.2.2	Implementierung des ersten Modells	76
5.2.3	Auswertung des ersten Modells	80
5.2.4	Verfeinerung der ersten Testimplementierung	86
5.2.5	Training mit aktuellem Leistungswert als zusätzlichen Input	92
5.2.6	Bemerkungen zu den multilayer Perzeptron basierten neuronalen Netzen	98
5.3	Elman rekurrente neuronale Netze	99
5.3.1	Onlinetraining ohne Windowing	100
5.3.2	Onlinetraining mit Windowing, ohne Information über Leistungswerte	103
5.3.3	Onlinetraining mit Windowing, mit Information über die Leistungs- werte	104
5.3.4	Bemerkungen zu den Elman rekurrenten Netzen	105
5.4	Echo State Netze	105
5.4.1	Einfaches Echo State Netz mit Onlinetraining, ohne Windowing, ohne Information über Leistungswerte	106
5.4.2	Erweitertes Echo State Netz mit Onlinetraining und Windowing, ohne Information über Leistungswerte	109
5.4.3	Erweitertes Echo State Netz mit Onlinetraining und Windowing, mit Information über die Erzeugungswerte	114
5.4.4	Bemerkungen zu den Echo State Netzen	114
5.5	Auswahl des regressionsbasierten Modells	114
5.6	Weitere Untersuchungen auf das ausgewählte regressionsbasierte Modell . .	115
5.6.1	Robustheit gegenüber realen Wetterprognosedaten	115
5.6.2	Prognose von weiteren Wasserlaufkraftwerken	121
5.7	Zusammenfassung	125
6	Physikalisches Modell	126
6.1	Evaluierung der Berechnung der Solarstrahlung auf geneigte und gedrehte Flächen	128
6.2	Evaluierung der Temperaturabschätzung von Paneelen	131
6.3	Verifizierung des Simulationsmodells	132
6.4	Anwendung vom Simulationsmodell auf ungemessene Photovoltaikanlagen .	134
6.5	Einsetzbarkeit des physikalischen Modells zur kurzfristigen Leistungspro- gnose von Photovoltaikanlagen	141
7	Schlussbemerkungen und Ausblick	144
7.1	Ergänzendes zu Wasserlaufkraftwerke	145
7.2	Ergänzendes zu Photovoltaikanlagen	145

Literaturverzeichnis	147
Anhang	151
A Implementierung der Simulationen, README	151
B Anhang, physikalische Modellierung	155
C Anhang, regressionsbasierte Modellierung	161
D Relevante Datenblätter, gemessene PVA <i>St. Veit</i>	183
E Abkürzungen	191

Abbildungsverzeichnis

1.1	Schematische Darstellung der Ansiedlung der Arbeit.	17
1.2	Typische Fehlerszenarien im Verteilernetz durch steigende Anzahl an dezentralen Erzeugungsanlagen.	18
2.1	Zeitreihen über Leistung, Solarstrahlungen, Peneeltemperatur und Umgebungstemperatur der PVA in <i>St. Veit</i>	23
2.2	Jahreszeitreihen der erzeugten Leistung von <i>Kirchbach</i> und <i>Radegund</i> für das Jahr 2010.	24
2.3	Jahreszeitreihen vom Jahr 2010 für <i>Kirchbach</i> und <i>Radegund</i> mit zugehörigem Jahresniederschlags- und Jahrestemperaturverlauf.	25
2.4	Hervorgehobener Bereich <i>A</i> aus Abbildung 2.3.	26
2.5	Hervorgehobener Bereich <i>B</i> aus Abbildung 2.3.	27
2.6	Kreuzkorrelationsfunktionen der Jahreszeitreihen 2010 der Kraftwerke <i>Kirchbach</i> und <i>Radegund</i> mit den Jahreszeitreihen der zur Region zugehörigen Temperatur und des Niederschlags.	29
2.7	Ausschnitte der Zeitreihen aus dem Jahr 2010 zur Demonstration von Datenanomalien.	30
3.1	Struktur eines multilayer Perzeptron basierten neuronalen Netzes, [ABIBM06, Fig. 1].	37
3.2	Blockdiagramm eines Echo State Netzes, [SOP ⁺ 07, Fig. 1].	41
3.3	Getrennte Hyperebenen (Bereiche der zwei Klassen), H_1 und H_2 , [Bur98, Fig. 5].	42
4.1	Veranschaulichung der Winkelbezeichnungen des Sonnenstandes, [Qua96, Abb. 3.1].	46
4.2	Solarstrahlung beim Durchlaufen der Athmosphäre, [Qua96, Abb. 3.4].	47
4.3	Veranschaulichung der Winkelbezeichnungen für den Sonnenstand auf eine geneigte Fläche, [Qua96, Abb. 3.3].	48
4.4	Strom- und Leistungsverlauf einer Solarzelle, [Bla06, Abb. 2.2].	50
4.5	Wirkungsgrad eines Wechselrichters in Abhängigkeit der Auslastung, [Bla06, Abb. 2.6].	51
4.6	Schematische Darstellung eines einfachen Perzeptrons, [Pfa01, Abbildung 3.1].	53
4.7	Perzeptron-Netz zur XOR Problemlösung, [Pfa01, Abbildung 4.5].	54
4.8	Ein dreischichtiges Perzeptron-Netz (mit einem hidden Neuron und hidden Layer) zur Lösung des XOR Problems, [Pfa01, Abbildung 4.6].	54

4.9	Eine Realisierung der <i>logsig</i> , <i>tansig</i> und der <i>purelin</i> Funktion.	55
4.10	Skizze eines Neurons.	56
4.11	Struktur eines multilayer Perzeptron basierten neuronalen Netz, [ABIBM06, Fig. 1].	56
4.12	Gradientenabstieg bei starken lokalen Minima.	58
4.13	Typische Elman Netzstruktur als eine Realisierung von einem rekurrenten neuronalen Netz, [CAB01, Fig. 4].	59
4.14	Input delayed neuronales Netz (ohne Rückkopplung) und time delayed neuronales Netz (mit Rückkopplung), [CAB01, Fig. 3].	60
4.15	Netzstruktur eines Echo State Netzes, [Jae10, Fig. 1].	65
4.16	Demonstration der Auswirkung unterschiedlicher Netzdimensionen und Spektralradien eines einfachen Echo State Netzes.	69
4.17	Demonstration der Auswirkung eines Spektralradius > 1	70
5.1	Teil des Trainingsdatensatzes. 2008 und 2009 der Leistung des WLKW <i>Kirchbach</i> mit zugehörigem Niederschlags- und Temperaturverlauf.	72
5.2	Testdatensatz bestehend aus den Datenreihen 2010 der Leistung des WLKW <i>Kirchbach</i> mit zugehörigem Niederschlags- und Temperaturverlauf.	73
5.3	Demonstration der unterschiedlichen Fälle bei unterschiedlicher Position von t_0	76
5.4	Inputs und Outputs des gesuchten multilayer Perzeptron basierten feedforward neuronalen Netze.	77
5.5	Simulation eines guten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes.	82
5.6	Simulation eines vermeintlich guten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes.	83
5.7	Simulation eines guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes.	85
5.8	Simulation eines vermeintlich guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes.	86
5.9	Simulation des gefundenen besten Netzes des verfeinerten Simulationsdurchlaufes.	89
5.10	Ausschnitt über die Wintermonate aus Abbildung 5.9.	91
5.11	Ausschnitt über die Sommermonate aus Abbildung 5.9 mit hervorgehobenem Bereich lokaler Gewitter.	92
5.12	Streudiagramm, Kraftwerk <i>Kirchbach</i>	93
5.13	Simulation des gefundenen besten Netzes mit aktuellem Leistungswert als weiteren Input.	95
5.14	Ausschnitt über Sommermonate aus Abbildung 5.13.	96
5.15	Simulation des gefundenen besten Netzes mit aktuellem Leistungswert als weiteren Input.	98
5.16	Ausschnitt aus Abbildung 5.15.	99
5.17	Streudiagramm, Kraftwerk <i>Kirchbach</i> , des gefundenen besten Netzes aus der Simulation mit aktuellem Wert der Leistung und der Vergangenheit der Leistungszeitreihe des Kraftwerks als weiteren Input, dargestellt in Abbildung 5.15.	100

5.18	Simulation des gefundenen besten Elman Netzes ohne Windowing.	102
5.19	Einfaches Echo State Netz ohne Rückkopplung und einem linearen Readout.	106
5.20	Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing.	108
5.21	Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing, mit geglättetem Temperaturverlauf.	110
5.22	Echo State Netz mit einem Perzeptron als Readoutelement.	111
5.23	Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing, mit einem Perzeptron als Ausgangsneuron anstatt dem linearen Readout.	111
5.24	Erweitertes Echo State Netz mit einem Perzeptron als Readoutelement.	112
5.25	Ausschnitt des Vergleiches der exakten (gemessenen) Wetterdaten mit den Wetterdaten mit Gaußschen additiven weißen Rauschen.	116
5.26	Ausschnitt des Vergleiches der exakten (gemessenen) Wetterdaten mit den Wetterdaten zufällig skaliert.	117
5.27	Boxplot der Simulationen mit den exakten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.	118
5.28	Boxplot der Simulationen mit den Gauß-verrauschten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.	119
5.29	Boxplot der Simulationen mit den zufällig nach einem Gaußrauschen skalierten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.	120
5.30	Simulation der Prognose des Kraftwerks <i>Radegund</i>	122
5.31	Simulation der Prognose des Kraftwerks <i>Kaningbach</i>	123
5.32	Simulation der Prognose des Kraftwerks <i>Winkl</i>	124
6.1	Simulationsmodell für die Simulationen der PV Anlagen.	127
6.2	Diffusstrahlungsanteil in Abhängigkeit des Klarheitsfaktors einer Messstation in Berlin für das Jahr 1960, Stundenmittelwerte, [Sch91, Fig.35].	129
6.3	Ausschnitt der Verläufe der gemessenen Solarstrahlung in der Paneelebene der Anlage <i>St. Veit</i> und der abgeschätzten Solarstrahlung auf die geneigte und gedrehte Paneelebene, auf Basis der Umrechnungsmodelle sowie der Messwerte horizontal gemessener Solarstrahlung der nächstgelegenen Messstation in <i>Klagenfurt</i>	130
6.4	Ausschnitt der Verläufe der gemessenen Paneeltemperatur der Anlage <i>St. Veit</i> und der abgeschätzten Paneeltemperatur, auf Basis des Umrechnungsmodells sowie der Messwerte der Umgebungstemperatur und einfallenden Solarstrahlung auf die geneigte und gedrehte Paneelfläche in <i>St. Veit</i>	131
6.5	Ausschnitt der Verläufe der gemessenen Leistung der PVA <i>St. Veit</i> und der abgeschätzten Leistung, bei nicht gemessener Paneeltemperatur und nicht gemessener Solarstrahlung in der Paneelebene.	134
6.6	Streudiagramm der Simulation mit ungemessener Paneeltemperatur und ungemessener Solarstrahlung in der Paneelebene.	135
6.7	Skizze der Datensituation für die Simulation ungemessener Anlagen.	136
6.8	Jahresverläufe der Solarstrahlung auf eine horizontale Fläche für <i>Wolfsberg</i> . Jahre: 2007, 2008, 2009 und 2010.	137

6.9	Jahressummen über α_{PV} und γ_{PV} einer $1kW - Peak$ Anlage am Standort <i>Wolfsberg</i> , für die Jahre 2007, 2008, 2009 und 2010.	138
6.10	Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 130425 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	139
6.11	Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 353243 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	140
6.12	Jahressummenverlauf einer nachgeführten $1kW - Peak$ Anlage, in Abhängigkeit der Neigung γ_{PV} für den Standort <i>Wolfsberg</i>	142
6.13	Sklaierter Jahressummen-Normverlauf, in Abhängigkeit der Neigung der Anlage Nr. 353243 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	143
B.1	Strukturgramm des SUNAE Algorithmus, [Qua96, Anhang], Teil 1.	156
B.2	Strukturgramm des SUNAE Algorithmus, [Qua96, Anhang], Teil 2.	157
B.3	Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 246959 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	158
B.4	Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 143344 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	159
B.5	Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 660867 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.	160
D.1	Datenblatt des <i>SMA Sunny Tripower 10000TL/12000TL/15000TL/17000TL</i> Wechselrichters, 1 von 2.	184
D.2	Datenblatt des <i>SMA Sunny Tripower 10000TL/12000TL/15000TL/17000TL</i> Wechselrichters, 2 von 2.	185
D.3	Datenblatt des <i>Kioto Photovoltaics KPV PE NEC 190/195/200/205/210</i> Wp Photovoltaikmoduls, 1 von 2.	186
D.4	Datenblatt des <i>Kioto Photovoltaics KPV PE NEC 190/195/200/205/210</i> Wp Photovoltaikmoduls, 2 von 2.	187
D.5	Datenblatt der <i>LSI DPAxxx-xxx Sternpyranometer</i> , 1 von 3.	188
D.6	Datenblatt der <i>LSI DPAxxx-xxx Sternpyranometer</i> , 2 von 3.	189
D.7	Datenblatt der <i>LSI DPAxxx-xxx Sternpyranometer</i> , 3 von 3.	190

Tabellenverzeichnis

5.1	Eigenschaften der tatsächlich guten regularisierten Netze nach dem ersten Simulationsdurchgang.	81
5.2	<i>eg_j</i> Werte der Prognoseintervalle des besten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.	81
5.3	<i>eg_j</i> Werte der Prognoseintervalle des angeführten vermeintlich guten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.	81
5.4	Eigenschaften der tatsächlich guten nicht regularisierten Netze nach dem ersten Simulationsdurchgang.	84
5.5	<i>eg_j</i> Werte der Prognoseintervalle des besten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.	85
5.6	<i>eg_j</i> Werte der Prognoseintervalle des angeführten vermeintlich guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.	87
5.7	Eigenschaften der besten 10 Netze nach dem verfeinerten Simulationsdurchlauf.	88
5.8	<i>eg_j</i> Werte der Prognoseintervalle des angeführten Netzes im verfeinerten Simulationslauf.	88
5.9	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Netzes der selben Netzkonfigurationen wie im verfeinerten Simulationslauf, aber zusätzlicher Randomisierung der Trainingsvektoren.	90
5.10	Eigenschaften der besten 10 Netze nach dem verfeinerten Simulationsdurchlauf und Randomisierung der Eingangsfolge.	90
5.11	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf mit aktuellem Leistungswert des Kraftwerks am Eingang.	94
5.12	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Netzes, mit dem aktuellen Leistungswert als weiteren Input.	94
5.13	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf mit aktuellem Leistungswert und vergangenen Leistungswerten des Kraftwerks am Eingang.	97
5.14	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Netzes, mit dem aktuellen und vergangenen Leistungswerten als weiteren Input.	98
5.15	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze ohne Windowing.	101

5.16	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Elman Netzes ohne Windowing.	102
5.17	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze mit Windowing, ohne Information über die Leistungswerte. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.	104
5.18	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Elman Netzes mit Windowing, ohne Information über die Leistungswerte des betrachteten Kraftwerks.	104
5.19	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze mit Windowing und mit Information der Leistungswerte. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.	105
5.20	<i>eg_j</i> Werte der Prognoseintervalle des besten gefundenen Elman Netzes mit Windowing und mit Information über Leistungswerte des betrachteten Kraftwerks.	105
5.21	Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Echo State Netze ohne Windowing. Aktivierungsfunktion Ausgang: purelin.	108
5.22	Eigenschaften der besten 10 Konfigurationen, gefunden über das globale Fehlermaß <i>eg</i> . Mit Windowing am Eingang, ohne Information über die Leistungswerte.	113
5.23	<i>eg</i> der besten Netze unterschiedlicher Methoden und Ansätze.	114
5.24	Trainingsdauer der besten Netze unterschiedlicher Methoden und Ansätze.	115
5.25	Mittlere <i>eg_j</i> Werte der 100 Simulationen des besten Netzes mit/ohne veräuschten Wetterdaten als Wettervorhersagedaten im Testset.	118
5.26	Mittlerer <i>eg</i> Wert der 100 Simulationen des besten Netzes mit/ohne veräuschten Wetterdaten als Wettervorhersagedaten im Testset.	119
5.27	<i>eg_j</i> Werte der Simulationen weiterer Wasserlaufkraftwerke.	121
5.28	<i>eg</i> Wert der Simulationen weiterer Wasserlaufkraftwerke.	122
6.1	Mittlerer absoluter Fehler und durchschnittliche Abweichung der Leistungsabschätzung vom maximalen Referenzwert, in Abhängigkeit unterschiedlicher Gegebenheiten für das Simulationsmodell.	133
6.2	Fehlermaß <i>eg</i> in Abhängigkeit unterschiedlicher Gegebenheiten für das Simulationsmodell.	133
6.3	Daten beispielhafter ungemessener Photovoltaikanlagen.	137
C.1	Eigenschaften der vermeindlich besten 100 regularisierten Netze nach dem ersten Simulationsdurchgang.	163
C.2	Eigenschaften der vermeindlich besten 100 nicht regularisierten Netze nach dem ersten Simulationsdurchgang.	166
C.3	Eigenschaften der besten 100 nicht regularisierten Netze aus der verfeinerten Simulation.	168
C.4	Eigenschaften der besten 100 nicht regularisierten Netze aus der Simulation mit dem aktuellen Leistungswert als Input.	171
C.5	Eigenschaften der besten 100 nicht regularisierten Netze aus der Simulation mit dem aktuellen und vergangenen Leistungswerten als Input.	173

C.6	Eigenschaften der 36 Elman Netze, ohne Windowing mit reinem online Training, aufsteigend sortiert nach dem mean squared Error.	174
C.7	Eigenschaften der 36 Elman Netze, mit Windowing und online Training, ohne Information über aktuelle und vergangene Leistungswerte, aufsteigend sortiert nach dem mean squared Error.	175
C.8	Eigenschaften der 36 Elman Netze, mit Windowing, online Training und mit Information über aktuelle und vergangene Leistungswerte, aufsteigend sortiert nach dem mean squared Error.	176
C.9	Eigenschaften der nach dem mean squared Error gefundenen besten Echo State Netzen, ohne Windowing mit reinem online Training und linearem Readout.	179
C.10	Eigenschaften der besten nach dem mean squared Error gefundenen Echo State Netzen unter Verwendung des sliding Windows als Input, ohne Information über die Leistungswerte des Kraftwerks.	182

Kapitel 1

Einleitung

Verteilernetze sind in Ihrer Struktur so aufgebaut, dass sie elektrische Energie an Verbraucher heranführen. Dazu ist es wichtig, die Anlagen so zu fahren, dass dabei eine stabile Spannungsversorgung innerhalb definierter Grenzen gewährleistet ist. Verteilernetze sind aus historischen Gründen heraus so konzipiert, dass sie ausgehend von zentralisierten Erzeugungsanlagen die Energie im Netz verteilen. Verteilernetzbetreiber müssen unter anderem die Last in einem Netz möglichst genau vorhersagen können, um mit den zentralen Erzeugungsanlagen soviel Energie bereitzustellen, dass die Spannungsstabilität gewährleistet ist. Die Prognose des Lastverhaltens dient zur Erstellung eines Fahrplans für Erzeugungsanlagen. Für die Lastprognose ist hierzu das Verbraucherverhalten in Abhängigkeit der Zeit das wichtigste zu untersuchende Kriterium. Um in gewisser Weise das Verbraucherverhalten beeinflussen zu können, wird aktuell am Thema Smart-Grids geforscht. [REG10] und [KKR⁺11] zeigt hierzu anschaulich ein Konzept für Smart-Grids. Konzepte dieser Art erfordern eine Kommunikation mit den Verbrauchern, und sehen diese nicht mehr als *passiv* an, um eine bessere Steuerung des Lastverhaltens von Verbrauchern zu erreichen. Die Konzepte reichen bis zu den sogenannten Micro-Grids, bei denen Netzteile autonom ihre Versorgungsqualität über intelligente Gerätschaften, dezentrale Stromerzeugung und virtuelle Energiespeicher regeln, siehe [HAIM07]. Die Verteilernetze können die Netzstabilität einerseits mit Ihren Anlagen durch Netzumkonfiguration bzw. Umschaltungen im Netz regeln. Andererseits kann über das Konzept des Smart-Grids ausgehend vom Verteilernetz das Verbraucherverhalten beeinflusst und optimiert werden.

Seit den letzten Jahren aber gibt es in Folge der Gesetzgebung und Förderungsmaßnahmen im Energiesektor einen starken Zuwachs an erneuerbarer Stromerzeugung [BWB⁺04] und somit steigt auch die Anzahl der dezentralen Erzeugungsanlagen (DEA). DEA sind kleine Anlagen, die Energie nahe den Verbrauchern in das Verteilernetz ein koppeln. Verschiedene Konzepte für DEA, vor allem Photovoltaikanlagen (PVA) betreffend, sind dazu beispielsweise in [CKI] anschaulich beschrieben. Auf Grund der vorliegenden gesetzlichen Lage können DEA dargebotsabhängig Energie in das Netz einspeisen. Das bedeutet, dass Verteilernetzbetreiber, z.B von PVA, dann Energie annehmen müssen, wenn die Betreiber von DEA einliefern möchten. Die Einspeisung wird sich in diesem Fall nach dem solaren Strahlungsangebot der Sonne richten. Analog wird es bei Wasserlaufkraftwerken (WLKW) der Fall sein, wobei die eingespeiste Energiemenge analog dem Niederschlag folgen wird. Bei dargebotsabhängiger Einspeisung wird die eingespeiste Energie zeitlich gesehen starke

Schwankungen aufweisen. Dies stellt die Verteilernetzbetreiber vor eine völlig neue Herausforderung, da zur Zeit nur mit eigenen Anlagenkapazitäten fehlende oder überschüssige Energie von DEA ausgeregelt werden kann. Einerseits müssen hierbei Konzepte von Smart-Grids und Micro-Grids greifen, andererseits ist es notwendig das Eigenleben von DEA zu kennen, um eine kurzfristige Prognose über ihre Leistung erstellen zu können, da Netzbetreiber nahezu blind bezüglich der Einspeisung sind.

1.1 Motivation

Derzeit ist eine Fahrplanerstellung seitens der DEA nicht üblich. Auf Grund der Dargebotsabhängigkeit dieser Anlagen ist es wichtig Werkzeuge zu erstellen, um die Stabilität der Spannung im Netz zu garantieren, damit keine elektrischen Probleme im Netz auftreten, trotz regional stark schwankender Einspeisung dieser Anlagen. Ein von Betreibern von DEA unabhängiges Werkzeug ist die kurzfristige Leistungsprognose. Wetterdaten und die technischen Daten bestimmen hierbei die Stromerzeugung. Dies führt zur Idee, dass Verteilernetzbetreiber selbst Fahrpläne über dezentrale Erzeuger generieren. Eine Grundlage für Prognosen dieser Art stellen meteorologische Wetterdaten dar. Hat man eine regionale Wetterprognose, so kann man die Leistung von DEA, wie PVA und WLKW, prognostizieren, indem man Prognosemodelle in Abhängigkeit der Wetterdaten erstellt.

1.2 Zielsetzung

Ziel der Arbeit ist es, verschiedene Prognosemethoden und Modelle auf ihre Eignung für die Aufgabenstellung zu untersuchen. Danach soll ein geeignetes Prognosemodell unter Heranziehung der prognostizierten Wetterdaten (Sonneneinstrahlung, Niederschlag) für einen bestimmten Zeitraum die Stromerzeugungsfahrpläne für die dezentralen Erzeuger erstellen. Dabei ist zwischen Sonnenkraft (Photovoltaik) und Wasserkraft zu unterscheiden. Die (prognostizierten) Fahrpläne für die einzelnen Erzeuger ergeben sich dann entsprechend dem Stromerzeugungstyp und seinem zugehörigen Wetterwert sowie der Nennleistung des Erzeugers.

1.3 Ansiedlung der Arbeit im System des Verteilernetzbetreibers

Die Arbeit soll Konzepte und Modelle zur kurzfristigen Leistungsprognose von DEA untersuchen. Sie wären in weiterer Folge, so wie die Lastprognose, ein wichtiger weiterer Bestandteil im Energiemanagement. Abbildung 1.1 zeigt schematisch, wo die Arbeit im bestehenden System eingreift. Wie bereits erwähnt, gibt es zentrale und dezentrale Stromerzeugungsanlagen, besser gesagt Kraftwerke, die Energieressourcen, wie Strahlung der Sonne, Wasserkraft, Windenergie, Öl, Kohle, uvm. in elektrische Energie umwandeln. Von zentralen Erzeugungsanlagen sind Fahrpläne bekannt. Außerdem ist die aktuelle Leistung bekannt, das heißt, es sind gemessene Kraftwerke. Es kann also eine Abstimmung der Kraftwerke untereinander erfolgen, sodass eine Spannungsstabilität im Netz erreicht werden kann. Dies wird durch Um- und Zuschaltungen von Kraftwerken gewährleistet. Verteilernetzbetreiber haben also Sicht auf das Verhalten, sowie Regelmechanismen zur

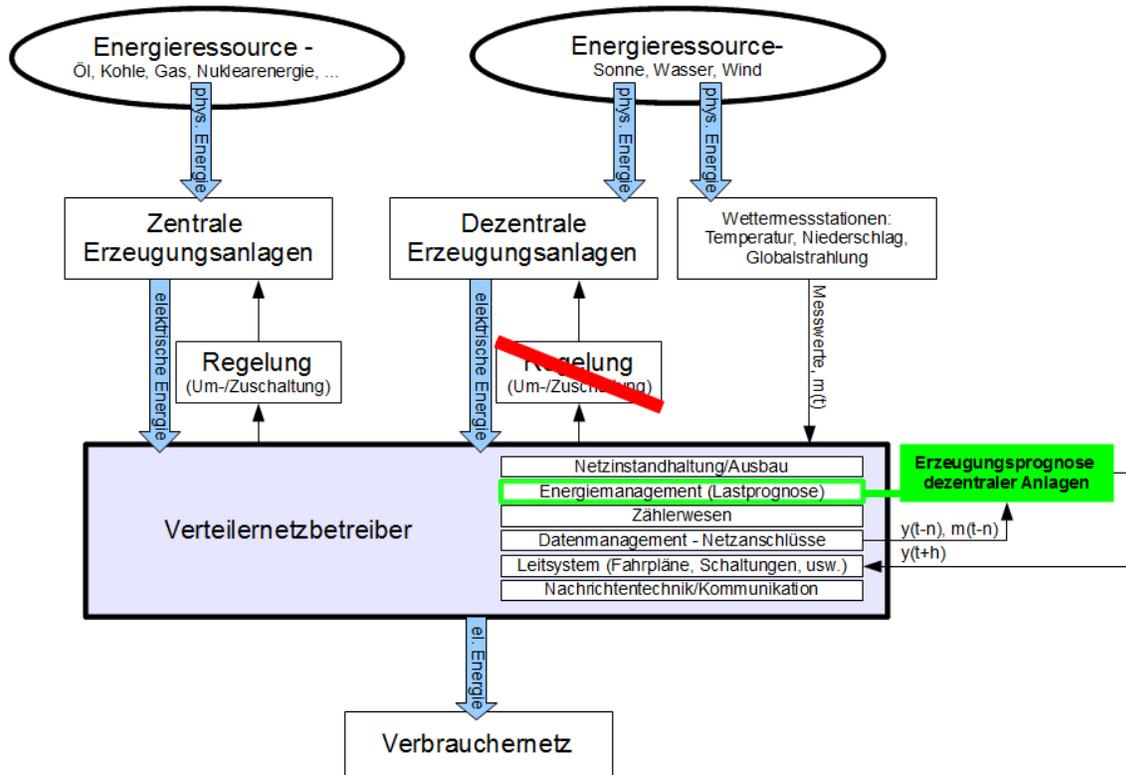


Abbildung 1.1: Schematische Darstellung der An siedlung der Arbeit.

Verfügung. Dies gilt für (kleine) DEA nicht. Hier haben Verteilernetzbetreiber seitens der DEA keine Mechanismen, um die eingespeiste Energie zu regeln. Verteilernetzbetreiber bestehen aus den Komponenten der Netzinstandhaltung und dem Netzausbau, der Nachrichten- und Kommunikationstechnik, dem Zählerwesen, dem Datenmanagement über Netzanschlüsse von Erzeugern und Verbrauchern, dem wichtigen System der Leittechnik für Netzschaltungen und Überwachung, sowie dem Energiemanagement zur Lastprognose und anderen wichtigen Datenverarbeitungen. Die Messdaten über Wetter, sowie über die Leistung gemessener Kraftwerke werden ebenfalls vom Datenmanagement verwaltet. Die Leistungsprognose von DEA muss also aus der Vergangenheit erfasste Wetterdaten und, sofern vorhanden, Daten über die Leistung von Anlagen aus dem Datenmanagement erhalten. Sie muss auch Wetterprognosedaten von außen erhalten. Nach erfolgter Verarbeitung liefert sie kurzfristige Prognosewerte über DEA, jeweils für die Zeit h voraus. Die Prognose ist letztendlich für den Betrieb im Leitsystem von entscheidender Bedeutung.

1.4 Einsatzmöglichkeit der Modellierung

Die kurzfristige Leistungsprognose von DEA wird auf Grund steigender Anlagenanzahl dieser Art immer notwendiger. Sie soll die Grundlage zur Vermeidung von Überlasten von Leitungen im Mittel- und Niederspannungsnetz, sowie Vermeidung der Unterdrückung der Abschaltung von Leistungsschaltern im Kurzschlussfall auf Leitungen schaffen. Diese

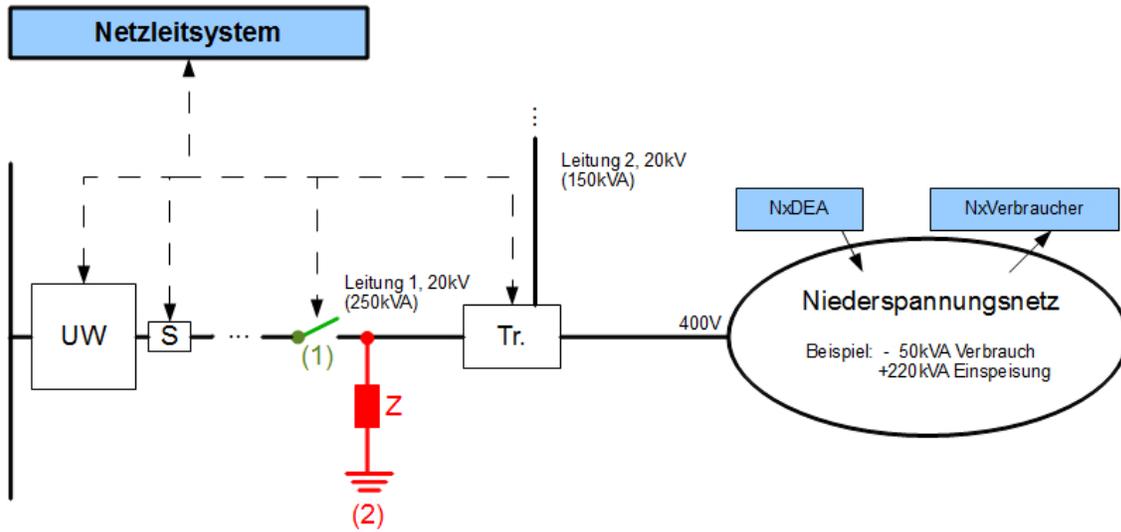


Abbildung 1.2: Typische Fehlerszenarien im Verteilernetz durch steigende Anzahl an dezentralen Erzeugungsanlagen. 1: Wegschalten von Leitungen, 2: Erdschluss.

zwei Fehlerszenarien sollen anhand der Abbildung 1.2 genauer erklärt werden. Dargestellt sei ein Ausschnitt aus der Netzstruktur eines Verteilernetzes. Ein Umspannungswerk, mit UW bezeichnet, stellt einen Abzweig vom Hochspannungsnetz dar und transformiert die Hochspannung auf das Mittelspannungsnetz. Hochspannungsnetze transportieren elektrische Energie über sehr weite Strecken. Mittelspannungsnetze versorgen dabei Regionen in Verteilernetzen. Im Mittelspannungsnetz transformieren Transformatoren, mit $Tr.$ bezeichnet, die Mittelspannung ins Niederspannungsnetz. Niederspannungsnetze sind meist örtlich begrenzt und versorgen beispielsweise Siedlungsgebiete. In Abbildung 1.2 hat der Transformator auf der Mittelspannungsseite zwei Leitungen, eine mit einer maximalen Belastbarkeit von $250kVA$ und eine mit einer maximalen Belastbarkeit von $150kVA$. Im Transformator sind sie über eine sogenannte *Sammelschiene* verbunden. Der Laststatus des Niederspannungsnetzes kann sich auf Grund folgender Umstände ergeben: Angenommen, es liegt ein klarer Frühlingtag mit viel Sonnenschein vor, dh. es kann ein hoher Globalstrahlungswert auf der Erdoberfläche gemessen werden. Die Temperatur sei für das menschliche Empfinden angenehm. Angenommen, es gab vor diesem Tag eine längere Niederschlagsperiode in Form von Regen. Im Niederspannungsnetz seien WLKW und PVA als Einspeiser vorhanden. Der Verbrauch im Niederspannungsnetz ist auf Grund abgeschalteter Heizungen, abgeschalteten Lichtquellen sowie noch nicht eingeschalteten Klimaanlage und diversen anderen Gründen niedrig. Der Verbrauch kann aus der Lastprognose abgeschätzt werden. Tatsächlich sei er hier $50kVA$. Die Leistung kann auf Grund der aktuellen und vergangenen Wetterlage hoch bzw. für die vorliegenden Anlagen auch maximal sein und sei hier tatsächlich $220kVA$.

- Fall 1: Dies ist ein sehr häufig auftretender Fall. Impedanz Z in Abbildung 1.2 sei für diesen Fall hochohmig. Auf Grund von Netzinstandhaltungs- oder Netzausbauarbeiten, müssen Leitungen, wie z.B. Leitung 1 in Abbildung 1.2, spannungsfrei geschaltet werden. Niederspannungsnetze können dabei weiterhin über andere Zuleitungen,

z.B. Leitung 2, versorgt werden. Betrachtet man aber den aktuellen Lastzustand des Niederspannungsnetzes in Abbildung 1.2, erkennt man schnell die Problematik die entsteht, wenn unter dem aktuellen Lastzustand Leitung 1 abgeschaltet werden würde. Vor der Abschaltung kann die Last, in diesem Fall ein Energieüberschuss von $220kVA - 50kVA = 170kVA$, bereits alleine durch Leitung 1 abgeführt werden, da in diesem Beispiel für sie eine maximale Belastbarkeit von 250kVA gelten soll. Wie sich der Lastfluss genau über Leitung 1 und 2 vor der Abschaltung von Leitung 1 aufteilt, sei hier unbedeutend. Nach erfolgter Abschaltung von Leitung 1, ist aber nur noch Leitung 2 als Zuleitung zum Niederspannungsnetz vorhanden. Für sie gelte in diesem Beispiel eine maximale Belastbarkeit von 150kVA. Sie wird nach der Abschaltung von Leitung 1 überlastet sein. Weitere Sicherheitsmechanismen werden eingreifen, und können bis zur kompletten Abschaltung der Energieversorgung zum und im Niederspannungsnetz führen.

- Fall 2: Dies ist ein eher selten auftretender Fall. Der Schalter in Abbildung 1.2 sei für diesen Fall geschlossen. Auf Grund eines Fehlers auf der Leitung kann es vorkommen, dass über eine niederohmige Impedanz Z ein Kurzschluss entsteht. Für diese Fehlerfälle sind Schutzmechanismen bei Leistungsschalter auf der Leitung gegen Überlast vorhanden, in Abbildung 1.2 mit S bezeichnet. Die Impedanz Z sei hier gerade so niederohmig, dass die Schutzvorrichtung ansprechen würde, wenn keine DEA im Mittelspannungsnetz vorhanden wären. Durch bestimmte Lastzustände im Mittelspannungsnetz kann es nun vorkommen, dass auf Grund der DEA, die gerade mit einer hohen Einspeisung fahren, der Strom aus dem Niederspannungsnetz ausreicht, um den Strom über den Kurzschluss zu decken. Der Kurzschluss wird unter diesen Umständen nicht erkannt, und stellt eine große Gefahr für Menschen und Umwelt im Bereich des Fehlerfalls dar.

Fall 1 ist mit einer qualitativen kurzfristigen Leistungsprognose von DEA von vorn herein vermeidbar, da Arbeiten an Leitungen auf Zeitpunkte verschoben werden können, bei denen die Leistung von DEA im Niederspannungsnetz niedriger ist. Fall 2 tritt nur unter bestimmten Bedingungen ein, stellt aber eine große Gefahr für Menschen und Umwelt dar. Lastzustände die im Falle eines Kurzschlusses diese Gefahr auftreten lassen würden, könnten aber mit zusätzlicher kurzfristiger Leistungsprognose von DEA frühzeitig erkannt werden.

1.5 Gliederung der Arbeit

In dieser Arbeit ist folgende Gliederung anzufinden:

- Kapitel 2 macht eine Analyse der Daten, die seitens des Verteilernetzbetreibers *KE-LAG Netz GmbH* vorhanden sind. Beispielhafte Datensätze werden vorgestellt und es werden erste Erkenntnisse aufgezeigt, sowie Vorgehensweisen für die Modellierung der Leistungsprognose von WLKW und PVA getroffen.
- Kapitel 3 zeigt aktuelle Methoden gleicher und analoger Problemstellungen auf. Hierbei werden verwandte Arbeiten präsentiert und erste Modellansätze aufgezeigt.

- Im Kapitel 4 werden notwendige Grundlagen der zu untersuchenden Modelle angeführt. Die Grundlagen stellen soweit eine Basis dar, um damit den Modellansatz nachvollziehen zu können.
- Kapitel 5 und Kapitel 6 präsentieren die Umsetzung bzw. Simulation der zu untersuchenden Modelle. Der Vorgang der Implementierung und Ergebnisse von Simulationen sowie Beobachtungen und Versuche mit den Modellen werden beschrieben.
- Kapitel 7 beinhaltet Schlussbemerkungen und eine Diskussion über die Aufarbeitung der Modelle. Außerdem wird ein Ausblick über mögliche zukünftige und auf diese Arbeit aufbauenden Untersuchungen gegeben.

Kapitel 2

Untersuchungen zur vorhandenen Datensituation

Die Netzbetreiberfirma *KELAG Netz GmbH* in Kärnten/Österreich hat neben Messdaten über Kraftwerke auch meteorologische Messstationen. Die WLKW sind nahezu alle online gemessen oder besitzen einen Lastprofilzähler, der jeden Tag einmal die $\frac{1}{4}h$ Leistungswerte ($P[MW]$) des gesamten Tages in das Datenmanagementsystem überträgt. Die dabei übermittelten Werte eines Tages umfassen den Zeitraum von 00:15 - 00:00 Uhr. Die Zeitreihen liegen dabei von jedem Kraftwerk bis ungefähr zum Beginn des Jahres 2007 zurück vor. Die meteorologischen Messstationen liefern im $\frac{1}{4}h$ Takt Niederschlagswerte [mm], Temperaturwerte $T[^\circ C]$ und Solarstrahlungswerte $E[\frac{W}{m^2}]$. Es gibt Daten von 25 Niederschlagsmessstationen, wobei mehr als die Hälfte davon z.B. bei Speicherkraftwerken im Gebirge aufgebaut sind, und daher nicht für andere Kraftwerke verwendet werden können. Von den Temperaturmessstationen sind 10 Stellen verfügbar, wobei 2 davon wiederum hoch gelegen sind, und daher für umliegende Talschaften nicht gültig sind. Es sind weiters 5 Solarstrahlungsmessstationen von horizontal ausgerichteten Sternpyranometern vorhanden. Zusätzlich sind Messwerte von einem in der Neigung der Paneele angebrachten Sternpyranometer bei einer großen online gemessenen PVA in *St. Veit* vorhanden.

Momentan liefert der Wetterdienst der Zentralanstalt für Meteorologie und Geodynamik (*ZAMG*) für Kärnten in sieben Regionen unterteilt, im $6h$ Raster für $48h$ in die Zukunft, immer von Mitternacht weg betrachtet, die zu erwartenden Niederschlagswerte in mm (integriert über $6h$), sowie die zu erwartenden Umgebungstemperaturwerte in $^\circ C$ (zu erwartenden Momentanwerte). Für die Solarstrahlung in $\frac{W}{m^2}$, wird lediglich für Ober- und Unterkärnten (Ost- bzw. Westkärnten) ebenfalls im $6h$ Raster, $48h$ in die Zukunft beginnend mit Mitternacht, der Bedeckungsgrad prognostiziert. Dies ist ein Wert der 8 mögliche Stufen hat, 1 – 8, und logarithmisch die extraterrestrische Solarstrahlung auf die Horizontale, siehe Kapitel 4.1.2 und 4.1.3, skaliert. Der Bedeckungsgrad ist ein mittlerer Wert für Ober- bzw. Unterkärnten.

2.1 Datenanalyse für die physikalische Modellierung von Photovoltaikanlagen

Die im Kapitel 2 erwähnte gemessene PVA ist die einzige umfangreich messtechnisch erfasste Anlage im Stromnetz des Netzbetreibers. Von ihr sind die Leistungswerte $P[kW]$, die Solarstrahlung $E[\frac{W}{m^2}]$ auf die Paneele (in der Neigung gemessen), die Paneeltemperatur $T[^\circ C]$ und die Umgebungstemperatur $T[^\circ C]$ im $\frac{1}{4}h$ Takt vorhanden. Von dieser Anlage sind außerdem die Größe $[m^2]$, der Wirkungsgrad $[\%]$, die Wirkungsgradreduktion $[\frac{\%}{\circ K}]$ in Abhängigkeit der Temperatur, die Wechselrichterwirkungsgrade, die Abweichung der Ausrichtung nach Süden und die Neigung der Paneele bekannt. Alle anderen PVA, die am Netz hängen, sind ungemessen und es sind lediglich die Jahressummenwerte sowie die kW-Peak Leistung bekannt. Über Wirkungsgrade, Ausrichtung, Neigung usw. gibt es keine Aufzeichnungen, da diese Anlagen meist von privaten Betreibern geführt werden. Von der gemessenen PVA liegen zum aktuellen Zeitpunkt Zeitreihen der Messungen lediglich 3-4 Wochen im Mai vor. Daher muss für die Modellierung der PVA ein physikalisches Modell gebildet werden. Die gemessene PVA kann dabei als Referenzanlage dienen, um das Modell zu verifizieren. Abbildung 2.1 zeigt die von der gemessenen Anlage in *St. Veit* vorhandenen Zeitreihen der Messdaten. Der direkte Zusammenhang zwischen Solarstrahlung auf die geneigte Fläche und der Leistung der Anlage ist deutlich zu erkennen. Der Zusammenhang der Strahlungen auf die Horizontale und auf die geneigte Ebene kann über ein statistisches Modell beschrieben werden, siehe [Qua96, Kapitel 3]. Vor allem geht es darum, die diffuse Strahlung aus der Summe der Strahlung (diffuse + direkte Strahlung) auf die Horizontale, auf die Strahlung der geneigten Fläche umzurechnen, da sich die diffuse Strahlung nicht einfach über den Winkel umrechnen lässt, und stark vom Sonnenstand zur Fläche abhängt, [Qua96, Kapitel 3]. Außerdem ist die direkte Abhängigkeit der Temperatur der Paneele zur Solarstrahlung und der Umgebungstemperatur zu erkennen. Diese Temperatur lässt sich auch über ein statistisches Modell aus diesen beiden Einflussgrößen errechnen, siehe [Bla06, Kapitel 2] und [Sch02, Kapitel 2]. Die Paneeltemperatur ist wichtig, da sie den Wirkungsgrad verändert, [Bla06].

Auf Grund dieser *einfachen* Abhängigkeiten kann ein physikalisches Modell mit den vorhandenen Daten erstellt werden.

2.2 Datenanalyse für Regressionsmodelle für kleine Wasserkraftanlagen

Im Stromnetz des Netzbetreibers *KELAG Netz GmbH* gibt es unterschiedliche Wasserkraftanlagen, die verteilt über das Bundesland Kärnten an Flüssen und Bächen liegen, aber auch künstlich angelget wurden, wie diverse Speicherseen. Zum einen ist die Unterscheidung zwischen Laufkraftwerk und Speicherkraftwerk zu treffen. Zum anderen ist die Größe eine Möglichkeit der Charakterisierung. In dieser Arbeit sind, wie in der Aufgabenstellung (Kapitel 1) beschrieben, nur WLKW zu betrachten. Es liegt die Vermutung nahe, dass die wichtigsten Einflussfaktoren *Temperatur und Niederschlag* auf WLKW direkt und unverzögert eingehen. Im Folgenden ist zu sehen, dass dies so nicht ganz korrekt ist und wesentlichen Einfluss auf die Konzeptionierung der Modellierung für eine Leistungsprognose haben wird.

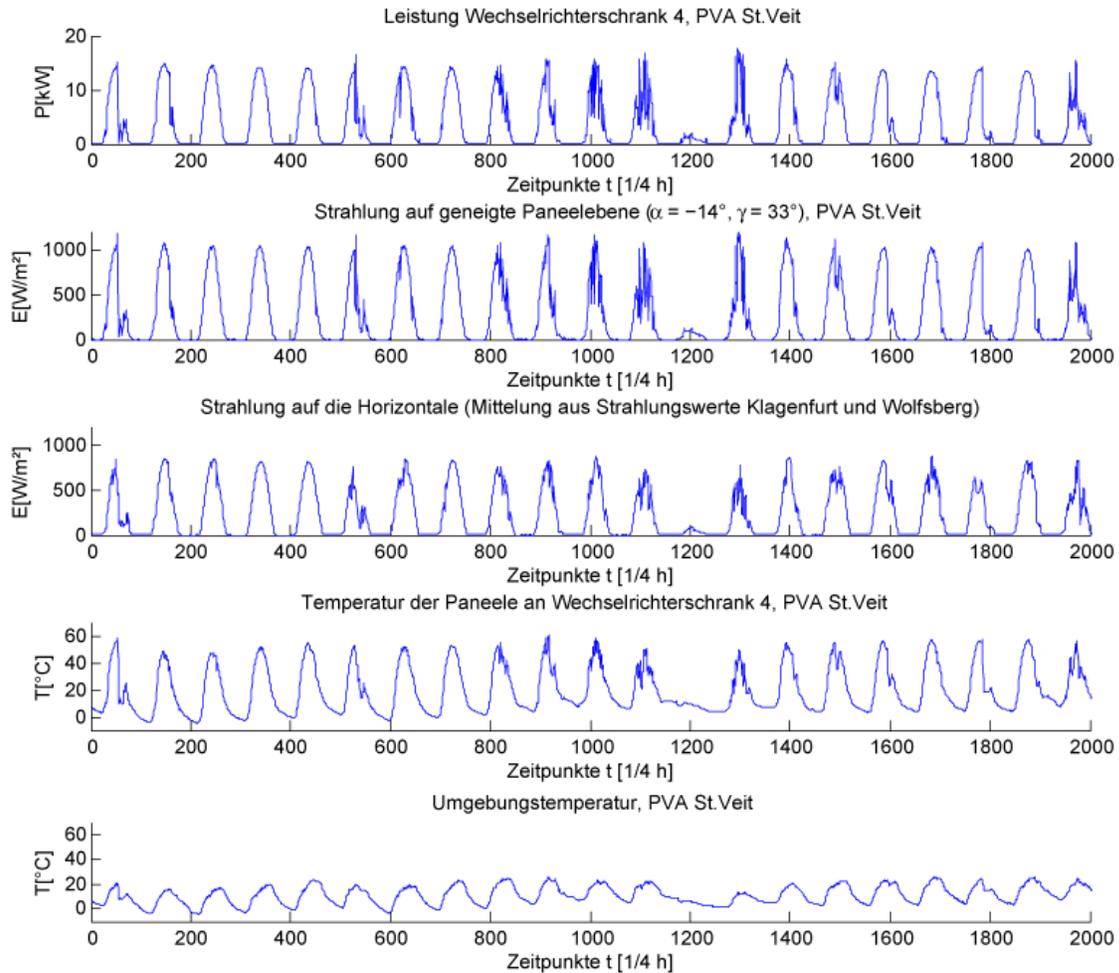


Abbildung 2.1: Zeitreihen über Leistung, Solarstrahlungen, Peneeltemperatur und Umgebungstemperatur der PVA in *St. Veit*.

Die Einflussfaktoren wie Bodenbeschaffenheit, Fallhöhe des Wassers, Einzugsgebiet des Niederschlags und Wirkungsgrad sind statische Größen und nur sehr aufwändig physikalisch erfassbar. Außerdem fehlen dazu diverse Messdaten. Sie fließen also statisch in die Ausgangsgröße *Leistung* ein. Einflussfaktoren wie Feuchtigkeitsgrad des Bodens, Grundwasserspiegel und andere dynamische Umweltbedingungen, hängen indirekt von Temperatur und Niederschlag, sowie deren Vergangenheit ab. Man kann also die zeitlichen Verläufe der zur Region zugehörigen Temperatur- und Niederschlagswerte als Haupteinflussfaktoren für das Verhalten der WLKW bestimmen. Von den jeweiligen WLKW liegen Messreihen der erzeugten Leistung vor, die zum Teil mehr als drei Jahre in die Vergangenheit zurück in einer Datenbank vorhanden sind. Jeder Wert beschreibt hierzu die gemittelte Leistung pro $\frac{1}{4}h$. Abbildung 2.2 zeigt beispielhaft jeweils eine Jahreszeitreihe, über die Leistung von zwei WLKW mit der Bezeichnung *Kirchbach* und *Radegund*. Jedes WLKW besitzt folgende für diese Arbeit charakteristischen Daten:

- Engpassleistung: ist die maximale elektrische Leistung, die ein Kraftwerk im Dauer-

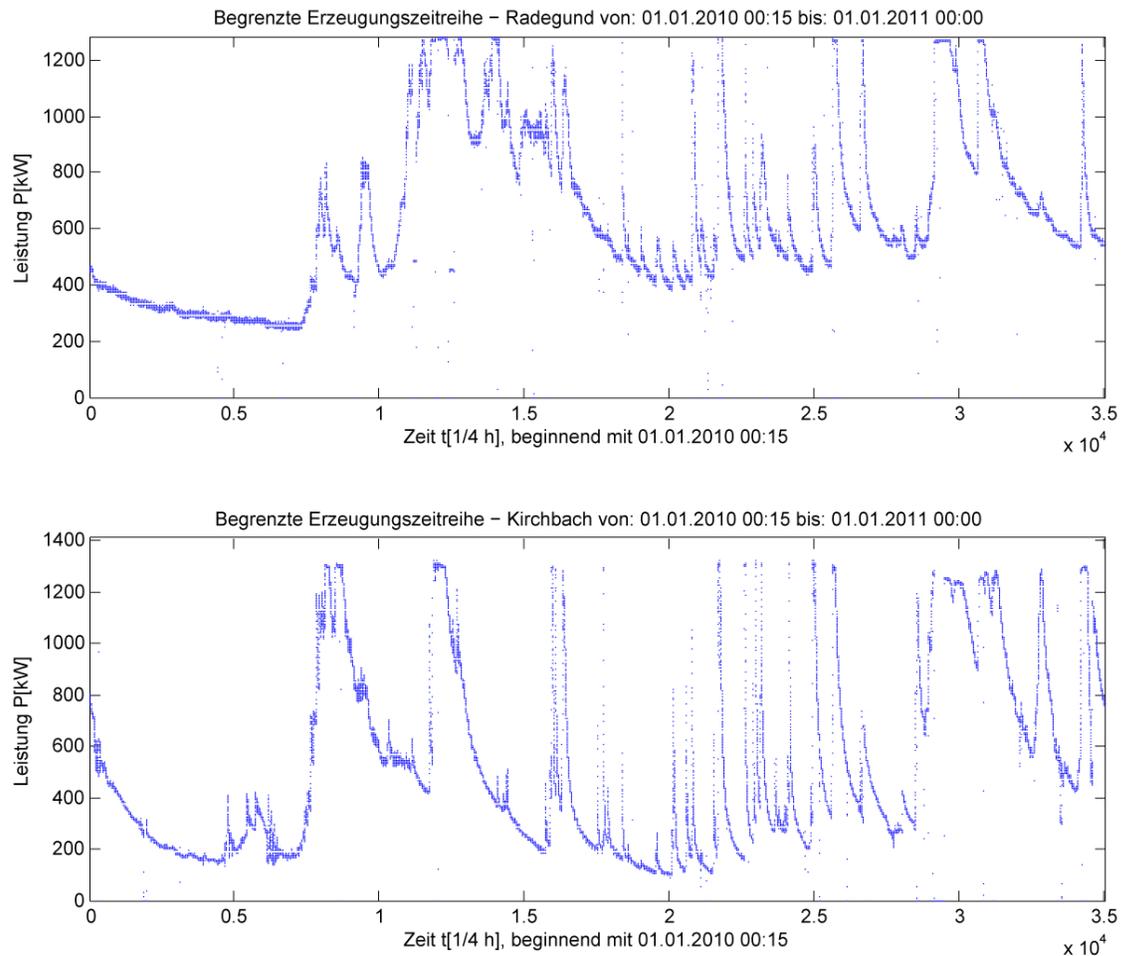


Abbildung 2.2: Jahreszeitreihen der erzeugten Leistung von *Kirchbach* und *Radegund* für das Jahr 2010. Datenpunkte sind $\frac{1}{4}h$ Mittelwerte der Leistung in *kW*.

betrieb unter Normalbedingungen abgeben kann. Wird in der Regel mit der Einheit *MW* angegeben.

- **Regelarbeitsvermögen:** gibt die durchschnittliche elektrische Energie an, die ein Kraftwerk in einem Jahr umwandelt. Wird in der Regel mit *Mio.kWh* angegeben.

Andere Daten, wie Bruttofallhöhe, Jahr der Inbetriebnahme, Turbinenanzahl, etc. sind für die Vergleichbarkeit für diese Arbeit nicht relevant. Die Kraftwerke *Kirchbach* und *Radegund* weisen beispielhaft folgende Daten auf:

- *Radegund:*
 - Engpassleistung: 1.41 *MW*
 - Regelarbeitsvermögen: 6.72 *Mio.kWh*
- *Kirchbach:*

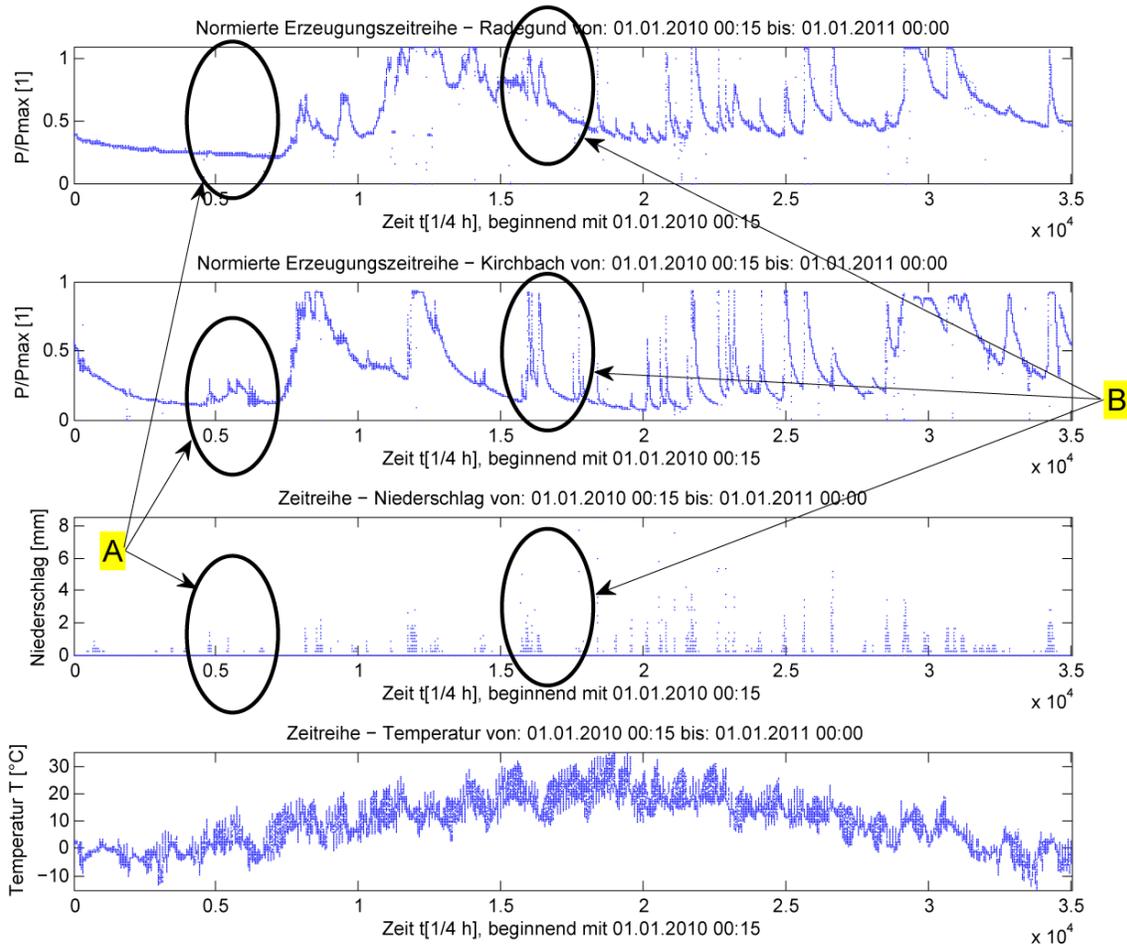


Abbildung 2.3: Jahreszeitreihen vom Jahr 2010 für *Kirchbach* und *Radegund* mit zugehörigem Jahresniederschlags- und Jahrestemperaturverlauf.

- Engpassleistung: 1.172 MW
- Regelarbeitsvermögen: 6.22 Mio.KWh

Die in Abbildung 2.2 dargestellten Zeitreihen wurden bereits von Ausreißer bereinigt, indem jene Werte, die weit über der Engpassleistung liegen, mit dem Wert der Engpassleistung ersetzt werden. Ihr Zustandekommen ist meist unbekannt. In den beiden Zeitreihen in Abbildung 2.2 waren jeweils weniger als 10 Ausreißer enthalten. Insgesamt hat eine Jahreszeitreihe (Jahr ohne Schalttag) eines WLKW $4 \times 24 \times 365 = 35040$ Datenpunkte. Die WLKW *Kirchbach* und *Radegund* liegen geographisch, bezogen auf die Größe der zugehörigen Region, relativ weit voneinander entfernt, haben aber annähernd den selben Wettereinfluss, da sie in der selben Wetterregion *Gailtal* liegen. Sie liegen nicht am selben Gewässer. Eine Wetterregion wird durch das Raster der Wetterprognose gebildet. Der Einfluss von Gewitterschauer sei hier vorerst ausgenommen. Abbildung 2.3 zeigt dazu die auf die jeweilige Engpassleistung normierten Zeitreihen und die meteorologischen Daten *Temperatur* und *Niederschlag*. Bei einer ersten Analyse fallen folgende interessante Aspekte

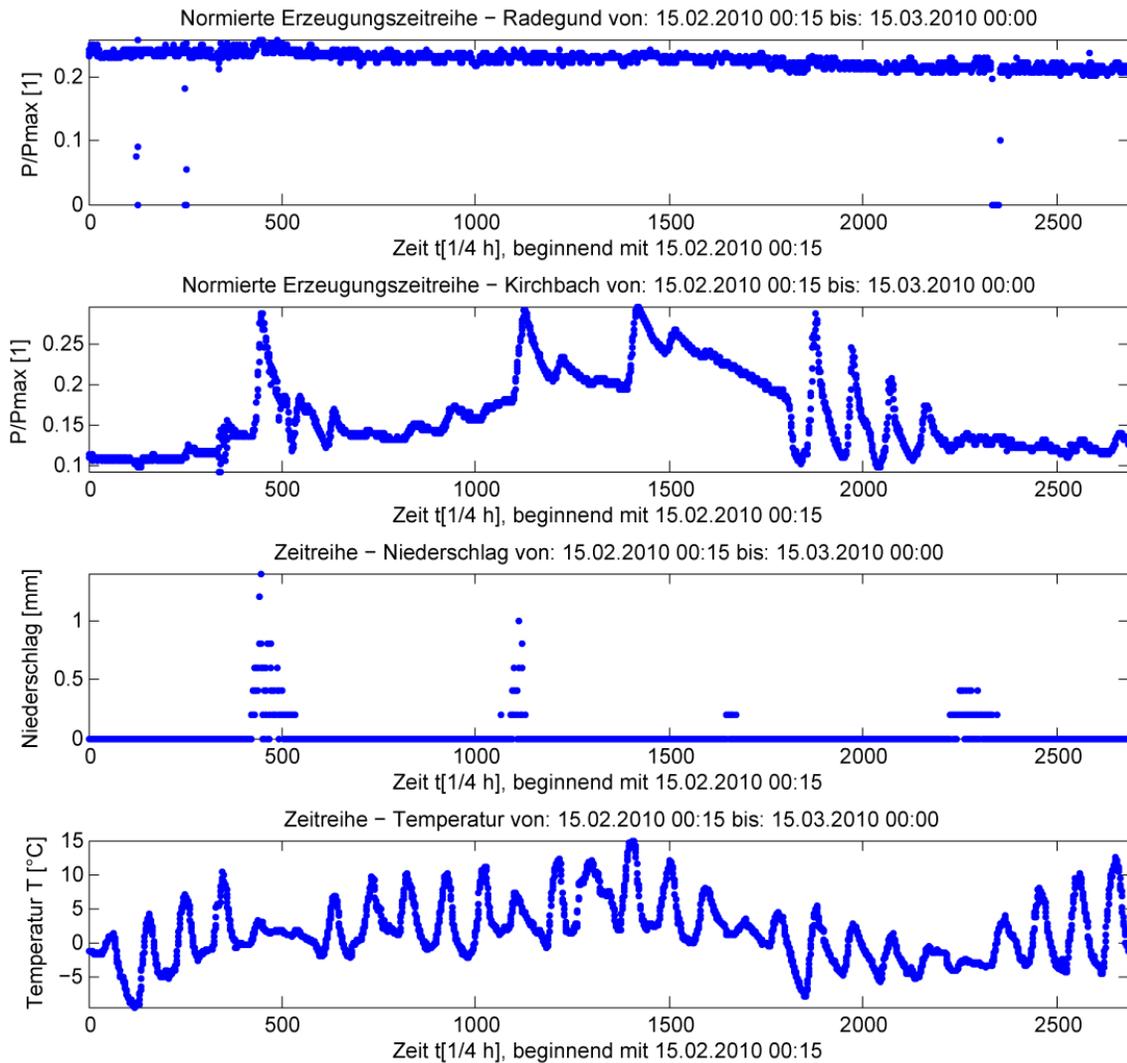


Abbildung 2.4: Hervorgehobener Bereich A aus Abbildung 2.3.

auf:

- Es gibt offensichtlich ein speicherndes Verhalten der Umwelt. Nach einem Niederschlag, klingt der Wasserzufluss (proportional zur Leistung) exponentiell ab.
- Es gibt ein deutlich unterschiedliches Verhalten beim Vergleich der Verläufe in den Sommer- und Wintermonaten, siehe markierte Bereiche mit der Bezeichnung A und B. Vor allem bis zu den ungefähr ersten 7500 Datenpunkten gibt es ein deutlich anderes Verhalten, als bei den Datenpunkten danach.
- Der Einfluss des Niederschlags kommt nach einer längeren Frostperiode teils komplett zum Erliegen, da der Niederschlag offensichtlich in Form von Schnee *gespeichert* wird. Es tritt aber ein anderes Verhalten beim WLKW *Kirchbach* als bei *Radegund* auf. In Abbildung 2.4 ist der Bereich A aus 2.3 hervorgehoben. *Kirchbach* zeigt hier-

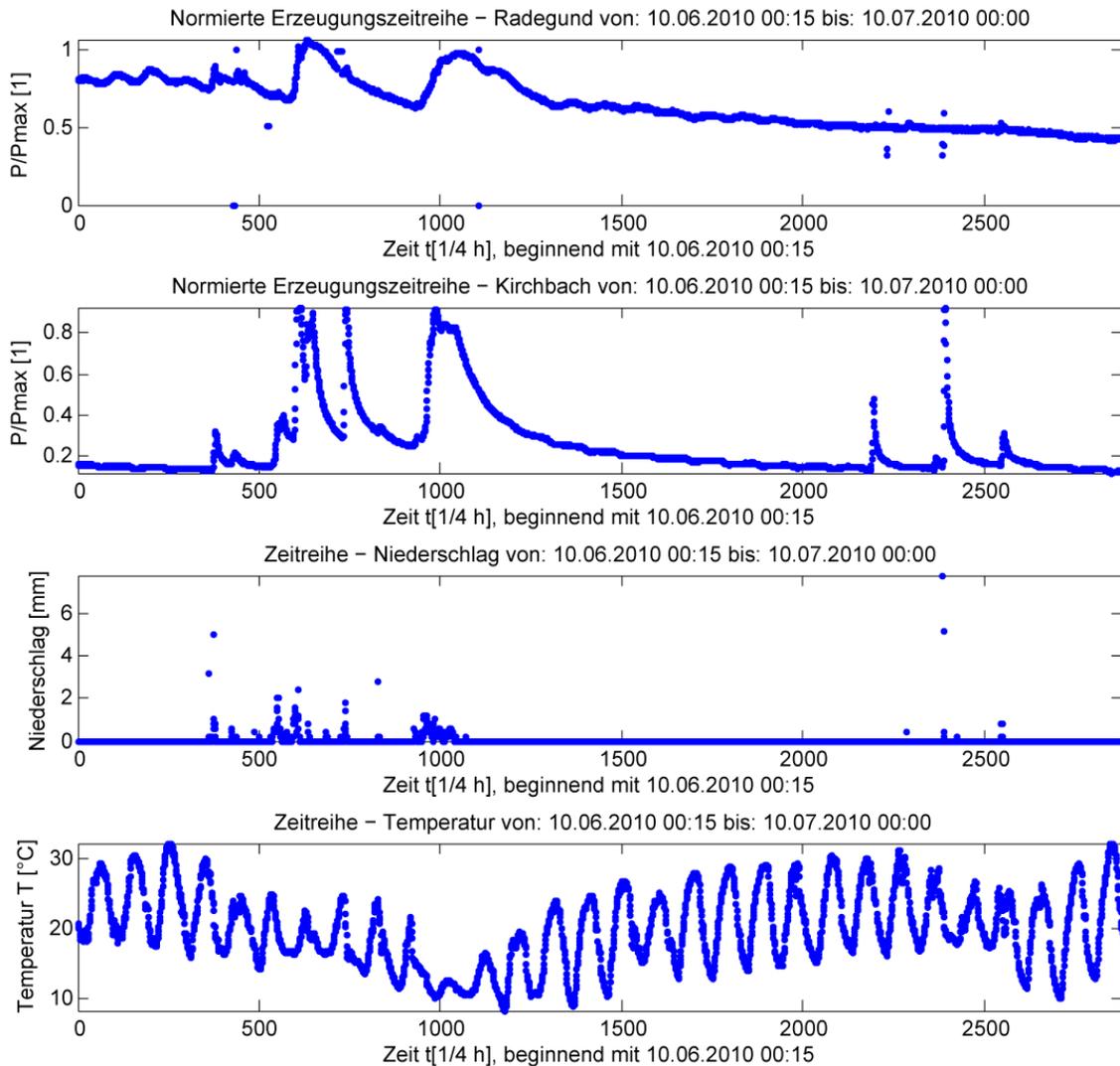


Abbildung 2.5: Hervorgehobener Bereich *B* aus Abbildung 2.3.

bei eine sofortige Reaktion auf Niederschlag, *Radegund* nicht. Wobei der Einfluss vom Niederschlag deutlich geringer ist und maximal $\frac{1}{4}$ gegenüber des Einflusses in den Sommermonaten beträgt.

- In den Sommermonaten gibt es eine sofortige Reaktion auf Niederschlag in Form eines Leistungsanstiegs beider Kraftwerke. Abbildung 2.5 zeigt hierbei den hervorgehobenen Bereich mit der Bezeichnung *B* aus Abbildung 2.3. Der Einfluss des Niederschlags ist bei beiden Kraftwerken groß.
- Der Verlauf in Abbildung 2.5 zeigt nun, dass die beiden Kraftwerke ein ähnliches aber nicht gleiches Verhalten aufweisen. Beide reagieren unmittelbar auf Niederschlag mit hohem Leistungsanstieg, aber das Abklingen der Leistung als Folge des Abklingens des Wasserzufflusses ist deutlich unterschiedlich. Auch der Verlauf des

Leistungsanstiegs bei Niederschlag ist verschieden. Die unterschiedlichen *Antworten* auf die *Erregung* weisen auf grundverschiedene Einflussfaktoren bzw. zeitliches Verhalten der Umwelt des Kraftwerks hin. Nennen wir hier die Umwelt und das Kraftwerk *Strecke*. Sind die beeinflussenden Parameter einer Strecke nicht bekannt, so kann man das Verhalten einer Strecke über die Sprungantwort beschreiben.

Unabhängig von der Lage der beiden Kraftwerke ist bereits jetzt zu erkennen, dass eine Parameterbestimmung je Kraftwerk notwendig sein wird.

Bereits an diesem Punkt kann gesagt werden, dass ein und die selbe Modellierungsart für alle WLKW gewählt werden kann (wie z.B. lineare Regression, multilayer Perzeptron (MLP) basierte neuronale Netze oder rekurrente neuronale Netze (RNN) wie z.B. Echo State Netze). Es ist aber eine Parameterbestimmung, bzw. im Falle von Lernverfahren ein Training pro Kraftwerk notwendig.

Es kann also nicht ein aus den normierten Leistungverläufen der einzelnen Kraftwerke einer Region *gemitteltes Ersatzkraftwerk* gebildet werden, um pro Wetterregion ein Modell mit definierten Parameterwerten bzw. einem Training zu erhalten. Ein Modell über ein gemitteltes Ersatzkraftwerk würde für gegebene Inputvektoren einen normierten Wert prognostizieren, der je nach charakteristischen Kraftwerksparametern (Engpassleistung) zurücknormiert werden könnte, um so den prognostizierten realen Leistungswert vom gewünschten Kraftwerk zu erhalten. Ein Vorteil daraus wäre, dass bei einem neu hinzukommenden Kraftwerk in einer Region sofort Prognose betrieben werden könnte, ohne zuvor zeitliche Verläufe für einen Trainingsdatensatz für das neue Kraftwerk aufzeichnen zu müssen. Das Prinzip des gemittelten Ersatzkraftwerks ist so nicht anwendbar, und würde zu große Fehler bei der Prognose erzeugen. Um das Problem mit neu hinzukommenden Kraftwerken in einer Region zu lösen, gäbe es den Ansatz, dass man sich ein Kraftwerk am selben Gewässer in der Nähe sucht, und das bekannte Verhalten als Verhalten für das neue Kraftwerk annimmt. Es kann also das trainierte bzw. parameter-bestimmte Modell eines Kraftwerks mit ähnlicher Streckencharakteristik vorläufig verwendet werden, bis eine genügend lange Zeitreihe für das neue Kraftwerk vorhanden ist, um ein eigenes Modell zu trainieren.

Eine einfache statistische Analyse gibt weiteren Einblick in die Zusammenhänge der als Inputparameter angenommenen Größen Temperatur und Niederschlag auf die beispielhaften WLKW. Abbildung 2.6 zeigt die Kreuzkorrelationsfunktion zwischen den Jahreszeitreihen der Kraftwerke *Kirchbach* und *Radegund*, sowie jeweils die Kreuzkorrelationsfunktion der Kraftwerke zur Temperatur und zum Niederschlag. Die Zeitreihen wurden dabei so normiert, dass sich bei identischem Verhalten (nur durch einen Skalierungsfaktor unterschiedlicher Verlauf) eine Korrelation 1 und bei absoluter Unkorreliertheit 0 ergibt.

Analysiert man die Kreuzkorrelationsfunktion in den Abbildungen 2.6, so fallen folgende interessante Aspekte auf:

- Es gibt zwischen den Kraftwerken *Kirchbach* und *Radegund* ein ähnliches Verhalten, aber auf keinen Fall gleiches Verhalten, dh. der Unterschied ist komplexer. Es gibt ein relativ hohes Maximum aber keine deutliche Spitze.
- Die Kraftwerke zeigen eine überraschend niedrige Übereinstimmung mit dem Niederschlagsverlauf, obwohl eine starke Übereinstimmung in Abbildung 2.3 zu erkennen ist. Dies hat den Grund, dass die beschriebene speichernde Funktion der Strecke

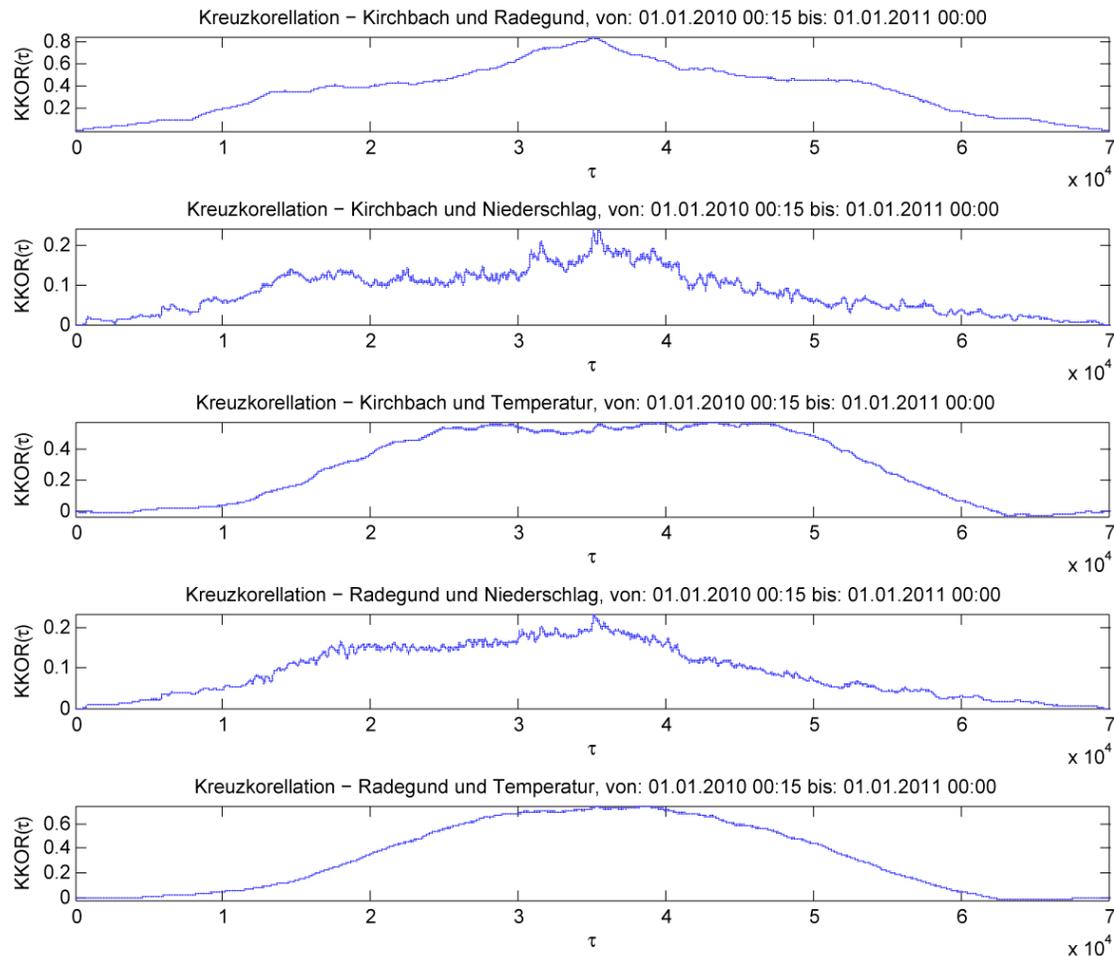


Abbildung 2.6: Kreuzkorrelationsfunktionen der Jahreszeitreihen 2010 der Kraftwerke *Kirchbach* und *Radegund* mit den Jahreszeitreihen der zur Region zugehörigen Temperatur und des Niederschlags.

(Umwelt) der Kraftwerke und das dadurch auftretende *Nachklängen* bei Niederschlag die Korrelation geringer werden lässt.

- Eine Korrelation mit der Temperatur ist bei beiden Kraftwerken vorhanden und hat einen langgezogenen Peak. Dies weist darauf hin, dass die Temperatur einen langanhaltenden Einfluss auf das Systemverhalten hat.

Wichtige Erkenntnisse an dieser Stelle sind:

- Die Modelle, die für die Prognose gebildet werden sollen, bzw. in Frage kommen, müssen die Vergangenheit mitberücksichtigen (können). Dies ergibt sich aus der Erkenntnis, dass die Strecken (die Umwelt) der Kraftwerke speicherndes Verhalten bezüglich Niederschlag und Temperatur aufweisen.
- Bei der Temperatur als Einfluss für die Prognosemodelle, muss zeitlich gesehen ein

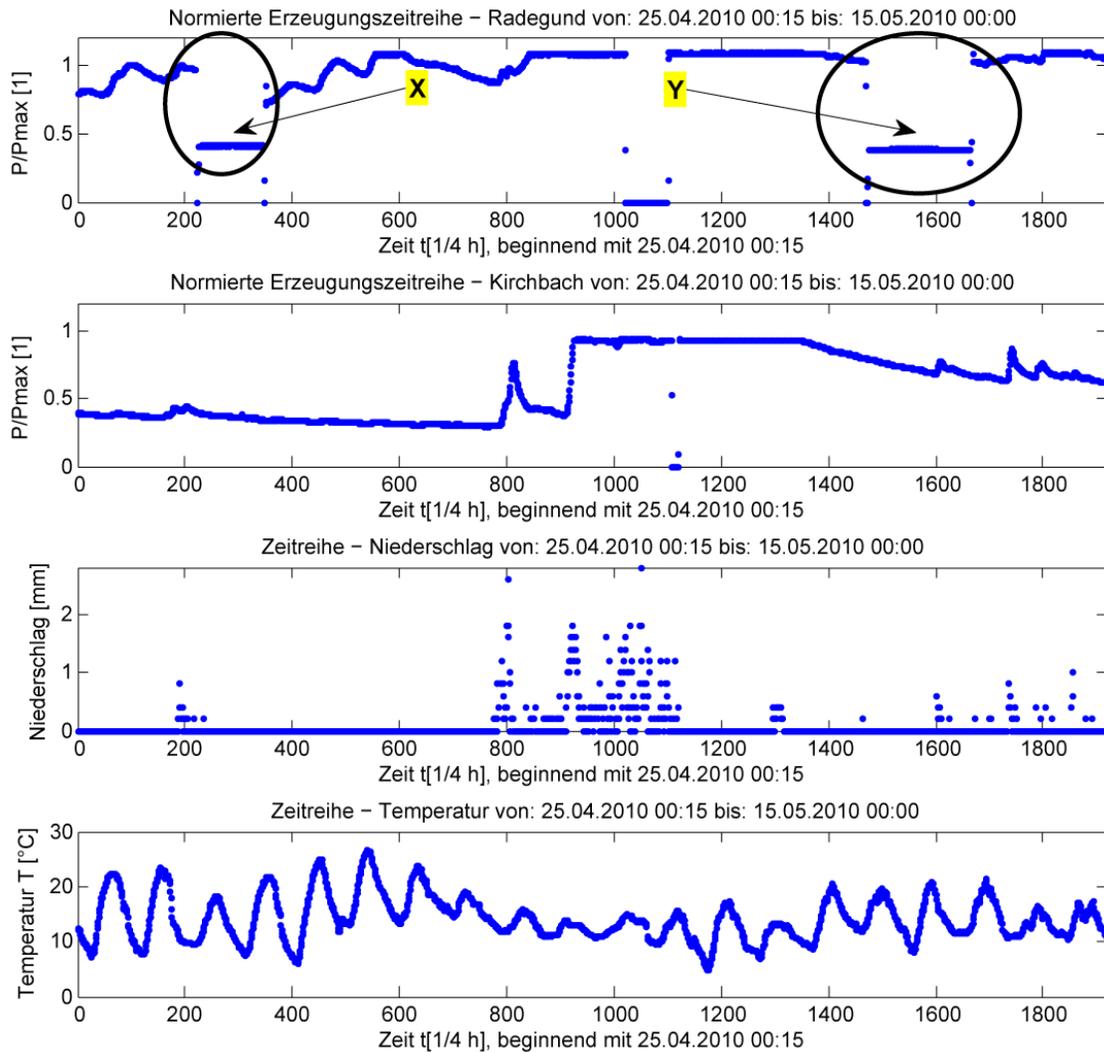


Abbildung 2.7: Ausschnitte der Zeitreihen aus dem Jahr 2010 zur Demonstration von Datenanomalien.

größeres Vergangenheitsfenster angewendet werden, als für die Betrachtung des Niederschlags.

2.2.1 Datenbereinigung

Neben der bereits erwähnten Bereinigung der Ausreißer, ist eine Datenbereinigung durch Tiefpassfilterung nicht geeignet, da bei zu starker Glättung wichtige Verhaltensmuster und somit Informationen über das Verhalten der Stecke eines Kraftwerks verloren gehen könnten.

In den Datensätzen kommen zum Teil Bereiche auf Grund unerklärter Ursachen vor, wie es in Abbildung 2.7 vom Kraftwerk *Radegund* zweimal hervorgehoben wird. Die Anomalien dauern ca. einen Tag (Bereich X) bzw. zwei Tage (Bereich Y) an. Es wurde hierzu ein Verlauf von ca. drei Wochen aus der Zeitreihe von 2010 im Frühling (25.04.2010 - 15.05.2010)

herausgehoben. Hier könnte ein Einflussfaktor zu tragen gekommen sein, der in der Regel nichts mit dem System des Kraftwerks zu tun hat (öffnen der Schleusen auf Grund einer Hochwassersituation, Reparaturen, kurze Abschaltung, Messfehler, etc.). Würde man z.B. in Abbildung 2.7 in den markierten Bereichen glätten, würde dieser *Fehler* etwas *versteckt* werden, aber durch Filterung höherer Frequenzen könnten Charakteristika beim Anstieg der Leistung auf Grund von Niederschlag verloren gehen. Auch eine manuelle Korrektur ist nicht ratsam, da ohne weitere Kenntnisse das Verhalten der Leistung in diesen Abschnitten gänzlich unbekannt ist. Die Auswirkungen auf den Lernerfolg durch solche Bereiche müssen durch genügend lange Vergangenheitszeitreihen kompensiert werden. Die Temperatur- und Niederschlagsverläufe sind durchwegs sehr gut und können direkt übernommen werden.

2.2.2 Methodeneingrenzung

Auf Grund der Datenanalyse kann man bei WLKW eine Eingrenzung der Methoden zur Modellierung eines Prognosemodells vornehmen. Die erste Frage, die bei der Erstellung von regressionsbasierten Prognosemodellen auftritt ist jene, ob man mit Hilfe der Zeitreihenanalyse einen stochastischen Prozess unterstellen kann, um dann mit AR (Auto Regressive/Auto Regression), MA (Moving Average/Gleitender Durchschnitt), ARMA (Auto Regressive-Moving Average) oder ARIMA (Auto Regressive-Integrated-Moving Average) Modellen die Zeitreihe erzeugen und somit eine Prognosemethode erstellen zu können.[SH01]

In diesem Fall ist es nicht sinnvoll, ein Modell mit einer reinen Zeitreihenanalyse zu erstellen, da mit Hilfe der Zeitreihenanalyse rein auf Grund des vergangenen Verhaltens der Zielgröße selbst auf das zukünftige Verhalten geschlossen wird. Es liegt aber eine eindeutige Abhängigkeit von Parametern wie Niederschlag und Temperatur vor, die ausschließlich das Verhalten der Zielgröße bestimmen.

Es kommen daher Regressionverfahren in Frage. Zu den Regressionverfahren kann man in gewisser Weise auch maschinelle Lernverfahren zählen. Neben der klassischen linearen Regression, können neuronale Netze und Erweiterungen davon angewendet werden, um das geforderte Prognosemodell erstellen zu können. Methoden, die in Fragen kommen würden, sind:

- Lineare Regression,
- Neuronale Netze wie MLP basierte neuronale Netze (mit der *windowing* Methode),
- RNN wie das Elman Netz oder Echo State Netze und
- Support Vector Regression (SVR) als Abwandlung einer Support Vector Machine (SVM) zur Klassifikation.

Kapitel 3

Verwandte Forschungsthemen und Projekte

Wie die Analyse der gegebenen Daten der Netzbetreiberfirma *KELAG Netz GmbH* in Kapitel 2 zeigt, kann für unterschiedliche Kraftwerkstypen nicht nur ein einziges Konzept zur Modellierung der kurzfristigen Leistungsprognose verwendet werden. Es sind in dieser Arbeit die Kraftwerkstypen Photovoltaik und Wasser-Laufkraft für die kurzfristige Leistungsprognose zu modellieren. Es gibt prinzipiell zwei unterschiedliche Herangehensweisen. Auf der einen Seite kann die Leistung eines Kraftwerks über ein physikalisches Modell voraus berechnet werden. Dazu werden externe Parameter, wie Sonnenstand zur Stunde h , vorhergesagter Bedeckungsgrad u.a., in einem mathematischen Zusammenhang formuliert. Dieser Formalismus ist eine Abbildung, die Parameter benützt, welche nicht von der Vergangenheit abhängig sind.

Im Gegensatz zu einem regressionsbasierten Modell, wie in [FKL09] beschrieben, bei dem ein systematischer Einfluss der erklärenden Variable x , z.B. Wetterdaten, auf die primär interessierende Zielvariable y , z.B. Leistung, vorliegt. [FKL09] legt dar, dass erklärende Variablen dabei als sog. *Regressoren oder Kovariablen* und die Zielvariablen als *abhängige* oder *zu erklärende Variable* bezeichnet werden. Die Abbildung der Regressoren auf die Zielvariable erfolgt dabei linear bzw. nichtlinear. Im linearen Fall werden die Regressoren über eine lineare Funktion mittels (geschätzter) Parameter auf die Zielvariable abgebildet. Im nichtlinearen Fall über eine nicht lineare Funktion. Wie die Abbildung im Speziellen aussieht, und welche Werte die Parameter annehmen, hängt von der Vergangenheit ab, dh. für ein regressionsbasiertes Modell muss immer eine Reihe des zeitlichen Verlaufes der Regressoren und Zielvariable in die Vergangenheit zurück vorhanden sein, um daraus die Abbildung zu schätzen, bzw. zu lernen. Als Begriff bedeutet Regression *rückläufige Tendenz*. Im weitesten Sinne kann man dies auch so verstehen, dass man *zur Tendenz zurück geht*, also den Verlauf des Mittelwertes folgt, und ihn somit kennen muss. [FKL09] bietet dazu eine sehr ausführliche und allgemeine Einführung in das Gebiet der Regression.

Es sei hier kurz die Formulierung der (multiplen) linearen Regression aus [FKL09] gegeben:

Die *Zielgröße* y_i sei metrisch und beschreibt die n Beobachtungen mit

$$y_i = \eta_i + \epsilon_i, i = 1, \dots, n \quad (3.1)$$

Die Fehler $\epsilon_1, \dots, \epsilon_n$ sind unabhängig und identisch verteilt (i.i.d.) mit

$$E(\epsilon_i) = 0, \text{Var}(\epsilon_i) = \sigma^2. \quad (3.2)$$

Der *Erwartungswert* beträgt

$$E(y_i) = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik} = \eta_i^{\text{lin}}. \quad (3.3)$$

Wobei

$$\beta_0, \beta_1, \dots, \beta_k \quad (3.4)$$

die geschätzten bzw. gelernten Parameter für die Regressoren

$$x_{i1}, x_{i2}, \dots, x_{ik} \quad (3.5)$$

sind, dh. externe Parameter zur Beobachtung i . Aus den Parametern und den Regressoren wird der *Prädiktor* η_i gebildet und entspricht hier dem Erwartungswert.

Zu den Regressionsverfahren gehören auch lernbasierte Modelle, wie künstliche neuronale Netze, siehe [Pfa01], SVR, siehe [SS04], und Echo State Netze, siehe [Jae] und [Jae10]. Input-Delayed neuronale Netze sowie RNN werden unter anderem in [CAB01] kurz beschrieben.

Daraus ist zu erkennen, dass für jeden (Zeit)punkt der Zielvariable y_i eine lineare Abbildung aus den Regressoren x_{ik} besteht. Da die Zielvariable über einen Mittelwertverlauf und einer Varianz beschreibbar ist, wird der Abbildung ein Rauschen additiv überlagert. Die Abbildung wird hier aus vergangenen Zeitreihen über diverse Verfahren geschätzt, ebenfalls in [FKL09] beschrieben.

Generell ist anzumerken, dass regressionsbasierte Modelle für beide Kraftwerkstypen zur Vorhersage geeignet sind, sofern Zeitreihen aus der Vergangenheit vorhanden sind. Wie in Kapitel 2 beschrieben, liegen diese Datensätze aber lediglich für die WLKW in befriedigender Form vor. Für PVA muss daher ein physikalisches Modell in Betracht gezogen werden. Diese Eingrenzung kann an dieser Stelle getroffen werden.

3.1 Physikalische Modelle

Für das physikalische Modell liegen hauptsächlich zwei Arbeiten als verwandte Arbeiten vor, die als Grundlage zur Modellierung des physikalischen Modells für PVA dienen sollen.

3.1.1 Quaschning et al. [Qua96]

[Qua96] behandelt das Thema der *Abschattungsverluste* bei solarelektrischen Systeme. Dabei geht es eher nicht um die elektrischen Effekte, sondern [Qua96] beschreibt dabei die Ertragseinbußen durch Abschattung von PVA. Die Dissertation ist so aufgebaut, dass nach einer grundlegenden elektrischen Beschreibung von Photovoltaikmodulen, ausgehend von Diodenmodellen eines Solarpaneels, die beeinflussenden Faktoren, wie solare Strahlung und deren Aufteilung in direkte, diffuse und reflektierte Strahlung, Wirkungsgradreduktion durch die Paneeltemperatur und Abschattungseffekte beschrieben werden. Verfahren zur Bestimmung der Abschattung durch umliegende Objekte werden angeführt. Dabei

wird das Modell der Schattenanalyse und virtuellen Bilder aufgezeigt. Darauf aufbauend werden sehr genau und detailliert die Einbußen des Ertrags auf Grund umliegender Objekte berechnet. Diese Arbeit ist als verwandte Arbeit interessant, da im Kapitel 3 Modelle zur Berechnung der Strahlung auf geneigten Flächen beschrieben werden, wenn die Strahlung auf die Horizontale gegeben ist. Interessant dabei ist das statistische Modell zur Bestimmung des Anteiles der diffusen Strahlung, da die gängigen Strahlungsmessstationen ein Sternpyranometer beinhalten, welches die gesamte Strahlung (direkte+diffuse+reflektierte Strahlung) erfasst. Die diffuse Strahlung hängt dabei stark von der Abweichung der Sonneneinstrahlung zum Normalvektor auf die geneigte Fläche ab. Außerdem ist der Anteil der diffusen Strahlung bei bedecktem Himmel um ein Vielfaches größer als bei klarem Himmel. [Qua96] zeigt auch die Berechnung des reflektierten Strahlungsanteils. Es wird auch eine Berechnung des Sonnenstandes auf Grund des Datums und der Uhrzeit beschrieben, der sog. *SUNAE* Algorithmus. Außerdem wird eine Möglichkeit zur Berechnung des Sonnenaufgangs und des Sonnenuntergangs präsentiert. Vor allem der Sonnenstand ist für die Berechnung der Strahlungsleistung auf geneigte Flächen der Panee das grundlegendste Werkzeug. Der Sonnenstand wird vor allem für die Berechnung der Wirksamkeit der direkten Strahlung auf geneigte Flächen benötigt. Ausführliche Modelle zur Bestimmung der diffusen Strahlung verwenden ebenfalls den Sonnenstand. Weitere Ausführungen der Abschattungsverluste werden für das zu bildende Modell nicht betrachtet.

3.1.2 Blank et al. [Bla06]

Die andere verwandte Arbeit, [Bla06], beschreibt eine Modellierung und Simulation von PVA in Verteilnetzen. Ein sehr ähnliches Themengebiet, jedoch zielt diese Arbeit auf die Bestimmung der Einflüsse von solarelektrischen Systemen in einem Verteilnetz ab. Ein Teil der Diplomarbeit von *Blank* handelt von der Bildung eines Matlabmodells einer PVA, sehr ähnlich jenem Modell, welches für diese Arbeit benötigt wird. Für das physikalische Modell dieser Arbeit ist vor allem Kapitel 2 aus [Bla06] interessant. Neben der Berechnung des Sonnenstandes und der Berechnung der Strahlung auf geneigte Flächen, bei einer Messung der Strahlung auf horizontale Flächen, ist vor allem die Berechnung des Temperatureinflusses auf den Paneelwirkungsgrad interessant. [Bla06] beschreibt auch kurz die Abhängigkeit des Wirkungsgrades eines Wechselrichters in Abhängigkeit der Auslastung.

3.2 Regressionsbasierte Modelle

Dieses Kapitel beschreibt State-of-the-Art Methoden geeigneter Regressionsverfahren ähnlicher/analoger Problemstellungen zur Modellierung von WLKW. Prinzipiell würden für die Modellierung einer Erzeugungsanlage auf Basis vergangener Daten folgende Methoden in Frage kommen:

- Reine Zeitreihenanalyse. Wird in dieser Arbeit nicht behandelt, da die Zeitreihenanalyse rein die zeitliche Abhängigkeit der Zielgröße selbst modelliert. Kann wie ein Filter betrachtet werden, das am Eingang die selbe Größe wie am Ausgang hat.
- Regression und daraus abgeleitete maschinelle Lernverfahren, wie neuronale Netze, RNN (z.B. Elman Netze), Echo State Netze und SVR. Die Regression bietet ein Werkzeug über Einflussgrößen (Regressoren) eine Zielgröße abzubilden.

Daher werden regressionsbasierte Modelle für die Modellierung von Erzeugungsanlagen auf Basis vergangener Leistungswerte untersucht.

3.2.1 Nicklass et al. [Nic10]

[Nic10] beschreibt in seiner Arbeit ein Problem aus der selben Problemklasse. Er beschreibt kurzfristige Prognoseverfahren, um die Konzentration des *Particulate Matters* (Feinstaub) mit einem aerodynamischen Durchmesser $< 10\mu\text{m}$ (PM_{10}) in der Luft vorherzusagen. Es gilt dabei ein Modell zu finden, welches auf Grund geeigneter externer Attribute, wie prognostizierte Wetterdaten (Niederschlag, Wind, etc.) und aktuellen Verkehrsdaten, einen numerischen Tagesmittelwert für den kommenden Tag prognostiziert. Es liegt also eine analoge Problemklasse vor. Die Prognosemethoden in dieser Arbeit basieren auf maschinellen Lernverfahren. Diese Arbeit ist auch eine Gegenüberstellung verschiedener state-of-the-art Methoden. Der Hauptaugenmerk in der Arbeit liegt darin, ein rasche Attributauswahl und Parametereinstellung für die Modelle zu finden, sowie eine Optimierung für die Prognosemodellerstellung zu finden. Dabei wird die Leistungsfähigkeit und die Qualität der Modelle experimentell untersucht. In dieser Arbeit werden folgende Methoden behandelt:

- Multiple lineare Regressionsverfahren:
 - stellt dies als Referenzmodell in den Raum,
 - schließt aus anderen Arbeiten, dass die Ergebnisse zufriedenstellend sind und
 - behauptet aber, dass der reale Zusammenhang nicht-linear ist, und daher mit geeigneten maschinellen Lernverfahren bessere Ergebnisse erzielt werden können.
- Maschinelle Lernverfahren:
 - MLP basierte neuronale Netze,
 - k-nearest-neighbour Verfahren für eine Regression und
 - SVR.

Weiters werden Verfahren zu Attributauswahl wie genetische Algorithmen behandelt. Für die Parameterfindung werden Rasteroptimierung und Evolutionsstrategien verwendet. Geeignete Bewertungsmethoden für die Leistung und Qualität der Prognosemodelle werden erklärt. Interessant ist auch der Teil über die Auswahl und Vorbereitung der Eingangsdaten. Hierbei ist vorab vor allem wichtig zu untersuchen:

- Welche Attribute kommen sinnvollerweise in Frage?
- Welche Daten kann man dazu aufbereiten?
- Wie weit in die Vergangenheit zurück liegen Daten vor?
- Wie müssen die Daten vorab hinsichtlich *Ausreißer*, Wertebereiche, fehlerhafte Messungen und außergewöhnlichen Einflüssenfaktoren bereinigt und gefiltert werden?

Es wird in dieser Arbeit deutlich gemacht, dass Vorsicht bei der Aussage über die Güte des Prognosemodells geboten ist, da Attribute wie Wetterwerte ebenfalls Prognosewerte sind und einem teils großen Prognosefehler unterworfen sind, und somit direkt Einfluss auf die Prognosegüte nehmen. Diese Problematik tritt bei der Leistungsprognose von WLKW analog auf.

3.2.2 Basurko et al. [ABIBM06]

[ABIBM06] behandelt die Prognose von O_3 und NO_2 Levels im Großraum Bilbao, Spanien. Die Prognose liefert dabei stündliche Vorhersagewerte über zwei Modellarten, lineare Regression und MLP basierte neuronale Netze. Wobei vom MLP basierten neuronalen Netz wiederum zwei Modellierungen gemacht werden. Die Prognose ist bis zu 8h voraus möglich. In dieser Arbeit kommen die Autoren zum Schluss, dass MLP basierte neuronale Netze in Summe bei 75% der Fälle im Vergleich zum linearen Regressionsmodell bessere Ergebnisse liefert. Die Parameter des linearen Regressionmodells werden mit einem Statistikprogramm evaluiert. Regressoren sind:

- Verkehr,
- Meteorologische Daten und
- O_3 und NO_2 Werte.

Vergangenheitszeitreihen aller Daten sind über zwei Jahre zurück vorhanden. Die Autoren kommen zum Schluss, dass beide Modellarten gut geeignet sind, jedoch das MLP basierte Modell wesentlich besser abschneidet. In [ABIBM06] wird auch zur jeweiligen Modellart die notwendige Theorie vermittelt, unter anderem auch jene für die MLP basierten neuronalen Netze.

Neuronale Netze sind in der Lage, generalisierte Patterns zu erlernen, dh. eine Abbildung von Inputvektoren auf Outputvektoren zu lernen, sodass der Fehler zur realen Abbildung minimal wird. Ein MLP basiertes neuronales Netz weist die Struktur wie in Abbildung 3.1 auf. Eine Inputschicht verteilt die Inputs gewichtet mit w_{NS}^h auf die S Neuronen der ersten Schicht im Hiddenlayer. Jedes Neuron besitzt einen Bias b und eine Aktivierungsfunktion f . Je nach Aktivierungsfunktion feuert ein Neuron ein Signal über die gewichteten Verbindungen zum nächsten Layer. Im Outputlayer gibt es L Neuronen, jeweils mit einem Bias und einer Aktivierungsfunktion, wobei innerhalb der Hiddenlayer ein feedforward Netz vorliegt. Die Signale der letzten Schicht des Netzes der Hiddenlayer werden auf jedes Neuron der ersten Outputschicht verteilt. Der Output y_k^o solch eines Netz kann mit

$$y_k^o = f_k^o \left(b_k^o + \sum_{i=1}^S w_{ik}^o y_i^h \right) \quad (3.6)$$

$$y_k^o = f_k^o \left(b_k^o + \sum_{i=1}^S w_{ik}^o f_i^h \left(b_i^h + \sum_{j=1}^N w_{ji}^h x_j \right) \right) \quad (3.7)$$

beschrieben werden, wobei $k = 1, \dots, L$ ist. Kompakter gefasst ist

$$n_i^h = b_i^h + \sum_{j=1}^N w_{ji}^h x_j \quad (3.8)$$

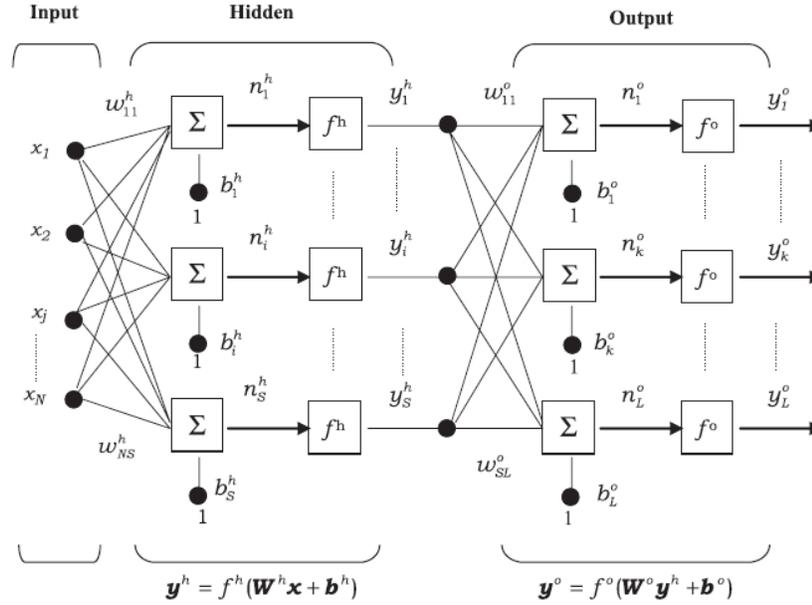


Abbildung 3.1: Struktur eines multilayer Perzeptron basierten neuronalen Netzes, [ABIBM06, Fig. 1].

$$n_k^o = b_k^o + \sum_{i=1}^S w_{ik}^o y_i^h \quad (3.9)$$

der jeweilige Output der Summenfunktion. n_i^h und n_k^o liegen an der Aktivierungsfunktion f_h bzw. f_o an. Die wichtigsten Aktivierungsfunktionen sind die logarithmische sigmoide Funktion *logsig*

$$\text{logsig}(x) = \frac{1}{1 + e^{-x}} \quad (3.10)$$

mit dem Wertebereich zwischen 0 und 1, die hyperbolische tangente Funktion *tansig*

$$\text{tansig}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3.11)$$

mit dem Wertebereich zwischen -1 und 1 , und die lineare Funktion *purelin*

$$\text{purelin}(x) = k \cdot x. \quad (3.12)$$

Zum Trainieren werden die Inputvektoren dem Netz auferlegt, der Ist-Output y_k mit dem Soll-Output t_k verglichen, und ein Fehlerwert gebildet. Die Fehlerfunktion kann unter anderem der *mean squared error (MSE)* sein:

$$E = \frac{1}{L} \sum_{k=1}^L (t_k - y_k^o)^2. \quad (3.13)$$

Die Anpassung der Gewichte erfolgt in der Regel mit dem *Backpropagation Algorithmus*. Hierbei wird *ermittelt*, welche Verbindung im Netz welchen Einfluss auf den Fehler hatte.

Sie wird anschließend in die notwendige Richtung korrigiert. Wenn der Fehler minimal wird, wird mit dem Training aufgehört. Dies nennt man *early stopping*. Mit dieser Methode wird allerdings nicht garantiert, dass ein globales Minimum gefunden wird.

3.2.3 Stadlober et al. [SHP08]

Hier wird das selbe Problem wie in [Nic10] verfolgt. Diese Arbeit kann als *Gegenstück* zu [Nic10] betrachtet werden. Es geht in beiden Arbeiten um die kurzfristige Prognose von PM_{10} Partikeln in den Luft, auf Grund externer Einflussparameter wie Wetter, Verkehr, Hausbrand etc. Dieses Problem liegt in der selben Problemklasse wie die Leistungsprognose von WLKW. In [Nic10] wird der Ansatz über neuronale Netze gewählt, da dort behauptet wird, dass es einen nichtlinearen Zusammenhang zwischen den Attributen und der Zielgröße gibt, und dieser Zusammenhang am *einfachsten* über maschinelle Lernverfahren gefunden werden kann. Hier in [SHP08] wird der nichtlineare Zusammenhang nicht bestritten. Dafür wird aber behauptet, dass für die Prognose des Mittelwerts für den Folgetag ein multiples lineares Regressionsmodell ausreichend sei. Vor allem in Hinblick auf die Prognose der Grenzwertüberschreitung seien nichtlineare Randeffekte nicht sehr relevant. Es gilt ein einheitliches Modell für drei Städte im Alpen-Adriaraum zu finden. Die Modellierung erfolgt über statistische Methoden, um ein multiples lineares Modell zu finden. Dazu ist eine sehr genaue Analyse über

- Daten,
- Einflussgrößen,
- Umweltsituation und Gegebenheit sowie
- Messdatenerfassung

durchzuführen, und wird auch in [SHP08] angeführt. Weiters wird die Wahl eines multiplen linearen Regressionsmodells damit begründet, dass diese Art von Modellen leichter nachvollziehbar sei, und seine *innere Struktur* preis gibt. Komplexe maschinelle Lernverfahren seien auf Grund der Blackboxeigenschaft nicht direkt nachvollziehbar und daher in ihrer inneren Struktur nicht transparent, heißt es sinngemäß in [SHP08]. Nach einer Untersuchung der Datenbestände mit statistischen Methoden, folgt eine genaue Beschreibung und Analyse der Gegebenheiten. Der Modellansatz wird beschrieben mit

$$\sqrt{PM10_d} = \beta_0 + \sum_{i=1}^m \beta_i z_d^{(i)} + \epsilon_d. \quad (3.14)$$

Als Ergebnis wird die Wurzeltransformierte angenommen. β_i sind die Parameter zu jedem Attribut z_i , wobei

$$\epsilon_d \sim^{(iid)} N(0, \sigma^2) \quad (3.15)$$

ist. Es wird auch die Vorhersage weiter als $n=1$ Tag beschrieben. Dieser Vorgang ist iterativ. Wäre die Wetterprognose exakt, so wäre die Vorhersage des PM_{10} Wertes exakt, sofern die lineare Abbildung gültig ist. Jedoch heißt das auch, dass die Vorhersagegüte des PM_{10} Wertes analog mit der Wettervorhersagegüte einhergeht.

[SHP08] ist eine geeignete verwandte Arbeit. Es ist jedoch daran zu denken, dass hiermit

eine lineare Abbildung gefunden wird. Es kann durchaus auch eine nichtlineare Regression angesetzt werden, wobei dann wiederum den maschinellen Lernverfahren der Vorzug zu geben sein wird, da sie im Prinzip nichts anderes sind, als nichtlineare Regressionen, speziell wenn man MLP basierte neuronale Netze betrachtet.

3.2.4 Kales et al. [Kal03]

[Kal03] beschreibt mathematische Ansätze zur kurzfristigen Lastprognose. Für die Energiewirtschaft ist es unumgänglich solche Prognosen über den Strombedarf für die kommenden Stunden bzw. kommenden Tage zu machen, um sich auf den Bedarf der Kunden im Netz einzustellen. Ebenfalls ist es für den Stromhandel von großer Bedeutung. Längerfristige Prognosen kommen eher bei der Langzeitnetzplanung zum Tragen. Die Ansätze, die in dieser Arbeit beschrieben werden, sind theoretische Ansätze der Zeitreihenanalyse, Regressionsmethoden, künstliche neuronale Netze, exponentielle Glättung und Zustandsraummodellierung. Die Lastprognose kann nicht so einfach auf die Leistungsprognose umgelegt werden, da vor allem Verhaltensmuster der Gesellschaft und der Industrie in Abhängigkeit von der Zeit (kurz-, mittel- und langfristig) eine ausschlaggebende Rolle spielen. Außerdem gibt diese Arbeit einen Überblick über viele Methoden der Lastprognose, und geht hauptsächlich auf die Methoden der Zeitreihenanalyse genauer ein, die für die Leistungsprognose eher uninteressant ist, da sie rein von den externen Parametern wie z.B. Niederschlag und Temperatur abhängt, jedoch sonst keine zeitlichen Periodizitäten oder Trends aufweist.

3.2.5 Coulibaly et al. [CAB01]

In Kapitel 2 wurden bereits zwei wesentliche Systemeigenschaften über das Zeitreihenverhalten sichtbar. Tritt Niederschlag auf, wird je nach Temperatur der Zufluss zu den WLKW ansteigen. Endet der Niederschlag, wird der Wasserzufluss nicht sofort enden, sondern auf Grund des speichernden Verhaltens der Umwelt und Natur langsam exponentiell abklingen.

Die zweite Eigenschaft ist ein Langzeitverhalten. Wie in Kapitel 2 erwähnt, wird der Niederschlag im Winter über lange Zeit als Schnee *gespeichert*, und erst bei längerfristig wärmeren Temperaturen tritt die Schneeschmelze auf. Es gibt also eine klare Differentierung des Verhaltens zwischen den Sommer- und den Wintermonaten.

Auf Grund dieser Eigenschaften muss man sich also überlegen, ob es sinnvoll ist, eine statische Abbildung der Regressoren *Niederschlag und Temperatur* auf die Zielgröße *Leistung* zu machen, vor allem wenn man das kurzzeitig speichernde Verhalten mit berücksichtigen muss. Dazu gibt es bezüglich des Problems der Leistungsprognose unter anderem die Arbeit von [CAB01]. Sie handelt von der Prognose der Zuflussmenge von Speicherkraftwerken in Abhängigkeit von Niederschlag und Temperatur. Im Speziellen geht es um ein Kraftwerk im Nordosten Kanadas. Das Problem kann folgendermaßen analog auf das zu untersuchende Problem der Leistungsprognose von WLKWs umgelegt werden:

- Der Wasserzufluss eines Speicherkraftwerks entspricht dem Wasserfluss ohne Wasserspeicher. Also ist das Problem proportional dem Problem der Prognose der Wasser(durch)flussmenge.

- Die Durchflussmenge hängt über einen linearen Faktor mit der erzeugten Leistung eines WLKWs zusammen.

[CAB01] eignet sich auch daher recht gut als verwandte Arbeit, da die zwei zuvor beschriebenen Verhaltenweisen auf Grund der ähnlichen klimatischen Gegebenheiten auch hier gegeben sind, vor allem wenn man das Langzeitverhalten betrachtet (Wintermonate zu Sommermonate). [CAB01] beschäftigt sich hierbei mit RNN, dh. es liegt nicht nur ein starres neuronales feedforward Netz vor, bei dem es nur Vorwärtsverbindungen gibt, sondern es gibt auch Rückkopplungen. In [CAB01] werden RNNs und input delayed neuronale Netze (IDNN) in ihrer Performance bei der Vorhersage verglichen. In dieser Arbeit werden teils sehr gute Prognoseergebnisse erzielt, und dies für Prognosen mehrere Tage voraus, wobei auch Datensätze über 20 Jahre in die Vergangenheit zurück vorliegen und für das Training verwendet werden.

Starre feedforward Netze bilden starre nichtlineare Abbildungen, und haben dabei keine dynamische Gedächtnisfunktion. Bei einem RNN wird zwischen folgenden Arten unterschieden:

- Rückkopplung vom Output zu den Inputneuronen (Jordan Netz),
- Rückkopplung von Hiddenneuronen zu den Inputneuronen (Elman Netz) und
- Lokale Rückkopplungen innerhalb der Hiddenneuronen (locally recurrent).

Als Ergänzung kann hier [WW] angeführt werden. In dieser Arbeit wird eine Einführung in die neuronalen Netze und Elman Netze gegeben. Mit Ihnen wird ein Prognosemodell für das zu erwartende Maximum der Ozon Konzentration im Osten Österreichs gebildet. Dabei werden die verschiedenen Modelle verglichen. Die Regressoren sind aktueller Ozonwert, Temperatur, Bedeckungsgrad und Wind.

[WRV09] beschreibt und vergleicht feedforward und feedback neuronale Netze miteinander, wobei sie hier zur kurzfristigen Prognose der Windgeschwindigkeit verwendet werden. Speziell werden die Netzarten MLP basierte neuronale Netze, Elman Netze und *simultaneous recurrent* Netze für diese Anwendung untersucht. Als Eingangsdaten werden gemessene meteorologische Daten aus der Vergangenheit der Temperatur, relative Luftfeuchtigkeit und Windgeschwindigkeit verwendet. Die Daten liegen im 15 Minuten Raster vor und die Prognose liefert einen Prognosewert für 15 Minuten voraus. In den Resultaten ist dabei zu erkennen, dass die Netzart der MLP basierten neuronalen Netze sogar in der Performance etwas besser ist. Sie weisen aber darauf hin, dass die Trainingsdauer bei RNN erheblich größer ist.

[KLSK] vergleicht MLP basierte neuronale Netze mit Elman Netze und FIR Filter für die Vorhersagemodelle von verschiedenen Zeitreihen. Die Autoren kommen zur Erkenntnis, dass der Lernalgorithmus, der verwendet wird, eine sehr wichtige Rolle spielt, und dass MLP basierte neuronale Netze trotz statischer Sicht auf die Vergangenheit bei korrekt gewähltem Lernalgorithmus gute Ergebnisse liefert. Auch sie meinen, dass neuronale Netze wesentlich kürzere Trainingszeiten haben. Anzumerken ist aber, dass die Autoren auffällig periodische Zeitreihen verwenden und vorhersagen möchten. Die Datenreihen hängen also stark von der Zeit ab. Bei der Leistungsprognose von beispielsweise den WLKW ist dies nicht der Fall, und die Leistung wird von den externen Parametern Niederschlag und Temperatur bestimmt.

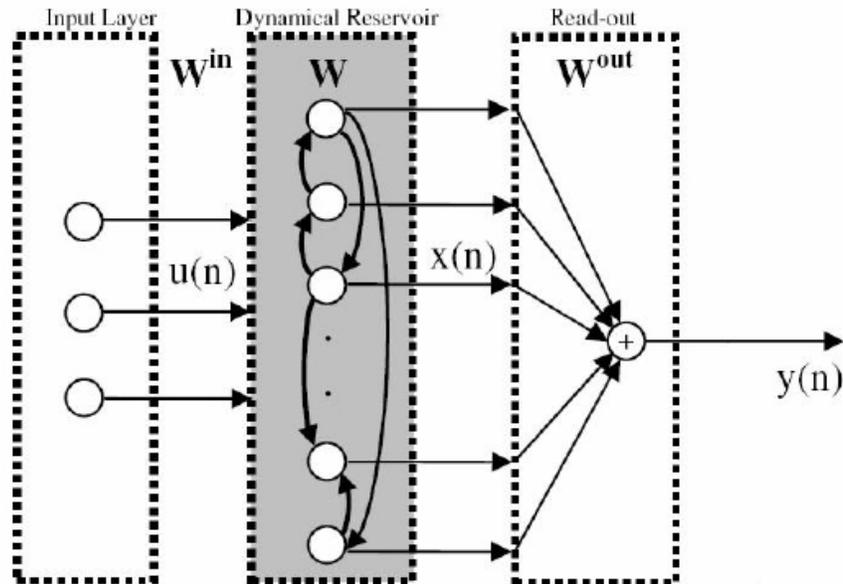


Abbildung 3.2: Blockdiagramm eines Echo State Netzes, [SOP⁺07, Fig. 1].

3.2.6 Sacchi et al. [SOP⁺07]

Als Ergänzung zu [CAB01] passt [SOP⁺07]. Auf Grund der Komplexität die ein RNN haben kann, können Probleme wie

- lange Trainingszeiten,
- Instabilitäten und
- großer Rechenaufwand

auftreten. Echo State Netze, wie in Abbildung 3.2 dargestellt, können hierbei Abhilfe verschaffen.

Laut [Jae10] ist das Dynamic Reservoir ein fix gewähltes rekurrentes Netz, mit fixen Gewichten der Verbindungsmatrix W . Im Inputlayer werden die notwendigen Inputs $u(n)$, wie z.B. Niederschlag, Temperatur, etc., definiert. Lediglich die Gewichtsmatrix W^{out} der Read-out Schicht kann trainiert werden. Das Training dieser Schicht kann analytisch gemacht werden und ist daher sehr schnell durchführbar. Wegen der rekurrenten Struktur des Dynamic Reservoir fließen Vergangenheitswerte in die aktuellen Inputwerte mit ein. States des Dynamic Reservoir werden *echo states* genannt. Diese echo states beinhalten Information über die Vergangenheit.

Man kann sich dies am einfachsten so vorstellen, dass bei einem Diracimpuls am Eingang eine Impulsantwort auf Grund der rekurrenten Struktur folgt. Die Impulsantwort kann dann als Echo des Diracimpulses gesehen werden. In [SOP⁺07] wird ein Prognosemodell für den Wasserzufluss eines brasilianischen Speicherkraftwerks gebildet. Dabei werden Echo State

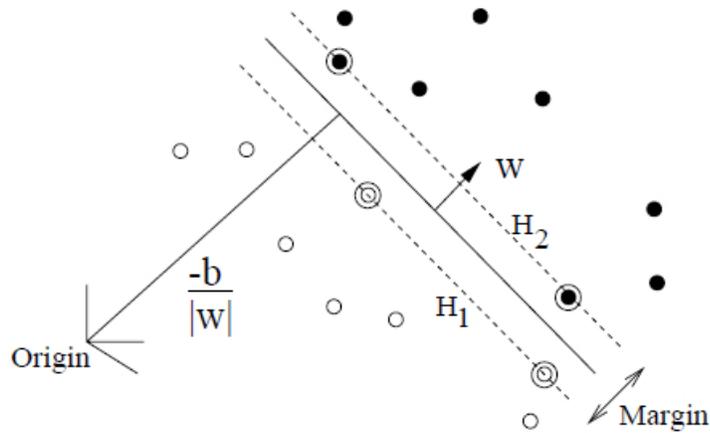


Abbildung 3.3: Getrennte Hyperebenen (Bereiche der zwei Klassen), H_1 und H_2 , [Bur98, Fig. 5].

Netze in den Fokus gestellt. Mit ihnen wird der Durchschnitt des Zuflusses für den kommenden Monat prognostiziert. Es sind hierzu Zeitreihen weit in die Vergangenheit zurück gegeben. Wie bereits in Kapitel 3.2.5 beschrieben, ist das Problem der Zuflussprognose analog dem Problem der Leistungsprognose für WLKW. In dieser verwandten Arbeit wird von der Behauptung ausgegangen, dass aufwändige nichtlineare Regressionsmodelle zwar gute Resultate für Prognose in dieser Problemklasse erzielen, jedoch nur mit fixem Fenster für die *Sicht auf die Vergangenheit*. Außerdem müsste Expertenwissen angewendet werden, um das Zeitfenster zu definieren. Vor allem bei den WLKW muss, wie bereits in der Datenanalyse beschrieben, pro Kraftwerk eine Parametereinstellung bzw. ein Training des Lernmodells vorgenommen werden. Ein Echo State Netz bildet die Breite des Fensters in die Vergangenheit implizit, und zudem kann eine dynamische Breite des Vergangenheitsfensters gefunden werden. Das Modell eines Echo State Netzes schneidet im Vergleich zu angeführten Regressionsmodellen bemerkenswert gut ab.

3.2.7 Zhao et al. [ZXDH]

SVR ist eine Erweiterung der SVM. Bei der SVM wird durch ein Optimierungsverfahren eine Hyperebene zur linearen Trennung zweier Klassen verwendet, wobei Datenpunkte über Vektoren dargestellt werden, siehe [VC] und [Bur98]. Datenpunkte werden, wie in Abbildung 3.3 dargestellt, in einem d -dimensionalen Raum linear in Klassen getrennt. Die Hyperebene wird dabei in die Ebenen H_1 und H_2 geteilt. Die Dimension d ist so groß wie nötig, um eine lineare Trennung durchführen zu können. Das Optimierungsverfahren findet eine sogenannte *optimal hyperplane/optimale Hyperebene* mit einem maximalen Abstand zu den n nächsten Punkten der Klassen, der sogenannte *optimal margin*. Die mathematische Herleitung für die Optimierung ist in [VC] angeführt.

In [VC] wird gezeigt, dass Datenpattern mit l Datenpunkten

$$(y_1, \mathbf{x}_1), \dots, (y_l, \mathbf{x}_l), y_i \in \{-1, 1\} \quad (3.16)$$

dann separierbar sind, wenn es eine Hyperebene beschrieben durch den Normalvektor \mathbf{w} und einem Skalar b gibt, und die Ungleichung

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq 1 \quad (3.17)$$

wenn $y_i = 1$, und

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \quad (3.18)$$

wenn $y_i = -1$, für alle Datenpunkte erfüllt sind. Das kann kompakt geschrieben werden mit

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq -1, i = 1, \dots, l. \quad (3.19)$$

Die optimale Hyperebene kann durch

$$\mathbf{w}_0 \cdot \mathbf{x} + b_0 = 0 \quad (3.20)$$

beschrieben werden. Die Distanz wird durch die Maximierung von

$$\rho(\mathbf{w}, b) = \min_{\{\mathbf{x}: y=1\}} \frac{\mathbf{x} \cdot \mathbf{w}}{|\mathbf{w}|} - \max_{\{\mathbf{x}: y=-1\}} \frac{\mathbf{x} \cdot \mathbf{w}}{|\mathbf{w}|} \quad (3.21)$$

gefunden.

Laut [Bur98] sind Klassen in der Regel nicht streng linear trennbar. Einerseits können gewisse *Ausreißer*, die in der jeweils anderen Klasse liegen, toleriert werden. Dies geschieht durch die Einführung einer *soft margin*. Andererseits kann eine Transformation in einen höherdimensionalen Raum gemacht werden, um dort eine lineare Trennung zu ermöglichen. Dabei wird eine Projektion

$$\Phi: \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}, \mathbf{x} \mapsto \Phi(\mathbf{x}) \quad (3.22)$$

angewendet. Die Punkte \mathbf{x} gehen in der Optimierung nur als Skalarprodukte ein, daher kann das Skalarprodukt des niederdimensionalen Raumes mit dem Skalarprodukt des hochdimensionalen Raumes ersetzt werden. Nach einer linearen Trennung im höherdimensionalen Raum wird wieder eine Rückprojektion gemacht, und somit die optimale Klassifizierung gefunden.

Die Projektion ist auf die Art sehr Rechenintensiv. Man kann mit Hilfe des sogenannten *Kernel Tricks* den Rechenaufwand deutlich senken. Dabei wird das hochdimensionale Skalarprodukt durch eine Kernelfunktion ersetzt.

Für die Klassifikation wird in (3.23) das Vorzeichen betrachtet, und $f(\mathbf{x})$ selbst beschreibt hierbei eine Funktion, die die zwei Klassen trennt. Also auf folgende Art:

$$f(\mathbf{x}) = \text{sgn}(\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b). \quad (3.23)$$

In [ZXDH] wird eine kurze Einführung in die SVR gegeben. Eine SVR macht im Prinzip dasselbe wie eine SVM, jedoch mit dem Unterschied, dass $f(\mathbf{x})$ nicht durch den Abstand zu den zwei Klassen gefunden wird, sondern durch die Datenpunkte, denen eine Funktion unterstellt wird. $f(\mathbf{x})$ beschreibt dann die Regression der Datenpunkte. In dieser Arbeit

wird die Methode der SVR und die Methode der neuronalen Netze zur Vorhersage von Windgeschwindigkeiten bearbeitet, um bessere Aussagen über die Stromproduktion durch Windenergie machen zu können. Die Autoren kommen bei dieser Arbeit zum Schluss, dass die SVR Methode bessere Ergebnisse als die Methode der neuronalen Netze liefert, beide jedoch sehr gute Prognoseergebnisse erzeugen. Wobei nicht näher auf das verwendete Konzept des Modells der neuronalen Netze eingegangen wird. Außerdem wird als Input rein der vergangene Verlauf verwendet. Es werden keine meteorologischen Daten mit einbezogen. Den großen Vorteil, dass die SVR Methode eine oberer Schranke des zu erwartenden Risikos (Fehler bei der Prognose) minimiert, heben die Autoren extra hervor. Dies nennt man die Methode der *strukturellen Risikominimierung*. Im Gegensatz zu den neuronalen Netzen, bei dem der Fehler auf das Trainingsset minimiert wird, ohne overfitting zu erhalten. Dies nennt man *empirische Risikominimierung*.

In [WCY08] wird jedoch deutlich hervorgehoben, dass die Methode der SVR bei vielen Datenpunkten schnell in ein exponentielles Wachstum der Trainingszeit mündet. In [LXZC] wird eine Prognose von Immobilienpreisentwicklungen auf Grund diverser Einflussfaktoren mit der Methode der SVR dargestellt. Die Autoren zeigen als Vergleich Ergebnisse von neuronalen Netzen, die über die Backpropagationmethode trainiert wurden. Sie meinen, dass der SVR Ansatz bessere Ergebnisse liefere, jedoch wird auch hier keine Aussage über das dahinterliegende Modell der neuronalen Netze gemacht. Es wird auch keine Aussage über die Trainingsdauer gemacht.

Kapitel 4

Grundlagen zu den Modellierungsansätzen

Für die praktischen Ausführungen der Modellierungen, werden in diesem Kapitel ausreichend notwendige Grundlagen präsentiert. Dabei werden einerseits die Grundlagen für den physikalischen Ansatz aufbereitet, andererseits Grundlagen für verschiedene mögliche regressionsbasierte Ansätze beschrieben.

4.1 Grundlagen zum physikalischen Ansatz

Für die praktische Realisierung des Modells einer PVA sind einige Berechnungen und andere Modelle notwendig, deren Grundlagen im Folgenden angeführt sind.

4.1.1 Sonnenstandsberechnung nach dem SUNAE - Algorithmus

Für den Sonnenstand sind zwei Winkelbezeichnungen maßgebend. Der erste Winkel ist der Azimut α_S , und beschreibt den Winkel zwischen Norden und dem Sonnenmeridian. Norden beschreibt dazu den Winkel von 0° . Der zweite Winkel ist die Sonnenhöhe γ_S , und beschreibt den Winkel zwischen der Horizontalen und dem Winkel der Einstrahlung der Sonne. Abbildung 4.1 veranschaulicht dazu die beiden Winkelbezeichnungen. Der SUNAE-Algorithmus wird hier nicht näher erklärt, ein Strukturgramm zur Implementierung des Algorithmus aus [Qua96, Anhang] ist im Anhang B unter Abbildung B.1 und B.2 zu finden.

4.1.2 Extraterrestrische Solarstrahlung

Die extraterrestrische Solarstrahlung schwankt im Laufe eines Jahres zwischen $1293 \frac{W}{m^2}$ und $1412 \frac{W}{m^2}$, auf Grund der elliptischen Bahn der Erde um die Sonne. [Bla06] führt die Berechnung für die extraterrestrische Solarstrahlung nach *Duffie und Beckman* wie folgt an:

$$F_e = 1 + 0.033 \cdot \cos\left(\frac{2 \cdot \pi \cdot DOY}{365}\right), \quad (4.1)$$

ist der Exzentritätsfaktor, wobei

$$DOY \in [1, 365] \quad (4.2)$$

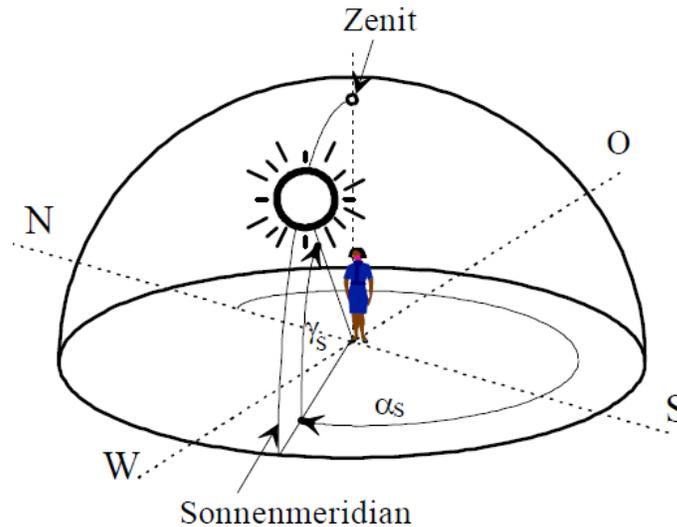


Abbildung 4.1: Veranschaulichung der Winkelbezeichnungen des Sonnenstandes, [Qua96, Abb. 3.1].

ist, und für die Abkürzung *day of year / Tag des Jahres* steht. Die extraterrestrische Solarstrahlung errechnet sich aus:

$$E_0 = F_e \cdot E_{sc}, \quad (4.3)$$

wobei E_{sc} mit $E_{sc} = 1367 \frac{W}{m^2}$ die Solarkonstante ist.

4.1.3 Berechnung der diffusen Solarstrahlung

Beim Durchlaufen der Solarstrahlung durch die Atmosphäre der Erde, tritt Reflexion, Absorption und Streuung auf. Dadurch entsteht eine indirekte Solarstrahlung, die sogenannte *diffuse* Solarstrahlung, die auf die Erde trifft, wie Abbildung 4.2 zeigt. Um in weiterer Folge die Solarstrahlung auf eine geneigte Fläche zu berechnen, ist es notwendig, die über ein horizontal ausgerichtetes Sternpyranometer gemessene terrestrische Globalstrahlung $E_{G,hor}$ in einen direkten und einen diffusen Strahlungsanteil zu unterteilen. Es gilt dabei der Zusammenhang

$$E_{G,hor} = E_{dir,hor} + E_{diff,hor}. \quad (4.4)$$

Dazu wird in [Qua96] unter anderem folgendes Modell beschrieben:

Die extraterrestrische Solarstrahlung auf die Erdoberfläche außerhalb der Erdatmosphäre ist abhängig von der Sonnenhöhe γ_s und beträgt

$$E_{0,hor} = E_0 \cdot \sin \gamma_s. \quad (4.5)$$

Über den sogenannten *Clearness-Faktor/Klarheitsfaktor* k_T kann über ein statistisches Modell der Anteil der diffusen Solarstrahlung ermittelt werden. [Qua96] fasst dabei mehrere statistische Modelle zusammen, wobei für Europa am ehesten jenes nach *Reindl, Beckmann*

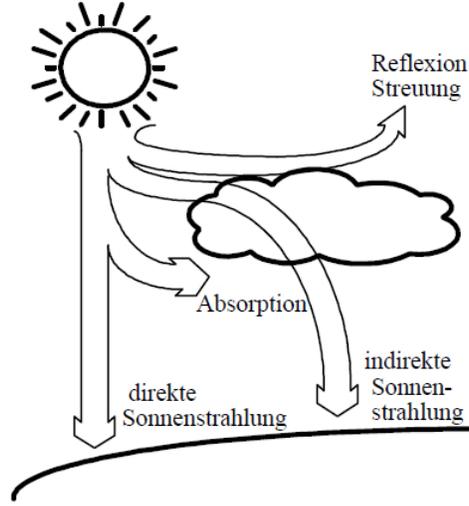


Abbildung 4.2: Solarstrahlung beim Durchlaufen der Atmosphäre, [Qua96, Abb. 3.4].

und Duffie zutrifft, da für die statistische Analyse dazu auch europäische Messstationen eingerichtet wurden.

$$k_T = \frac{E_{G,hor}}{E_{0,hor}} \quad (4.6)$$

berechnet den Klarheitsfaktor. Das in [Qua96] angeführte Modell von Reindl, Beckmann und Duffie berechnet die diffuse Solartstrahlung auf die Horizontale wie folgt:

$$E_{diff,hor} = E_{G,hor} \cdot (1.020 - 0.254 \cdot k_T + 0.0123 \cdot \sin\gamma_S), k_T \leq 0.3 \quad (4.7)$$

$$E_{diff,hor} = E_{G,hor} \cdot (1.400 - 1.749 \cdot k_T + 0.1770 \cdot \sin\gamma_S), 0.3 < k_T < 0.78 \quad (4.8)$$

$$E_{diff,hor} = E_{G,hor} \cdot (0.486 - 0.182 \cdot \sin\gamma_S), k_T \geq 0.78 \quad (4.9)$$

In [Sch91] wird unter anderem das Modell von Erbs, Klein und Duffie beschrieben:

$$E_{diff,hor} = E_{G,hor} \cdot (1.0 - 0.09 \cdot k_T), 0 \leq k_T \leq 0.22 \quad (4.10)$$

$$E_{diff,hor} = E_{G,hor} \cdot (0.9511 - 0.1604k_T + 4.388k_T^2 - 16.638k_T^3 + 12.336k_T^4), \quad (4.11)$$

$$0.22 < k_T \leq 0.8$$

$$E_{diff,hor} = E_{G,hor} \cdot (0.165), k_T > 0.8. \quad (4.12)$$

Auch für dieses Modell wurden unter anderem Messdaten von Messstationen aus Europa verwendet.

4.1.4 Solarstrahlung auf geneigte Flächen

Liegen die Solarstrahlungswerte auf eine horizontale Fläche vor (gemessen über ein horizontal montiertes Sternpyranometer), so gilt lt. [Qua96] für die Solarstrahlung auf die geneigte Fläche:

$$E_{G,gen} = E_{dir,gen} + E_{diff,gen} + E_{refl,gen}. \quad (4.13)$$

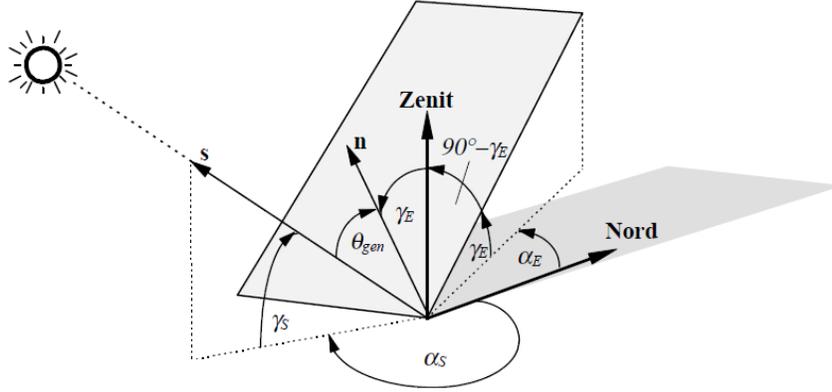


Abbildung 4.3: Veranschaulichung der Winkelbezeichnungen für den Sonnenstand auf eine geneigte Fläche, [Qua96, Abb. 3.3].

Der Reflektionsanteil $E_{refl,gen}$ ist der Strahlungsanteil, der von der Umgebung in Richtung der geneigten Fläche reflektiert wird. Es gilt, dass der Winkel der Abweichung der Sonne vom Normalvektor auf eine geneigte Fläche, siehe Abbildung 4.3,

$$\Theta_{gen} = \arccos(\mathbf{s} \cdot \mathbf{n}) \quad (4.14)$$

und weiters

$$\Theta_{gen} = \arccos(-\cos\gamma_S \cdot \sin\gamma_E \cdot \cos(\alpha_S - \alpha_E) + \sin\gamma_S \cdot \cos\gamma_E) \quad (4.15)$$

beträgt. α_E und γ_E beschreiben dabei die Drehung, bezogen auf Norden, und die Neigung der Fläche aus der Horizontalen.[Qua96] fasst ebenfalls Methoden für die Berechnung der Solarstrahlungsanteile auf eine geneigte Fläche zusammen. [Qua96] führt dazu zuerst die Berechnung vom direkten Solarstrahlungsanteil nach *Duffie und Beckman* an:

$$E_{dir,gen} = E_{dir,hor} \cdot \left\{ \max\left(0, \frac{\cos\Theta_{gen}}{\sin\gamma_S}\right) \right\}. \quad (4.16)$$

[Qua96] führt danach die Berechnung vom reflektierten Solarstrahlungsanteil auf Grund der jeweiligen Umgebung nach *Perez* an:

$$E_{refl,gen} = E_{G,hor} \cdot A \cdot \frac{1}{2} \cdot (1 - \cos\gamma_E). \quad (4.17)$$

A ist dabei die sog. *Albedo-Konstante*, und hat für unterschiedliche Umgebungen unterschiedliche Werte. Bei einer Umgebung aus Rasen hat A z.B. den Wert 0,18...0,23, Gras den Wert von $A = 0,25$, Beton den Wert $A = 0,30$, usw. Für die diffuse Solarstrahlung auf eine geneigte Fläche gilt lt. [Qua96] nach *Temps und Coulson*:

$$E_{diff,gen} = E_{diff,hor} \cdot \frac{1}{2} \cdot (1 + \cos\gamma_E) \cdot \left(1 + F \cdot \sin^3\frac{\gamma_E}{2}\right) \cdot \left(1 + F \cdot \cos^2\Theta_{gen} \cdot \cos^3\gamma_S\right), \quad (4.18)$$

mit

$$F = 1 - \left(\frac{E_{diff,hor}}{E_{G,hor}}\right)^2. \quad (4.19)$$

Bei dieser Berechnung wird auch die Nähe des Normalvektors zur Sonnenscheibe beachtet, da die diffuse Solarstrahlung in der Nähe der Sonne anders verteilt ist. Damit ist eine Berechnungskette gegeben, wie aus der Solarstrahlung auf eine Horizontale die Solarstrahlung auf eine geneigte und aus Norden gedrehte Fläche geschätzt werden kann.

4.1.5 Temperatureinfluss auf Solarpaneele

Die Erwärmung der Solarpaneele spielt eine wesentliche Rolle für den Wirkungsgrad eines Solarpaneels. Der angegebene Wirkungsgrad eines Solarpaneels gilt immer für die sog. *Standard Test Conditions (STC) / Standard Test Bedingungen*. Sie sind für kristalline Solarpaneele in der *IEC61215* Norm und für dünnfilm Solarpaneele in der *IEC61646* Norm definiert. STC liegen vor, wenn eine senkrechte direkte Solarstrahlung von $1000 \frac{W}{m^2}$, einer konstant gehaltenen Paneeltemperatur von $25^\circ C$ und einem Luftmassenindex von $AM = 1.5$ vorliegt. Sie treten im normalen Betrieb praktisch nie auf und können auf Dauer nur im Labor erzeugt werden. Die Paneeltemperatur wird dabei durch Kühlung konstant gehalten. Der Luftmassenindex gibt die Atmosphärendicke an, die das Licht beim Durchlaufen der Atmosphäre zurücklegen muss. $AM = 1$ bedeutet, dass das Licht senkrecht auf die Atmosphäre trifft, und nur einen Weg einer Atmosphärendicke zurücklegen muss, bis es auf die Erdoberfläche trifft. Der AM Wert ist dann interessant, wenn man die Spektralverteilung der Solarstrahlung auf Grund des Einfallswinkels des Sonnenlichtes mitberücksichtigen möchte. Die Leistung, die von einem Paneel unter STC abgegeben wird, nennt man auch die *kW-Peak Leistung* bzw. *Nennleistung*. Sie wird um so mehr reduziert, je höher die Paneeltemperatur ist. Dieser Temperaturkoeffizient wird $\alpha_{P, mpp}$ genannt, und liegt bei den modernen Paneelen zwischen $-0.4 \frac{\%}{K}$ und $-0.5 \frac{\%}{K}$. Die für den Temperaturkoeffizienten wirksame Temperatur ist die Differenz $T_{akt} - 25^\circ C$. T_{akt} ist die momentane Paneeltemperatur. Die Paneeltemperatur hängt von der Solarstrahlung auf das Paneel und von der Umgebungstemperatur ab. Dabei ist auch eine Zeitkonstante zu berücksichtigen, da sich die Paneeltemperatur nicht sprunghaft ändern kann. Wird die Paneeltemperatur nicht direkt an den Paneelen gemessen, so kann über die in [Sch02, Kapitel 2] angeführte Berechnung eine Temperaturabschätzung getroffen werden:

Dazu wird die sog. *Normal Operation Cell Temperature (NOCT) / Nennbetriebstemperatur* unter standardisierten Normalbedingungen ($E_{dir} = 800 \frac{W}{m^2}$, Windgeschwindigkeit $v_W = 1 \frac{m}{s}$, einer Umgebungstemperatur von $\vartheta_U = 20^\circ C$ und $AM = 1.5$) verwendet. Sie liegt bei heutigen Paneelen zwischen $42^\circ C$ und $50^\circ C$. Wie in [Sch02] angeführt ist, kann die Paneeltemperatur ϑ_P auf folgende Weise berechnet werden:

$$\vartheta_P(t) = \vartheta_U(t) + c \cdot E_{G,gen}(t - 0.25h). \quad (4.20)$$

$E_{G,gen}$ ist die gesamte Solarstrahlung auf das Paneel. Für ϑ_U wird auf Grund der Temperaturträgheit eines Paneels der Wert $t - 0.25h$ verwendet. $c \left[\frac{^\circ C m^2}{W} \right]$ ist dabei ein Koeffizient, der über die NOCT berechnet werden kann. Die Nennbetriebstemperatur sei dazu

$$\vartheta_P = 45^\circ C. \quad (4.21)$$

Aus Gleichung 4.20 folgt, dass

$$45^\circ C = 20^\circ C + c \cdot 800 \frac{W}{m^2} \quad (4.22)$$

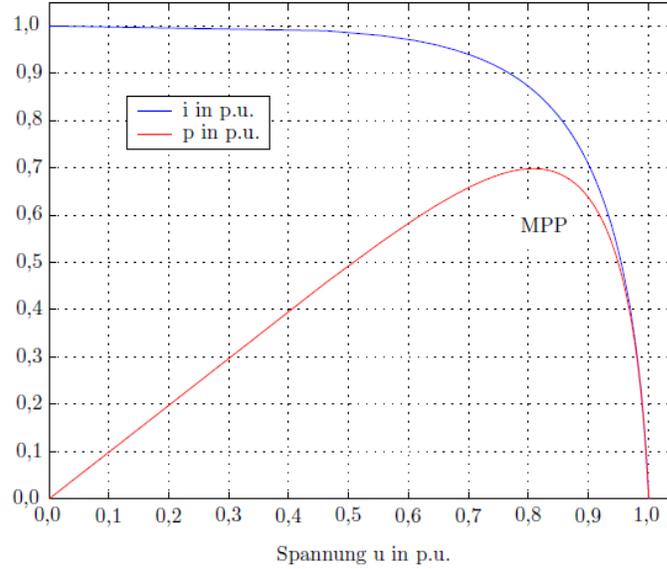


Abbildung 4.4: Strom- und Leistungsverlauf einer Solarzelle, [Bla06, Abb. 2.2].

ist. c ergibt sich daraus ungefähr mit

$$c = 0.03 \left[^\circ C \frac{m^2}{W} \right] \quad (4.23)$$

Für die Berechnung der Paneeltemperatur zum Zeitpunkt t gilt dann

$$\vartheta_P(t) = \vartheta_U(t) + 0.03 \cdot E_{G,gen}(t - 0.25h). \quad (4.24)$$

Für den Temperatur-Reduktionsfaktor k_{PV} zum Zeitpunkt t des Wirkungsgrades unter STC η_{STC} gilt

$$k_{PV}(t) = \alpha_{P,mpp} \cdot (\vartheta_P(t) - 25^\circ C). \quad (4.25)$$

Es ergibt sich ein reduzierter Wirkungsgrad η_{red} zum Zeitpunkt t von

$$\eta_{red}(t) = \eta_{STC} \cdot (1 - k_{PV}(t)). \quad (4.26)$$

4.1.6 Wechselrichter Wirkungsgrad

[Bla06, Kapitel 2] fasst die Grundlagen zum Wirkungsgrad eines Wechselrichters kompakt zusammen:

Die maximale Leistung, die von einem Paneel abgegeben werden kann, ist die Leistung beim sog. *Maximum Power Point (MPP) / Arbeitspunkt der maximalen Leistung* P_{MPP} , graphisch in Abbildung 4.4 veranschaulicht. Die Aufgabe eines modernen Wechselrichters besteht darin, die Spannung der Paneele so zu regeln, dass dieser Arbeitspunkt vorliegt, um die maximale Leistung zu erbeuten. Dabei gilt, dass

$$P_{MPP} = U_{MPP} \cdot I_{MPP} \quad (4.27)$$

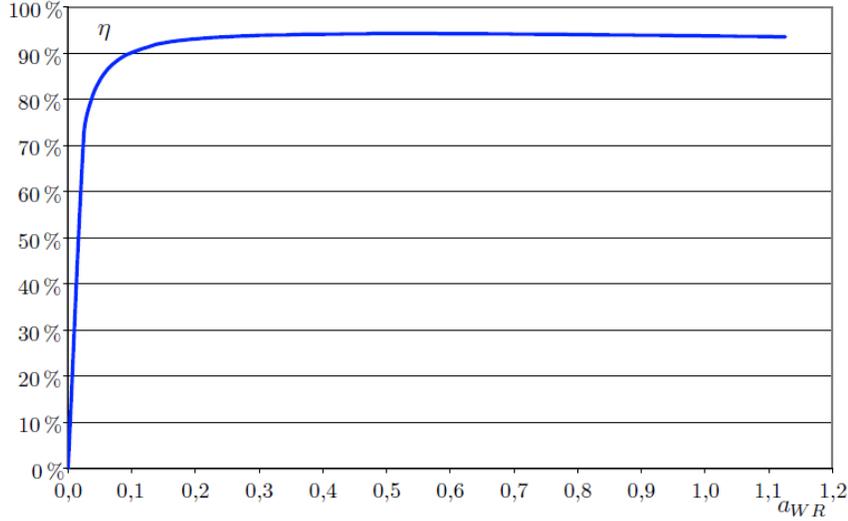


Abbildung 4.5: Wirkungsgrad eines Wechselrichters in Abhängigkeit der Auslastung, [Bla06, Abb. 2.6].

ist. U_{MPP} kann geregelt werden, I_{MPP} ist von der Solarstrahlung auf ein Paneel abhängig. Zusätzlich kommt hinzu, dass der Wechselrichterwirkungsgrad von der Auslastung eines Wechselrichters abhängig ist. Dabei benötigt man den sog. *Dimensionierungsfaktor* k_{DC} eines Wechselrichters. Er errechnet sich aus dem Verhältnis der Nennleistung auf der Gleichspannungsseite zur Nennleistung auf der Wechselspannungsseite, mit

$$k_{DC} = \frac{P_{DC,Nenn}}{P_{AC,nenn}}, \quad (4.28)$$

und liegt bei heutigen modernen Wechselrichtern bei

$$k_{DC} \approx 1.2. \quad (4.29)$$

Weiters benötigt man den sog. *DC-Auslastungsfaktor* a_{DC} , mit

$$a_{DC} = k_{PV} \cdot \frac{E_{G,gen}}{1000 \frac{W}{m^2}}. \quad (4.30)$$

$E_{G,gen}$ ist dabei die solare Einstrahlung, errechnet in Gleichung 4.13, in Kapitel 4.1.4. Der sog. *Wechselrichterauslastungsfaktor* a_{WR} errechnet sich mit

$$a_{WR} = a_{DC} \cdot k_{DC}. \quad (4.31)$$

[Bla06] gibt dazu einen beispielhaften Verlauf des Wirkungsgrades η eines Wechselrichters in Abhängigkeit der Auslastung an, siehe Abbildung 4.5.

Typischerweise ist im Datenblatt des jeweiligen Wechselrichters der Wirkungsgradverlauf auf Grund des Auslastungsfaktors

$$a = \frac{P_{AC,ist}}{P_{AC,Nenn}} \quad (4.32)$$

angegeben. Wird die Leistung $P_{AC,ist}$ eines Wechselrichters nicht gesondert gemessen, da sehr häufig mehrere Ausgangsleistungen unterschiedlicher Wechselrichter und Anlagengrößen zu einer Summe zusammengefasst werden, kann je Wechselrichter der zugehörige sogenannte *Euro-Wirkungsgrad* η_{EU} für den *durchschnittlichen* Wirkungsgrad für Berechnungen verwendet werden. Er wird unter typischen Lastverhältnissen ermittelt.

Ein moderner Wechselrichter hat auch die Möglichkeit, einen Wirkleistungsfaktor $\cos\varphi$ einzustellen. Standardmäßig ist $\cos\varphi = 1$.

4.2 Grundlagen zum regressionsbasierten Ansatz

Die Methode der klassischen linearen Regression wird hierbei nicht behandelt, da sich auf Grund der Analyse der aktuellen Methoden in Kapitel 3 gezeigt hat, dass maschinelle Lernverfahren sehr gerne eingesetzt werden, da es vor allem in der Wasserzuflussprognose, siehe [CAB01] und [SOP⁺07] (analog zum Problem der Wasserflussprognose und somit der Leistung von WLKW), sehr viele Einflussfaktoren gibt, die einen teils sehr komplexen Zusammenhang zwischen dem Wetter und der Leistung bilden. Verwandte Arbeiten unterstellen einen nichtlinearen Zusammenhang. Die Datenanalyse in Kapitel 2 hat auch gezeigt, dass es immer wiederkehrende aber in der Dauer dynamische Verhaltensmuster gibt, die nicht mit einer starren Abbildung beschrieben werden kann. Hierzu sollen sich RNN gut eignen. Außerdem ist ein MLP basiertes neuronales Netz nichts anderes als eine nichtlineare Regression. Daher sollen Modelle mit MLP basierten neuronalen Netzen untersucht werden. [SHP08] bekundet die vergleichbaren Ergebnisse zwischen neuronalen Netzen und der klassischen linearen Regression in der Problematik der Feinstaubprognose, betont aber auch, dass die klassischen linearen Regressionsmodelle *intuitiver* zu verstehen sind, und keine *blackbox* Charakteristik besitzen. Die Daten, welche für ein klassisches Regressionsmodell notwendig wären, sind aber nur teilweise bis nicht vorhanden. Es sind lediglich ausreichende Zeitreihen vom Niederschlag und der Temperatur gegeben.

Die Methode der SVR wird in dieser Arbeit auf Grund des Umfangs nicht untersucht. Zusammenfassend führt dies dazu, dass die Methoden

- Feedforward MLP basierte neuronale Netze,
- RNN wie Elman Netze und
- Echo State Netze

auf ihre Eignung zur Leistungsprognose von WLKW untersucht werden. Daher werden in diesem Kapitel die notwendigen Grundlagen dazu festgehalten.

4.2.1 Multilayer Perzeptron basierte neuronale Netze

Als Grundlage für ein MLP basiertes neuronales Netz dient ein einfaches Perzeptron, siehe Abb. 4.6. Ein Perzeptron kann vereinfacht als ein Schwellwertgatter gesehen werden. Ein Perzeptron hat hierzu mehrere Eingänge x_i , dessen Verbindungen mit w_i gewichtet sind. Die gewichteten Eingangssignale werden im Perzeptron aufsummiert und je nach Summenwert ergibt sich am Ausgang der Wert 0 oder 1. Zusätzlich kann zu den gewichteten Eingängen ein Offset, der sogenannte *Bias* b , angelegt werden. Der Ausgang y eines

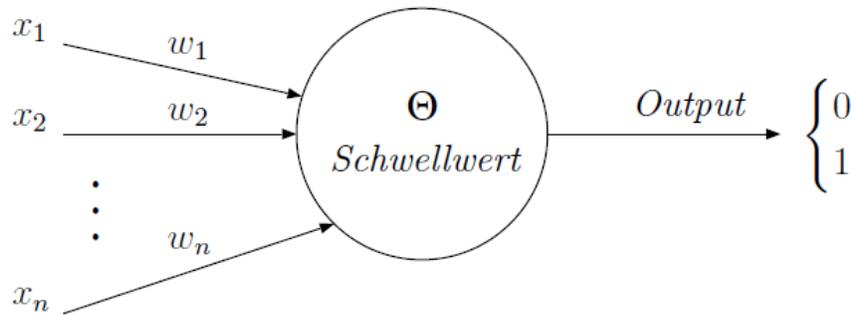


Abbildung 4.6: Schematische Darstellung eines einfachen Perzeptrons, [Pfa01, Abbildung 3.1].

Perzeptrons kann somit über

$$y = \Theta \left(\sum_{i=1}^N (w_i \cdot x_i) + b \right) \quad (4.33)$$

beschrieben werden. [Pfa01] zeigt, wie eine Funktion eines Perzeptrons geometrisch interpretiert werden kann. Ein Perzeptron mit einer Treppenfunktion als Schwellwertgatter ist somit für eine lineare Trennung zweier Klassen gut geeignet. [Pfa01] zeigt auch das *XOR-Problem*, bei dem eine Funktionsnachbildung des logischen XORs nicht mit einem einfachen Perzeptron lösbar ist, und dazu eine weitere (versteckte) Schicht eingeführt werden muss, um ein Netz zu bilden, wie es in Abbildung 4.7 dargestellt ist. Der Gewichtsvektor für die Realisierung der XOR Funktion würde dazu $(x_1, x_2, x_3) = (1, 1, -2)$ betragen. Das Problem kann aber auch mit einem sogenannten *hidden/versteckten Neuron* gelöst werden, wie es Abbildung 4.8 zeigt. Netze mit versteckten Neuronen werden auch als *Backpropagation*-Netze bezeichnet, da die Methode des Backpropagation eine Methode des Netztrainings ist.

Die Schwellwertfunktion Θ ist bei neuronalen Netzen zur Regression ist keine reine Treppenfunktion, sondern kann dabei z.B. die *tansig*, die *logsig*, oder auch die *purelin* Funktion sein, siehe beispielhafte Verläufe in Abbildung 4.9.

$$\text{tansig}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (4.34)$$

$$\text{logsig}(x) = \frac{1}{1 + e^{-x}} \quad (4.35)$$

$$\text{purelin}(x) = k \cdot x \quad (4.36)$$

Aufgrund der stetigen und differenzierbaren Schwellwertfunktionen, sind auch die Signale stetig und differenzierbar. Je nach Summenwert liegt dann am Ausgang ein Wert an. Ein Perzeptron mit n Inputs, n Gewichten und einer stetigen Aktivierungsfunktion und einem Output wird als *Neuron* bezeichnet. Ein Netz, bestehend aus solchen Neuronen, wird als neuronales Netz bezeichnet. Neuronale Netze werden trainiert, um eine Abbildung eines Inputvektors auf einen Outputvektor zu lernen.

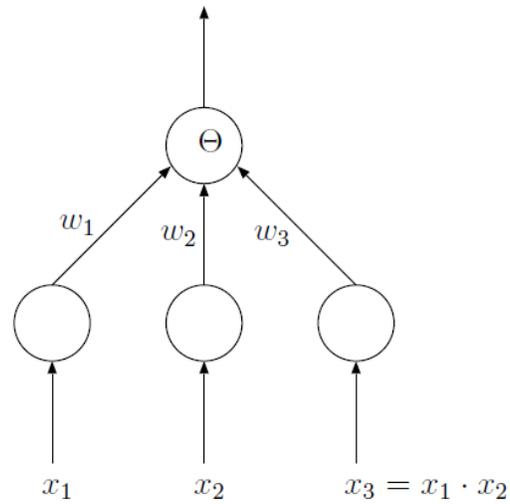


Abbildung 4.7: Perzeptron-Netz zur XOR Problemlösung, [Pfa01, Abbildung 4.5].

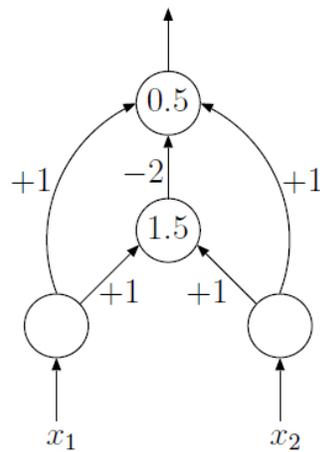
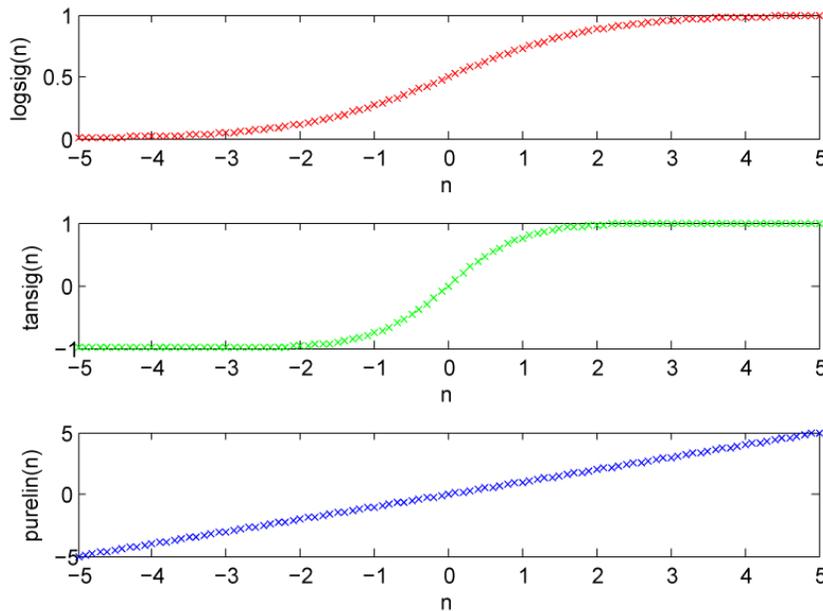


Abbildung 4.8: Ein dreischichtiges Perzeptron-Netz (mit einem hidden Neuron und hidden Layer) zur Lösung des XOR Problems, [Pfa01, Abbildung 4.6].

Abbildung 4.9: Eine Realisierung der *logsig*, *tansig* und der *purelin* Funktion.

Trainingsansatz

(Die Informationen dieses Unterkapitels stammen aus [Maa07])

Abbildung 4.10 zeigt ein Neuron aus der j -ten Schicht eines (feedforward) neuronalen Netzes aus Abbildung 4.11. Es hat n Inputs (x_1, \dots, x_n) , ein Summenglied

$$net_j = \sum_{i=1}^n x_i \cdot w_{ij} \quad (4.37)$$

und ein Funktionselement

$$o_j = f(net_j), \quad (4.38)$$

wobei o_j der Output des Neurons ist. Zusätzlich kann ein Bias b_j in das Neuron in der Schicht j eingebracht werden. Das Ziel beim Trainieren ist es, z.B. das MSE Fehlermaß, im Folgenden als *Fehler* bezeichnet, bei l Inputvektoren und gegebenem Gewichtsvektor \mathbf{w}

$$MSE(H_{\mathbf{w}}) = \frac{\sum_{i=1}^l (\mathbf{w} \cdot \mathbf{x}_i - y_i)^2}{l} \quad (4.39)$$

zu minimieren, um die optimale *Hypothese* H , dh. optimale Abbildungsfunktion von Input des Netzes zum Output des Netzes, zu finden. l ist die Anzahl der Inputvektoren zu den Targetvektoren $\langle \mathbf{x}_i, \mathbf{y}_i \rangle$ des Trainingsdatensatzes. Der Fehler lässt sich grundsätzlich auf zwei Arten minimieren:

- Exakte Lösung - nur im Falle einer linearen Hypothese H möglich, und
- Gradientenabstieg.

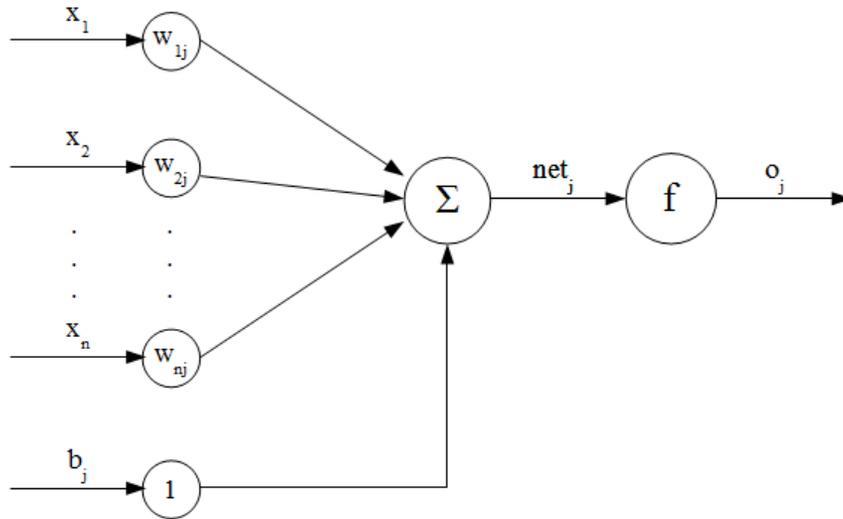


Abbildung 4.10: Skizze eines Neurons.

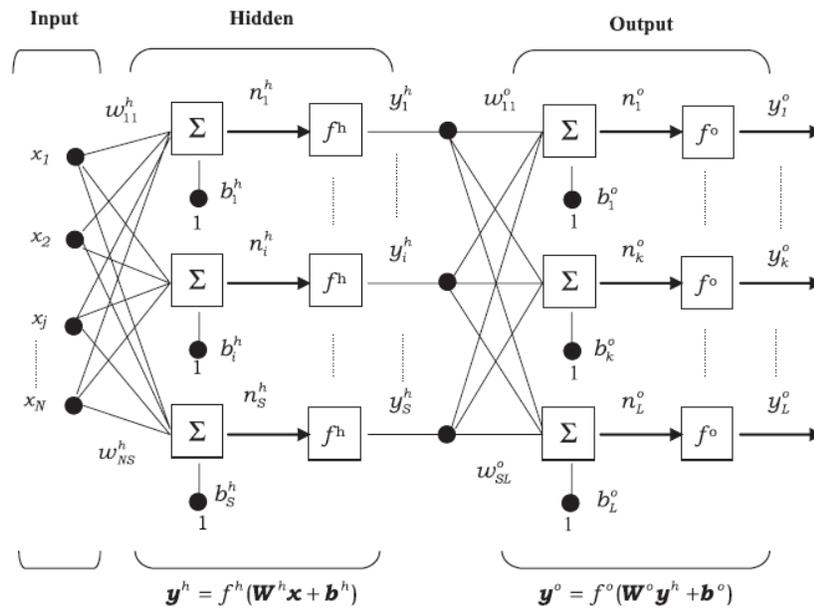


Abbildung 4.11: Struktur eines multilayer Perzepton basierten neuronalen Netz, [ABIBM06, Fig. 1].

Die exakte Lösung für den optimalen Gewichtsvektor \mathbf{w}_{opt} im Falle einer linearen Hypothese H ist die Lösung der klassischen linearen Regression mittels eines *Kleinsten-Quadrate (KQ) Schätzer* und ergibt

$$\mathbf{w}_{\text{opt}} = (X^T \cdot X)^{-1} \cdot X^T \cdot \mathbf{y}. \quad (4.40)$$

Die Lösung erfolgt direkt durch 0 setzen von

$$\frac{\delta}{\delta w_j} MSE(H\mathbf{w}) = \frac{2 \cdot \sum_{i=1}^l (\mathbf{w} \cdot \mathbf{x}_i - y_i) \cdot x_{ji}}{l}, \quad (4.41)$$

dh.

$$\frac{\delta}{\delta w_j} MSE(H\mathbf{w}) = 0. \quad (4.42)$$

Die Lösung für \mathbf{w}_{opt} folgt dann aus

$$\nabla_{\mathbf{w}} \cdot (X \cdot \mathbf{w} - \mathbf{y})^T \cdot (X \cdot \mathbf{w} - \mathbf{y}) = 2 \cdot X^T \cdot (X \cdot \mathbf{w} - \mathbf{y}), \quad (4.43)$$

wobei X die Matrix mit den Inputvektoren ist. \mathbf{y} ist der Vektor der Targets.

Der Gradientenabstieg bei Netzen ohne einer versteckten Schicht erfolgt durch iteratives Vorgehen. Dabei erfolgt das Training über mehrere *Epochen*, dh. die Inputvektoren werden mehrfach in das Netz gespeist. Nach jeder Epoche wird der Gewichtsvektor auf folgende Art *korrigiert*:

$$\mathbf{w}_{\text{neu}} = \mathbf{w}_{\text{alt}} + \Delta \mathbf{w}, \quad (4.44)$$

wobei der Korrekturwert für \mathbf{w}_{neu}

$$\Delta \mathbf{w} = -\alpha \cdot \nabla E(\mathbf{w}) \quad (4.45)$$

ist.

$$E(\mathbf{w}) = MSE(H\mathbf{w}) \quad (4.46)$$

α nennt man dabei die *Schrittweite* oder *Lernrate*. Diesen Faktor kann man als *Impuls* in Richtung des Gradienten verstehen. Es folgt, dass jedes Gewicht durch

$$w_j^{\text{neu}} = w_j^{\text{alt}} - \alpha \cdot \sum_{i=1}^l (\mathbf{w}^{\text{alt}} \cdot \mathbf{x}_i - y_i) \cdot x_{ji} \quad (4.47)$$

korrigiert werden kann. Hier treten zwei unterschiedliche Arten des Trainings zu Tage:

- Training im batch mode - Update der Gewichte nach einer Abbildung aller Inputvektoren, also erst nach einer Epoche.
- Online Training - Update der Gewichte nach jedem Input (die Abbildungsreihenfolge geht in die Hypothesenfindung mit ein).

Beim Gradientenabstieg ist nicht garantiert, dass das globale Minimum des Fehlers gefunden wird. Dies tritt vor allem dann zu Tage, wenn starke lokale Minima vorhanden sind, beispielhaft in Abbildung 4.12 skizziert. Um kleine lokale Minima zu *überwinden* muss α geeignet gewählt werden. Der Gradientenabstieg bei Netzen mit versteckten Schichten erfolgt unter anderem mit der *Backpropagation* Methode. Dabei wird der Fehler auf eine

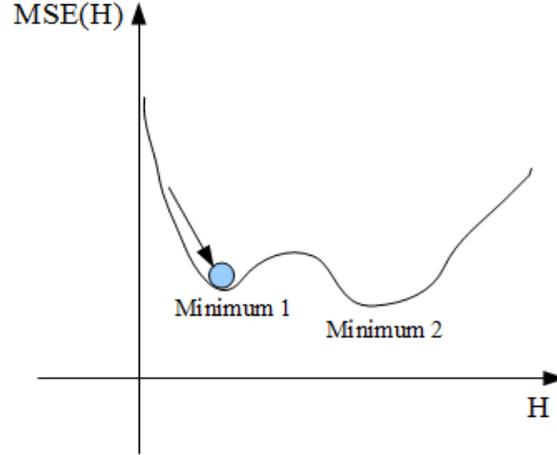


Abbildung 4.12: Gradientenabstieg bei starken lokalen Minima.

spezielle Art *zurückverfolgt*. Es erfolgt sozusagen eine *Schuldzuweisung* an jede gewichtete Verbindung. Die gewichteten Verbindungen eines Neurons innerhalb eines Netzes mit versteckten Schichten, wie es in 4.11 gezeigt ist, erfolgt ebenfalls durch die Methode des Gradientenabstiegs, jedoch ist der Gradient des Fehlers nun nur mehr über eine verkettete Ableitung zu finden. Dh., wie bei 4.44 lautet nun

$$w_{ji}^{neu} = w_{ji}^{alt} - \alpha \cdot \nabla E(\mathbf{w}). \quad (4.48)$$

Es gilt, dass

$$o_j^{neu} = f(net_j) \quad (4.49)$$

ist.

$$\nabla E(\mathbf{w}) = d_j \cdot o_j \quad (4.50)$$

setzt sich zusammen aus der Fehlerzuweisung d_j und dem Output des Neurons, wobei die Berechnung für d_j abhängig davon ist, ob ein Neuron in der Ausgabeschicht oder in einer versteckten Schicht ist: Für den Fall dass das Neuron in der Ausgabeschicht liegt, ist

$$d_j = f'(net_j) \cdot (y_i - o_j), \quad (4.51)$$

wobei das im net_j enthaltene Gewicht w_{ij} das alte Gewicht ist. Andernfalls gilt, dass

$$d_j = f'(net_j) \cdot \sum_k d_k \cdot w_{jk}^{alt}. \quad (4.52)$$

k bezeichnet dabei das k -te Neuron in einem Hiddenlayer. Auch hier ist das im net_j enthaltene Gewicht w_{ij} das alte Gewicht. Die Verkettete Ableitung lautet dann:

$$\nabla E(\mathbf{w}) = \frac{\delta E(\mathbf{w})}{\delta w_{ij}} = \frac{\delta E(\mathbf{w})}{\delta o_j} \cdot \frac{\delta o_j}{\delta net_j} \cdot \frac{\delta net_j}{\delta w_{ij}} \quad (4.53)$$

Diese Grundlage vom Backpropagation wird von diversen Algorithmen zum Training von neuronalen Netzen verwendet.

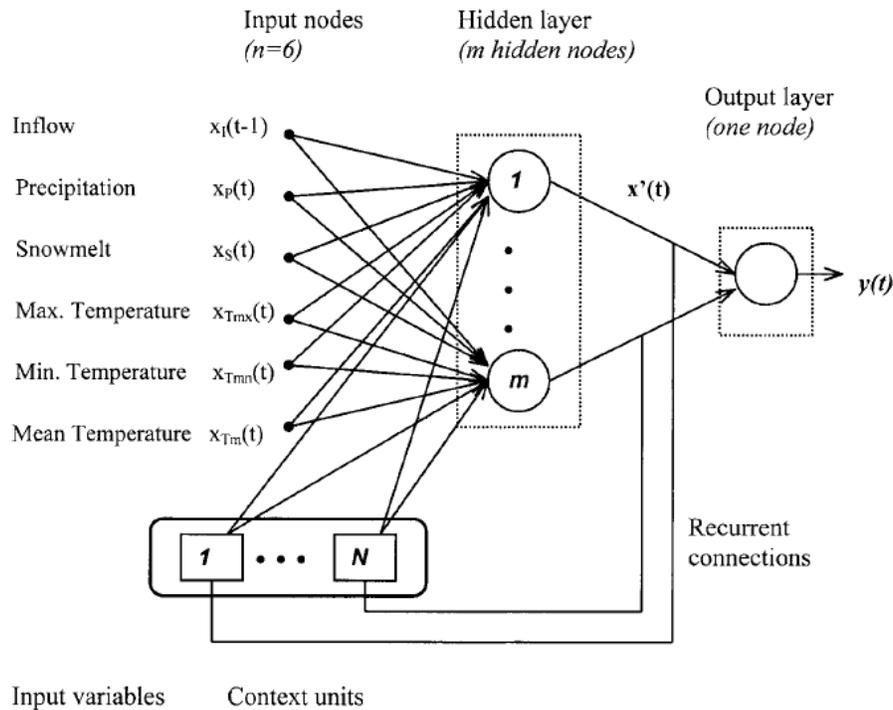


Abbildung 4.13: Typische Elman Netzstruktur als eine Realisierung von einem rekurrenten neuronalen Netz, [CAB01, Fig. 4].

Zum Schluss sei hier noch ganz kurz das Prinzip der Regularisierung erwähnt. Ein Netz mit hoher Neuronenanzahl neigt dazu, das Trainingsset schnell auswendig zu lernen, das heißt, die Gewichte im Netz werden so angepasst, dass das Netz exakt die Trainingsbeispiele abbildet, aber keine generalisierte, allgemein gültige Abbildung bildet. Es tritt sogenanntes *Overfitting* auf, [CLG]. Eine Möglichkeit dies zu verhindern ist, *Kosten* für die Verbindungsgewichte eines Netzes einzuführen. Dieses Prinzip nennt man im einfachsten Fall *weight decay*, [KH95]. Der MSE wird dadurch z.B. auf folgende Weise modifiziert:

$$E(\mathbf{w})_{decay} = E(\mathbf{w}) + \lambda \sum_{i,j} w_{ij} \quad (4.54)$$

λ beschreibt hierbei den *Strafterm* für hohe Gewichte. Zusätzlich gibt es das Prinzip des *early stopping*, [CLG]. Dabei wird mit dem Trainieren aufgehört, wenn der Fehler auf das Testset markant zu steigen beginnt. Dies ist ebenfalls ein Zeichen für den Beginn von *Overfitting*. λ wird auch oft als α angegeben, wobei in Matlab ein sehr kleines α eine starke Regularisierung bedeutet.

4.2.2 Elman rekurrente neuronale Netze

Elman Netze, siehe Abbildung 4.13, sind sehr häufig verwendete Netzstrukturen bei RNN und eignen sich wie alle anderen RNN zur Modellierung eines Netzes mit dynamischen

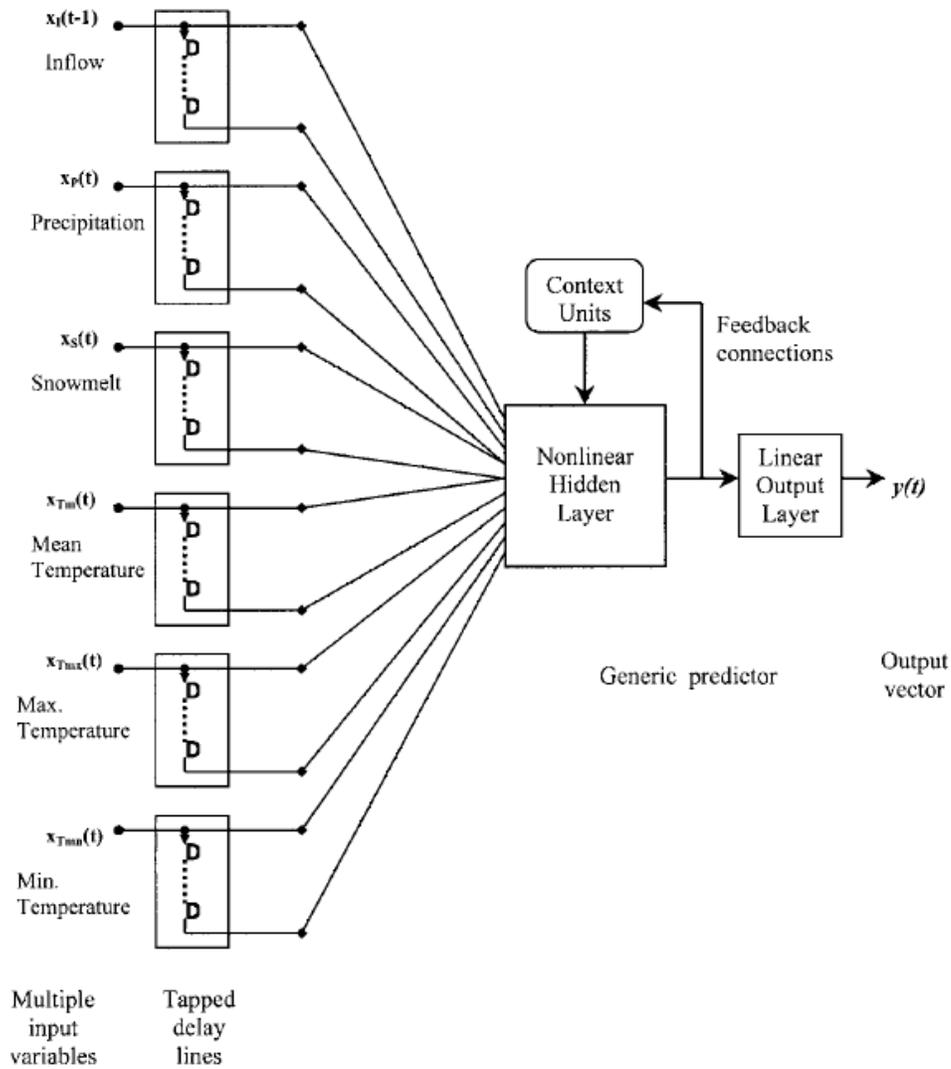


Abbildung 4.14: Input delayed neuronales Netz (ohne Rückkopplung) und time delayed neuronales Netz (mit Rückkopplung), [CAB01, Fig. 3].

Gedächtnis. [CAB01] beschreibt dies noch etwas genauer, und zeigt den Unterschied von IDNN und zeitverzögerte/time delayed rekurrente neuronale Netze, siehe Abbildung 4.14, und meint dazu, dass sich ein time delayed rekurrentes neuronales Netz von einem RNN dadurch unterscheidet, dass ein time delayed rekurrentes neuronales Netz zusätzlich zum adaptiven Gedächtnis (Speicher), auch zusätzlich einen statischen Speicher besitzt. Ein IDNN besitzt also nur ein fixes statisches Vergangenheitsfenster. Dies ist bei reinen feed-forward neuronalen Netzen der Fall. Ein IDNN wendet nur das Prinzip des *sliding Windows/schleifenden Fensters* an. Eine Methode, bei der immer der aktuelle Moment und ein bestimmter Ausschnitt mit einer bestimmten Dauer in die Vergangenheit als Input in das neuronale Netz geschickt wird. Beim Training genauso wie im Betrieb. Somit können kurzzeitige Abhängigkeiten mit modelliert werden, eignen sich jedoch nicht dafür, zeitliche Patterns zu erkennen, die in ihrer Dauer variabel sein können.

Bei einem RNN wird also das zeitliche Verhalten implizit und nicht explizit modelliert. Jedoch hat ein RNN einen hohen Trainingsaufwand. Einerseits hat man die selben Freiheitsgrade wie bei einem feedforward neuronalen Netz:

- Inputs,
- Anzahl der Hiddenlayer und Neuronen pro Layer,
- Gewichte der Verbindungen zwischen den Neuronen und
- Aktivierungsfunktionen.

Andererseits führt die Rückkopplung (recurrent connections) dazu, dass zum Trainieren die Gradientenbildung vom Error E nicht trivial ist, da auch der Fehler eine Funktion des zeitlichen Verhaltens des Netzes ist, wie die Autoren von [CAB01] meinen.

Der Error E ist hierbei, wie bei den MLP basierten neuronalen Netzen, der Unterschied des Outputs der erlernten Funktion zum Output der Trainingsfunktion bei gegebenem Input. *Context Units*, wie sie in Abbildung 4.13 eingezeichnet sind, sind Neuronen für die *Verzögerung und Verteilung* eines rückgekoppelten Signals auf $1, \dots, m$ Neuronen im Hiddenlayer. Sie merken sich den letzten Zustand von $t - 1$, der vom Zustand $t - 2$ abhängt, usw.

Das Elman Netz ist nach *Jeffrey L. Elman* benannt, der dieses Netz in [Elm90] vorstellt. Beim Elman Netz, Abbildung 4.13, sind die Verbindungen von den Inputs und von den Contextneuronen zu den Neuronen im ersten Hiddenlayer gewichtet. Ebenfalls bestehen Gewichte von den Neuronen im letzten Hiddenlayer zu den Outputneuronen / zum Outputneuron. Die Verbindungen der Rückkopplungen sind stets mit dem Gewicht 1 beaufschlagt. Somit wird der vergangene Zustand im Hiddenlayer nicht bereits beim *Übertragen* in den Zwischenspeicher verändert, sondern erst bei der Zufuhr zurück ins Netz.

[CAB01] beschreibt dazu das Netz in Form einer Zustandsraumbeschreibung:

$$y_j(t+1) = A \cdot x'(t+1), \quad (4.55)$$

bzw.

$$y_j(t+1) = \Theta[A \cdot x'(t+1)], \quad (4.56)$$

und

$$x'(t+1) = g[W_m x'(t) + W_{m_0} x(t)], \quad (4.57)$$

wobei gilt dass:

- $y_j(t+1)$ ist der nächste Output zum Zeitpunkt $t+1$.
- A ist die Matrix der Gewichte von den Hiddenneuronen zu den Outputneuronen.
- $x'(t+1)$ sind die Signale der Hiddenneuronen zum Zeitpunkt $t+1$.
- $g[\cdot]$ ist die Aktivierungsfunktion der Hiddenneuronen.
- W_m ist die Gewichtsmatrix der Contextneuronen zu den Hiddenneuronen (Eingangsseitig).
- W_{m_0} ist die Gewichtsmatrix der Inputs zu den Hiddenneuronen.

Trainingsansatz

In [WP90] wird der Ansatz des *backpropagation through time* und der Lernalgorithmus *real time recurrent learning* grundlegend erklärt, sowie darauf aufbauende Algorithmen skizziert. Der folgende Absatz gibt [WP90] inhaltlich wieder:

Hierbei wird *backpropagation through time* und darauf aufsetzende Lernalgorithmen für allgemeine RNN beschrieben. Zuerst werden einige Annahmen und Definitionen gegeben:

- Es wird ein RNN mit semilinearen Einheiten angenommen.
- Die Zeitreihe der Input/Outputvektoren ist diskret.
- Es gibt n Units/Netzeinheiten.
- Es gibt m externe Inputs.
- $\mathbf{y}(t)$ beschreibt den Outputvektor zur Zeit t (n -tuple).
- $\mathbf{x}^{net}(t)$ beschreibt den Inputvektor zur Zeit t (m -tuple).
- $\mathbf{x}(t)$ sei $\langle \mathbf{x}^{net}(t), \mathbf{y}(t) \rangle$.
- Die Menge $U = 1, \dots, k$. x_k ist dabei die k -te Komponente von \mathbf{x} . x_k ist der Output einer Einheit im Netz. Eine Einheit kann dabei ein einzelnes Neuron oder ein Netz im Netz sein.
- Die Menge $I = 1, \dots, k$. x_k ist dabei k -te externe Input einer Einheit, wobei
 - $x_k(t) = x_k^{net}(t)$, wenn $k \in I$,
 - $x_k(t) = y_k(t)$, wenn $k \in U$.
- Die Gewichtsmatrix \mathbf{w} eines Netzes beschreibt dabei die Gewichte zwischen der j -ten Einheit zur i -ten Einheit mit den einzelnen Gewichten w_{ij} , wobei die Gewichte von den Inputs zu den zugehörigen Units 1 ist. Dabei ist $j \in U$ oder $j \in I$, je nachdem ob das Signal von einem Unitoutput oder von einem Netzininput stammt. Anmerkung: Bei Elman Netze ist die Rückkopplungen von den Hiddenunits zu den Contextunits stets mit 1 gewichtet.

- Ein Bias wird dadurch berücksichtigt, in dem ein Inputsignal mit dem Wert 1 gewichtet mit w_{ij} in das Netz gekoppelt wird.

Der *net*-Input der k -ten Einheit zur Zeit $t + 1$ wird mit

$$s_k(t+1) = \sum_{l \in U \cup I} w_{kl} \cdot x_l(t) \quad (4.58)$$

beschrieben. Der Output einer Einheit zur Zeit $t + 1$ wird durch

$$y_k(t+1) = f_k(s_k(t+1)) \quad (4.59)$$

beschrieben. f_k sei (typischerweise) die logsig Funktion:

$$f_k(s_k(t)) = \frac{1}{1 + e^{-s_k(t)}}. \quad (4.60)$$

Es gilt, dass:

$$f'_k(s_k(t)) = y_k(t) \cdot [1 - y_k(t)] \quad (4.61)$$

ist. Außerdem gilt, dass der erste Einfluss auf das Netz bei einem Input zur Zeit t frühestens zur Zeit $t + 1$ erfolgen kann. Die ist z.B. durch eine 1-Schritt-Zeitverzögerung (=Contextunit) realisierbar.

Lernansätze verfolgen hierbei den Ansatz, dass das notwendige Δw_{ij} gesetzt wird, damit der Fehler zwischen dem Output des Netzes und dem Target minimal wird, wie bei den MLP basierten feedforward neuronalen Netzen. $T(t)$ sei ein Set von $k \in U$ für die ein Targetvalue $d_k(t)$ (Outputs aus den Trainingsdaten) existiert.

$$e_k(t) = d_k(t) - y_k(t), \quad (4.62)$$

wenn $k \in T(t)$, ansonsten 0. $d_k(t)$ ist dabei der Targetoutput einer Einheit, $y_k(t)$ der Istoutput. Daraus wird ein Fehlermaß

$$E(t) = \frac{1}{2} \sum_{k \in U} [e_k(t)]^2 \quad (4.63)$$

definiert. Ein Lernalgorithmus versucht nun das Fehlermaß zu minimieren, indem eine *Schuldzuweisung* an die einzelnen Gewichte erfolgt, und gegen den Gradienten die Δw_{ij} Werte gesetzt werden. Dies ist der Vorgang von Backpropagation, wie auch in Kapitel 4.2.1 für MLP basierte feedforward neuronale Netze beschrieben wurde. Dh.

$$E^{total}(t', t) = \sum_{\tau=t'+1}^t E(\tau) \quad (4.64)$$

ist der gesamte Fehler des Netzes im Zeitintervall t' bis t , und muss z.B. mittels Backpropagation minimiert werden. Der Gradient dazu errechnet sich mit

$$\nabla_{\mathbf{w}} E^{total}(t', t) = \sum_{\tau=t'+1}^t \nabla_{\mathbf{w}} E(\tau). \quad (4.65)$$

Die Gewichtsänderungen der Verbindungen werden gegen den mit einem Impuls α gewichteten Gradienten geändert (Gradientenabstieg):

$$\Delta w_{ij} = -\alpha \frac{\delta E^{total}(t', t)}{\delta w_{ij}}. \quad (4.66)$$

α beschreibt die Lernrate, wie in Kapitel 4.2.1. Die Änderung der Gewichte erfolgt nun bei jeder Abbildung des Inputvektors auf den Targetvektor beim Training. Dies ist auch der Grund, wieso das Training im Vergleich zu feedforward Netzen um ein Vielfaches länger dauert. Die Lernalgorithmen versuchen nun möglichst effizient die Δw_{ij} Gewichtsänderungen zu berechnen.

Ein bekannter Ansatz ist der zuvor erwähnte real time recurrent learning Algorithmus. Dieser wird ebenfalls in [WP90] beschrieben. Er wird hier im Folgenden inhaltlich aus [WP90] wiedergegeben:

Es wird dabei ein Set von Werten

$$p_{ij}^k = \frac{\delta y_k(t)}{\delta w_{ij}} \forall k \in U, i \in U, j \in U \cup I \quad (4.67)$$

in jedem Zeitschritt berechnet und zur Gewichtsveränderung verwendet, wobei alle p_{ij}^k zu Beginn mit 0 initialisiert werden. Diese Werte sind Gewichtsänderungsakkumulatoren. Die Änderung dieser Akkumulatoren errechnen sich für einen darauffolgenden Zeitschritt mit

$$p_{ij}^k(t+1) = f'_k(s_k(t+1)) \cdot \left(\sum_{l \in U} p_{ij}^l(t) \cdot w_{kl} + \delta_{ik} x_j(t) \right), \quad (4.68)$$

wobei δ_{ik} das Kronecker-Delta ist, dh. $\delta_{ik} = 1$ wenn $i = k$, sonst 0. Die Gewichtsänderungen errechnen sich mit

$$\Delta w_{ij} = -\alpha \cdot \frac{\delta E(t)}{\delta w_{ij}} = \alpha \cdot \sum_{k \in U} e_k(t) \cdot p_{ij}^k(t). \quad (4.69)$$

Die Autoren von [WP90] meinen dazu, dass dieser Ansatz bei n Einheiten im Netz $O(n^2)$ Gewichte, einen Speicheraufwand von $O(n^3)$ und eine Laufzeit von $O(n^4)$ besitzt. Verbesserungen sind z.B. Ansätze wie:

- Hinzufügen einer feedforward Struktur in jedem Zeitschritt, wobei dann epochenweise, wie bei *normalen* feedforward Strukturen, Backpropagation gemacht werden kann.
- Oder durch parallele Änderung der Gewichte und Bestimmung vom nächsten Gradienten. Dazu ist aber erhöhter Speicheraufwand notwendig, da in jedem Schritt das komplette Netz im Speicher abgebildet werden muss.

4.2.3 Echo State Netze

Laut [Jae10] besteht ein Echo State Netz aus einem Input (Inputs und Gewichtungsmatrix der Inputs auf die Zustandsneuronen), einem sogenannten Reservoir (Zustandsneuronen

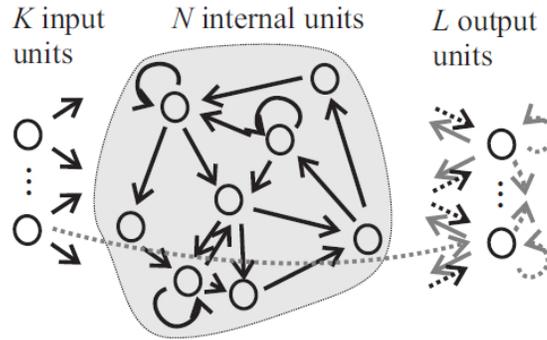


Abbildung 4.15: Netzstruktur eines Echo State Netzes, [Jae10, Fig. 1].

und Verbindungsmatrix untereinander) und einem Readout (Outputgewichtung und Outputneuronen). Ein Echo State Netz ist in Abbildung 4.15 schematisch dargestellt. Die Inputs können direkt auf die Outputs *durchgreifen*. Die Outputs können auf die Zustandsneuronen im Reservoir zurückgekoppelt werden. Die Formale Beschreibung laut [Jae10] lautet wie folgt:

Im Folgenden sollen Bezeichnungen und Gegebenheiten aufgezählt werden.

- Es gibt K Inputs, L Outputs und N interne Einheiten (Zustandsneuronen). N bezeichnet auch die Netzdimension vom Reservoir.
- Ein Inputvektor wird mit $\mathbf{u}(\mathbf{t}) = (u_1(t), \dots, u_K(t))$ bezeichnet.
- Der interne Zustand wird mit $\mathbf{x}(\mathbf{t}) = (x_1(t), \dots, x_N(t))$ bezeichnet.
- Der Outputvektor wird mit $\mathbf{y}(\mathbf{t}) = (y_1(t), \dots, y_L(t))$ bezeichnet.
- Generell kann ein Neuron beliebige Aktivierungsfunktionen annehmen, wie z.B. die purelin-, logsig- oder tansig-Funktion.
- Es gibt eine $N \times K$ Matrix für die Inputgewichtung, normalverteilt mit $\mu = 0$ und $\sigma^2 = 1$. Sie wird mit einer bestimmten Eingang-Verbindungswahrscheinlichkeit, sogenannte *inputconnectivity* erzeugt. Im Folgenden sei sie als $W^{in} = (\mathbf{w}_{ij})^{in}$ bezeichnet.
- Es gibt eine $N \times N$ Matrix für die Gewichtung der Verbindungen der Zustandsneuronen, normalverteilt mit $\mu = 0$ und $\sigma^2 = 1$. Sie wird mit einer bestimmten Verbindungswahrscheinlichkeit, sogenannte *connectivity*, erzeugt. Im Folgenden sei sie als $W = (\mathbf{w}_{ij})$ bezeichnet. Es sind auch Rückkopplungen eines Zustandsneurons auf sich selbst erlaubt.
- Es gibt eine $L \times (K+N+L)$ Matrix für die Verbindungsgewichte vom Reservoir zu den Outputneuronen. Ihre Gewichte werden trainiert. Die Verbindungswahrscheinlichkeit vom Reservoir zum Output ist mit 100% festgelegt. Sie wird im Folgenden $W^{out} = (\mathbf{w}_{ij})^{out}$ genannt.
- Außerdem gibt es eine $N \times L$ Matrix für die Verbindungsgewichte der Rückkopplungen von den Outputneuronen zum Reservoir, gleichverteilt und mit $\mu = 1$ erzeugt.

- Alle Verbindungsgewichte können reelle Werte annehmen.

Die Inputsequenz U für ein Echo State Netz muss eine kompakte Eingangsfolge sein. Ein Echo State Netz muss die *echo-state* Eigenschaft aufweisen. Das bedeutet, dass sich ein aktueller Zustand eines nicht zurückgekoppelten Netzes aus

$$\mathbf{x}_n = \mathbf{E}(\dots, \mathbf{u}(n-1), \mathbf{u}(n)) \quad (4.70)$$

ergibt. \mathbf{E} beschreibt hierbei die Echo-Funktion. Vergleichbar mit einer Impulsantwort eines Filters.

$$\mathbf{E} = (e_1, \dots, e_N), \quad (4.71)$$

wobei gilt, dass

$$e_i : U^{-N} \rightarrow \mathbb{R}, \quad (4.72)$$

U^{-N} bezeichnet eine kompakte linksseitige Statesequenz bestehend aus N Zeitschritten. Damit das Netz nicht instabile States generiert, soll gelten, dass der Spektralradius der Matrix W kleiner 1 ist, dh. dass

$$|\lambda_{max}(W)| \leq 1 \quad (4.73)$$

gelten muss, wobei $\lambda_{max}(W)$ der größte Eigenwert der Matrix W ist. Ist diese Bedingung erfüllt, kann eine Abbildung vom Input zum Output gefunden werden. Ausgehend von einem initialen Zustand der Zustandsneuronen \mathbf{x} , wird der darauffolgende Zustand des Echo State Netzes mit

$$\mathbf{x}(n+1) = \mathbf{f} \left(W^{in} \cdot \mathbf{u}(n+1) + W \cdot \mathbf{x}(n) + W^{back} \cdot \mathbf{y}(n) \right) \quad (4.74)$$

beschrieben. $\mathbf{f} = (f_1, \dots, f_N)$ beschreibt hierbei die Aktivierungsfunktion des jeweiligen Zustandsneurons. Der Ausgang eines Echo State Netzes kann dann mit

$$\mathbf{y}(n+1) = \mathbf{f}^{out} \left(W^{out} \cdot (\mathbf{u}(n+1), \mathbf{x}(n+1), \mathbf{y}(n)) \right) \quad (4.75)$$

gefunden werden. $\mathbf{f}^{out} = (f_1^{out}, \dots, f_L^{out})$ ist die Aktivierungsfunktion der Outputneuronen (Readoutfunktion).

Der große Vorteil, der bei Echo State Netzen im Vergleich zu anderen neuronalen Netzen vorliegt, ist jener, dass das Trainieren und Lernen trivial und schnell ist.

Trainingsansatz

Gegeben sei eine Trainingsinputsequenz und eine zugehörige Outputsequenz

$$(\mathbf{u}_{teach}(n), \mathbf{y}_{teach}(n))_{n=0, \dots, n_{max}} \quad (4.76)$$

aus einem kompakten Set U .

- Die Zustandsneuronen werden mit $\mathbf{x}(0)$ initialisiert.
- Am Eingang eines Echo State Netzes, erstellt wie in 4.2.3 beschrieben, wird die Trainingsinputsequenz Zeitschritt für Zeitschritt abgebildet.
- Dabei wird ein Update der Zustandsneuronen, wie in 4.2.3 beschrieben, durchgeführt.

- Um transiente Zustände vergehen zu lassen, wird erst nach hinreichend vielen Inputs ein Abbild der Zustandsneuronen, Zeitschritt für Zeitschritt in einer Matrix zwischengespeichert. Sie sei im Folgenden als *state collect Matrix*

$$W_{collect} = (x_{collect}(n_{min}), \dots, x_{collect}(n_{max})) \quad (4.77)$$

bezeichnet, wobei sich $x_{collect}(n)$ aus

$$\mathbf{x}_{collect}(n+1) = \mathbf{f} \left(W^{in} \cdot \mathbf{u}_{teach}(n+1) + W \cdot \mathbf{x}_{collect}(n) + W^{back} \cdot \mathbf{y}_{teach}(n) \right) \quad (4.78)$$

ergibt.

- Es gilt, dass

$$\mathbf{y}_{teach}(n) = (y_{teach,1}(n), \dots, y_{teach,L}(n)). \quad (4.79)$$

- Für jedes $j = 1, \dots, L$ kann dann w_{ij}^{out} über Minimierung des MSE errechnet werden

$$mse_{train,j} = 1 / (n_{max} - n_{min}) \cdot \sum_{n=n_{min}}^{n_{max}} \left((f_j^{out})^{-1} - \sum_{i=1}^N w_{ji}^{out} x_i(n) \right)^2. \quad (4.80)$$

- Um daraus die Gewichtsmatrix $W^{out} = w_{ji}^{out}$ zu erhalten, wird lineare Regression angewendet. Im praktischen Fall bedeutet dies, dass z.B W^{out} über die Pseudoinverse von $W_{collect}$ berechnet werden kann:

$$W^{out} = \mathbf{u}_{teach} \cdot (W_{collect})^{-1}. \quad (4.81)$$

- Für den Fall eines zurückgekoppelten Netzes, kommt der Output als weiterer Input in das Echo State Netz. Für das Training wird dazu $y_{teach}(n)$ in die Outputneuronen geprägt, dh. \mathbf{u}_{teach} mit \mathbf{y}_{teach} zu einem Vektor zusammengefasst. Weiters wird W^{in} und W^{back} zu einer Matrix zusammengefasst.

Eine Minimierung des MSE kann aber auch über eine weitere elegante Methode erreicht werden, in dem man alle Verbindungen vom Reservoir zum Output auf ein Perzeptron hängt, und über wenige Backpropagationsschritte im Training zu einem Minimum gelangt. Der Vorteil daraus ist, dass man dabei am Readout die Logsig- bzw. auch die Tansigfunktion anwenden kann.

Ein einfaches Beispiel zur Demonstration: In Matlab wird ein Echo State Netz, wie es in diesem Kapitel beschriebenes ist, siehe Abbildung 4.15, erstellt und trainiert, wobei keine Rückkopplung auf das Reservoir vorhanden ist. Als Trainingsset wird für den Input eine Zufallsfolge von 1 und -1 erstellt. Die Ausgangssequenz ist die um eine konstante Zeit verschobene Eingangssequenz. Dasselbe gilt für das Testset, wobei die selbe konstante Zeitverschiebung vorhanden ist. Dieses Beispiel soll das speichernde Verhalten eines Echo State Netzes aufzeigen. In Abbildung 4.16 wird im obersten Graph ein Ausschnitt der Eingangs- und Ausgangsfolge des Trainingssets gezeigt. Danach folgen vier Graphen. Zwei davon mit einer Netzdimension von 80 und zwei mit 150. Je Netzdimension wird ein Spektralradius mit 0.1 und 0.9 gewählt. Zu erkennen ist, dass die Netzdimension von 80 mit

einem Spektralradius von 0.1 kein ausreichendes Speichervermögen besitzt. Um den Input auf den Output korrekt zu verzögern. Mit der Netzdimension von 80 und einem Spektralradius von 0.9, werden einige Outputs richtig erstellt. Erst das Netz mit einer Dimension von 150 und einem Spektralradius von 0.9 weist genügend großes Speichervermögen im Reservoir auf, wie der unterste Graph zeigt. Das Netz hat im Training offensichtlich auch Muster erlernen können, und tendiert bereits vor dem Wechsel eines Zustandes, den kommenden zu kennen. Es schätzt offenbar den wahrscheinlichen nächsten Zustand ab, wie in Abbildung 4.16 Bereich A deutlich zu sehen ist. Im Bereich B wird die Zukunft falsch geschätzt, und das Netz tendiert zu einem Zustandswechsel, kehrt aber in den aktuellen Zustand zurück.

Abbildung 4.17 zeigt das selbe Netz (Dimension 150), aber mit einem Spektralradius von 2, also größer 1. Es wird kaum eine korrekte Lösung gefunden. Dies zeigt deutlich die Verletzung der Regel, dass der Spektralradius kleiner 1 sein muss, damit das Netz eine Lösung für eine Abbildung Input zu Output finden kann. Im vorliegenden Fall gibt es instabile Zustände, die nicht *vergessen* werden.

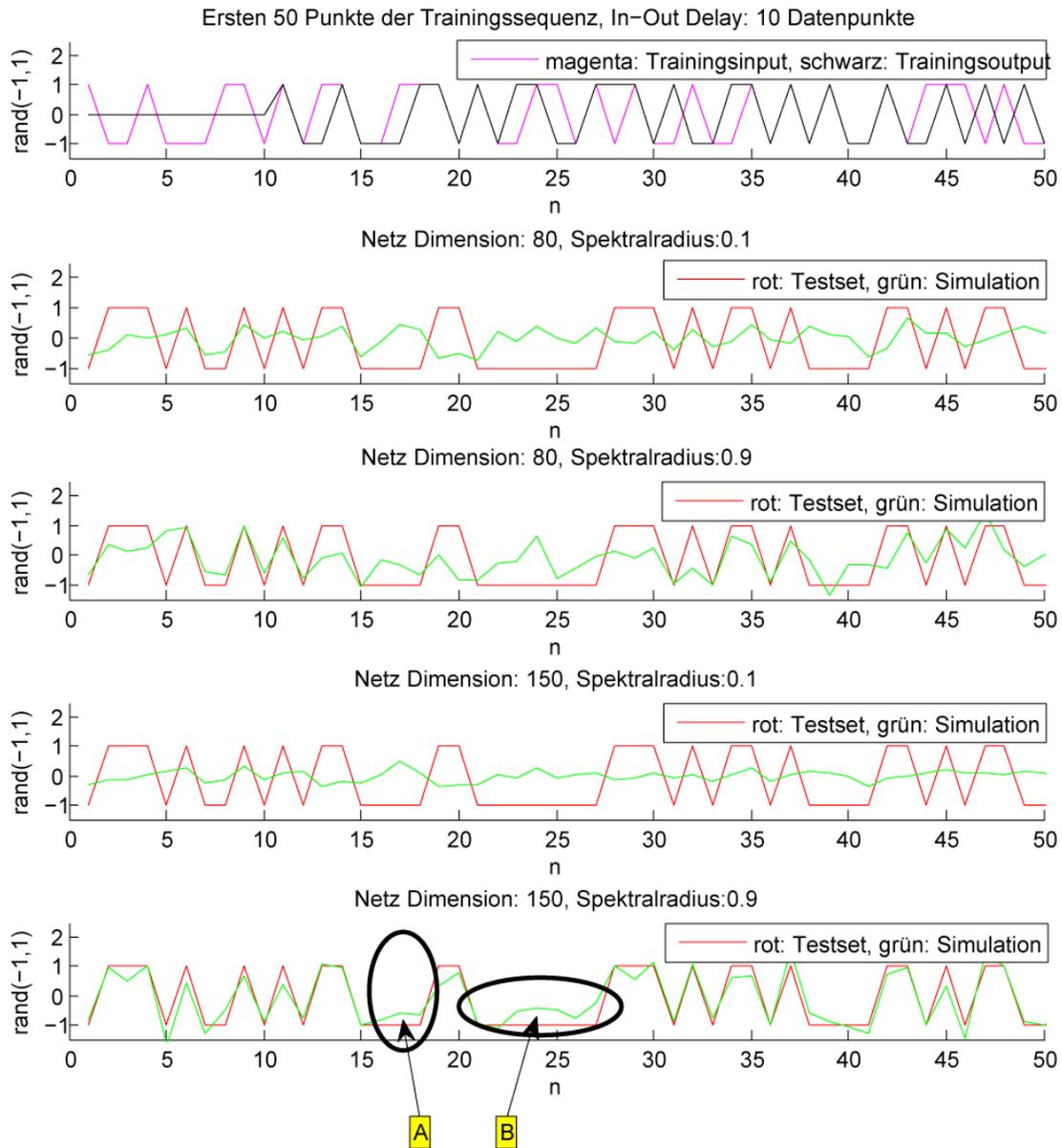


Abbildung 4.16: Demonstration der Auswirkung unterschiedlicher Netzdimensionen und Spektralradien eines einfachen Echo State Netzes.

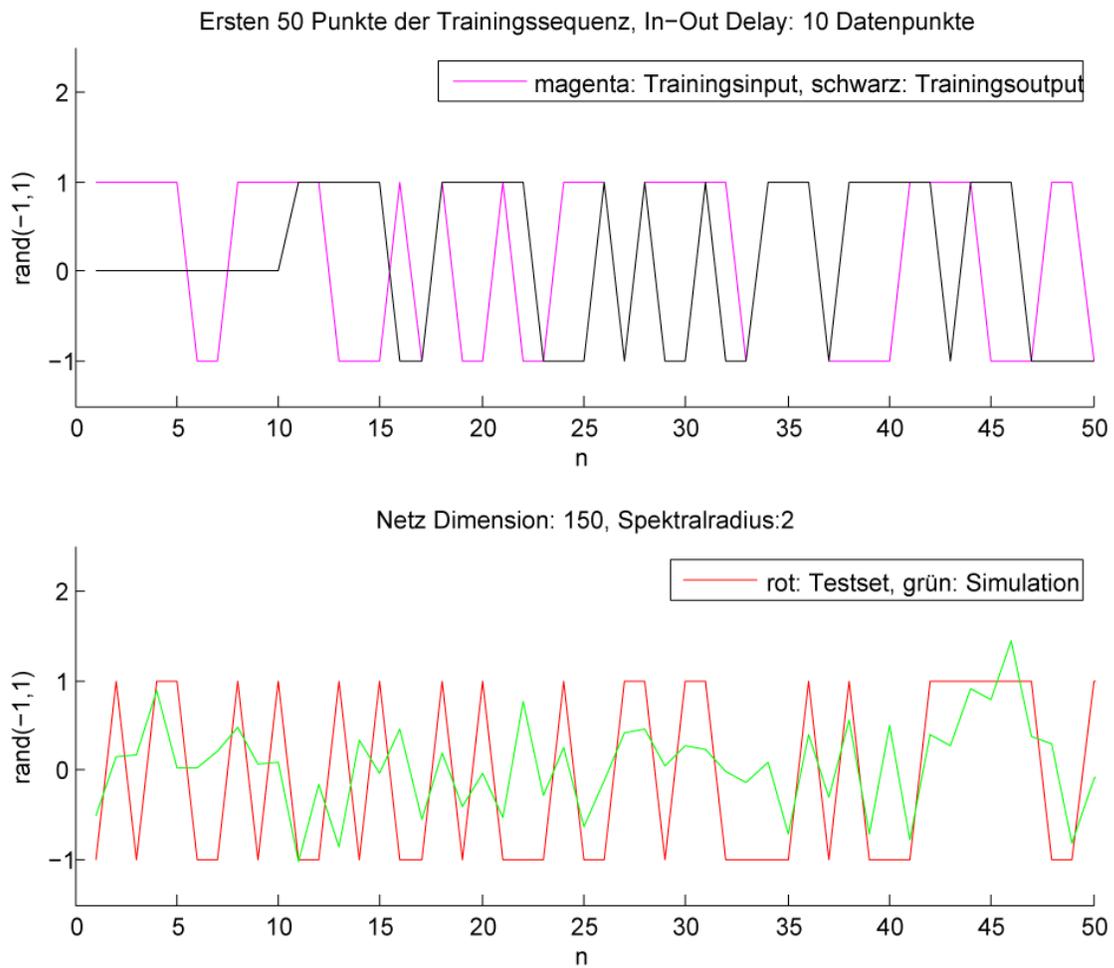


Abbildung 4.17: Demonstration der Auswirkung eines Spektralradius > 1 .

Kapitel 5

Untersuchungen regressionsbasierter Modelle

[KLSK] meint, dass ein RNN bessere Ergebnisse als die MLP basierten neuronalen Netze liefert, da sie auch dynamische Patterns abbilden können, und zeigt dies im Vergleich MLP basierte feedforward neuronale Netze mit RNN und FIR Filtern für die Prognose von Zeitreihen. Es ist aber anzumerken, dass in den vorgestellten Modellen aus verwandten Arbeiten immer der aktuelle Wert jener Zeitreihe als Input mit eingeht, die auch prognostiziert werden soll. Daher müssen die Modelle nur die Änderung vom Zeitpunkt t auf $t+1$ abbilden. In dieser Arbeit soll zuerst der aktuelle Leistungswert bzw. Vergangenheitsreihen des Leistungswertes nicht als Input mit in das Modell fließen, da der Leistungswert zum Zeitpunkt t bei vielen WLKW nicht direkt vorhanden ist, sondern die vergangenen Werte der Leistungswerte jeden Tag um Mitternacht in eine Datenbank übertragen werden. Die Prognose des Leistungswertes soll also rein aus Niederschlag und Temperatur erfolgen. Somit wird rein das Streckenverhalten des Systems vom Modell gefunden. Danach soll erst die Information vergangener Leistungswerte mit in das Modell einfließen.

MLP basierte feedforward neuronale Netze haben gegenüber den RNN aber den Vorteil, dass Sie schneller trainiert werden können. Dies wird ebenfalls durch [WRV09] bestätigt. Hierbei wird eine kurzfristige Windprognose mit feedforward- und feedback-Netzen gemacht, und die Ergebnisse zeigen, dass RNN (feedback Netze) besser abschneiden. Sie weisen einen geringeren MSE auf als feedforward Netze. Ebenfalls zeigt sich, dass RNN längere bis deutliche längere Trainingszeiten benötigen.

In diesem Kapitel werden zuerst erste Modellierungskonzepte gebildet und Testimplementierungen vorgestellt, welche in Matlab umgesetzt werden. Wie in Kapitel 2 beschrieben, ist bei den regressionsbasierten Modellen (nahezu) je Kraftwerk ein Training notwendig. Für die folgenden Testimplementierungen und Untersuchungen werden beispielhaft daher Datensätze des WLKW *Kirchbach* genommen. Der zugehörige Niederschlagsverlauf ist von der Messstation *Mauten* und der zugehörige Temperaturverlauf von der Messstation *Villach*. Als Trainingsdatensatz werden die Jahresverläufe 2007, 2008 und 2009, siehe Abbildung 5.1 (zur Übersicht nur mit den Jahresverläufen 2008 und 2009), und als Testdatensatz die Datenreihe 2010, siehe Abbildung 5.2, gewählt.

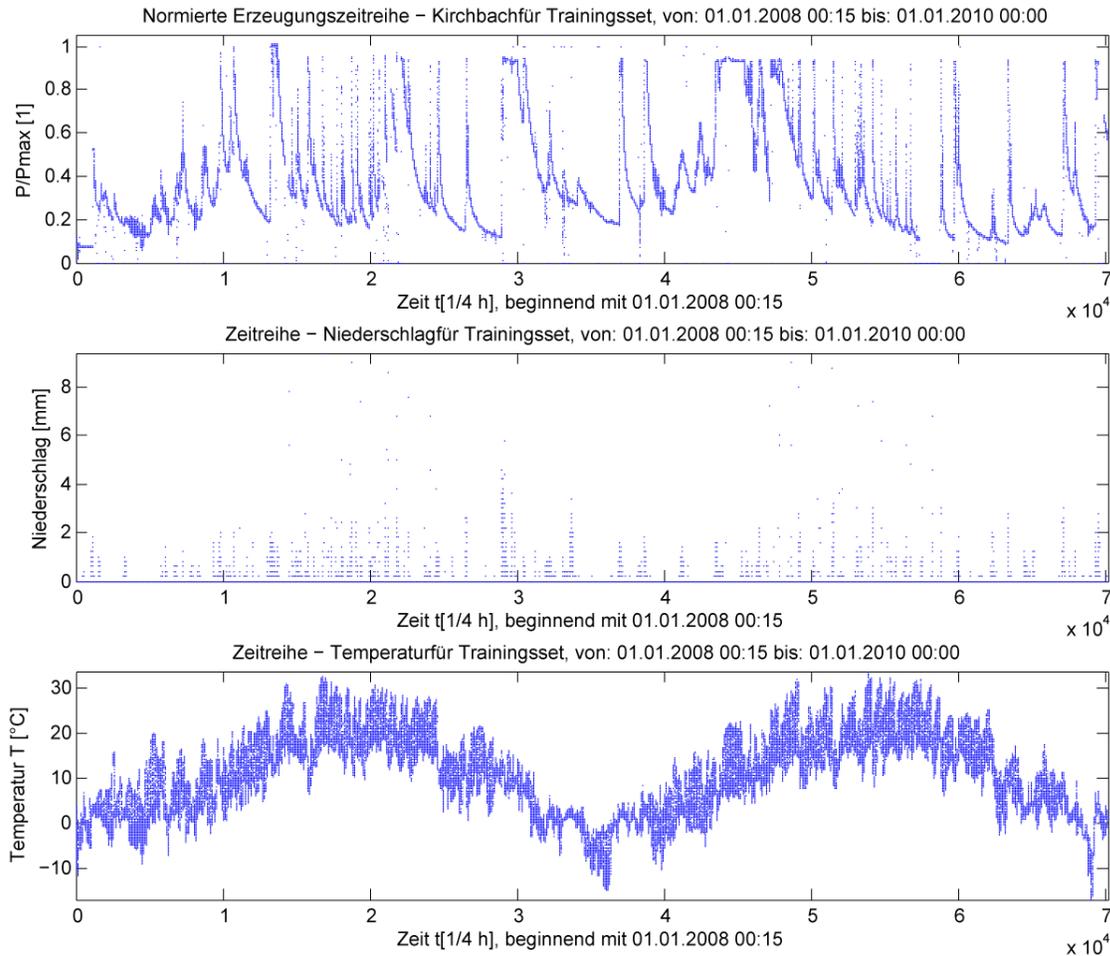


Abbildung 5.1: Teil des Trainingsdatensatzes. 2008 und 2009 der Leistung des WLKW *Kirchbach* mit zugehörigem Niederschlags- und Temperaturverlauf.

5.1 Globales Fehlermaß für unterschiedliche Modelle

Um verschiedene Modelle miteinander vergleichen zu können, wird ein globales Fehlermaß (*eg*) eingeführt. Es zeigt die Güte der Prognose auf, ohne wissen zu müssen, wie ein Ergebnis zustande gekommen ist. Um die in der Einleitung, Kapitel 1.5, beschriebenen Fehlerszenarien behandeln zu können, muss eine Aussage über die Prognosegüte getroffen werden. Dazu wird für die Abweichung der Prognose zum bekannten Wert eine Grenze von 10% nach oben und nach unten festgelegt. Dh., bei kleinen Referenzwerten ist das Band sehr eng, in dem sich die Prognose bewegen darf, um als *gut* eingestuft zu werden. Umgekehrt ist das Band bei hohen Referenzwerten größer. Maximal ist das Band aber höchstens 10% der Engpassleistung nach oben und nach unten breit. Alle Prognosewerte die die *Margin* überschreiten, sind als *schlecht* zu werten.

Es wird also neben dem MSE, siehe Gleichung 4.39, auch folgendes Fehlermaß zwischen dem bekannten Testverlauf und simulierten Prognoseverlauf festgelegt: Es werden alle jene Abweichungen des Prognosewerts zum Testwert in jedem Zeitpunkt gezählt, die um 10%

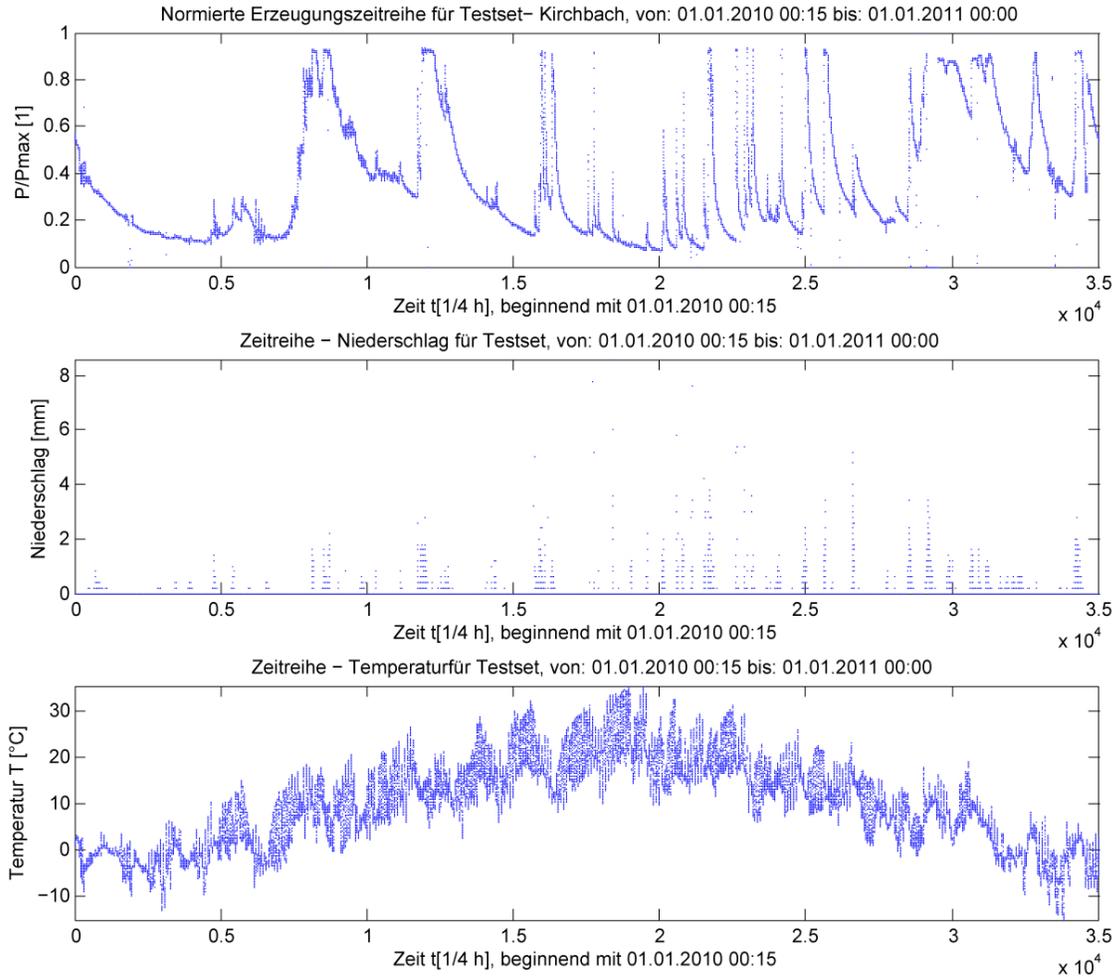


Abbildung 5.2: Testdatensatz bestehend aus den Datenreihen 2010 der Leistung des WLKW *Kirchbach* mit zugehörigem Niederschlags- und Temperaturverlauf.

vom Testwert abweichen, das heißt

$$c = \sum_{i=1}^N \text{COUNTIF} (|y_i - t_i| \cdot 100 > 10), \tag{5.1}$$

wobei y_i und t_i bereits auf die Engpassleistung normierte Werte sind. Danach wird die Anzahl der Überschreitungen in % zur Länge der Testzeitreihe angegeben, dies entspricht dem globalen Fehler

$$eg_j = \frac{c}{N} \cdot 100, \tag{5.2}$$

für $j = 1, \dots, U$, wobei U die Anzahl der vorhergesagten Intervalle (z.B. $6h$, $12h$, $18h$, etc.) ist. Diese Anzahl ist auch die Anzahl der Outputs eines Modells. Nachdem bei den Modellen mehr als ein Intervall prognostiziert wird (in der Regel $8 \times 6h$ Intervalle), wird

der Mittelwert aller eg_j gebildet:

$$eg = \frac{1}{U} \sum_{j=1}^U eg_j. \quad (5.3)$$

eg ist das verwendete globale Fehlermaß.

5.2 Multilayer Perzeptron basierte neuronale Netze

Eine Möglichkeit eine Vorhersage von WLKW zu machen, ist die Anwendung von MLP basierten neuronalen Netzen mit einer feedforward Struktur. Die notwendigen theoretischen Grundlagen dazu wurden im Kapitel 4.2.1 beschrieben. Da ein Leistungszustand eines WLKW auf Grund des speichernden Verhaltens der Natur von der Vergangenheit abhängt, siehe Kapitel 2, muss die *windowing* Methode angewendet werden. Dabei wird immer ein Fenster mit der Sicht über eine bestimmte Dauer zurück in die Vergangenheit als Inputvektor in das Netz geschickt. Auf Grund der Patterns, die beim Training dadurch erlernt werden, bildet das Netz auf eine Leistung ab. Da diese Art von Netzen eine statische Abbildung von den Regressoren zur Zielgröße macht, ist die Wahl der Inputvariablen und des Vergangenheitszeitfensters für die Qualität der Prognose von entscheidender Bedeutung. Es sind daher viele geeignete Netzvarianten zu testen. Im Folgenden wird das Modellierungskonzept und die Suche nach einem guten Netz dieser Art vorgestellt.

Primär, wie die Aufgabenstellung fordert, wird ein Modell gesucht, dass die Leistung eines WLKW rein auf Grund des Inputs der Regressoren Niederschlag und Temperatur prognostiziert. Zusätzlich wird danach kurz die Eigenschaften eines Modells aufgezeigt, welches auch den aktuellen Leistungswert bzw. vergangene Leistungswerte zur Prognose als Input erhält.

5.2.1 Modellierungskonzept

Als erstes muss dem Problem der enormen Datenmenge des Trainingssets bzw. Testsets entgegengekommen werden, um den Aufwand des Trainings zu reduzieren. Im Kapitel 2 wurde gezeigt, dass sich eine Datenreihe aus $\frac{1}{4}h$ Mittelwerten der Leistung eines WLKW zusammensetzt. Eine Jahresdatenreihe hat somit $4 \times 24 \times 365 = 35040$ bzw. $4 \times 24 \times 366 = 35136$ Datenpunkte! Nachdem aber eine Prognose in $\frac{1}{4}h$ Intervallen nicht notwendig ist, kann man mehrere $\frac{1}{4}h$ Werte zu einem Intervall durch Mittelwertbildung zusammenfassen. Ein sinnvolles Intervall ist z.B. ein $I = 6h$ breites Intervall. Damit würde nur mehr $\frac{1}{24}$ der Datenmenge vorhanden sein. Somit werden, je nach Breite des Vergangenheitsfensters, unterschiedlich viele $6h$ Mittelwerte in das Netz abgebildet.

- Es wird somit der durchschnittliche Wert über $6h$ der Temperatur und der Leistung geliefert. Dh. für jede $\frac{1}{4}h$ im Intervall von $6h$ gilt ein Wert.
- Der Niederschlag ist eine integrale Größe. Macht man eine Mittelung aller $\frac{1}{4}h$ Summenwerte über die Dauer von $6h$, heißt das, dass im Schnitt für jede $\frac{1}{4}h$ der gebildete Wert gilt. Wäre der mittlere Wert beispielsweise $5mm$, dann würde dies bedeuten, dass im Intervall von $6h$, $24 \times 5mm$ an Niederschlagsmenge gefallen sind. Dh. wiederum, dass der $6h$ Summenwert $120mm$ wäre. Für das Training eines Modells macht

es keinen Unterschied, da bei viel Niederschlag auch der Schnitt groß ist und umgekehrt. Dieser Punkt ist lediglich bei der Interpretation zu beachten, da vor allem vom Wetterdienst $6h$ Summenwerte angegeben werden. Würde man diese Niederschlagsangaben für ein Modell verwenden, müsste man die $6h$ Summenwerte auf $\frac{1}{4}h$ Werte gleichmäßig verteilen.

Um aber die unmittelbaren, sehr kurzfristig vergangenen Ereignisse nicht gänzlich außer acht zu lassen, kann man einige $\frac{1}{4}h$ Intervalle ungemittelt berücksichtigen. Interessant ist dies vor allem beim Niederschlag, da die Zeitreihe innerhalb kurzer Zeit von wenig auf viel Niederschlag springen kann. Ein weiterer wichtiger Input in das Netz sind Niederschlagswerte auf längere Sicht in die Vergangenheit. Die Dauer des Abklingens der Leistung nach Niederschlag, siehe z.B. Abbildung 5.2, dauert unterschiedlich viele Tage, je nach Menge des Niederschlags. Ein sinnvolles Fenster in die Vergangenheit könnte z.B. 7 Tage groß sein. Mit der selben Zeitdauer in die Vergangenheit werden die $6h$ Mittelwerte von der Temperatur genommen. Um Frostperioden oder Hitzeperioden zu erkennen, muss zusätzlich der langfristige Einfluss der Temperatur mit modelliert werden. Eine Möglichkeit ist das hineinnehmen des Langzeit-Temperaturmittelwertes. So ein Langzeit-Temperaturmittelwert kann z.B. über Monate zurück gebildet werden. Zusätzlich kann ein Langzeit-Trend der Temperatur über den selben Zeitraum mit betrachtet werden.

Der Output des Netzes soll zumindest ein I breites Intervall in die Zukunft sein. Wird ein Netz nur mit einem Output trainiert, kann damit zur Prognose zum Zeitpunkt der Gegenwart t_0 (in Abbildung 5.3 mit *jetzt* bezeichnet) für das erste Prognoseintervall t_{0+1} , als Input die notwendigen Vergangenheitswerte verwenden. Für weitere Prognoseintervalle t_{0+2}, t_{0+3}, \dots können dann die *fehlenden Vergangenheitswerte* durch meteorologische Prognosedaten ersetzt werden, siehe Abbildung 5.3b und c. Fehlende Vergangenheitswerte ergeben sich dadurch, dass die Gegenwart t_0 in die Zukunft rückt, und es z.B. bei der *angenommenen* Gegenwart bei t_{0+3} , den von dort aus gesehenen Vergangenheitswert t_{0-1} nicht gibt. Man prognostiziert die zukünftigen Leistungswerte eines Kraftwerks interaktiv mit ein und dem selben Netz. Ratsam ist es aber, ein Netz mit mehreren Outputs (z.B. 8 x $6h$ Intervallen) zu trainieren.

Eine weitere grundsätzliche Möglichkeit ist, dem Netz als zusätzlichen Input den von der aktuellen Gegenwart aus gesehenen Prognosewert für Temperatur und Niederschlag zu geben, um damit eventuell bessere Prognoseergebnisse für die Leistung zu erzielen. Dabei ist aber unbedingt zu beachten, dass im Falle des Hinzugebens des ersten Prognoseintervalls der meteorologischen Prognosewerte, beim Training und Testen bzw. Simulieren mit dem Testset, nur die *exakten* Prognosewerte genommen werden. Die als *zukünftig* angenommenen Werte in den Vergangenheitszeitreihen sind bekannt, wie es Abbildung 5.3a im Vergleich zu Abbildung 5.3b und c darstellt. Vom Zeitpunkt *jetzt* retour in die Vergangenheit ist alles bekannt. Es gibt aber soweit keine Aufzeichnungen der prognostizierten meteorologischen Daten, wie sie in der Vergangenheit lauteten. Dadurch bekommt man Ergebnisse der Qualität des Modells, die rein von der Güte des Modells abhängen und nicht zusätzlich von der Qualität der prognostizierten Wetterdaten vom Wetterdienst. Dies stimmt aber auch nur dann, sofern die aufgezeichneten Wetterdaten mit dem tatsächlichen Wetterverlauf vor Ort beim betrachteten Kraftwerk übereinstimmen. Der Idealfall ist jener, dass eine Wettermessstation, die zu einem Kraftwerk zugehörige Wetterdaten aufzeichnen soll, in unmittelbarer Umgebung des Kraftwerks positioniert ist. Daher nimmt

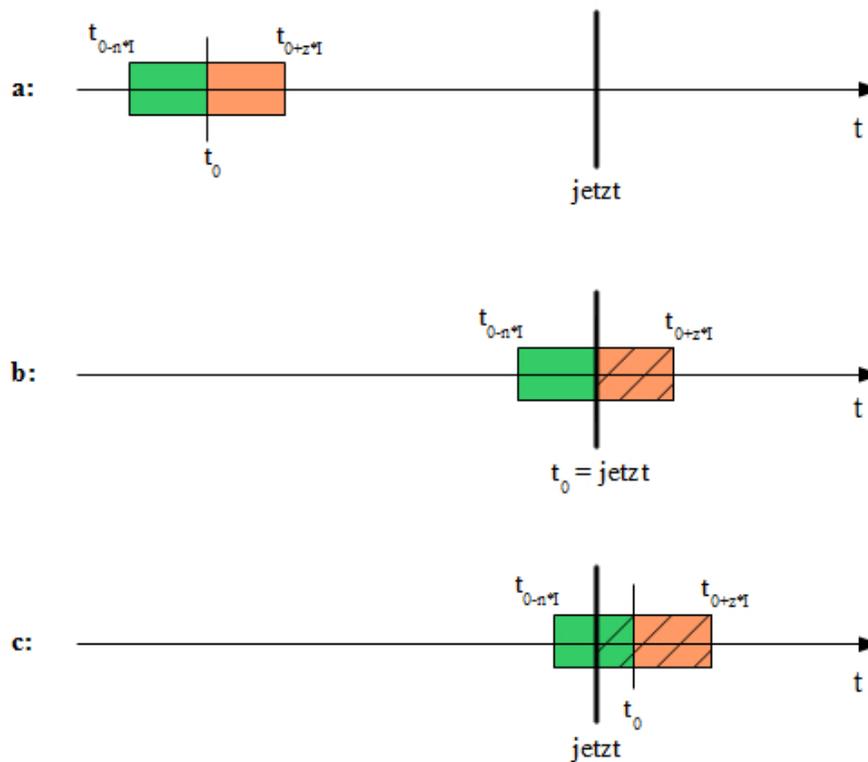


Abbildung 5.3: Demonstration der unterschiedlichen Fälle bei unterschiedlicher Position von t_0 . **a:** t_0 befindet sich in der Vergangenheit. **b:** t_0 entspricht dem Gegenwartszeitpunkt *jetzt*. **c:** t_0 rückt in die Zukunft.

man von t_0 aus gesehen, siehe z.B. Abbildung 5.3a, für die Prognose die exakten Werte aus der Zukunft. Die Sicht auf die Vergangenheit kann wie in Abbildung 5.4 dargestellt aussehen. Abbildung 5.4 skizziert eine Möglichkeit für die Input- und Outputkonfiguration eines neuronalen Netzes.

5.2.2 Implementierung des ersten Modells

Die Testimplementierung hat den Zweck, viele sinnvolle Netzkonfigurationen sowie Input/Outputkonfigurationen zu testen. Da es aber nicht möglich ist, alle Varianten bis ins tiefste Detail zu variieren, wird eine grobe Stufung interessanter Einstellungen vorgenommen. Mit dem erstellten Matlabskript kann einerseits die Struktur und Trainingsvariante des neuronalen Netzes variieren. Andererseits kann man auch die Struktur der Input- und Outputvektoren, wie in Abbildung 5.4 skizziert, variieren. Man kann den Trainings- und Testdatensatz als CSV Datei angeben, oder einen Ausschnitt aus den CSV Dateien ziehen. Vor dem Simulationsstart kann eine Konfigurationsdatei erstellt werden. Danach werden die entsprechenden unterschiedlichen Input- und Outputvektoren aus dem Trainings- und Testdatensatz geladen und das Training mit den unterschiedlichen Netzvarianten gestartet. Die Ergebnisse zu allen Netzen eines Sets von Input/Outputkonfigurationen werden

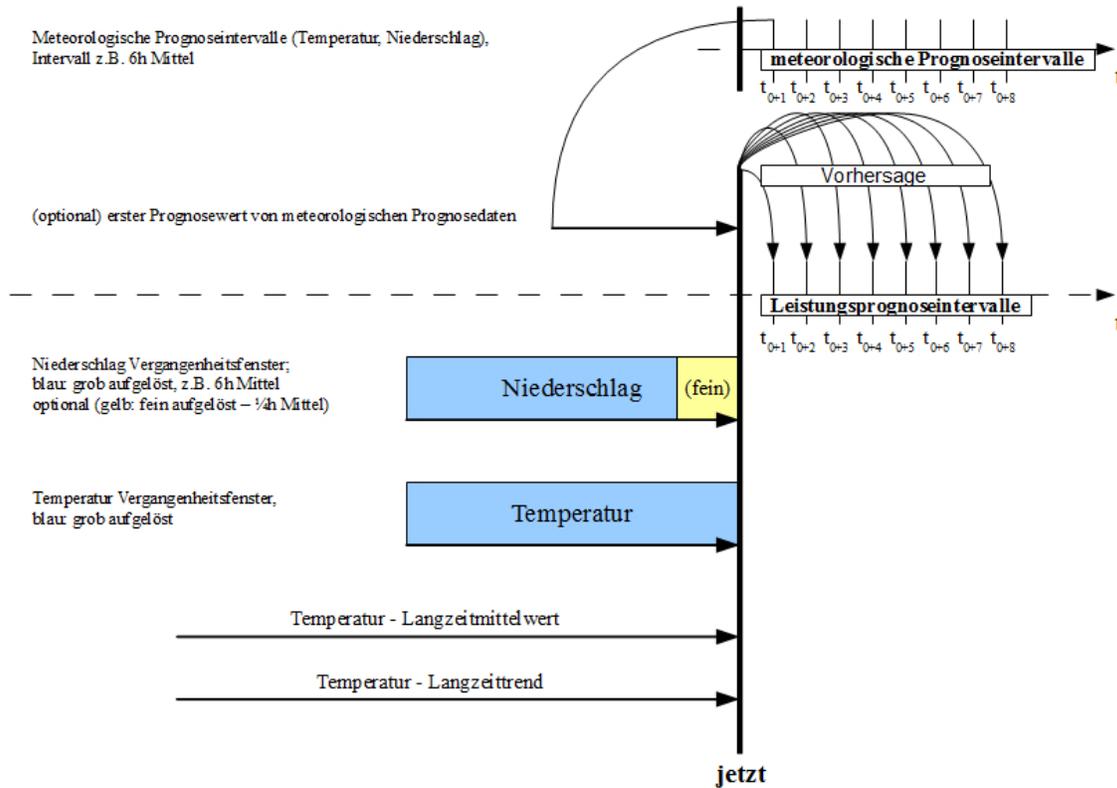


Abbildung 5.4: Inputs und Outputs des gesuchten multilayer Perzeptron basierten feed-forward neuronalen Netze.

in einer Struktur gespeichert. Nach dem Training kann dann eine Auswertung gemacht werden. Input/Outputkonfigurationen können beispielsweise folgendermaßen aussehen:

```
setkonf.nr = n;
setkonf.incl_first_forecastval = 1;
setkonf.step = 24;
setkonf.past_steps = 28;
setkonf.from_past_steps_direkt = 1;
setkonf.forecast_steps = 8;
setkonf.meanTemp_duration = 100;
tr_t_setkonfigs{n} = setkonf;
```

Die Variablen beschreiben dabei:

- *incl_first_forecastval = 1* - Bedeutet, dass der erste meteorologische Vorhersagewert für Temperatur und Niederschlag als Input in das neuronale Netz gegeben wird, wie in Abbildung 5.4 gezeigt wird. Der Wert 0 würde dies deaktivieren.
- *step* - Gibt dabei die Anzahl der $\frac{1}{4}$ h Intervalle an, die zu einem gemittelten Intervall zur Datenreduktion zusammen gefasst werden.
- *past_steps* - Gibt an, wie weit das Temperatur- und Niederschlagsfenster in die Vergangenheit *sicht* (in Abbildung 5.4 blau eingezeichnet). Der Zahlenwert gibt die zur

Datenreduktion gemittelten Intervalle, $\text{step} \times \frac{1}{4}\text{h}$, an. Hier wäre der Wert 28 gleichbedeutend mit 7 Tage, $24 \times \frac{1}{4}\text{h} \times 28 = 7 \text{ Tage}$.

- *from_past_steps_direkt* - Gibt an, wie viele *past_steps* vom Zeitpunkt *jetzt*, zurück in die Vergangenheit fein aufgelöst als Input in das Netz gegeben werden. Hier bedeutet 1, dass das erste 6h Intervall fein aufgelöst dem Netz gegeben wird, also die ersten $24 \times \frac{1}{4}\text{h}$ Werte.
- *meanTemp_duration* - Gibt das Zeitfenster an, über das ein Langzeitmittelwert und linearer Trend von der Temperatur erstellt wird.
- *tr.t.setkonfigs* - Ist die Struktur, die eine Konfiguration aufnimmt und somit alle Input/Outputkonfigurationen speichert.

Die Konfiguration für die zu testenden Netze (Input/Outputkonfigurationen) kann beispielsweise folgendermaßen aussehen:

```
netkonf.nr = n;
netkonf.is_regulated = 0;
netkonf.alpha = 0.2;
netkonf.hiddenlayer_num = [50 50];
netkonf.is_mapstd_normalized = 1;
netkonf.is_early_stopped = 1;
netkonf.performance_fcn = 'mse';
netkonf.activation_fcn = {'tansig','tansig','logsig'};
netkonf.algorithm_train = 'traingda';
netkonf.learn_fcn = 'learngdm';
netkonf.max_epochs = 700;
netkonf.epochs_show = 700;
netkonf.showGUI = false;
netkonfigs{n} = netkonf;
```

Dabei bedeuten die Variablen folgendes:

- *is_regulated=0* - Heißt, dass das neuronale Netz ohne Regularisierung trainiert wird. 1 würde die Regularisierung aktivieren, und der Wert *alpha* (α) würde als Regularisierungsterm verwendet werden.
- *hiddenlayer_num* - Gibt dabei in Form eines Vektors an, wie die Hiddenlayerschichten des neuronalen Netzes aufgebaut sind, dh., wie viele Neuronen in jeder versteckten Schicht erstellt werden sollen. Die Input- und Outputneuronenanzahl ergibt sich aus der Länge des Inputvektors bzw. Outputvektors (=Targetvektor).
- *is_mapstd_normalized = 1* - Gibt an, ob die Daten vor dem Training mit der *mapstd* Funktion von Matlab standardisiert werden, oder nicht (0);
- *is_early_stopped = 1* - Gibt an, ob beim Training *earlystopping* verwendet werden soll, um Overfitting beim Training zu vermeiden.
- *performance_fcn* - Gibt die Fehlerfunktion an, die beim Training verwendet werden soll.

- *activation_fcn* - Gibt die Übergangsfunktionen der Hiddenlayer und des Outputlayers an.
- *algorithm_train* - Gibt den zu verwendenden Trainingsalgorithmus an.
- *learn_fcn* - Gibt die zu verwendende Lernfunktion an.
- *max_epochs* - Gibt die maximalen Trainingsepochen an.
- *epochs_show* - Gibt die Anzahl der Epochen an, bei dem ein Status auf die Console geschrieben werden soll.
- *showGUI* - Schaltet die GUI ein oder aus.
- *netkonfigs* - Ist die Struktur, die eine Netzkonfiguration aufnimmt.

Für den folgenden Simulationslauf werden bezüglich der Input/Outputkonfiguration folgende Variablen fixiert:

- *step* = 24, heißt 6h Mittelung der Trainings- und Testsets.
- *forecast_steps* = 8, heißt 8 x 6h Prognoseintervalle in die Zukunft.

Folgende Variablen werden variiert:

- *incl_first_forecastval* = [1,0],
- *from_past_steps_direkt* = [1,0],
- *past_steps* = [20,30,40,50], bedeutet ein Vergangenheitsfenster von 5, 7.5, 10 und 12.5 Tage,
- *meanTemp_duration* = [50,100,150,200], heißt 12.5, 25, 37.5 und 50 Tage Langzeitmittelwert und Trend.

step kann fixiert werden, da die meteorologischen Prognoseintervalle die selbe Breite haben. Mehr als eine 8 x 6h Vorhersage ist vorerst nicht notwendig.

Bezüglich der Netzkonfiguration sind folgenden Variablen fixiert:

- *is_mapstd_normalized* = 1,
- *is_early_stopped* = 1,
- *activation_fcn* = tansig für hiddenlayers und logsig für Outputlayer,
- *algorithm_train* = traingda,
- *learn_fcn* = learngdm,
- *max_epochs* = 700,
- *epochs_show* = 700,
- *showGUI* = false.

Alle Trainingsdaten werden normalisiert. Ebenfalls wird bei allen Netztrainingsvorgängen *earlystopping* verwendet, um zum Einen schneller im Training zu sein, und zum Anderen Auswendiglernen (Overfitting) zu vermeiden. Die Trainingsdaten sind, bis auf die Temperatur, alle positiv. Sie sind aber nach der Normalisierung mittelwertfrei gemacht, und können dadurch negativ werden. Daher wird in jedem Hiddenlayer die tansig Aktivierungsfunktionen verwendet. Die Outputs sind positive Werte, daher wird die *logsig* Aktivierungsfunktion verwendet. Trainingsalgorithmus und Fehlermaß sind ebenfalls fixiert. Variiert werden folgende Variablen:

- *is_regulated* = [0,1],
- *hiddenlayer_num* = [20,50,80,110,140,170] jeweils für die Hiddenlayeranzahl [1,2,3],
- *alpha* = [0.2,0.5,0.8],
- *performance_fcn* = mse für unregulierte Netze, msereg für regulierte Netze.

Wobei im Falle der Regularisierung eine Hiddenneuronenanzahl von 180 Neuronen gesetzt wird, und jeweils mit der Layeranzahl 1,2 und 3 trainiert wird. In Summe ergibt dies 64 Input/Outputkonfigurationen, 18 unregulisierte Netze und 9 regulisierte Netze. Daraus folgen $64 \times 27 = 1728$ Netzvarianten, die trainiert werden müssen.

5.2.3 Auswertung des ersten Modells

Die Eigenschaften der nach dem MSE gefundenen *vermeintlich* besten 100 Netze, die sich nach dem Training ergeben, sind in Tabelle C.1 im Anhang C für regulisierte Netze und Tabelle C.2 im Anhang C für nicht regulisierte Netze angeführt. Die Tabellen beinhalten nur *vermeintlich* gute Netze, da der MSE noch nichts über die Güte der Prognose aussagen kann. Hier kommt das globale Fehlermaß *eg* für eine qualitative Auswertung zum Tragen. Obwohl der MSE zum Teil niedrig ist, zeigt der *eg* Fehlerwert, der nach dem Schema im Kapitel 5.1 errechnet wird, teils große Abweichungen. Der *eg* Fehlerwert ist jener Wert, den man zur Auswahl tatsächlich guter Netze nimmt. Sortiert man nun die besten 100 regulisierten vermeintlich guten Netze nach dem geringsten *eg* Wert, so erhält man als beste 10 Netzkonfigurationen für die regulisierten Netze jene lt. Tabelle 5.1. Die Simulation mit dem regulisierten Netz mit der Konfiguration mit der Nummer 2 aus Tabelle C.1, ergab bei der Simulation den geringsten *eg* Fehlerwert. Dh., dass bei diesem Netz die meisten Simulationspunkte in allen 8 Intervallen innerhalb der gegebenen Schranke liegen. Diese Simulation ist in Abbildung 5.5 dargestellt, wobei das erste Prognoseintervall (1 x 6h), oberer Verlauf, des fünfte Prognoseintervall (5 x 6h), mittlerer Verlauf, und achte Prognoseintervall (8 x 6h), unterer Verlauf, abgebildet ist. Die bekannte Leistung der Testzeitreihe (Jahr 2010) ist dementsprechend verschoben, um den Verlauf des jeweiligen Prognoseintervalls mit der Leistungszeitreihe des Testsets überlagern zu können. Dieses Netz *eg_j* Werte, wie sie in Tabelle 5.2 aufgelistet sind. Netze mit hohem *eg* Wert sind unter anderem Netze mit der Konfiguration mit der Nummer 22,31,36,45,73,78,80,83,85 und 97 aus Tabelle C.1. Nummer 22 ist dazu in Abbildung 5.6 dargestellt. Die Simulation weist einen geringen MSE auf, aber *eg* ist hoch. Das Verhalten, das bei den vermeintlich guten Netzen erlernt wurde, bildet einen optimalen mittleren Verlauf mit starkem Rauschen ab. Der MSE scheint dadurch gering zu werden, aber *eg* nicht. Dieses Netz hat *eg_j* Werte, wie

Nr	1. Vorher-sagewert	fein aufgelöst	Langzeitfenster	Verg.-fester	α	Hidden-layeranzahl	min(MSE)	Zeit Training[s]	eg
2	0	1	100	50	0.2	1	0.021629	31.793	54.321
82	1	0	150	50	0.2	1	0.025311	26.077	54.615
1	1	1	150	50	0.2	1	0.020924	27.75	55.061
47	1	1	50	50	0.2	1	0.024174	27.905	56.112
33	0	1	150	50	0.2	1	0.023628	29.853	56.21
28	0	1	50	50	0.2	2	0.023427	43.864	57.529
52	0	0	150	50	0.2	1	0.024403	28.937	57.938
14	0	1	50	50	0.2	1	0.022823	26.934	58.019
30	0	1	100	40	0.2	1	0.02351	28.604	58.873
77	0	0	50	50	0.2	1	0.025205	28.125	59.48

Tabelle 5.1: Eigenschaften der tatsächlich guten regularisierten Netze nach dem ersten Simulationsdurchgang.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
46.4718	47.3984	49.7505	50.4633	64.5046	51.9601	53.5994	70.4205

Tabelle 5.2: eg_j Werte der Prognoseintervalle des besten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
70.0641	72.1311	68.1397	71.2758	73.9130	73.0577	72.2737	71.7748

Tabelle 5.3: eg_j Werte der Prognoseintervalle des angeführten vermeintlich guten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.

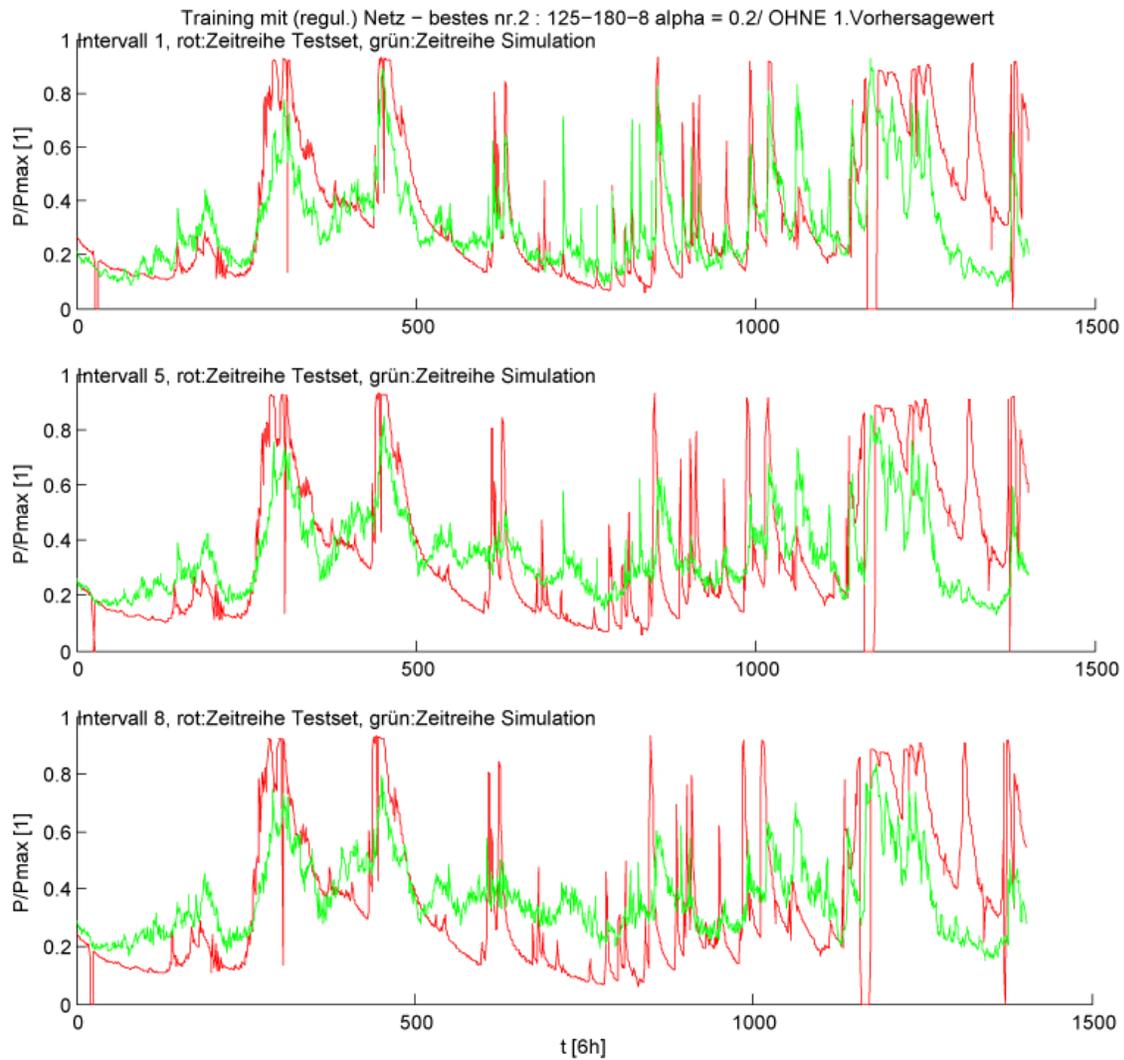


Abbildung 5.5: Simulation eines guten regularisierten multilayer Perzeptron basierten feed-forward neuronalen Netzes (oben: Intervall 1, mitte: Intervall 5, unten: Intervall 8).

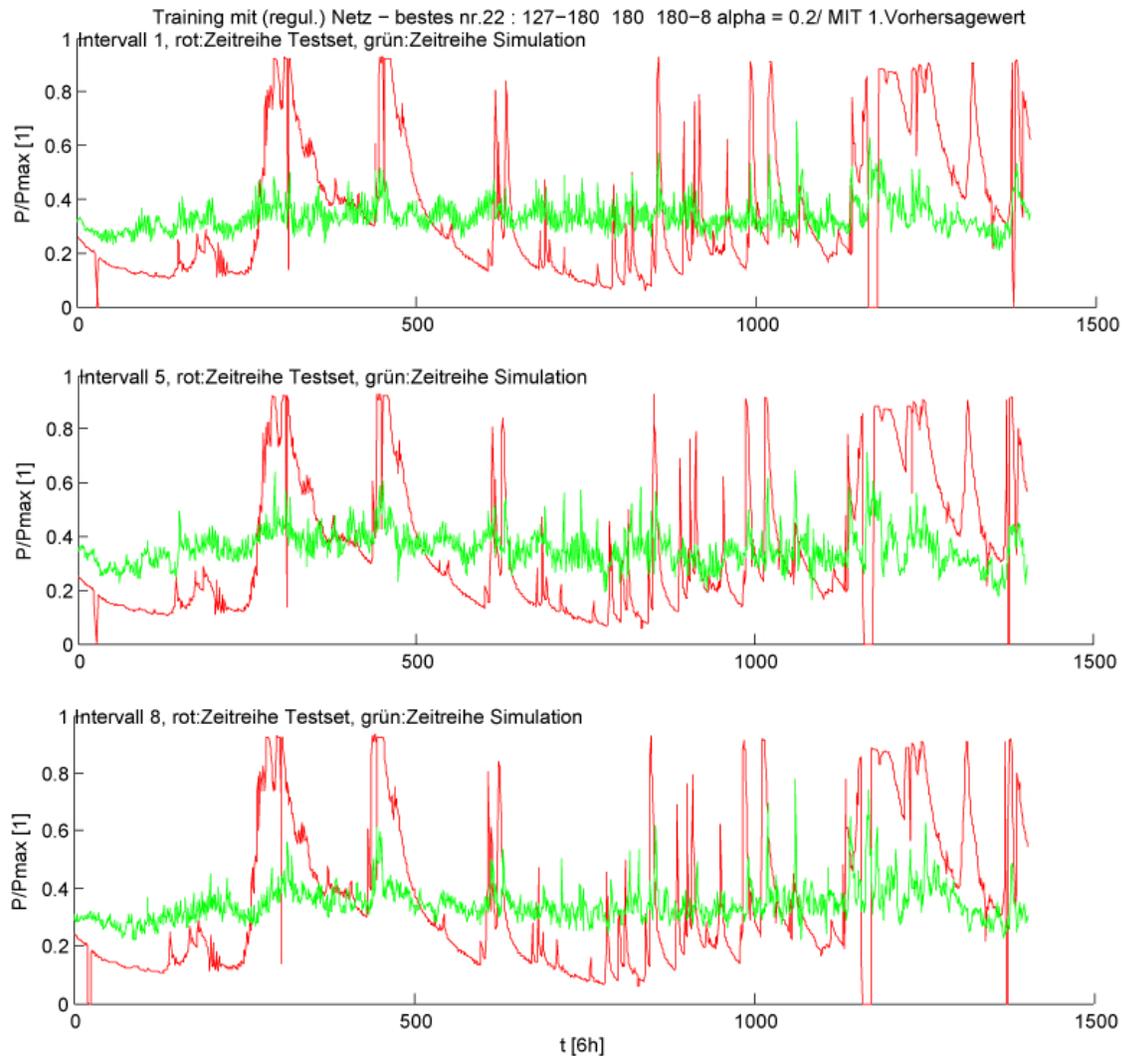


Abbildung 5.6: Simulation eines vermeintlich guten regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes (oben: Intervall 1, mitte: Intervall 5, unten: Intervall 8).

Nr	1. Vorher-sagewert	fein aufgelöst	Langzeitfenster	Verg.-fester	Hidden-layer Anz.	Hidden n	min(MSE)	Zeit Training[s]	<i>eg</i>
53	1	0	150	30	1	170	0.042293	27.161	51.845
14	0	0	150	50	2	20	0.040905	6.8437	52.254
61	1	1	150	30	1	170	0.042445	21.554	54.067
30	1	0	150	30	1	110	0.041483	16.976	54.252
62	0	0	150	40	1	170	0.042459	23.612	54.414
17	1	1	100	50	2	20	0.040999	6.3729	54.562
1	1	0	150	40	3	20	0.038656	12.106	54.609
85	1	0	150	30	1	80	0.042967	12.711	54.656
52	1	0	100	50	1	170	0.042261	19.946	54.695
45	1	1	150	50	3	50	0.04203	17.998	54.802

Tabelle 5.4: Eigenschaften der tatsächlich guten nicht regularisierten Netze nach dem ersten Simulationsdurchgang.

sie in Tabelle 5.3 aufgelistet sind. Folgende Punkte fallen bei den regularisierten Netzen mit geringem *eg* Wert bei dieser Simulation auf:

- Alle tatsächlich guten Netze sind jene mit nur einem Hiddenlayer.
- Nahezu alle Netze mit 3 Hiddenlayer sind laut MSE gut, aber in Wirklichkeit lt. *eg* Wert schlecht.
- Das erste Intervall fein aufgelöst, scheint zur Güte beizutragen.
- Die Hineinnahme des ersten Vorhersageintervalls der Wetterdaten scheint vorerst nicht offensichtlich etwas zur Güte beizutragen.
- Alpha ist sehr gering. Bei allen guten Netzkonfigurationen.
- Die Trainingszeit liegt meist unter einer Minute.

Sortiert man nun die besten 100 **nicht** regularisierten vermeintlich guten Netze nach dem geringsten *eg* Wert, so erhält man als beste 10 Netzkonfigurationen für die nicht regularisierten Netze, jene lt. Tabelle 5.4. Die Simulation mit dem nicht regularisierten Netz mit der Konfiguration mit der Nummer 53 aus Tabelle C.2, ergab bei der Simulation den geringsten *eg* Wert. Dh., dass bei diesem Netz die meisten Simulationspunkte in allen 8 Intervallen innerhalb der gegebenen Schranke liegen. Diese Simulation ist in Abbildung 5.7 dargestellt, wobei das erste Prognoseintervall (1 x 6h), oberer Verlauf, das fünfte Prognoseintervall (5 x 6h), mittlerer Verlauf, und achte Prognoseintervall (8 x 6h), unterer Verlauf, abgebildet ist. Die bekannte Leistung der Testzeitreihe (Jahr 2010) ist wie zuvor dementsprechend verschoben, um den Verlauf des jeweiligen Prognoseintervalls mit der Leistungszeitreihe des Testsets überlagern zu können. Dieses Netz hat *eg_j* Werte, wie sie in Tabelle 5.5 aufgelistet sind. Netze mit hohem *eg* Wert, sind unter anderem Netze mit der Konfiguration mit der Nummer 9,13,18,27,38,42,66,90 und 93 aus Tabelle C.2. Nummer 27 ist dazu in Abbildung 5.8 dargestellt. Dieses Netz hat *eg_j* Werte, wie sie in Tabelle 5.6

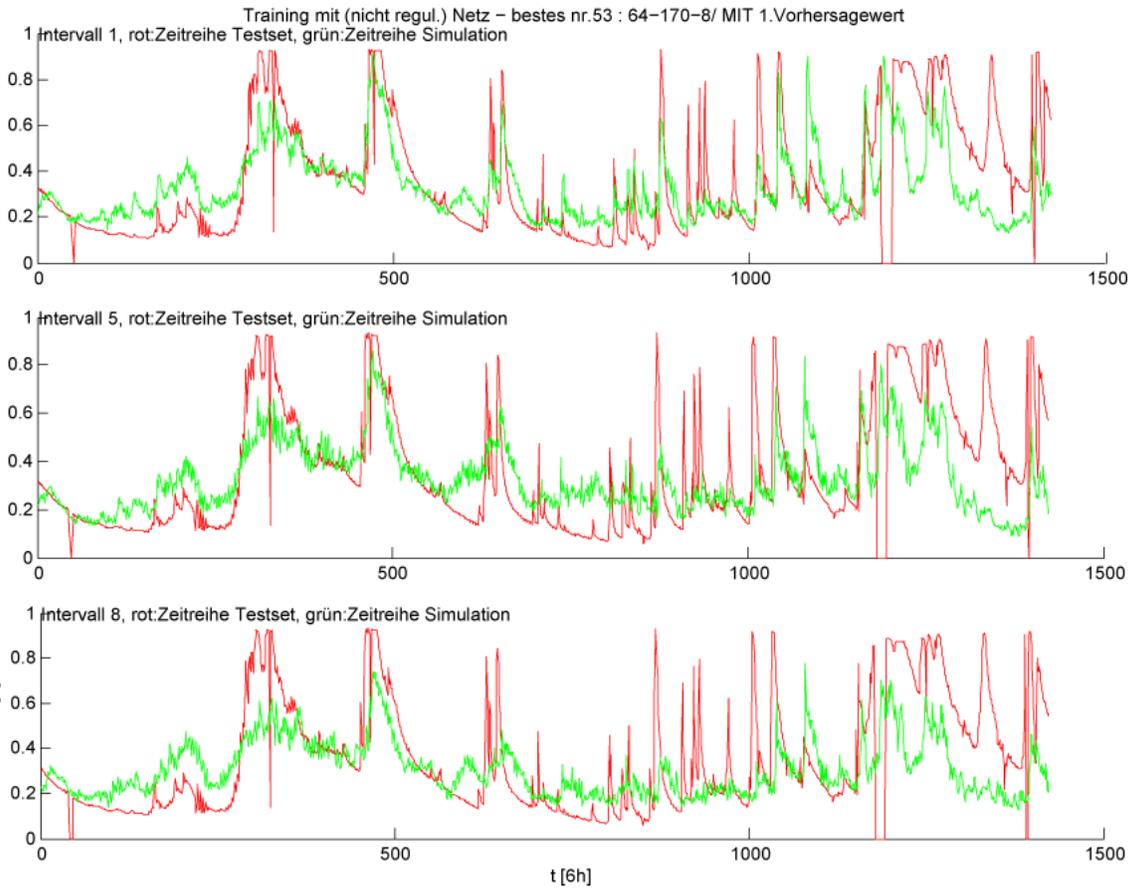


Abbildung 5.7: Simulation eines guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes (oben: Intervall 1, mitte: Intervall 5, unten: Intervall 8).

aufgelistet sind. Folgende Punkte fallen bei den nicht regularisierten Netzen mit geringem eg Wert bei dieser Simulation auf:

- Die eg Werte sind durchwegs geringer, als bei regularisierten Netzen, obwohl die eg_j Einzelwerte zum Teil höher sein können.
- Sieht man sich alle Simulationsverläufe der besten 100 nicht regularisierten Netze an, fällt auf, dass die simulierten Verläufe *besser* dem Testverlauf/Referenzverlauf folgen, als es regularisierte Netze tun. Dies spiegelt sich in den jeweiligen eg Werten wider.
- Die tatsächlich guten Netze haben eine Hiddenlayerschicht mit hoher Neuronenan-

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
50.5271	50.8784	47.3647	48.9108	55.3760	50.2460	57.5545	53.9002

Tabelle 5.5: eg_j Werte der Prognoseintervalle des besten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.

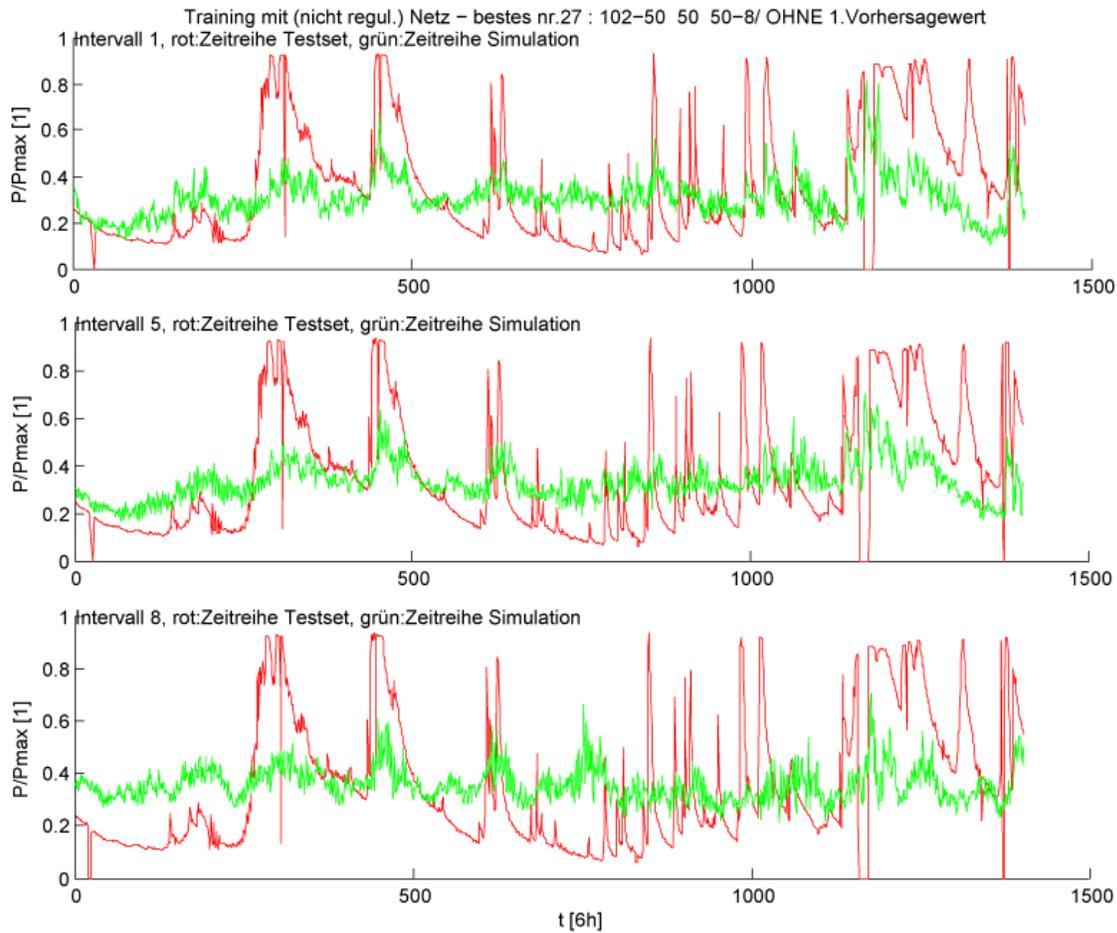


Abbildung 5.8: Simulation eines vermeintlich guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes (oben: Intervall 1, mitte: Intervall 5, unten: Intervall 8).

zahl ($n = 170$), aber auch zwei Hiddenlayerschichten mit geringer Neuronenanzahl ($n=20$).

- Das Langzeitfenster für die Temperatur ist auf der hohen/langen Seite.
- Ob eine feine Auflösung des ersten Intervalls oder das Hineinnehmen des ersten Vorhersageintervall als Input eine klare Verbesserung bringt, geht nicht klar hervor.

5.2.4 Verfeinerung der ersten Testimplementierung

Nun wird mit den gewonnenen Kenntnissen aus dem Abschnitt 5.2.3 ein weiterer verfeinerter Simulationslauf gemacht. Zusätzlich soll bei den Outputneuronen das Netz einmal mit der *logsig* und einmal mit der *purelin* Aktivierungsfunktion trainiert werden. Außerdem wird die Implementierung geringfügig erweitert, indem nicht nur das erste Vorhersageintervall der meteorologischen Vorhersagedaten mit in den Inputvektor genommen wird, sondern die ersten 8. Deshalb genau 8, da der Netzbetreiber *KELAG Netz GmbH* von der

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
64.2195	71.2758	67.9259	75.2673	66.5716	77.1917	71.6322	73.6992

Tabelle 5.6: eg_j Werte der Prognoseintervalle des angeführten vermeintlich guten nicht regularisierten multilayer Perzeptron basierten feedforward neuronalen Netzes, im ersten Simulationslauf.

ZAMG jeweils am Morgen eines Tages 8 x 6h Vorhersageintervalle für die Niederschlagsmenge [mm] und der Temperatur [°C] für die einzelnen Wetterregionen in Kärnten erhält. Das beste Netz, das aus folgender Simulation folgt, wird dann nicht wie zuvor nur mit dem ersten Leistungsprognosewert ausgewertet, sondern mit allen 8 Intervallen. Daher ist es hier notwendig, mehr meteorologische Vorhersageintervalle in den Inputvektor der zu testenden Netze mit hinein zu nehmen.

Simuliert werden folgende Kombinationen:

- $step = 24$,
- $forecast_steps = 8$,
- $incl_forecastvals = 1$,
- $from_past_steps_direkt = 1$,
- $past_steps = [40,45,50,55,60,65]$,
- $meanTemp_duration = [150,175,200,225]$.

Für die Netze werden folgende Koffigurationen erstellt:

- $is_mapstd_normalized = 1$,
- $is_early_stopped = 1$,
- $activation_fcn = \text{tansig}$ für Hiddenlayer und $[\text{logsig}, \text{purelin}]$ für Outputlayer,
- $algorithm_train = \text{traingda}$,
- $learn_fcn = \text{learngdm}$,
- $max_epochs = 700$,
- $epochs_show = 700$,
- $showGUI = \text{false}$,
- $is_regulated = 0$,
- $hiddenlayer_num = [140,155,170,185,200,215,230]$ jeweils für die Hiddenlayeranzahl 1,
- $performance_fcn = \text{mse}$.

Nr	Langzeitfenster	Verg.-fester	Akt.fkt.	Hidden n	Hidden-layer Anz.	min(MSE)	Zeit Training[s]	eg
59	150	50	1	230	1	0.040732	34.747	45.634
74	150	45	1	200	1	0.041108	31.439	47.408
39	225	40	1	215	1	0.039948	39.122	48.001
81	200	45	1	215	1	0.041628	30.811	48.322
45	150	40	1	185	1	0.040272	27.013	48.77
62	150	55	1	185	1	0.040763	28.99	48.793
29	150	45	1	230	1	0.039719	31.916	48.935
80	225	40	1	230	1	0.041603	35.06	48.938
86	175	55	1	215	1	0.041672	29.416	49.025
8	225	50	1	215	1	0.038753	34.692	49.216

Tabelle 5.7: Eigenschaften der besten 10 Netze nach dem verfeinerten Simulationsdurchlauf. Aktivierungsfunktion: 1='logisg', 2='purelin'.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
43.5495	42.7655	45.0463	47.4697	44.4761	48.3963	49.4654	43.9059

Tabelle 5.8: eg_j Werte der Prognoseintervalle des angeführten Netzes im verfeinerten Simulationslauf.

Dies ergibt in Summe 336 Netzvarianten, die trainiert werden müssen. Ausgewertet werden die besten 100 Netze, wie sie in Tabelle C.3 aufgelistet sind. Sortiert man wiederum die besten Netze nach dem geringsten eg Wert, so erhält man als beste 10 Netzkonfigurationen jene lt. Tabelle 5.7. Daraus ergibt sich aus dieser Simulation das beste Netz mit der Konfiguration mit der Nummer 59, dargestellt in Abbildung 5.9. Dieses Netz hat eg_j Werte, wie sie in Tabelle 5.8 aufgelistet sind. Auffällig ist:

- Dass die einzelnen eg_j Wert und der eg Fehlerwert massiv sinken, im Vergleich zur ersten Simulation nicht regularisierter Netze. Vor allem scheint eine Hiddenneuronenanzahl, die größer ist als zuvor, besser zu sein. Wobei der Input jetzt auf Grund weiterer Vorhersageintervalle der Wetterdaten größer geworden ist und somit auch die Hiddenneuronenanzahl etwas größer werden muss. Es scheint die Anzahl der Hiddenneuronen um etwa 200 die gesuchte Anzahl zu sein, wenn so wie in diesem Fall 8 Wettervorhersageintervalle als Input mit hinein genommen werden. Wären noch mehr Intervalle mit im Input, so müsste die Neuronenanzahl noch etwas größer werden.
- Die Hineinnahme von mehr als einem Wettervorhersagewert verbessert die Qualität stark. Offensichtlich *erkennt* das Netz dadurch eine zukünftige Tendenz im Wettergeschehen, und kann somit den Leistungswert besser prognostizieren.
- Das Vergangenheitsfenster mit etwa $50 \times 6h = 12.5$ Tagen scheint für dieses Kraftwerk geeignet zu sein.

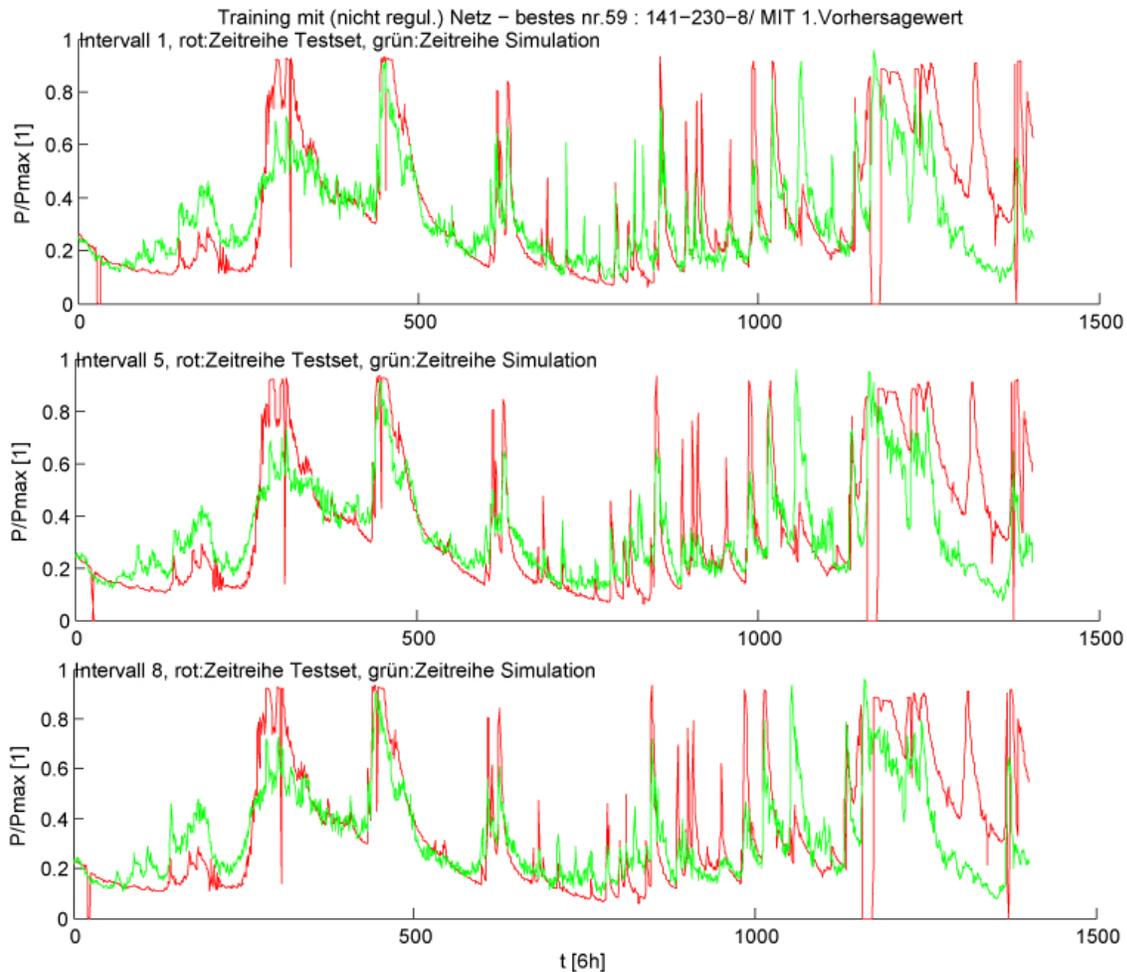


Abbildung 5.9: Simulation des gefundenen besten Netzes des verfeinerten Simulationsdurchlaufes (oben: Intervall 1, mitte: Intervall 5, unten: Intervall 8).

- Das Langzeitfenster für die Mittelwertbildung der Temperatur im hohen/langen Bereich scheint sich auch hier zu bestätigen.
- Die Aktivierungsfunktion ist klar die *logsig* Funktion. Der Grund liegt darin, dass der Output immer positiv ist. *logsig* hat immer einen Output der positiv ist. Die Leistung kann keinen negativen Wert annehmen. Die *purelin* erzeugt dadurch offensichtlich zu große Fehler, da auch negative Outputwerte mit ihr möglich sind.

Rein des Versuches wegen, werden die selben Netzkonfigurationen (336 unterschiedliche) mit randomisierten Trainingsdatensätzen trainiert. Dadurch wird getestet, ob dieser Modellansatz bessere Ergebnisse erzielt, wenn nicht ein sliding Window angewendet wird, sondern *reines* Windowing. Zu erwarten ist dass es keinen massiven Unterschied geben wird, da ein sliding Window nur bei der Variante des online Trainings, wie sie beispielsweise bei einem RNN verwendet wird, wichtig ist. Das Ergebnis dieses Versuchs bestätigt auch diese Vermutung. Das Modell wird nicht merklich besser. Tabelle 5.10 zeigt das Er-

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
46.8526	43.2761	45.4936	47.2818	45.2074	43.4192	44.1345	47.3534

Tabelle 5.9: eg_j Werte der Prognoseintervalle des besten gefundenen Netzes der selben Netzkonfigurationen wie im verfeinerten Simulationslauf, aber zusätzlicher Randomisierung der Trainingsvektoren.

Nr	Langzeitfenster	Verg.-fester	Hidden-layer Anz.	Hidden n	Akt.-fkt.	min(MSE)	Zeit Training[s]	eg
50	150	55	1	185	1	0.040234	22.803	45.377
3	150	50	1	215	1	0.037	32.416	45.438
9	150	45	1	230	1	0.037724	36.237	47.408
37	150	40	1	215	1	0.039835	28.045	48.62
89	200	45	1	230	1	0.04143	30.46	48.704
11	150	55	1	170	1	0.037864	24.21	48.864
53	175	50	1	170	1	0.040342	25.577	49.323
61	225	55	1	215	1	0.040562	27.033	49.607
60	150	55	1	215	1	0.040528	27.336	49.723
78	150	55	1	200	1	0.041061	23.341	49.839

Tabelle 5.10: Eigenschaften der besten 10 Netze nach dem verfeinerten Simulationsdurchlauf und Randomisierung der Eingangsfolge. Aktivierungsfunktion: 1='logisg', 2='purelin'.

gebnis des daraus folgenden besten Netzes, nach Sortierung nach dem eg Fehlermaß der besten 100 Netze, die durch den MSE gefunden wurden. Analog wie bei den zuvor herangegangenen Simulationen.

In Abbildung 5.11 ist ein Ausschnitt über den Zeitraum der Sommermonate (Im Bereich vom August und September) aus Abbildung 5.9 vergrößert dargestellt. In Abbildung 5.10 ist ein Ausschnitt über die Winter-/Frühjahrsmonate (Jänner bis Ende März) dargestellt. Während der Wintermonate ist die Tauwetterphase (Datenpunkte 200 bis 300) deutlich zu erkennen. Das Netz zeigt auch, dass es die Tauwetterperiode gut erkennt und erlernt hat. Dies ist eine ganz wichtige Eigenschaft einer guten Prognose für WLKW. Dies sollte ja bei der Modellierung vor allem mit dem Langzeitvergangenheitsfenster und daraus in weiterer Folge erstellten Durchschnittstemperatur und des Langzeittrends der Temperatur erreicht werden. In den Sommermonaten treten offensichtlich gröbere Abweichungen auf als in den Wintermonaten. Dies hat aber vor allem jenen Grund, dass Gewitter starke Fehler produzieren. Vor allem in der Lernphase, da der Fall Auftreten kann, dass Gewitter in der direkten Umgebung einer Messstation Niederschlag erzeugen, im Bereich des Kraftwerks aber nicht. Ebenfalls gilt die umgekehrte Variante, die einen systematischen Fehler erzeugen. Dies ist vor allem dann der Fall, wenn das Kraftwerk und die zugehörige Messstation geographisch gesehen nicht nahe genug aneinander liegen. So kann es vorkommen, dass das Netz korrekt prognostiziert, aber der Input für die reale Wettergegebenheit vor Ort beim Kraftwerk nicht korrekt ist. In den angeführten Simulationen werden ja vor allem exakte Wetterprognosedaten in das Netz als Input gegeben. Die exakten Pro-

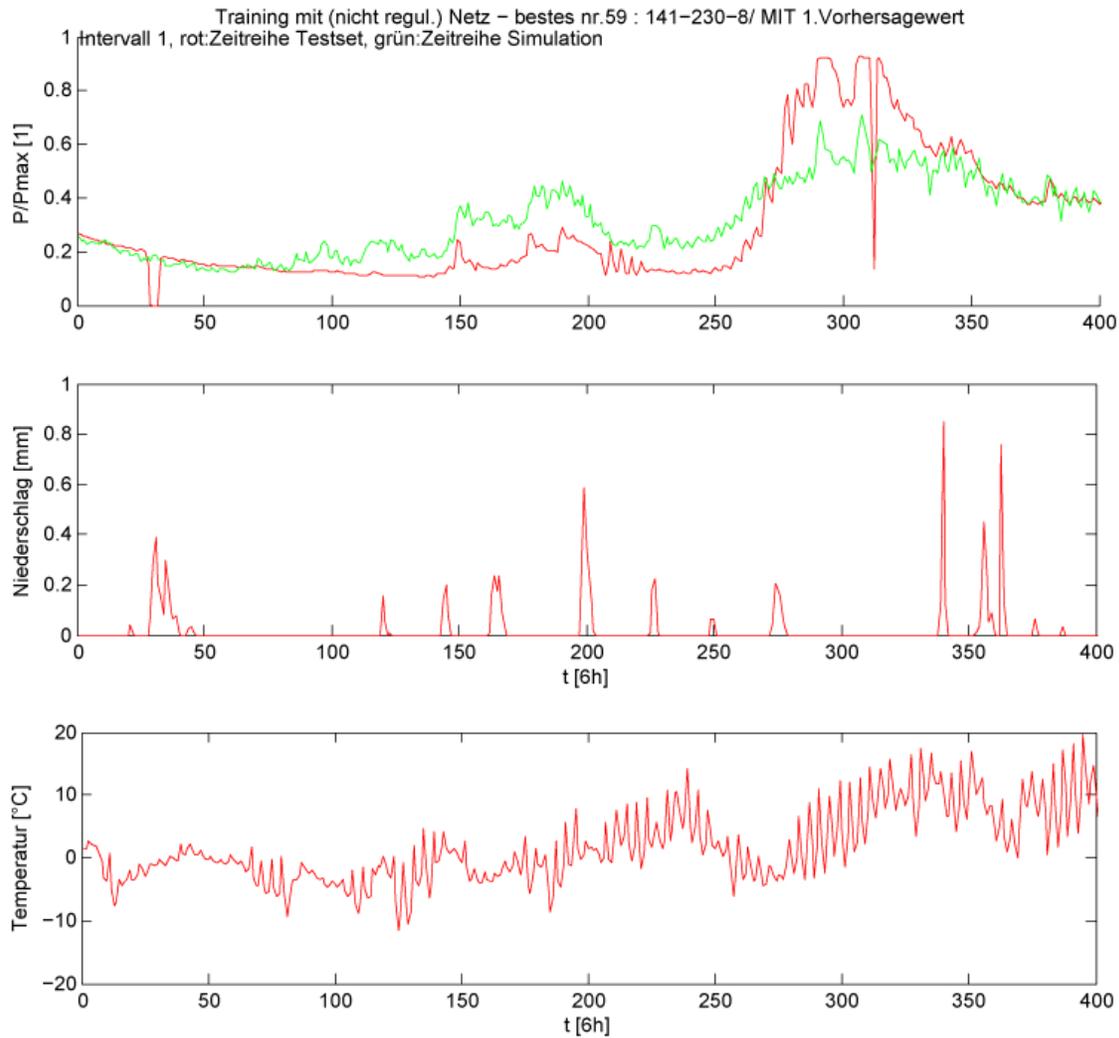


Abbildung 5.10: Ausschnitt über die Wintermonate aus Abbildung 5.9.

gnosedaten, hervorgehend aus einer Zeitreihe aus Messdaten einer Messstation, zeigen Gewitterschauer am Ort der Messstation an. Das Kraftwerk aber könnte dabei am Rande des selben Gewitterschauers liegen, und bekommt kaum bis gar keinen Niederschlag ab. Dieses Phänomen der örtlich eng begrenzten Ausdehnung einzelner Gewitter tritt vor allem im Alpenraum verstärkt auf und lässt im vorliegenden Modell Prognosefehler, wie in Abbildung 5.11 im markierten Bereich A gezeigt, auftreten. Dies führt zwangsläufig zu einer Erhöhung des *eg* Fehlerwertes. Dieser negative Einfluss kann nur mit einer erhöhten Dichte an Messstationen vermindert werden. Das Streudiagramm des Intervalls 1 und 8 des gefunden Prognosenetzes für das Kraftwerk *Kirchbach* aus der verfeinerten Simulation ohne Randomisierung, siehe Abbildung 5.9, ist in Abbildung 5.12 dargestellt.

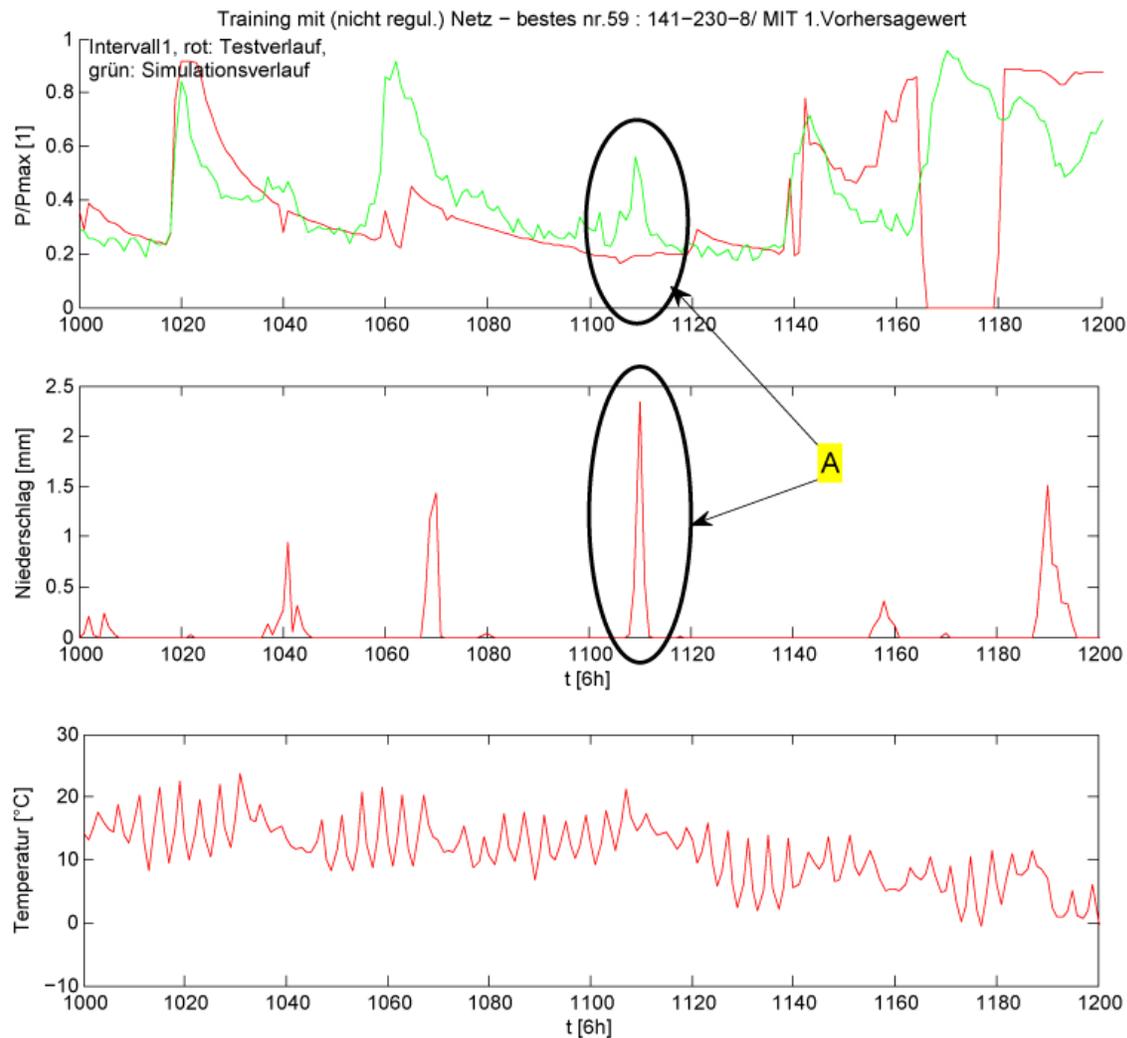


Abbildung 5.11: Ausschnitt über die Sommermonate aus Abbildung 5.9 mit hervorgehobenem Bereich lokaler Gewitter.

5.2.5 Training mit aktuellem Leistungswert als zusätzlichen Input

Der folgende Trainingsansatz erweitert den Input um eine wichtige Information: Es wird die Information des aktuellen Leistungswertes von Kraftwerken als weiterer Input in das Netz am Eingang gegeben. Das Netz sollte somit nur die Änderungen vom aktuellen Leistungswert prognostizieren müssen. Dies stellt einen grundlegenden anderen Sachverhalt dar, denn nun ist das Netz nicht *blind* bezüglich des Kraftwerks. Bisher wurde das Verhalten der reinen Strecke des Systems nachgebildet. Nun weiß das Netz aber, was am Ausgang der Strecke passiert. Simuliert werden folgende Kombinationen eines unregulierten Netzes:

- $step = 24$,
- $forecast_steps = 8$,

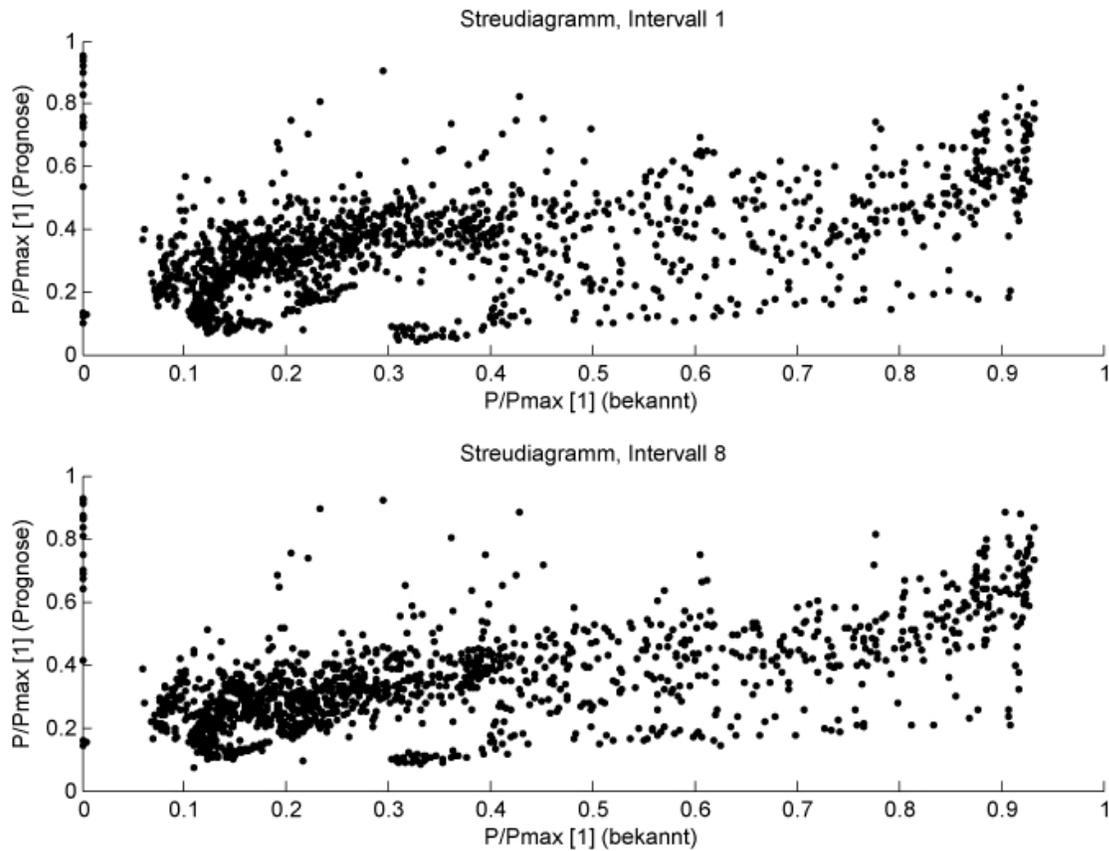


Abbildung 5.12: Streudiagramm, Kraftwerk *Kirchbach*, Intervall 1 (oben) und Intervall 8 (unten) des gefundenen besten Netzes aus der verfeinerten Simulation ohne Randomisierung, dargestellt in Abbildung 5.9.

- $incl_forecastvals = 1$,
- $from_past_steps_direkt = 1$,
- $past_steps = [20,30,40,50,60,70]$,
- $meanTemp_duration = [50,100,150,200]$,

Für die Netze werden folgende Konfigurationen erstellt:

- $is_mapstd_normalized = 1$,
- $is_early_stopped = 1$,
- $activation_fcn = \text{tansig}$ für hiddenlayers und logsig für Outputlayer,
- $algorithm_train = \text{traingda}$,
- $learn_fcn = \text{learngdm}$,
- $max_epochs = 700$,

Nr	Langzeitfenster	Verg.fester	Hiddenlayer Anz.	Hidden n	Akt.fkt.	min(MSE)	Zeit Training[s]	eg
45	100	30	1	80	1	0.016984	25.681	17.085
6	50	20	1	140	1	0.016068	40.468	17.106
2	50	20	1	100	1	0.015591	32.247	17.158
5	100	20	1	100	1	0.016025	31.625	17.359
18	50	20	1	120	1	0.016477	35.806	17.498
50	50	30	1	200	1	0.017173	66.025	17.7
21	100	40	2	60	1	0.016565	36.976	17.764
43	50	30	1	180	1	0.016981	53.795	18.052
23	150	30	1	180	1	0.016583	53.028	18.157
93	100	60	1	180	1	0.017792	51.7	18.288

Tabelle 5.11: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf mit aktuellem Leistungswert des Kraftwerks am Eingang.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
7.2382	11.7358	14.9684	16.3739	20.8011	20.7309	21.9958	22.8391

Tabelle 5.12: eg_j Werte der Prognoseintervalle des besten gefundenen Netzes, mit dem aktuellen Leistungswert als weiteren Input.

- $epochs_show = 700$,
- $showGUI = false$,
- $is_regulated = 0$,
- $hiddenlayer_num = [20,40,60,80,100,120,140,160,180,200]$ jeweils für die Hiddenlayeranzahl 1, 2 und 3,
- $performance_fcn = mse$.

Dies ergibt in Summe 720 Simulationen. Die Eigenschaften, der nach dem MSE gefundenen vermeintlich besten 100 Netze, die sich nach dem Training ergeben, sind in Tabelle C.4 im Anhang C aufgelistet. In Tabelle 5.11 sind die besten 10 aus den vermeintlich besten 100 Netzen nach dem globalen Fehlerwert eg aufgelistet. Das Netz mit der Nummer 45 ist in Abbildung 5.13 dargestellt. Das globale Fehlermaß ist mit diesem Ansatz massiv gesunken. Die einzelnen eg_j Werte sind in Tabelle 5.12 angeführt. Der eg_j Wert des ersten Intervalls ist sehr gut.

Eines ist jedoch sehr auffällig, wie ein vergrößert dargestellter Bereich aus Abbildung 5.13 zeigt. In Abbildung 5.14 erkennt man, dass die Prognose nicht optimal ist. Im Simulationsverlauf des ersten Intervalls, Bereiche mit der Markierung A, ist deutlich ein Versatz des Outputs zum Input zu erkennen. Bei einem Leistungsanstieg wird erst ein Intervall später in der Prognose auch ein Anstieg verzeichnet. Dies deutet auf eine zu starke Gewichtung des Inputs mit dem aktuellen Leistungswert hin. Dieser Versatz ist auch im Intervall 8 gut zu erkennen. Wenn im Referenzverlauf eine Leistungsspitze auftritt, tritt diese Spitze

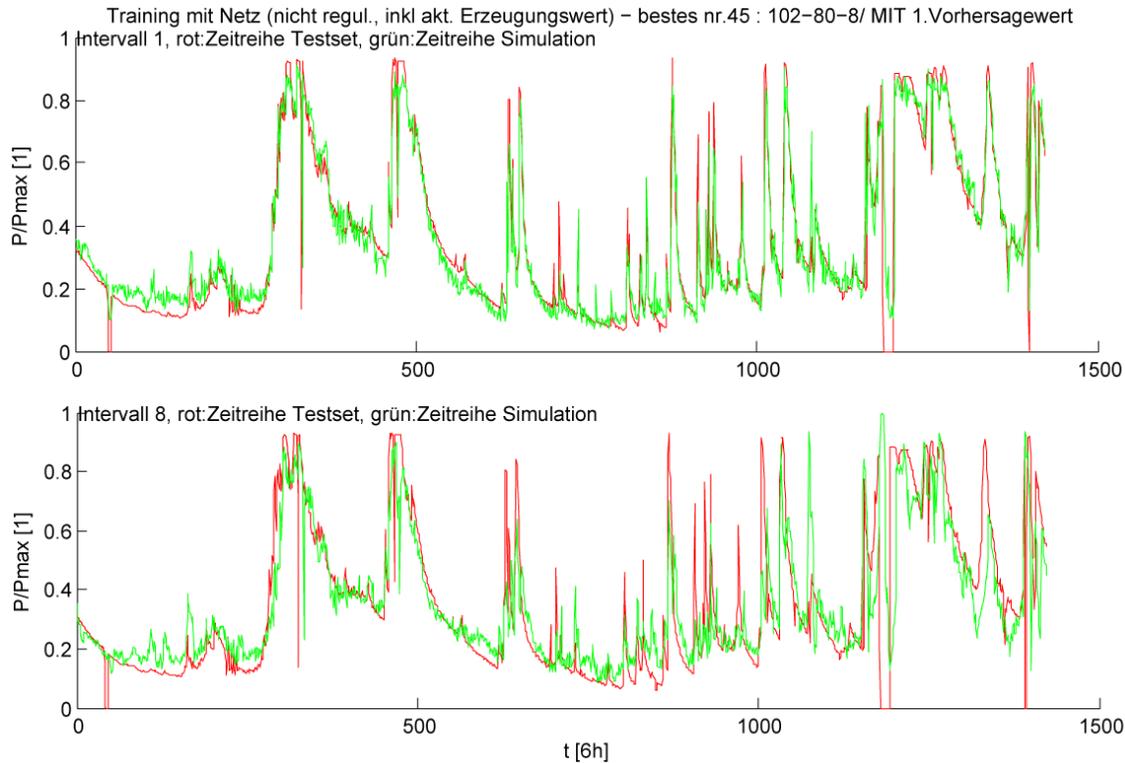


Abbildung 5.13: Simulation des gefundenen besten Netzes mit aktuellem Leistungswert als weiteren Input (oben: Intervall 1, unten: Intervall 8).

genau acht Zeitschritte später im Simulationsverlauf vom Intervall 8 erneut stark auf. Der Referenzverlauf des Intervalls 8, Abbildung 5.14 (untere Darstellung, rot) deutet auf die kommende Leistung in acht Zeitintervallen hin. Die Prognose prognostiziert dies zum Teil gut. Aber acht Zeitintervalle später tritt diese Spitze tatsächlich am Input auf und da die Gewichtung des aktuellen Leistungswertes durch das Netz zu stark ist, erfährt der Ausgang des Netzes einen zu großen Fehler. Dieser Sachverhalt ist in den markierten Bereichen B gut zu erkennen. Im ersten Intervall machen diese Fehler wenig aus, da die tatsächlich eintretenden Spitzen knapp nach der Prognose eintreten. Im Intervall 8 jedoch erzeugt dieser Fehler starke Abweichungen vom Soll.

Um dem Netz mehr Information über das Kraftwerk zu geben, wird nun nicht nur der aktuelle Leistungswert als Input in das Netz gegeben, sondern zusätzlich auch die vergangenen Leistungswerte. Dabei ist das Sichtfenster auf die Vergangenheit des Leistungsverlaufs dasselbe wie bei der Temperatur. Simuliert werden folgende Kombinationen eines unregularisierten Netzes:

- $step = 24$,
- $forecast_steps = 8$,
- $incl_forecastvals = 1$,

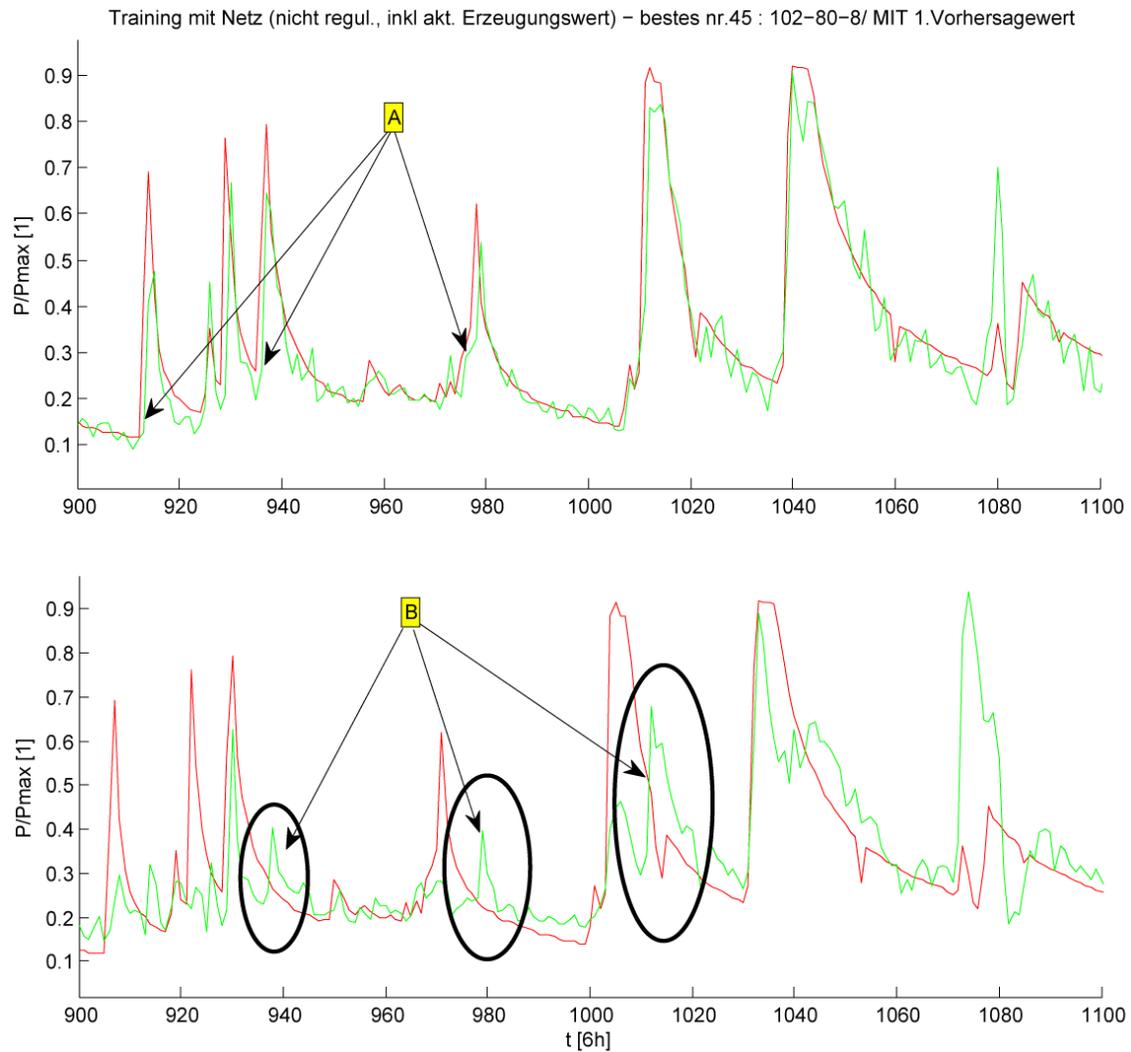


Abbildung 5.14: Ausschnitt über Sommermonate aus Abbildung 5.13.

- $from_past_steps_direkt = 1$,
- $past_steps = [20,30,40,50,60,70]$,
- $meanTemp_duration = [50,100,150,200]$,

Für die Netze werden folgende Konfigurationen erstellt:

- $is_mapstd_normalized = 1$,
- $is_early_stopped = 1$,
- $activation_fcn = \text{tansig}$ für Hiddenlayers und logsig für den Outputlayer,
- $algorithm_train = \text{traingda}$,

Nr	Langzeitfenster	Vergangenes Fenster	Hiddenlayer Anz.	<i>Hidden</i> n	Akt.fkt.	min(MSE)	Zeit Training[s]	<i>eg</i>
7	100	20	1	150	1	0.016086	42.135	18.397
5	50	20	1	120	1	0.015869	33.566	18.519
18	50	20	1	270	1	0.016687	69.762	18.528
64	50	30	1	150	1	0.017356	40.312	18.693
26	50	20	1	180	1	0.016847	45.905	18.929
3	50	20	1	90	1	0.015774	26.27	18.972
11	100	20	1	180	1	0.016346	47.18	18.981
88	100	20	1	270	1	0.017534	67.3	19.138
40	100	20	1	240	1	0.017136	59.232	19.182
78	150	30	1	300	1	0.017484	75.986	19.238

Tabelle 5.13: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf mit aktuellem Leistungswert und vergangenen Leistungswerten des Kraftwerks am Eingang.

- *learn_fcn* = `learn_gdm`,
- *max_epochs* = 700,
- *epochs_show* = 700,
- *showGUI* = `false`,
- *is_regulated* = 0,
- *hiddenlayer_num* = [60,90,120,150,180,210,240,270,300] jeweils für die Hiddenlayeranzahl 1, 2 und 3,
- *performance_fcn* = `mse`.

Dies ergibt in Summe 648 Netzvarianten. Die Eigenschaften der nach dem MSE gefundenen vermeintlich besten 100 Netze, die sich nach dem Training ergeben, sind in Tabelle C.5 im Anhang C aufgelistet. In Tabelle 5.13 sind die besten 10 aus den vermeintlich besten 100 Netzen nach dem globalen Fehler *eg* aufgelistet. Deutlich zu erkennen ist, dass das Vergangenheitsfenster nun nicht so groß sein muss, wie im Fall des weglassenen Leistungswertes als Input. Auch das Langzeitfenster liegt im niederen Bereichen. Die Neuronenzahl ist nicht eindeutig in einem Bereich. Der Großteil liegt aber über 100 Neuronen. Die besten Netze besitzen aber eindeutig nur einen Hiddenlayer. Das beste Netz, das nach dem globalen Fehlermaß gefunden wird, ist hier das Netz mit der NR. 7. Die Simulation des Intervalls 1 und 8 von diesem Netzes ist in Abbildung 5.15 dargestellt. Die globalen Fehlerwerte der einzelnen Intervalle sind in Tabelle 5.14 aufgelistet. Der globale Fehler der Prognosen ist auch hier sehr klein. Steigt aber bei den einzelnen Intervallen ebenfalls recht stark an. Hebt man nun wieder einen Bereich aus dem Simulationsverlauf heraus, siehe Abbildung 5.16 und vergleicht man den Ausschnitt mit jenen in Abbildung 5.14 von zuvor, erkennt man, dass die Prognose im Detail besser ist. Anstiege und Abfälle werden nicht verzögert prognostiziert, siehe Bereiche mit der Markierung A. Auch die verzögerten

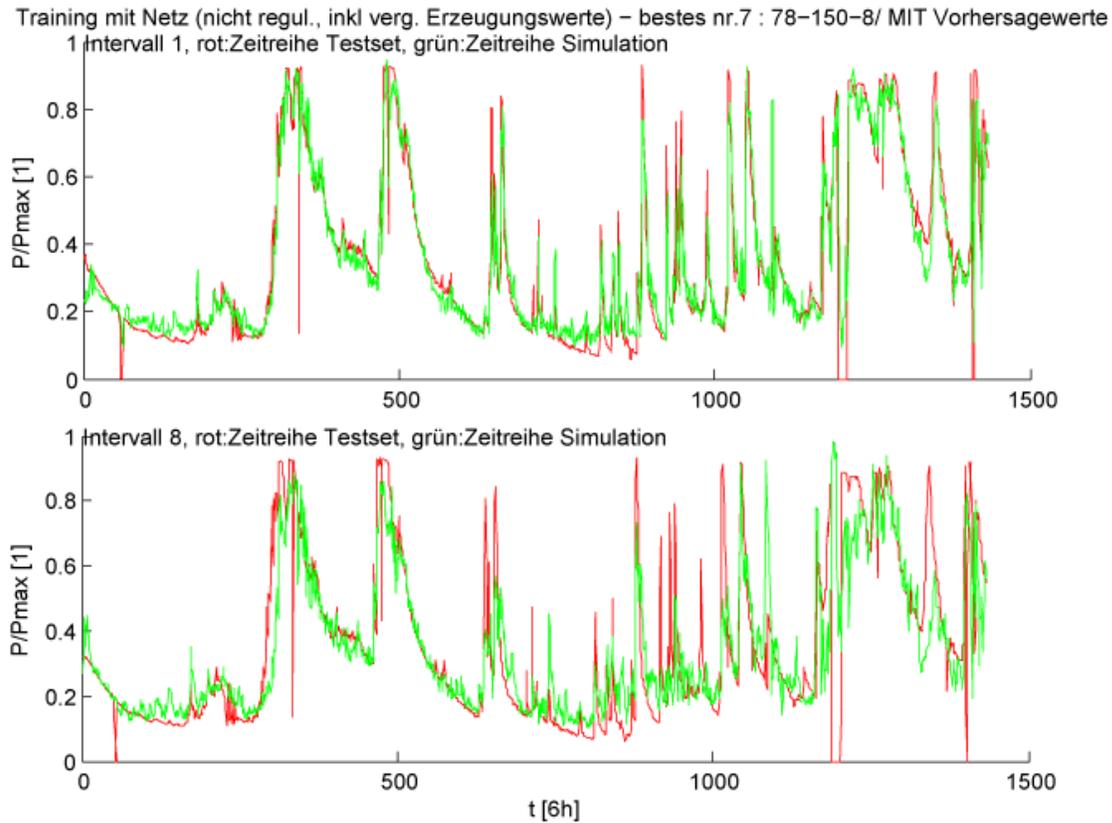


Abbildung 5.15: Simulation des gefundenen besten Netzes mit aktuellem Leistungswert als weiteren Input (oben: Intervall 1, unten: Intervall 8).

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
10.7467	13.6776	15.8409	18.8416	19.8186	20.3768	23.3775	24.4941

Tabelle 5.14: eg_j Werte der Prognoseintervalle des besten gefundenen Netzes, mit dem aktuellen und vergangenen Leistungswerten als weiteren Input.

Spitzen sind nur in abgeschwächter Form vorhanden. Das Netz erhält nun offenbar aus dem Vergangenheitsverlauf des Leistungswertes wichtige Informationen, wie Trends und andere Größen, die für eine optimalere Prognose notwendig sind. Abbildung 5.17 zeigt das Streudiagramm der Simulation des besten gefundenen Prognosenetzes für das Kraftwerk *Kirchbach* mit der Information über die Leistungswerte eines Kraftwerks. Es zeigt, wie erwartet, eine viel geringere Streuung, als beim Training ohne Information über die Leistungswerte eines Kraftwerks.

5.2.6 Bemerkungen zu den multilayer Perzeptron basierten neuronalen Netzen

Das Prinzip des Onlinetrainings wird bei RNN und Echo State Netzen angewendet werden, und wird hier bei den feedforward neuronalen Netzen ausgelassen. Die mit hineingenomme-

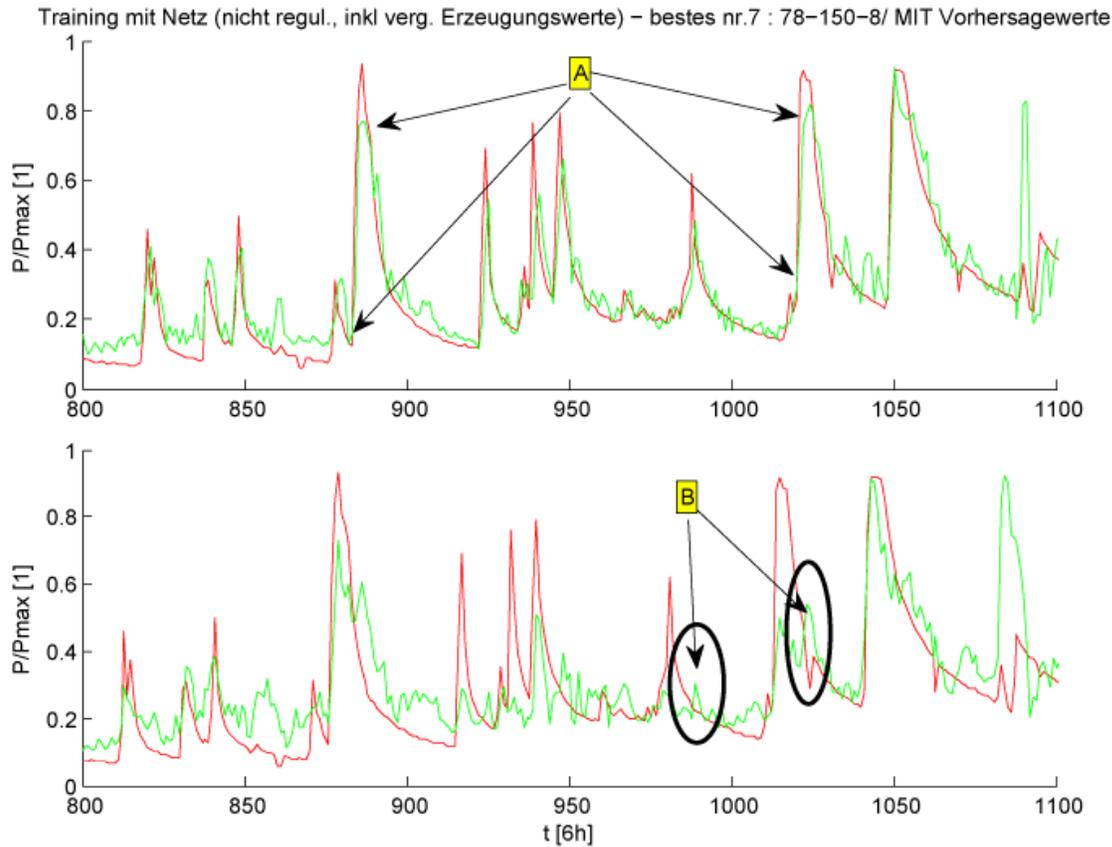


Abbildung 5.16: Ausschnitt aus Abbildung 5.15.

nen meteorologischen Wetterdaten sind beim Training immer die exakten Prognosedaten, dies ist bei der Güte der Prognose zu beachten. Die Ergebnisse, die bei den getätigten Simulationen aufgetreten sind, sind jene, die bei exakter Prognose des Wetters durch einen Wetterdienst auftreten würden. Der Einfluss verrauschter Wetterdaten wird dazu in einem eigenen Kapitel, siehe Kapitel 5.6, behandelt. Die Trainingszeiten sind bei dieser Art von neuronalen Netzen sehr kurz und liegen im Minutenbereich. Es ist auch deutlich zu machen, dass in den verwandten Arbeiten immer Modelle entwickelt wurden, die die Information über die aktuelle und vergangene Leistung am Input erhalten.

5.3 Elman rekurrente neuronale Netze

Eine weitere Möglichkeit, eine Prognose mit neuronalen Netzen zu erstellen, ist die Anwendung von RNN, wie hier anhand der Elman Netze gezeigt wird. Die theoretischen Grundlagen zu Elman Netzen sind im Kapitel 4.2.2 angeführt.

In diesem Abschnitt soll zuerst untersucht werden, ob der dynamische Speicher eines Elman Netzes ausreichend ist, um das speichernde Verhalten einer Strecke (Umwelt und Kraftwerk) abbilden zu können. Dazu wird in der Modellierung als Input jeweils nur der aktuelle Niederschlags-, Temperatur- und Leistungswert, sowie 8 Prognoseintervalle vom

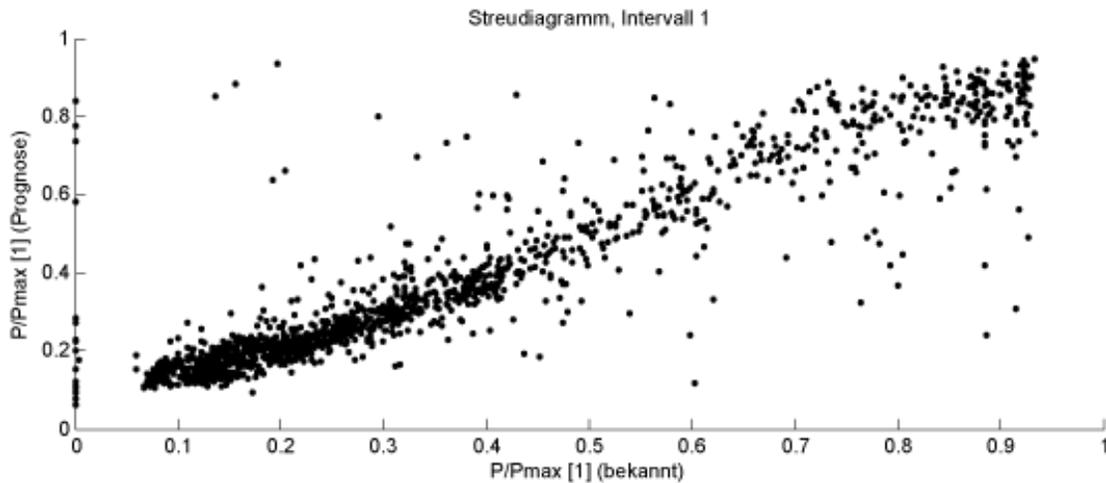


Abbildung 5.17: Streudiagramm, Kraftwerk *Kirchbach*, des gefundenen besten Netzes aus der Simulation mit aktuellem Wert der Leistung und der Vergangenheit der Leistungszeitreihe des Kraftwerks als weiteren Input, dargestellt in Abbildung 5.15.

Niederschlag und Temperatur genommen. Der Output besteht wiederum aus 8 Intervallen, wie es durchwegs bei den feedforward neuronalen Netzen gemacht wurde. Das Training erfolgt im Onlinemodus, dh. dass ein Inputvektor mit zugehörigem Targetvektor dem Netz präsentiert wird und gleich danach gelernt wird. Dies erfolgt Zeitpunkt für Zeitpunkt. Onlinetraining bedeutet, dass die Reihenfolge der Präsentation der Inputvektoren eine wesentliche Rolle spielt. Dazu ist es notwendig, eine Sequenz von Inputvektoren zu bilden. Dies ist auch bei RNN sinnvoll, da sich im Netz ein State *aufbauen* kann.

Danach soll online mit der Windowingmethode trainiert werden, dies stellt die zweite Modellierungsmethode dar. Das Windowingkonzept soll dabei dasselbe wie beim Modell der feedforward neuronalen Netzen sein, siehe Abschnitt 5.2.1. Vorerst soll auch hier die Information über den aktuellen Leistungswert und den vergangenen Leistungswerten für den Input weggelassen werden.

Zum Schluss soll ein Modell getestet werden, das online mit der Windowingmethode trainiert wird, wobei es Information über Leistungswerte erhalten soll.

5.3.1 Onlinetraining ohne Windowing

Onlinetraining mit Backpropagation und Rekurrenz im Netz, wie es bei Elman Netze der Fall ist, lässt enorme Trainingszeiten bei den Simulationen dieser Netze auftreten. Dies wurde bereits in den verwandten Arbeiten hervorgehoben. Hierbei traten teils Trainingszeiten auf, die pro Netz oft bis zu 1 Stunde dauern. Vergleiche dazu die Trainingszeiten von feedforward neuronalen Netzen, sie lagen im 30 Sekunden Bereich je Netzvariante. Deshalb ist es für diese Arbeit nicht möglich, mehrere 100 Netzvarianten und Konfigurationen zu simulieren, wie es bei den feedforward neuronalen Netzen gemacht wurde.

Die Simulation wird wieder in Matlab gemacht. Dazu wurde ein Skript implementiert, mit dem man mehrere Inputvektoreinstellungen und Netzkonfigurationen in einer Abfolge simulieren und auswerten kann. Die Implementierung ist ähnlich jener der feedforward

Nr	Hidden-layer Anz.	Hidden n	Akt.-fkt.	Inkl. Wettervorhersage	Zeit Training [s]	<i>eg</i>	MSE
4	2	40	1	1	2768.7	56.715	0.047218
1	1	40	1	1	2391.5	57.576	0.04432
5	1	20	1	1	2394.3	57.653	0.047802
7	2	20	1	1	2720.1	58.824	0.050749
10	1	12	1	1	2367.4	60.305	0.051435
9	2	12	2	1	2699.6	60.356	0.051432
12	1	40	2	1	2358.3	60.503	0.051946
2	2	12	1	1	2729.9	61.286	0.045911
11	1	20	2	1	2343.8	61.338	0.051782
14	2	40	2	1	2731.4	61.501	0.053536

Tabelle 5.15: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze ohne Windowing. Aktivierungsfunktion Ausgang: 1=tansig, 2=purelin, 3=logsig.

neuronalen Netzen. Für diesen Simulationslauf werden folgende Einstellungen für die Input/Outputkonfigurationen gewählt:

- *step* = 24,
- *forecast_steps* = 8,
- *incl_forecastvals* = [0,1],

Die Einstellungen für die Vergangenheitsfenster fallen hier weg, da es kein Windowing gibt. Für die Netze werden folgende Konfigurationen erstellt:

- *is_mapstd_normalized* = 1,
- *activation_fcn* = tansig für hiddenlayers und ['tansig', 'logsig', 'purelin'] für die Outputneuronen,
- *max_epochs* = 600,
- *showGUI* = false,
- *hiddenlayer_num* = [12,20,40] jeweils für die Hiddenlayeranzahl 1 und 2,
- *performance_fcn* = mse.

Dies ergibt in Summe eine Anzahl von 36 Simulationsdurchläufen. Tabelle C.6 im Anhang C zeigt dazu, aufsteigend nach dem MSE sortiert, die Ergebnisse zu den einzelnen Einstellungen. Die besten 10 Netzvarianten, gefunden über das globale Fehlermaß, sind in Tabelle 5.15 angeführt. Auffällig ist dabei, dass als Aktivierungsfunktion am Ausgang deutlich die tansig Funktion die besseren Ergebnisse liefert. Auch Netze mit zwei Hiddenlayer liefern gute Ergebnisse. Das globale Fehlermaß *eg* zeigt mittelmäßige Qualität des Netzes. Die Qualität ist jedoch bemerkenswert, wenn man den Punkt beachtet, dass kein Windowing

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
56.0606	55.2342	55.3719	55.9917	57.1625	57.0248	58.3333	58.5399

Tabelle 5.16: eg_j Werte der Prognoseintervalle des besten gefundenen Elman Netzes ohne Windowing.

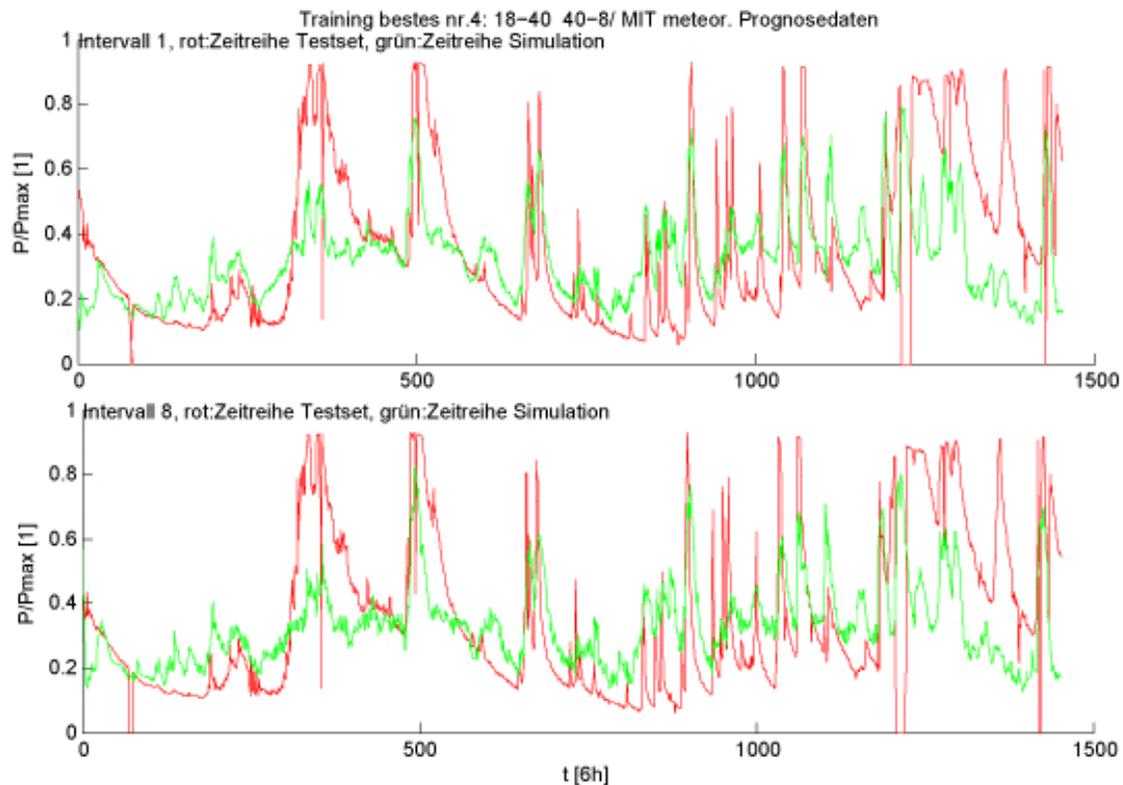


Abbildung 5.18: Simulation des gefundenen besten Elman Netzes ohne Windowing (oben: Intervall 1, unten: Intervall 8).

angewendet wird und die Sicht auf die Vergangenheit nur durch die Speicherung im internen Zustand erfolgt. Die eg_j Werte der einzelnen Intervalle, siehe Tabelle 5.16, zeigen, dass die einzelnen Fehler über die Intervalle im selben Bereich liegen. Interessant ist auch, dass bei den Netzen 25-36, siehe Tabelle C.6 im Anhang C, der Fehler massiv höher ist. Sie alle haben dabei die logsig Funktion als Aktivierungsfunktion im Output. Der Fehler ist so groß, dass offenbar keine Lösung gefunden wird. Die Simulation des Intervalls 1 und 8 des Netzes mit der Nr. 4 ist in Abbildung 5.18 dargestellt. Zu sehen ist, dass das Netz den Übergang der Frostperiode in die Tauwetterperiode ($t=250 \times 6h$ bis $400 \times 6h$) zumindest erkennt, auch wenn der Speicher nicht ausreicht, um den Niederschlag von zuvor auf akkumulieren zu können. Leistungsverläufe bei kurzen Niederschlagsperioden werden gut prognostiziert. Offenbar kann mit einem Elman Netz eine Prognose mit reinem dynamischen Speicher im internen Zustand gemacht werden, jedoch stößt man bei komplexeren Systemen mit komplexeren Speicherverhalten recht bald an die Grenzen.

5.3.2 Onlinetraining mit Windowing, ohne Information über Leistungswerte

Hierbei werden die Inputs wieder sequentiell dem Netz präsentiert und ein Training im Onlinemodus durchgeführt. Die Inputvektoren werden über das Windowingkonzept gewonnenen, wie es bei den feedforward neuronalen Netzen, siehe Kapitel 5.2, angewendet wurde. Im Input ist keine Information über aktuelle oder vergangene Leistungswerte enthalten. Simuliert werden folgende Kombinationen:

- $step = 24$,
- $forecast_steps = 8$,
- $incl_forecastvals = 1$,
- $from_past_steps_direkt = 1$,
- $past_steps = [30,50,70]$,
- $meanTemp_duration = [100,140,180]$,

Für die Netze werden folgende Konfigurationen erstellt:

- $is_mapstd_normalized = 1$,
- $activation_fcn = \text{tansig}$ für Hidden- und Outputlayer,
- $max_epochs = 600$,
- $showGUI = \text{false}$,
- $hiddenlayer_num = [200,300]$ jeweils für die Hiddenlayeranzahl 1 und 2,
- $performance_fcn = \text{mse}$.

In Summe ergibt das 36 Einzelsimulationen. Es können bei diesem Testlauf nicht allzu viele Variationen vorgenommen werden, da die Trainingszeit für solche Netzkonfigurationen bis zu 2 Stunden je Einzelsimulation beträgt. Tabelle C.7 im Anhang C zeigt die einzelnen Ergebnisse zu den Einstellungen. Tabelle 5.17 zeigt die besten 10 Netzeinstellungen, bewertet nach dem globalen Fehlermaß eg . Da zu wenig Variationen simuliert wurden, kann sehr schwer eine eindeutige Tendenz zu einer Konfiguration gefunden werden. Die Wettervorhersagedaten wurden bei allen Netzvarianten mit in den Input genommen. Das beste Netz (Nr. 1) hat zu den einzelnen Intervallen die Fehlermaße, wie sie in Tabelle 5.18 angeführt sind. Die Prognose über das Elman Netz liegt hier bezüglich der Qualität im selben Bereich, wie mit feedforward neuronalen Netzen. Die Rekurrenz zeigt keine merklichen Verbesserungen. Die Trainingsperformance ist deutlich schlechter. Elman Netze brauchen beim Training ca. 100mal länger als feedforward neuronale Netze. Die Abbildung der Simulation des besten Netzes ist hier nicht eigens angeführt.

Nr	Hidden-layer Anz.	Hidden n	Inkl. Wettervorhersage	Verg.-heitsfenster	Langzeitfenster	Zeit Training [s]	eg	MSE
1	1	200	1	30	140	2653.2	48.537	0.036855
7	1	200	1	70	140	2690.4	51.171	0.037863
6	2	300	1	30	140	5438.2	51.524	0.037765
24	1	200	1	50	140	2678.8	51.98	0.039816
3	2	300	1	70	140	5642.4	52.049	0.03735
5	2	200	1	70	140	3676.4	52.118	0.03757
23	2	200	1	30	140	3616.2	52.247	0.0398
12	1	200	1	50	100	2679.7	52.436	0.038881
4	1	300	1	70	100	3082.1	52.462	0.037455
13	1	200	1	70	100	2685	52.841	0.038904

Tabelle 5.17: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze mit Windowing, ohne Information über die Leistungswerte. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
47.8650	47.8650	47.8650	47.8650	47.3829	48.4160	50.8264	50.2066

Tabelle 5.18: eg_j Werte der Prognoseintervalle des besten gefundenen Elman Netzes mit Windowing, ohne Information über die Leistungswerte des betrachteten Kraftwerks.

5.3.3 Onlinetraining mit Windowing, mit Information über die Leistungswerte

Wie bei den feedforward neuronalen Netzen, Kapitel 5.2.5, soll ein Versuch unternommen werden, bei dem der Inputvektor Informationen über die Höhe der Leistung des Kraftwerks erhält. Dazu wird analog wie bei den feedforward neuronalen Netzen vorgegangen. Das Netz erhält zusätzlich ein Fenster mit der Sicht auf den aktuellen Leistungswert sowie vergangenen Leistungswerten. Die Länge des Fensters ist dieselbe, wie bei der Temperatur und dem Niederschlag. Die Konfiguration für die Simulationen ist dieselbe wie im Kapitel 5.3.2. Tabelle C.8, Anhang C, listet zur Simulation die Ergebnisse auf. In Tabelle 5.19 sind die Konfigurationen der besten 10 Netze aufgelistet, die nach dem globalen Fehlermaß gefunden wurden. Vom besten gefundenen Netz dieses Abschnitts, Nr. 1, sind die einzelnen Fehlerwerte der jeweiligen Prognoseintervalle in Tabelle 5.20 angeführt. Auch hier ist die selbe Charakteristik wie bei den feedforward neuronalen Netzen zu erkennen. Die eg_j Werte des ersten Intervalls sind sehr niedrig. Die Werte steigen aber stark mit steigender Intervallanzahl an. Dies deutet, wie bei den feedforward neuronalen Netzen mit gleichem Input, darauf hin, dass der aktuelle Leistungswert stark ins Gewicht fällt und verzögert auf den Output wirkt. Folgende interessante Punkte können bei den Simulationsergebnissen beobachtet werden:

- Wie es scheint, reicht ein Netz mit einem Hiddenlayer für eine gute Prognose aus.
- Es kommt kein Vergangenheitsfenster größer 50 x 6h vor.

Nr	Hidden-layer Anz.	Hidden n	Inkl. Wettervorhersage	Verg.-heitsfenster	Langzeitfenster	Zeit Training [s]	eg	MSE
1	1	200	1	30	180	2662.7	19.611	0.017753
3	1	200	1	30	140	2659.6	19.792	0.017918
17	1	300	1	50	100	3100.2	20.687	0.019124
9	2	200	1	30	100	3663.5	20.808	0.018615
6	1	200	1	50	140	2689	20.876	0.018385
11	1	200	1	30	100	2671.9	20.911	0.018779
10	1	200	1	50	100	2707.4	21.04	0.018773
4	2	300	1	30	180	5525	21.092	0.018332
13	1	300	1	30	140	2990.4	21.212	0.018831
2	1	200	1	50	180	2700.3	21.358	0.017844

Tabelle 5.19: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Elman Netze mit Windowing und mit Information der Leistungswerte. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
9.6419	14.0496	17.3554	19.1460	21.4187	23.6915	24.7245	26.8595

Tabelle 5.20: eg_j Werte der Prognoseintervalle des besten gefundenen Elman Netzes mit Windowing und mit Information über Leistungswerte des betrachteten Kraftwerks.

- Über die Langzeitfenster und Hiddenlayer kann nichts ausgesagt werden.
- Die Trainingszeiten sind sehr lang. Teilweise betragen sie knapp 2h je Netz!
- Das Fehlermaß eg liegt im vergleichbaren Bereich, wie bei feedforward neuronalen Netzen mit gleichem Input, siehe Tabelle 5.14.
- Der Simulationsverlauf des besten Netzes ist sehr ähnlich dem Verlauf des besten feedforward neuronalen Netzes, wird daher nicht eigens angeführt.

5.3.4 Bemerkungen zu den Elman rekurrenten Netzen

Elman Netze liefern gegen den Erwartungen keine merklichen besseren Ergebnisse. Obwohl bei der Simulation des reinen Onlinetrainings ohne Windowing bereits sehr vielversprechende Ergebnisse aufgetreten sind, konnte mit den Elman Netzen die Prognosequalität verglichen zu den feedforward neuronalen Netzen nicht verbessert werden. Die Trainingszeiten steigen im Vergleich zum Training von feedforward neuronalen Netzen stark an. Dies macht mehrfache Simulationen sehr mühsam und zeitaufwändig.

5.4 Echo State Netze

Die dritte ausgewählte Methode eine Leistungsprognose von WLKW mit einem neuronalen Netz zu machen, ist die Methode mit Echo State Netzen. Die Theorie wurde dazu

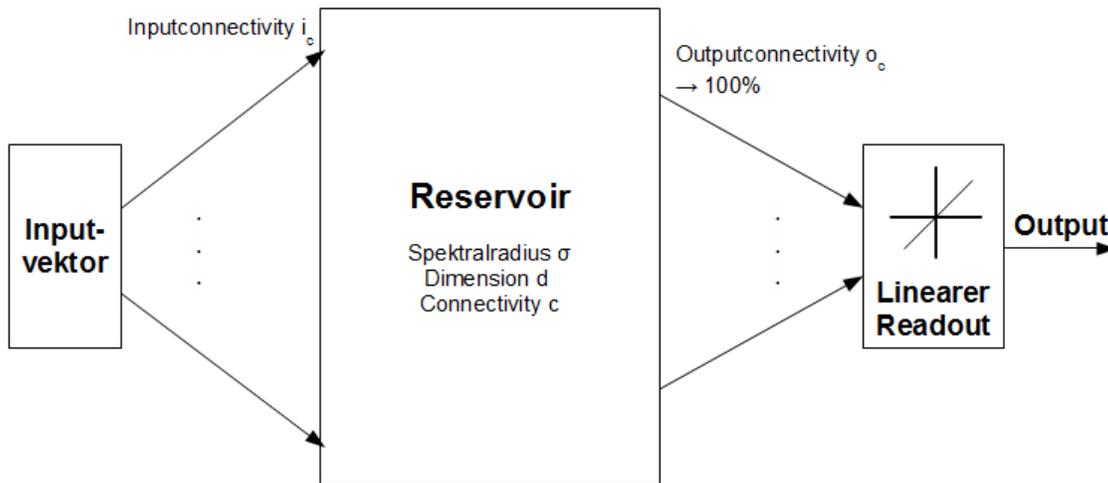


Abbildung 5.19: Einfaches Echo State Netz ohne Rückkopplung und einem linearen Readout.

bereits im Kapitel 4.2.3 angeführt. Die Implementierung folgt analog der angegebenen Formalismen und beschriebenen Theorie.

Der erste Teil in diesem Abschnitt zeigt die Simulationsergebnisse eines Modells eines Echo State Netzes mit einem einfachen Reservoir, mit einem linearen Readout und ohne Windowing. Dazu werden mehrere Konfigurationen simuliert. Es wird auch der Unterschied zu der Variante aufgezeigt, wenn anstatt des linearen Readouts, ein Perzeptron mit der logsig-Funktion als Outputregression trainiert wird.

Der zweite Teil präsentiert die Ergebnisse einer Simulation eines Echo State Netzes mit 3-fachen Reservoir mit unterschiedlichen Spektralradien und einem Perzeptron als Ausgangsneuron. Hierzu wird das Konzept des Windowings angewendet, jedoch ohne Information über den Verlauf der Leistungswerte des Kraftwerks.

Im dritten Teil soll dann in den Inputvektor zusätzlich Information über die Leistungswerte einfließen.

5.4.1 Einfaches Echo State Netz mit Onlinetraining, ohne Windowing, ohne Information über Leistungswerte

Hierbei wurde ein Echo State Netz ohne Rückkopplung des Outputs auf das Reservoir mit einem linearen Readout simuliert, siehe Abbildung 5.19. Das Netz besitzt dabei eine bestimmte Eingangsverbindungswahrscheinlichkeit (inputconnectivity) i_c , und eine 100%ige Outputverbindungswahrscheinlichkeit (outputconnectivity) o_c . Für Verbindungswahrscheinlichkeiten gilt, dass z.B. 0.1 (= 10%) der Verbindungen nicht mit 0 gewichtet sind. Die Lösung über den linearen Readout wird, wie in 4.2.3 beschrieben, über die Inverse der sogenannten Zustandssammelmatrix (state-collect Matrix) gefunden. Das Reservoir hat einen Spektralradius σ , eine bestimmte interne Verbindungswahrscheinlichkeit (connectivity) c und eine Dimension d . Die Simulation wurde wiederum in Matlab erstellt. Für den Simulationslauf werden folgende Einstellungen für die Input/Outputkonfigurationen gewählt:

- $step = 24$,
- $forecast_steps = 8$,
- $incl_forecastvals = [0,1]$,

Die Einstellungen für die Vergangenheitsfenster fallen hier weg, da es kein Windowing gibt. Für die Netze werden folgende Konfigurationen erstellt:

- $netDim = [50,100,200,500]$; beschreibt die Werte für die Variationen der Netzdimension.
- $specRad = [0.1,0.3,0.5,0.7,0.9,0.99]$; beschreibt die Werte für die Variationen des Spektralradius.
- $connectivity = [0.1,0.15,0.2,0.5]$; beschreibt die Werte der Variationen der internen Verbindungswahrscheinlichkeit.
- $inputconnectivity = [0.1,0.4,0.5]$; beschreibt die Werte der Variationen der Eingangsverbindungswahrscheinlichkeit.
- $inputscale = [0.5,1,2]$; beschreibt die Werte für die Variationen der Eingangsskalierung. Zusätzlich wird der Input gewichtet. Dadurch wirkt ein Inputvektor einmal schwach und einmal stark auf den Zustand des Reservoirs.
- $act_fcn = \text{tansig}$; beschreibt die Aktivierungsfunktion der internen Neuronen und ist standardmäßig die tansig-Funktion.
- $output_fcn = \text{purelin}$; beschreibt den linearen Readout.
- $shift_transient_len = 150$; bedeutet, dass die ersten 150 Zustände nicht in die Zustandssammelmatrix aufgenommen werden, um transiente Vorgänge beim *Einschwingen* des Reservoirs auszulassen.

In Summe ergibt dies 1728 verschiedene Netzvarianten. Tabelle C.9 im Anhang C zeigt dazu die besten nach dem MSE bewerteten 100 Netze. Die besten 10 Netzvarianten, gefunden über das globale Fehlermaß eg , sind in Tabelle 5.21 angeführt. Zu erkennen ist Folgendes:

- Die Eingangsverbindungswahrscheinlichkeit soll offensichtlich klein sein.
- Die Connectivity scheint keine klare Tendenz zu einem Wert hin zu zeigen.
- Sind die Wettervorhersagewerte nicht im Input enthalten, so soll der Spektralradius sehr groß sein. Generell werden aber für diesen Fall dann gute Lösungen gefunden, wenn die Wettervorhersagewerte mit im Input enthalten sind.
- Die Netzdimension zeigt unterschiedliche Werte, jedoch fast immer über 100.

Nr	Netz Dimension	σ	c	i_c	input scale	inkl. Wetter Vorhersage	Zeit Training [s]	eg	MSE
43	500	0.99	0.15	0.1	0.5	0	64.998	59.005	0.062354
44	500	0.1	0.1	0.1	1	1	65.339	59.874	0.06242
81	200	0.3	0.15	0.1	1	1	25.59	60.098	0.063601
13	200	0.5	0.15	0.1	2	1	25.596	60.176	0.06107
23	200	0.5	0.1	0.1	2	1	25.566	60.176	0.061484
25	100	0.1	0.2	0.1	2	1	13.38	60.365	0.061557
100	500	0.3	0.15	0.1	1	1	65.008	60.563	0.064084
3	200	0.3	0.1	0.1	2	1	25.614	60.623	0.059271
91	200	0.9	0.5	0.1	1	0	25.074	60.735	0.063759
72	500	0.3	0.15	0.1	2	1	65.105	60.838	0.063355

Tabelle 5.21: Eigenschaften der besten 10 Netze nach dem Simulationsdurchlauf der Echo State Netze ohne Windowing. Aktivierungsfunktion Ausgang: purelin.

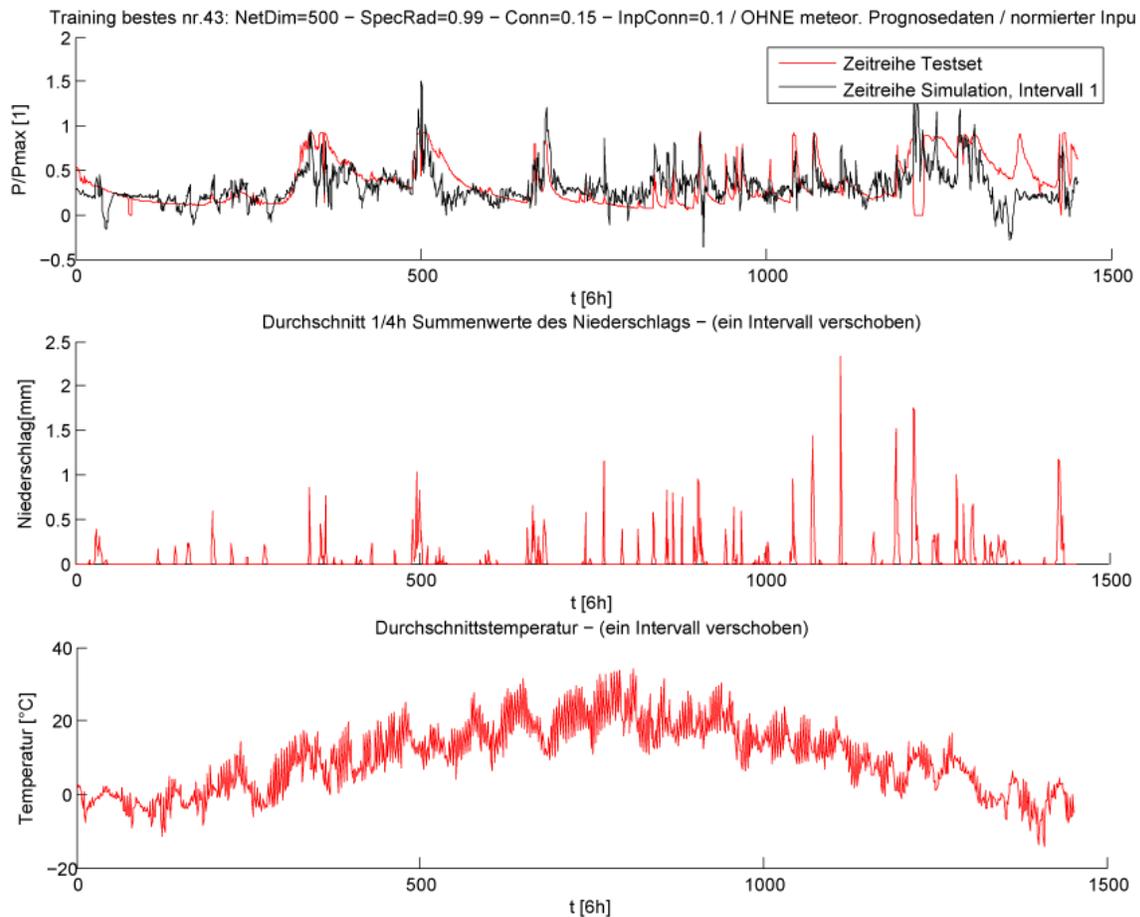


Abbildung 5.20: Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing.

Offensichtlich gibt es für diese Art von Netzen mehrere gute Lösungen. Die *eg* Fehlerwerte sind aber prinzipiell recht hoch. Dies zeigt, dass zwar mehrere Lösungen gefunden werden, diese aber im Vergleich zu den bisher angeführten Methoden und Ergebnisse schlecht sind. Abbildung 5.20 zeigt dazu die Simulation des ersten Intervalls des besten nach dem globalen Fehlermaß gefundenen Netz mit der Nr. 43. Zusätzlich ist in der Abbildung auch der Niederschlags und Temperaturverlauf dargestellt, um die Einflüsse besser sichtbar zu machen. Es wird der Übergang der Frostperiode zum Tauwetter ($t=250 \times 6h$ bis $400 \times 6h$) erkannt. Bei kurzen Niederschlägen im Sommer kann das Netz tendenziell richtige Prognosewerte liefern. Die Auswirkung des Niederschlags im Winter wird korrekterweise als gering prognostiziert. Der Prognoseverlauf weist einen sehr verrauschten Verlauf auf. Dies kann einerseits an der *purelin*-Funktion am Ausgang liegen, andererseits könnte das Rauschen der Temperatur auf den Ausgang *durchgreifen*. Prinzipiell ist auch im Falle des Echo State Netzes der dynamische Speicher zu gering, um ein ausreichendes Speicherverhalten der Umwelt nachzubilden, wie es bei den Elman Netzen ohne Windowing der Fall war, siehe Kapitel 5.3.1.

Ein Versuch, bei dem das selbe Netz (Netz Nr. 43) mit geglättetem Temperaturverlauf simuliert wird, wird in Abbildung 5.21 dargestellt. Die Temperatur wurde hierzu mit einem gleitendem Durchschnitt mit einem 20 Datenpunkte langen Fenster geglättet. Der *eg* Fehlerwert liegt bei diesem Versuch bei $eg = 64.428$. Die *eg* Werte sind im Vergleich zur ungeglätteten Temperatur etwas größer geworden. Das Rauschen des Verlaufes ist aber kaum geringer geworden. Der sehr *unruhige* Verlauf dürfte auf Grund der *purelin*-Funktion am Readout zustande kommen.

Um zu überprüfen, ob der unruhige Verlauf aufgrund der *purelin*-Funktion des Readouts zustande kommt, wird das Echo State Netz so umgebaut, dass am Ausgang ein Perzeptron mit der *logsig*-Funktion als Aktivierungsfunktion die Regressoin bildet, siehe Abbildung 5.22. Anstatt der Invertierung der Zustandssammelmatrix, wird das Perzeptron über wenige Backpropagationsschritte trainiert. Abbildung 5.23 zeigt das beste Netz, erneut mit einem Perzeptron als Ausgangsneuron. Der Fehler *eg* liegt nach dieser Simulation bei 70.665. Der Fehler ist zwar deutlich höher, aber der Verlauf ist deutlich *ruhiger*. Der Fehler ist vermutlich deshalb noch sehr groß, weil

- kein Windowing verwendet wird und
- ein einfaches Reservoir mit nur einem Spektralradius bei einer Strecke mit sehr komplexem Speicherverhalten an dessen Grenzen stößt.

5.4.2 Erweitertes Echo State Netz mit Onlinetraining und Windowing, ohne Information über Leistungswerte

Um komplexeres speicherndes Verhalten der Strecke abbilden zu können, wird in diesem Abschnitt ein Echo State Netz mit einem 3-fachen Reservoir getestet. Die Reservoirs besitzen unterschiedliche Spektralradien, dh. $\sigma_1 < \sigma_2 < \sigma_3$. Sie können unterschiedliche Verbindungswahrscheinlichkeiten und Dimensionen aufweisen, sollen aber in dieser Simulation gleiche Werte erhalten. Am Ausgang wird wieder ein Perzeptron zur Regression verwendet, wobei alle Outputs aller drei Reservoirs auf das Perzeptron als Input gelegt werden. In Abbildung 5.24 ist das erweiterte Echo State Netz schematisch dargestellt.

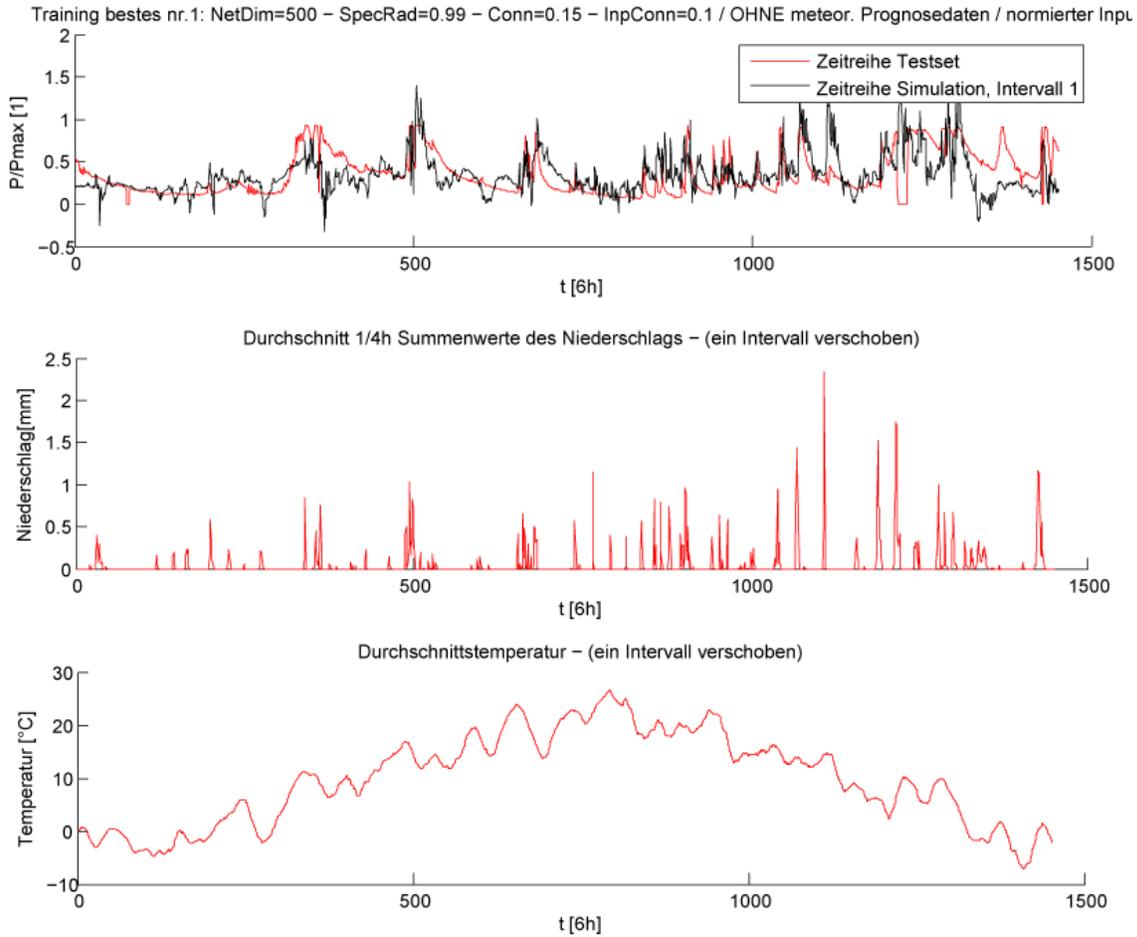


Abbildung 5.21: Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing, mit geglättetem Temperaturverlauf.

Außerdem wird in diesem Abschnitt das Windowingkonzept, siehe 5.2.1, für weitere Inputparameter angewendet, wobei die Information über den aktuellen Leistungswert und vergangene Leistungswerte nicht in den Input mit einfließen. Die Implementierung ist analog zu jener mit nur einem Reservoir. Dazu werden drei eigenständige Reservoirs und drei eigenständige Zustandssammelmatrizen gebildet. Der Inputvektor wird auf jedes Reservoir auf die gleiche Art abgebildet. Für die Simulation werden folgende Einstellungen für die Input/Outputkonfigurationen gemacht:

- $step = 24,$
- $forecast_steps = 8,$
- $incl_forecastvals = 1,$
- $past_steps = [40,75],$
- $meanTemp_duration = [150,200],$

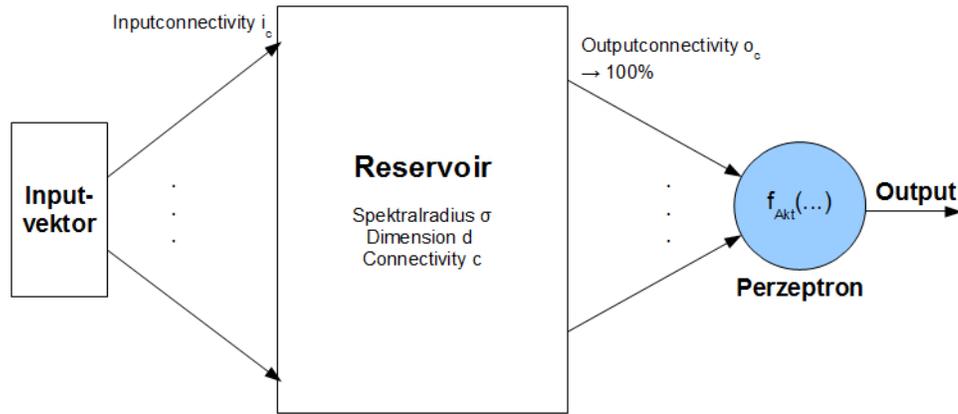


Abbildung 5.22: Echo State Netz mit einem Perzeptron als Readoutelement.

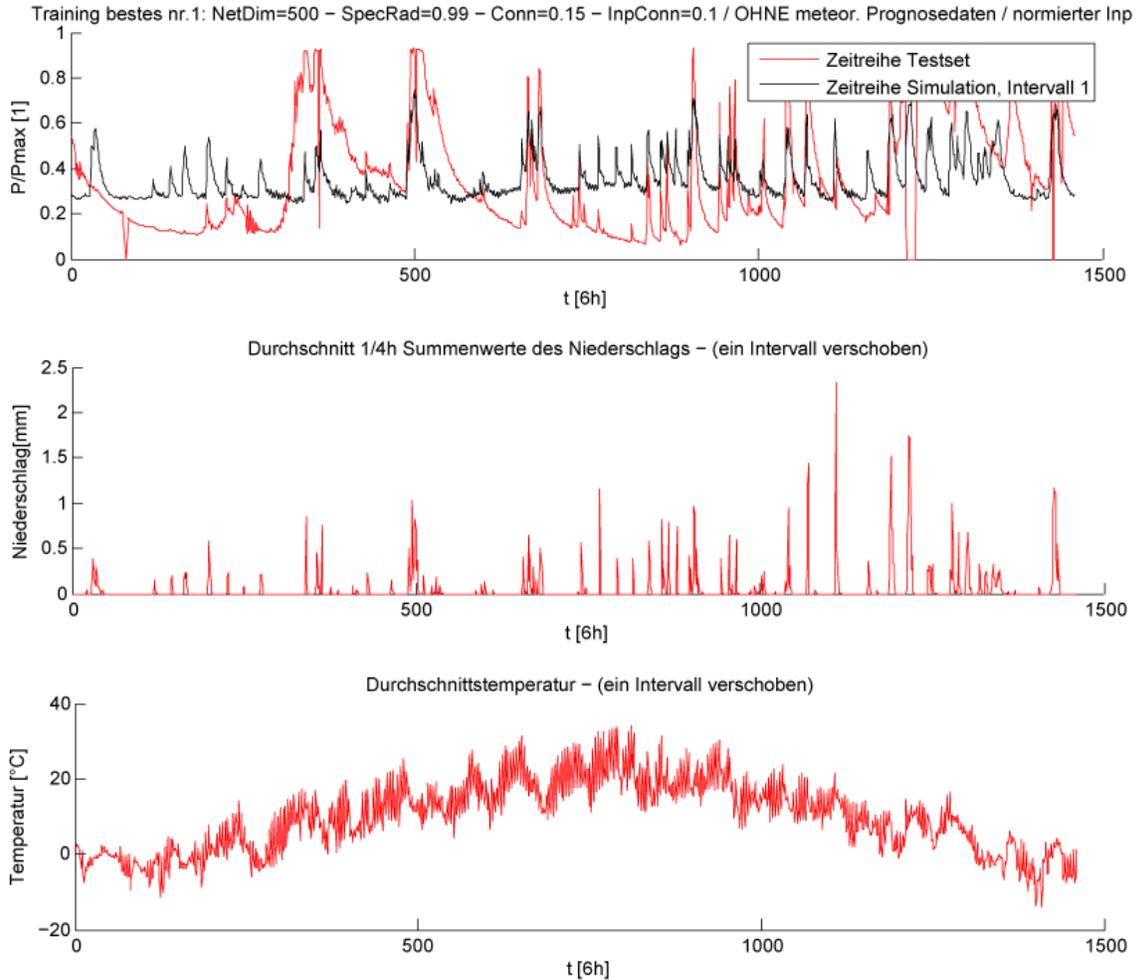


Abbildung 5.23: Simulation des besten Netzes der Variationen aus Echo State Netzen mit linearem Readout, ohne Windowing, mit einem Perzeptron als Ausgangsneuron anstatt dem linearen Readout.

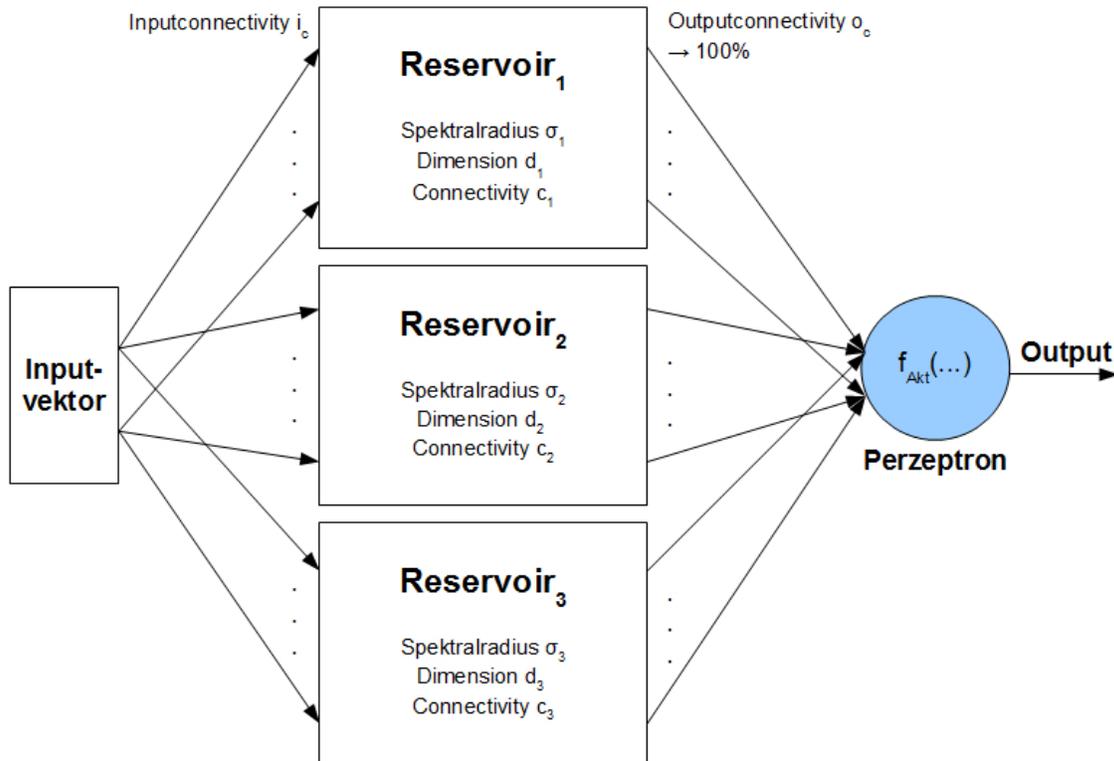


Abbildung 5.24: Erweitertes Echo State Netz mit einem Perzeptron als Readoutelement.

Für die Netze werden folgende Konfigurationen erstellt:

- $netDim = [50,100,250]$; beschreibt die Netzdimension eines Reservoirs,
- $[specRad\ 1, specRad\ 2, specRad\ 3] = [[0.05;0.15;0.3] , [0.05;0.1;0.9] , [0.05;0.5;0.99] , [0.1;0.9;0.99] , [0.5;0.75;0.99]]$; Werte für die Variationen und Kombinationen der Spektralradien der Reservoirs,
- $connectivity = [0.1,0.2]$,
- $inputconnectivity = [0.1,0.4,0.6]$,
- $inputscale = [0.8,1,2]$,
- $act_fcn = \text{tansig}$; beschreibt die Aktivierungsfunktion der internen Neuronen und ist standardmäßig die tansig-Funktion,
- $output_fcn = \text{logsig}$; beschreibt die Aktivierungsfunktion des Perzeptrons am Ausgang,
- $shift_transient_len = 200$; shift der Transienten.

Das ergibt 1080 verschiedene Simulationsvarianten. Tabelle C.10 im Anhang C zeigt dazu die besten nach dem MSE bewerteten 100 Netze. Die besten 10 Netzvarianten, gefunden

Nr	N_D	σ_1	σ_2	σ_3	c	i_c	in_s	i_W	pS	mT	Zeit Tr. [s]	eg	MSE
17	250	0.05	0.5	0.99	0.1	0.4	0.8	1	75	200	69.602	57.223	0.040505
12	100	0.05	0.1	0.9	0.2	0.1	2	1	75	150	27.511	58.772	0.039955
69	50	0.05	0.1	0.9	0.1	0.1	0.8	1	40	150	14.398	58.867	0.042437
7	250	0.1	0.9	0.99	0.1	0.1	0.8	1	75	150	69.068	59.978	0.039334
72	100	0.05	0.15	0.3	0.1	0.1	0.8	1	75	200	27.888	59.995	0.042471
9	250	0.1	0.9	0.99	0.2	0.6	0.8	1	40	150	69.173	61.183	0.039632
58	100	0.5	0.75	0.99	0.2	0.1	1	1	40	150	28.363	61.217	0.042169
35	250	0.1	0.9	0.99	0.2	0.1	0.8	1	75	150	68.674	61.226	0.041463
32	250	0.5	0.75	0.99	0.1	0.6	1	1	75	150	71.761	61.303	0.0414
89	250	0.05	0.1	0.9	0.1	0.1	1	1	75	150	68.868	61.846	0.042872

Tabelle 5.22: Eigenschaften der besten 10 Konfigurationen, gefunden über das globale Fehlermaß *eg*. Mit Windowing am Eingang, ohne Information über die Leistungswerte.

- N_D ... Netz/Reservoir Dimension
- σ_n ... Spektralradius vom Reservoir n
- c ... Connectivity der Reservoirs
- i_c ... Inputconnectivity zu den Reservoirs
- in_s ... Inputskalierung
- i_W ... inklusive Wettervorhersagewerte (ja 1 / nein 0)
- pS ... Länge des Vergangenheitsfenster
- mT ... Langzeittemperaturfenster

über das globale Fehlermaß, sind in Tabelle 5.22 angeführt. Folgendes kann beobachtet werden:

- Für gute Konfigurationen liegen die Spektralradien weit auseinander, dh. ein kombiniertes Netz besitzt einen sehr kleinen, einen mittleren und einen großen Spektralradius. Das bedeutet, dass sehr kurze, mittelmäßig lange und sehr lange Einflüsse vorliegen.
- Für die Verbindungswahrscheinlichkeit der internen Verbindungen scheint eine sehr lose Verbindung angebracht zu sein.
- Ebenfalls sollte die Inputskalierung eher kleiner 1 gewählt werden.
- Ein eher längeres Vergangenheitsfenster scheint die Qualität zu verbessern.
- Die Trainingszeit ist für RNN bemerkenswert gut und liegt im Minutenbereich. Dies ist deutlich besser als bei den Elman Netzen.
- Das globale Fehlermaß liegt hier aber deutlich unter den Erwartungen und unter jenen der anderen angeführten Methoden.

Methode	<i>eg</i> , inkl. Leistungswerte	<i>eg</i> , exkl. Leistungswerte
Feedforward neuronales Netz	18.397	45.377
Elman Netz	19.611	48.537
Echo State Netz	30.622	57.223

Tabelle 5.23: *eg* der besten Netze unterschiedlicher Methoden und Ansätze.

5.4.3 Erweitertes Echo State Netz mit Onlinetraining und Windowing, mit Information über die Erzeugungswerte

Führt man die selbe Simulation mit der selben Netzstruktur und den selben Konfigurationen wie im Kapitel 5.4.2 erneut durch und nimmt aber aktuelle und vergangene Leistungswerte in den Input mit hinein, so ergibt sich für das beste nach dem globalen Fehlermaß gefundene Netz ein $eg = 30.622$, mit einer Trainingsdauer von $t = 27.775s$. Diese Methode liefert bezüglich der Güte deutlich schlechtere Ergebnisse, als die anderen beschriebenen Ansätze mit der selben Inputkonfiguration.

5.4.4 Bemerkungen zu den Echo State Netzen

Der Ansatz mit Echo State Netzen liefert Ergebnisse, die unter den Erwartungen liegen. Die Streckencharakteristik der WLKW scheint zu komplex zu sein, um sie mit Echo State Netzen mit relativ einfachen Ansätzen nachbilden zu können. Mit sogenannten *Leaky-Integrator* Echo State Netzen, die einen Ansatz mit einem *internen* Integrator zur Nachbildung von Lanzeiteinflüssen verwenden, sowie Echo State Netzen mit Rückkopplungen, könnten eventuell bessere Ergebnisse erzielt werden. Diese Ansätze werden hier nicht weiter verfolgt.

5.5 Auswahl des regressionsbasierten Modells

Ein einfacher Vergleich der unterschiedlichen Methoden zeigt, dass die feedforward neuronalen Netze mit den Elman Netze in der Prognosequalität gleich auf liegen und die Echo State Netze ausstechen, siehe Tabelle 5.23. Tabelle 5.24 zeigt den Vergleich der Trainingszeiten der unterschiedlichen Arten und Methoden. Dabei liegen die feedforward neuronalen Netze mit den Echo State Netzen gleich auf. Die Elman Netze haben, wie auch in sämtlichen Literaturen erwähnt, enorme Trainingszeiten. Auf Grund der Simulationen und Ergebnisse, kann die Aussage getroffen werden, dass feedforward neuronale Netze für die Prognose von WLKW zu bevorzugen sind. Dies ist etwas gegen den Erwartungen, da in den verwandten Arbeiten eher die RNN das Rennen auf Grund ihrer dynamischen Speicherfähigkeit machen. Offensichtlich ist die Streckencharakteristik der WLKW zu komplex, um über Methoden der Zustandsraummodellierung eine sehr gute Prognose zu erreichen. Offenbar ist die Methode der Mustererkennung auf Grund der Komplexität besser. Feedforward neuronale Netze mit der Windowing-Methode für die Generierung der Inputs ist eine Form der Mustererkennung, indem Featurevektoren erkannt werden und der zugehörige Output erzeugt wird.

Methode	t[s], inkl. Leistungswerte	t[s], exkl. Leistungswerte
Feedforward neuronales Netz	42.135	22.803
Elman Netz	2662.7	2653.2
Echo State Netz	27.775	69.602

Tabelle 5.24: Trainingsdauer der besten Netze unterschiedlicher Methoden und Ansätze.

5.6 Weitere Untersuchungen auf das ausgewählte regressionsbasierte Modell

Es gilt die Robustheit gegenüber realen Wetterprognosedaten zu testen. Die aufgezeichneten Zeitreihen von Niederschlag und Temperatur stellen hierzu exakte Wetterdaten dar. Diese exakten Wetterdaten wurden für die bisherigen Untersuchungen auch für die Werte der Wetterprognosedaten verwendet. Von den realen Wetterprognosedaten sind keine Zeitreihen vorhanden. Man kann sie aber simulieren, indem man *Gaußsches adaptives weißes Rauschen* auf die exakten Wetterdaten legt, oder die Zeitreihen Zeitpunkt für Zeitpunkt *zufällig skaliert*. Das Trainingsset erhält neben allen anderen Inputdaten die exakten Wetterdaten für die Wettervorhersagedaten, da beim Training Messdaten über die Wetterdaten vorhanden sind. Das Testset erhält aber nicht die exakten, sondern die verrauschten Wettermesswerte. Damit wird der Fall simuliert, dass fehlerhafte Wetterprognosedaten auftreten.

Außerdem soll das gefundene Modell dazu verwendet werden, Prognose auf weitere WLKW durchzuführen, wobei für die Vergleichbarkeit mit dem WLKW für die Suche des besten Modells wiederum die exakten Wetterwerte für die Wetterprognosedaten am Input verwendet werden sollen.

5.6.1 Robustheit gegenüber realen Wetterprognosedaten

Dazu werden die exakten Vergangenheitszeitreihen der Wetterdaten auf zwei Arten manipuliert:

- Addieren von Gaußschen weißen Rauschen.
- Zufällige Skalierung.

Abbildung 5.25 zeigt einen Ausschnitt der Wetterdaten, gegenübergestellt mit den Wetterdaten mit Gaußschen additiven weißen Rauschen. Zu den Niederschlagsmesswerten wird ein Rauschen mit

$$\mu = 0, \sigma^2 = 1 \tag{5.4}$$

addiert. Dadurch entstehende negative Werte bei den Niederschlagsmesswerten wurden mit 0 ersetzt. Zu den Temperaturmesswerten wird ein Rauschen mit

$$\mu = 0, \sigma^2 = 10 \tag{5.5}$$

addiert. Abbildung 5.26 zeigt einen Ausschnitt der zufällig skalierten Wetterdaten, gegenübergestellt mit den exakten Wetterdaten, wobei die jeweiligen Skalierungsfaktoren

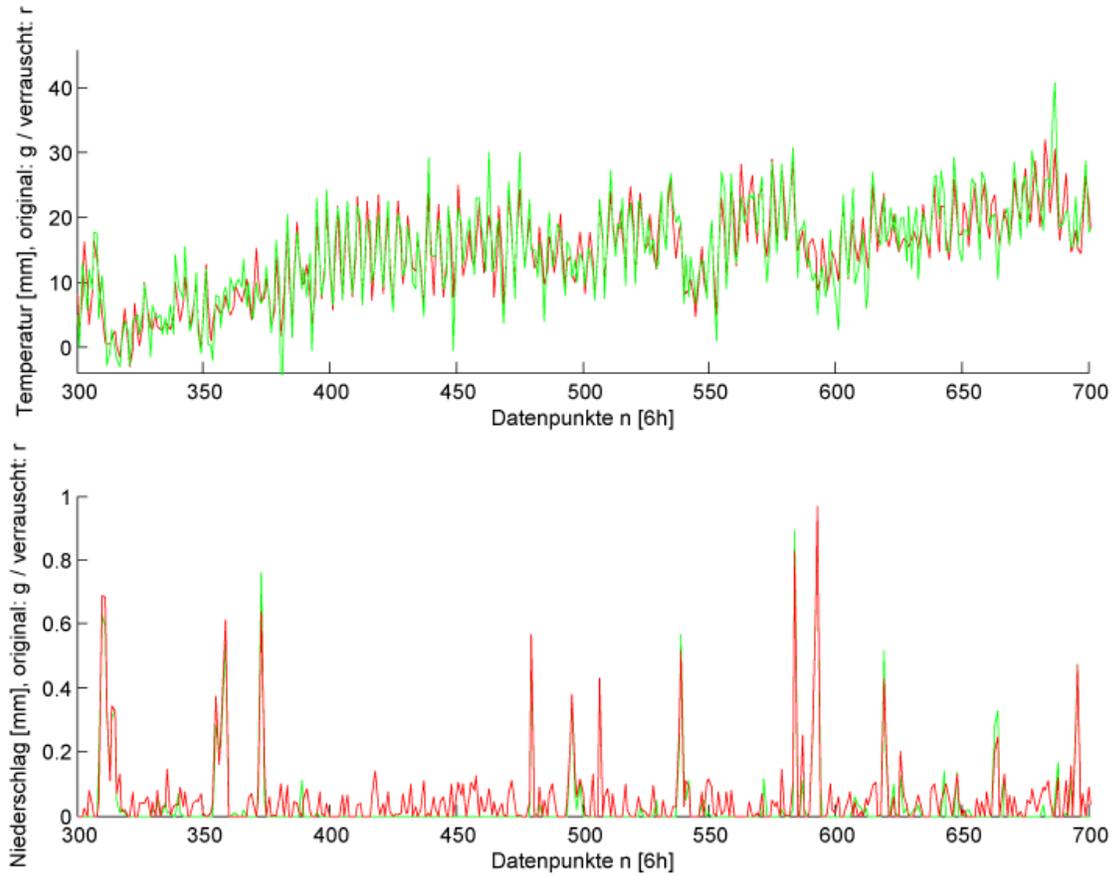


Abbildung 5.25: Ausschnitt des Vergleiches der exakten (gemessenen) Wetterdaten mit den Wetterdaten mit Gaußschen additiven weißen Rauschen.

aus einer Gaußverteilung folgen. Für die Niederschlagsmesswerte wird mit einem Rauschen mit

$$\mu = 0, \sigma^2 = 0.05 \tag{5.6}$$

skaliert. Für die Temperaturmesswerte wird mit einem Rauschen mit

$$\mu = 0, \sigma^2 = 0.02 \tag{5.7}$$

skaliert. Das Modell erhält im Training die exakten Wettervorhersagedaten, und im Test die verrauschten Wetterdaten als Wettervorhersagedaten. Jeweils für die Modellart mit der Information über die Leistungszeitreihe bzw. ohne Information darüber. Beide Arten von Rauschen werden verwendet. Jede Modellart mit jeder Art des Rauschens wird 100mal simuliert. Die verrauschten Datenreihen bleiben dabei konstant, dh. die Wetterdaten werden nicht jedesmal erneut verrauscht. Das Modell mit bzw. ohne Information über die Leistungszeitreihe, jeweils mit den exakten Wetterprognosedaten für das Trainings- und Testset, wird ebenfalls 100mal simuliert. Der Grund der mehrfachen Simulation ist, dass der Gradientenabstieg sowie die zufällige Initialisierung der Gewichte des neuronalen Netzes nicht deterministisch sind. Bei mehrfacher Simulation sollte sich aber ein relativ

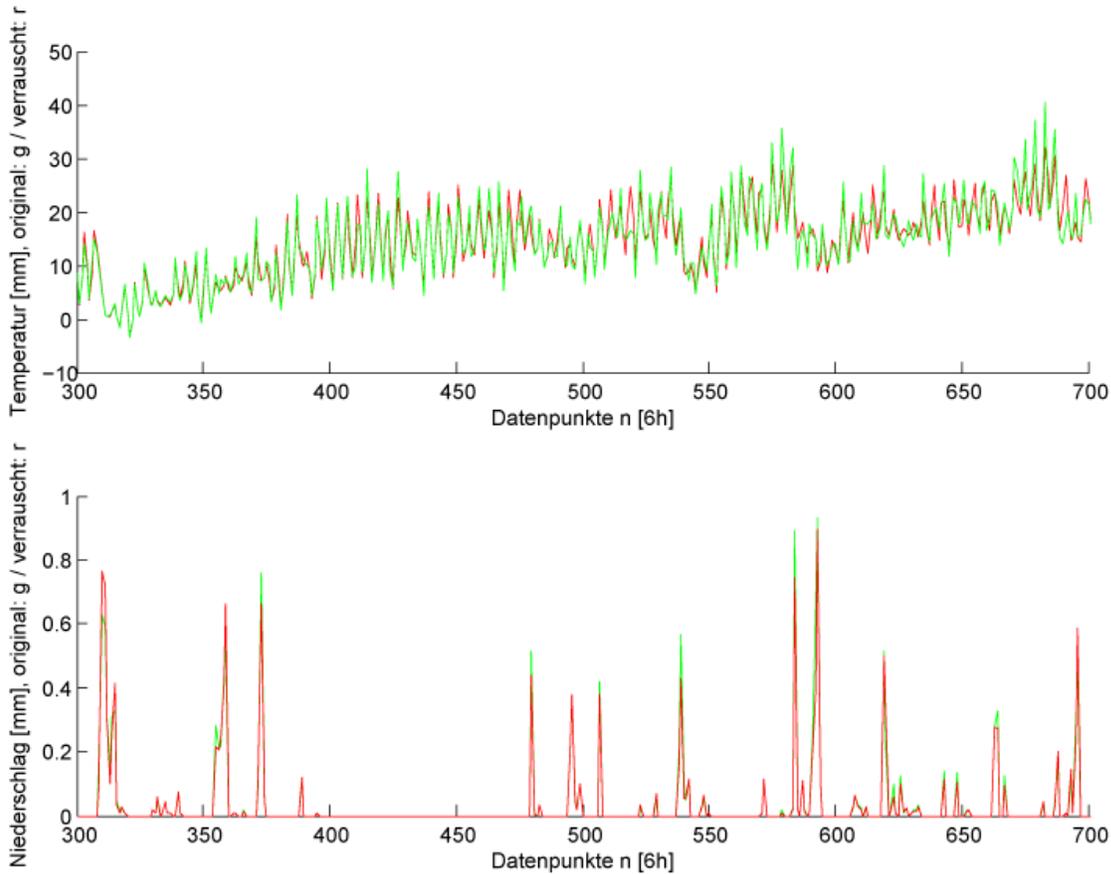


Abbildung 5.26: Ausschnitt des Vergleiches der exakten (gemessenen) Wetterdaten mit den Wetterdaten zufällig skaliert.

konstanter Mittelwert des Gütemaßes jedes Intervalls einstellen, somit ist das Modell mit den verrauschten Wetterdaten, mit jenem der exakten Wetterdaten vergleichbar. Die Darstellung erfolgt mit Hilfe eines *Boxdiagramms* bzw. *Boxplots*. Der Boxplot in Abbildung 5.27 zeigt dabei die Lage der Fehler der einzelnen Intervalle der Netze mit den exakten Wettervorhersagedaten im Testset. Der obere Plot zeigt dabei die Netze mit der Information über die Leistungswerte, der untere Plot jene ohne dieser Information. Abbildung 5.28 zeigt dazu den Boxplot der Simulationen mit den additiv Gauß-verrauschten Wetterdaten als Wettervorhersagedaten im Testset. Abbildung 5.29 zeigt den Boxplot zu den Simulationen mit zufällig, nach einem Gaußrauschen, skalierten Wetterdaten als Wettervorhersagedaten. Tabelle 5.25 beinhaltet zu den Boxplots die Mittelwerte der Fehlermaße eg_j der einzelnen Intervalle und Tabelle 5.26 das Fehlermaß eg . *inkl.* bedeutet dabei die Netzart mit der Information über die Leistungswerte im Inputvektor, *exkl.* bedeutet dabei die Netzart ohne Information darüber. Die Ausreißer in den Boxplots lassen sich dadurch erklären, dass hin und wieder beim neuronalen Netz die Initialisierung der Gewichte so getroffen wird, dass keine sinnvolle Lösung über einen Gradientenabstieg zu finden ist. Zu beobachten ist, dass das Fehlermaß etwas höher wird, wenn die Wetterprognosedaten

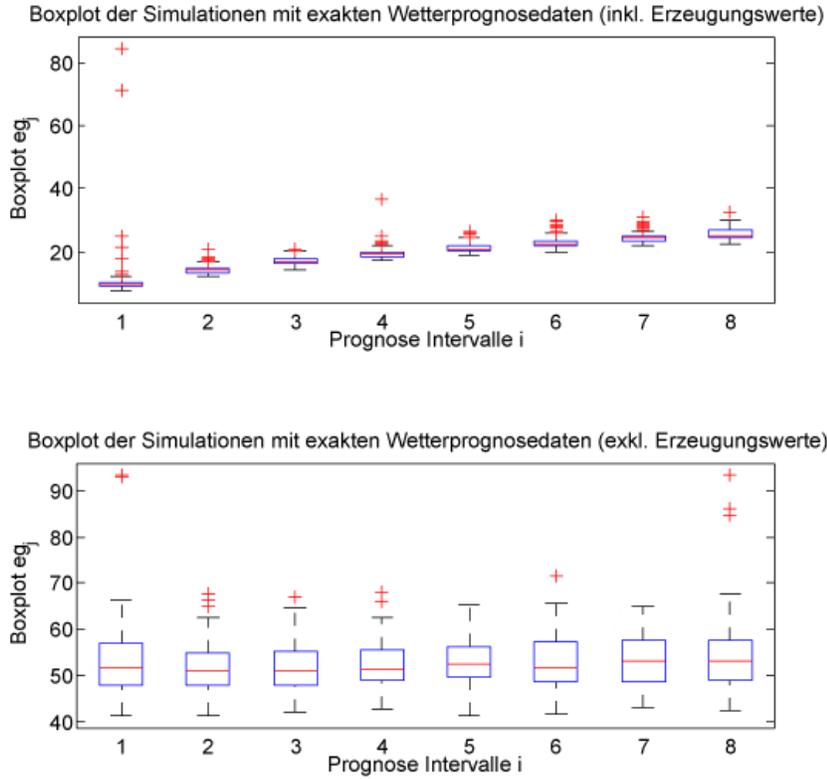


Abbildung 5.27: Boxplot der Simulationen mit den exakten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.

Rauschen	eg_j I1	eg_j I2	eg_j I3	eg_j I4	eg_j I5	eg_j I6	eg_j I7	eg_j I8
Exakt/inkl.	11.402	14.242	17.090	19.406	21.312	23.039	24.526	25.445
Gauß add./inkl.	12.760	15.946	19.087	21.611	24.225	26.340	28.312	29.591
Gauß skal./inkl.	12.475	15.117	17.529	19.738	22.062	23.861	25.247	26.503
Exakt/exkl.	53.106	51.876	51.819	52.341	53.342	52.891	53.454	54.354
Gauß add./exkl.	53.158	52.223	53.407	53.389	53.508	55.098	56.381	57.073
Gauß skal./exkl.	53.516	51.014	52.185	51.084	50.955	52.105	52.920	53.347

Tabelle 5.25: Mittlere eg_j Werte der 100 Simulationen des besten Netzes mit/ohne ver-rauschten Wetterdaten als Wettervorhersagedaten im Testset.

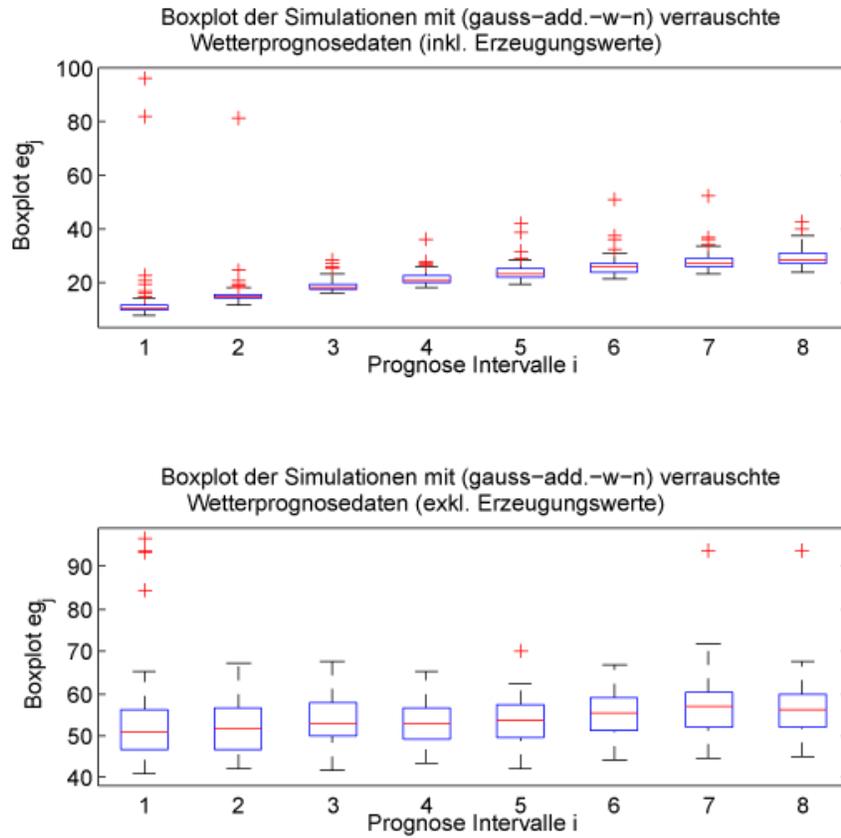


Abbildung 5.28: Boxplot der Simulationen mit den Gauß-verrauschten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.

Rauschen	<i>eg</i>
Exakt/inkl.	19.5581
Gauß add./inkl.	22.2344
Gauß skal./inkl.	20.3171
Exakt/exkl.	52.8982
Gauß add./exkl.	54.2802
Gauß skal./exkl.	52.1411

Tabelle 5.26: Mittlerer *eg* Wert der 100 Simulationen des besten Netzes mit/ohne ver-räuschten Wetterdaten als Wettervorhersagedaten im Testset.

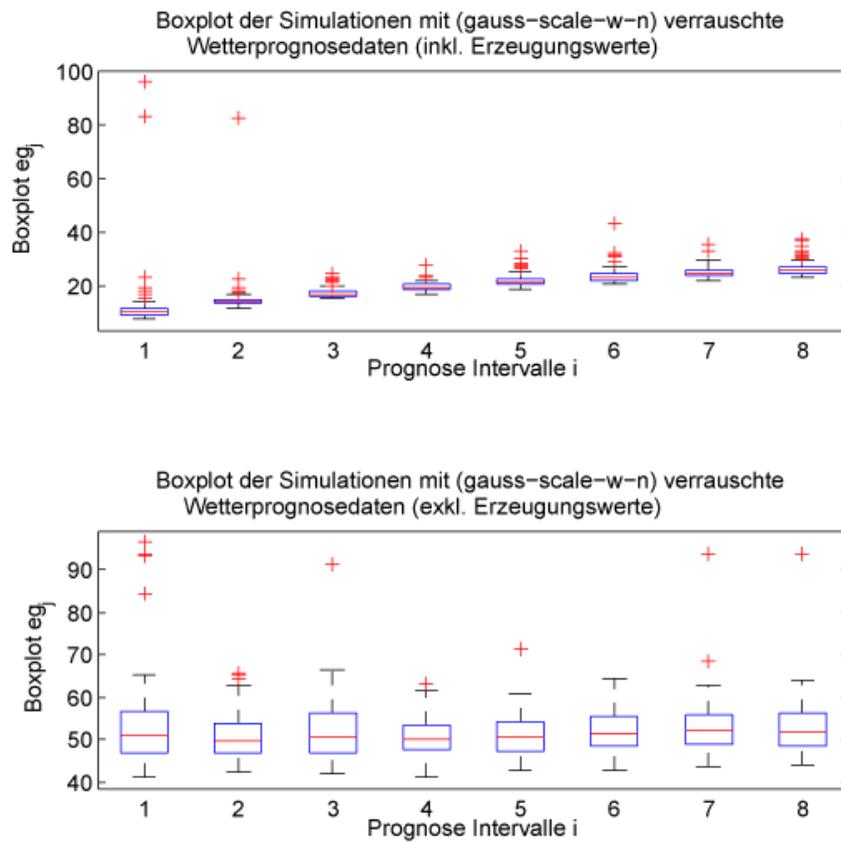


Abbildung 5.29: Boxplot der Simulationen mit den zufällig nach einem Gaußrauschen skalierten Wetterdaten für die Wettervorhersagedaten für den Testdatensatz.

WLKW	eg_j I1	eg_j I2	eg_j I3	eg_j I4	eg_j I5	eg_j I6	eg_j I7	eg_j I8
Radegund /inkl.	15.422	21.563	22.540	25.471	25.191	34.542	30.216	29.378
Kaningbach /inkl.	14.166	13.398	14.515	18.004	22.261	25.610	19.748	27.773
Winkl /inkl.	11.305	14.515	13.049	19.260	17.306	20.516	24.982	21.493
Radegund /exkl.	54.312	64.575	58.374	66.571	56.094	60.085	55.310	61.368
Kaningbach /exkl.	62.722	61.511	59.158	64.932	64.861	61.867	63.649	62.366
Winkl /exkl.	62.651	54.953	60.156	58.660	59.372	62.010	61.368	58.161

Tabelle 5.27: eg_j Werte der Simulationen weiterer Wasserlaufkraftwerke.

nicht exakt sind, jedoch ist der Unterschied im ersten Intervall sehr gering. Bei Gaußschen additiven Rauschen steigt der Fehler mit steigender Intervallanzahl stärker an, als im Falle des Rauschen mit der Gauß-Skalierung. Im achten Intervall ist vor allem bei additiven Gaußschen Rauschen ein deutlich höherer Fehler vorhanden. Das Modell ist aber recht robust gegenüber Rauschen auf den Wettervorhersagedaten bei der Prognose. Dies könnte daran liegen, dass das Modell bereits ausreichend Information aus den Werten der Vergangenheit erhält, um eine sinnvolle Prognose zu erstellen. Die Boxplots visualisieren die Lage der Fehlerwerte und weisen darauf hin, dass die meisten Fehlerwerte eng aneinander liegen, vor allem bei den Netzen mit der Information über die Leistungswerte.

5.6.2 Prognose von weiteren Wasserlaufkraftwerken

Leistungsverläufe und zugehörige Temperatur und Niederschlagszeitreihen von drei weiteren WLKW werden herausgenommen, und darauf wird das mit den Datensätzen zum Kraftwerk *Kirchbach* gefundene beste feedforward neuronale Netz trainiert und simuliert. Für die Wettervorhersagedaten wurden die exakten Wetterdaten verwendet, um eine Vergleichbarkeit mit dem WLKW *Kirchbach* zu schaffen. Herausgenommen werden die Kraftwerke

- *Radegund*,
- *Kaningbach* und
- *Winkl*,

wobei nur die Zeitreihen 2008 und 2009 zum Trainieren verwendet werden, da in der Zeitreihe 2007 bei diesen und vielen anderen WLKW eine Menge Datenfehler und Lücken aufgetreten sind. Abbildungen 5.30, 5.31 und 5.32 zeigen dazu jeweils den Vergleich der Simulation des ersten Intervalls, mit den Testzeitreihen des Jahres 2010 der genannten Kraftwerke. Zusätzlich ist die zugehörige Niederschlags- und Temperaturzeitreihe dargestellt. Tabelle 5.27 gibt dabei die eg_j Werte der Simulationen an, Tabelle 5.28 die eg Werte. Die Fehler sind hier jeweils etwas größer im Vergleich zum WLKW *Kirchbach*. Dies hat einerseits den Grund, dass hier nur die Zeitreihen 2008 und 2009 zum Trainieren verwendet wurden. Beim WLKW *Kirchbach* wurde eine Jahreszeitreihe mehr zum Trainieren verwendet. Andererseits ist das Netz, das hier verwendet wird, das Beste gefundene für *Kirchbach*. Es ist nicht sinnvoll eine so umfangreiche Variation der Parameter wie in

WLKW	<i>eg</i>
Radegund /inkl.	25.5408
Kaningbach /inkl.	19.4348
Winkl /inkl.	17.8036
Radegund /exkl.	59.5866
Kaningbach /exkl.	62.6336
Winkl /exkl.	59.6668

Tabelle 5.28: *eg* Wert der Simulationen weiterer Wasserlaufkraftwerke.

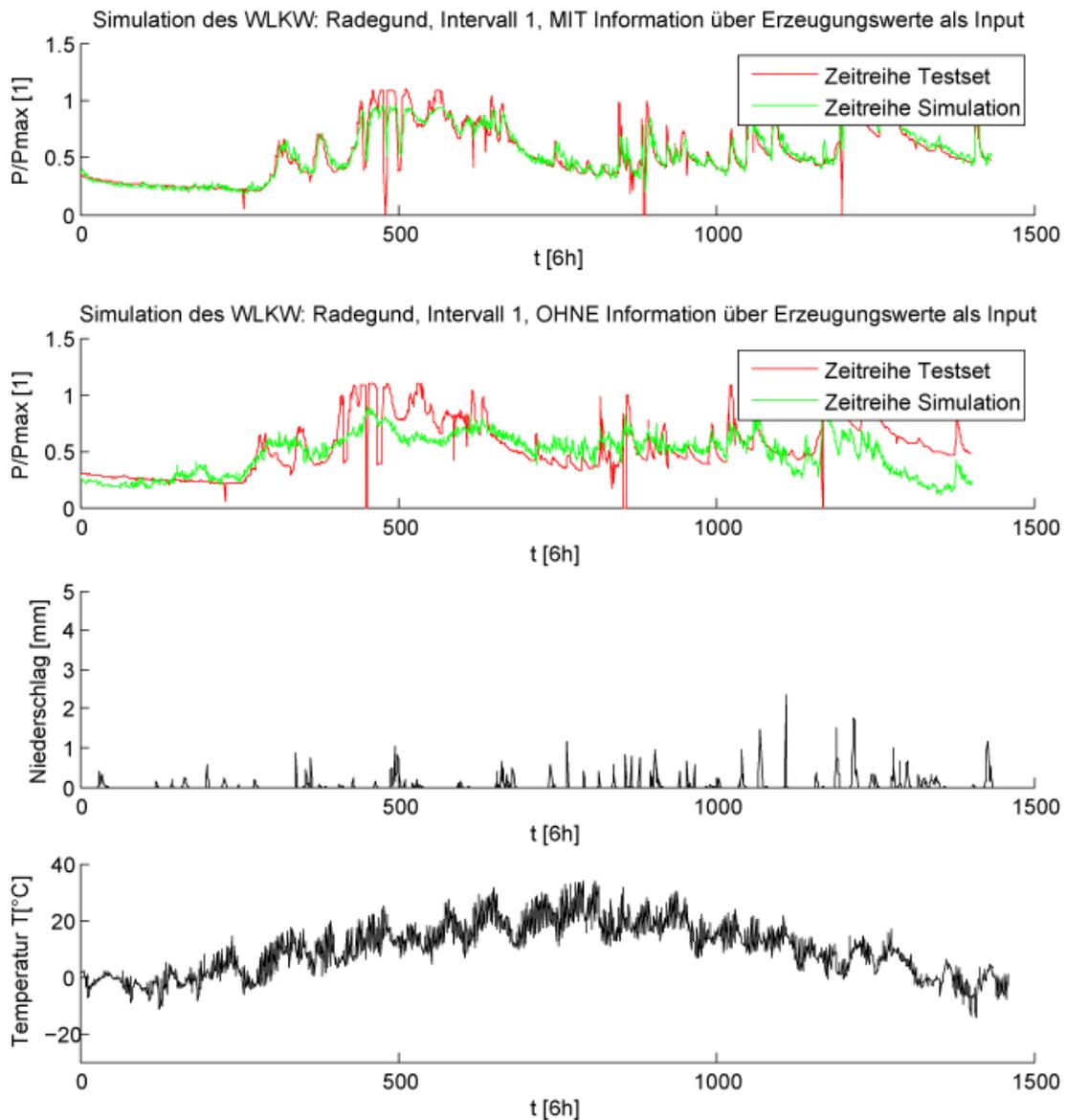
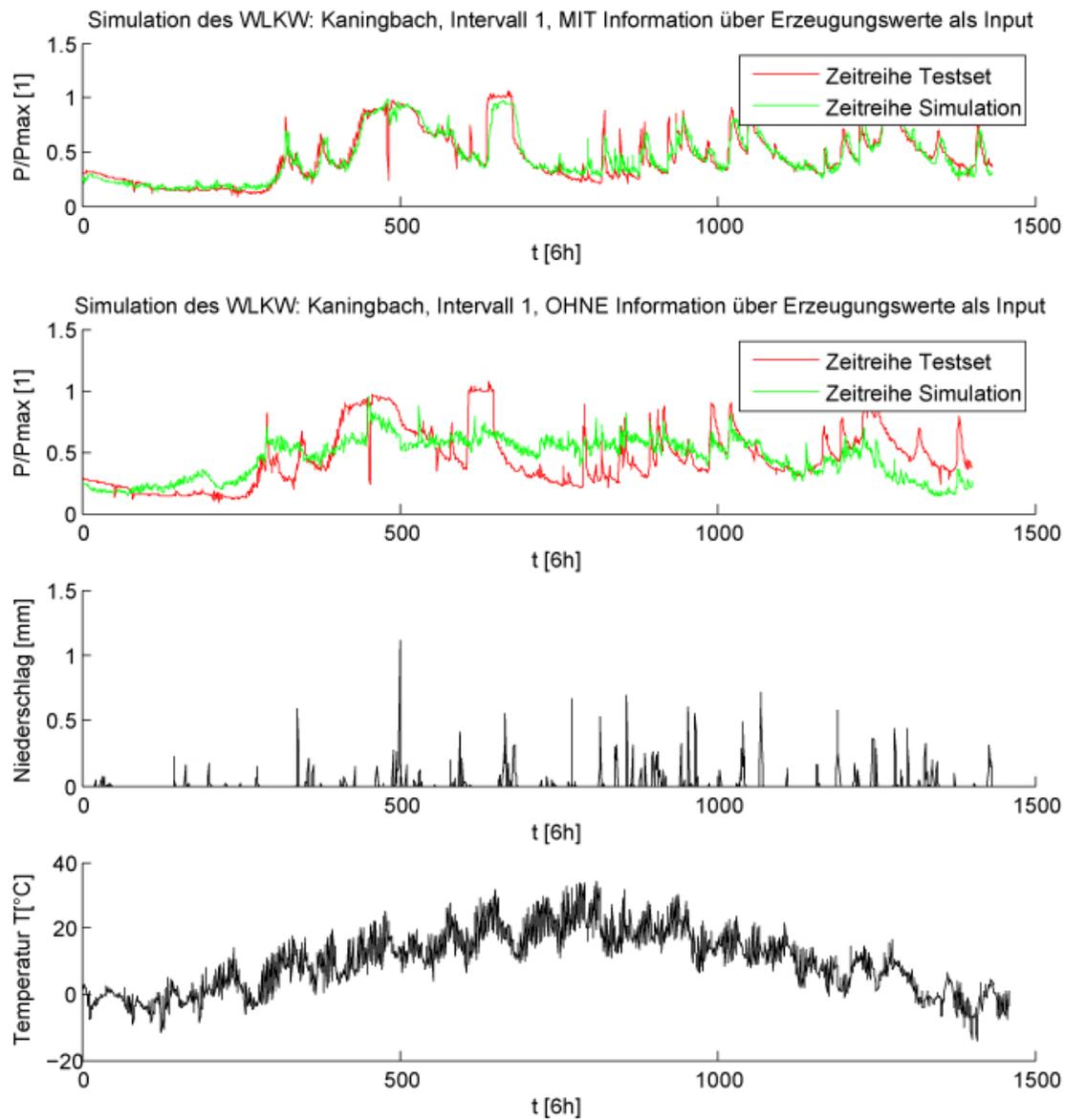
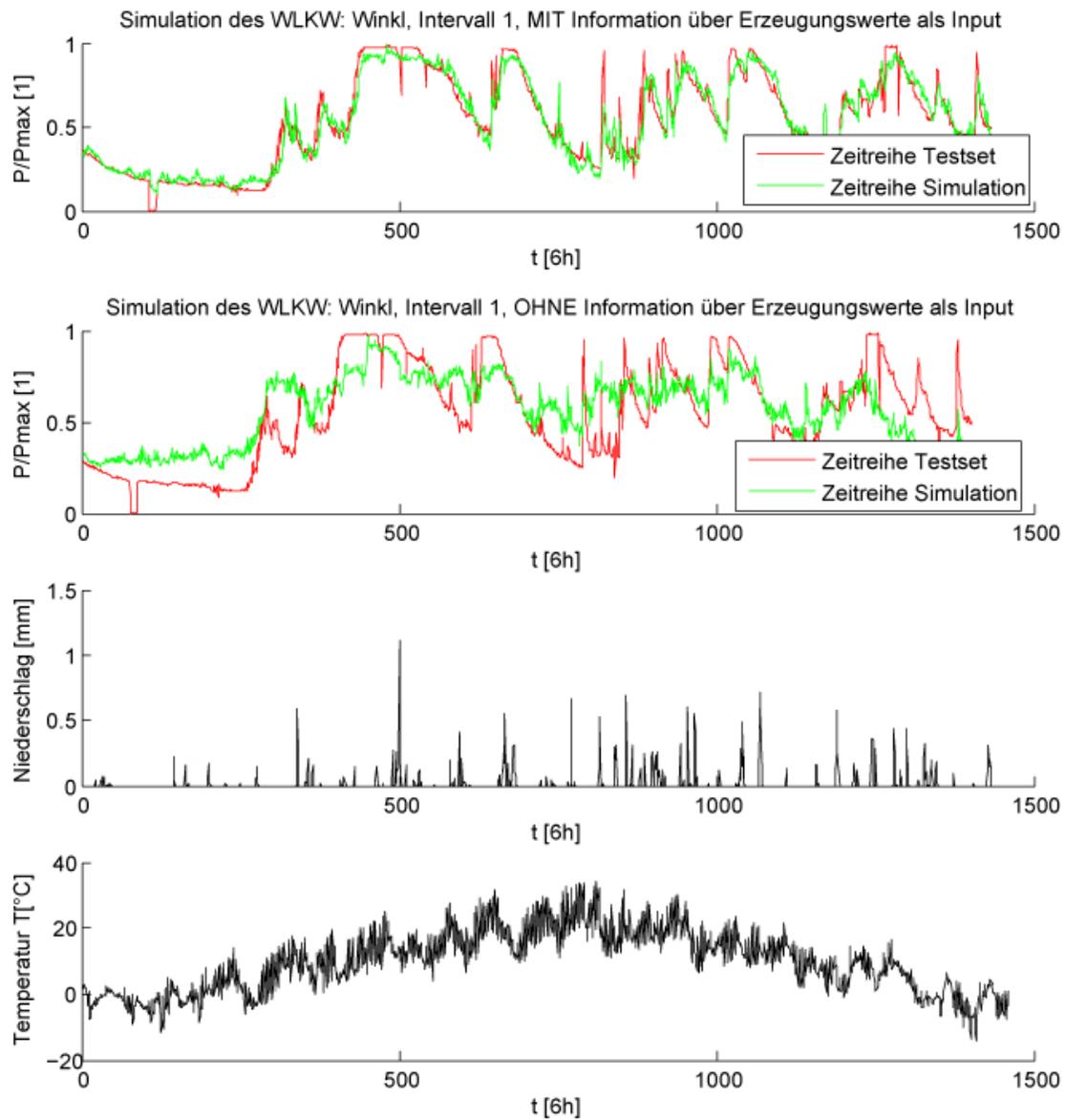


Abbildung 5.30: Simulation der Prognose des Kraftwerks *Radegund*.

Abbildung 5.31: Simulation der Prognose des Kraftwerks *Kaningbach*.

Abbildung 5.32: Simulation der Prognose des Kraftwerks *Winkl*.

Kapitel 5.2.3 und 5.2.4 für jedes Kraftwerk zu machen, aber eine leichte Variation der Parameter, wie Anzahl der Hiddenlayer und Neuronen im Hiddenlayer, sowie Länge des Vergangenheitsfensters ist je Kraftwerk schon sinnvoll, da jedes womöglich eine deutlich andere Streckencharakteristik besitzt. Bei vielen WLKW mag es sein, dass die Prognose nicht besser möglich ist. Außerdem sind die zugehörigen Zeitreihen vom Niederschlag oft aus einer anderen Talschaft oder Region, und bringen auf diese Art große Fehler bereits in das Training vom Modell mit hinein. Auch das Problem der örtlich begrenzten Gewitterschauer, wie bereits im Kapitel 5.2.4 kurz diskutiert, tritt dabei verstärkt auf.

5.7 Zusammenfassung

Aufgrund des in Tabelle 5.23 angeführten Vergleichs des globalen Fehlermaßes eg der unterschiedlichen regressionsbasierten Ansätze und des Vergleichs der in Tabelle 5.24 angeführten Trainingszeiten, sowie den Ausführungen im Kapitel 5.5 zur Auswahl des geeigneten Modellansatzes, ist der regressionsbasierte Ansatz über feedforward neuronale Netze zur Realisierung der kurzfristigen Leistungsprognose für WLKW vorzuziehen.

Die Untersuchungen mit verrauschten Wetterdaten zur Simulation exakter Wetterdaten im Kapitel 5.6 zeigen, dass der Ansatz über feedforward neuronale Netze sehr robust gegenüber Störeinflüsse durch reale Wetterprognosedaten ist. In Summe gesehen, kann der Ansatz mit feedforward neuronalen Netzen für WLKW verwendet werden, bei denen der aktuelle Leistungswert vorhanden bzw. nicht vorhanden ist. Dieser Ansatz eignet sich also für beide Varianten sehr gut. Vor allem für den Fall, bei dem das Modell blind bezüglich der aktuellen Leistung eines WLKW ist, wird trotz geringer Anzahl an Niederschlagsmessstationen eine gute Prognosegüte erzielt.

Kapitel 6

Physikalisches Modell

Auf Grund der vorhandenen Daten, siehe Kapitel 2.1, und den angeführten Grundlagen, siehe 4.1, kann ein Simulationsmodell wie in Abbildung 6.1 dargestellt ist, in Matlab erstellt werden. Auf der einen Seite gibt es die gemessene PVA, die als Referenzanlage dienen soll. Mit dieser Anlage soll das Modell verifiziert werden. Auf der anderen Seite soll das verifizierte Modell auf die ungemessenen Anlagen angewendet werden. Das Modell ist so ausgelegt, dass man mit den vorhandenen Daten für ungemessenen Anlagen trotzdem eine sinnvolle Simulation machen kann.

Das Modell, wie es in 6.1 dargestellt ist, errechnet im ersten Schritt die extraterrestrische Solarstrahlung zu einem gegebenem Datum und der angegebenen Solarkonstante. Die Berechnung der extraterrestrischen Solarstrahlung berücksichtigt auch den Einfallswinkel der Sonne auf die Athmosphärenoberfläche. Die einfallende extraterrestrische Solarstrahlung gelangt durch die Athmosphäre, welche in diesem Modell nicht modelliert ist, zu einem Globalstrahlungsmessgerät (z.B. über ein Sternpyranometer). Sie wird von dem Teil der *Strahlungsberechnung* benötigt, um den Klarheitsfaktor zu bestimmen. Das Globalstrahlungsmessgerät liefert die einfallende Energie auf eine horizontale oder geneigte Fläche, je nachdem wie der Sensor ausgerichtet ist. Der Teil der *Strahlungsberechnung* ermittelt die Solarstrahlungsenergie auf die geneigte Fläche der Photovoltaikpaneele, unter Berücksichtigung der Neigung und Drehung der Paneele. Hierbei gilt, dass

- $\alpha_{PV} = \alpha_E$, und
- $\gamma_{PV} = \gamma_E$

ist, um mit den Grundlagen, Kapitel 4.1.4, evident zu bleiben. Dieser Simulationsteil benötigt auch die Kenntnis über den Sonnenstand, welcher in einem eigenen Modul auf Grund des Datums und der Uhrzeit berechnet wird, und hat als weitere Eingangswerte den Wert der Solarstrahlung auf eine geneigte oder horizontale Fläche. Wird die Solarstrahlung in der geneigten und gedrehten Fläche gemessen, so ist der Output der *Strahlungsberechnung* gleich der Input der Globalstrahlung. Wird die Globalstrahlung über eine Horizontale gemessen, so wird die Umrechnung über ein statistisches Modell, wie in Kapitel 4.1 beschrieben, angewendet. Der Teil der Sonnenstandsrechnung muss auch die Möglichkeit eines Schaltjahres und der Sommerzeitverschiebung berücksichtigen können.

Der Teil der *Photovoltaik Energieumwandlung* simuliert die Energieumwandlung durch die

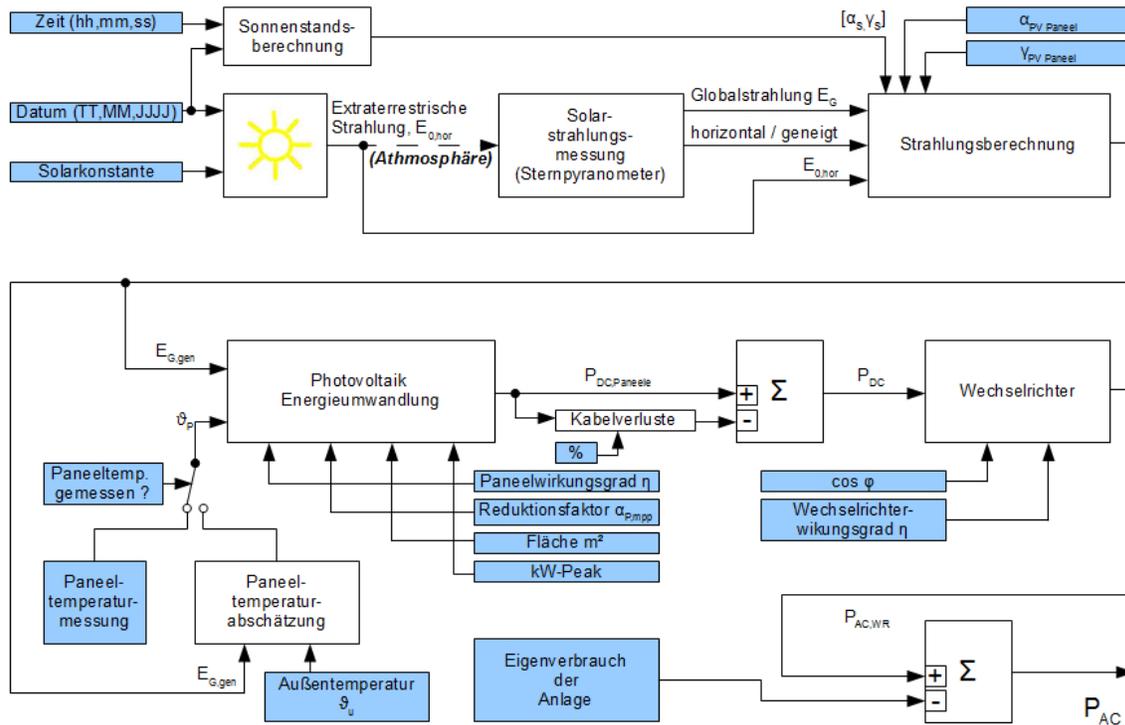


Abbildung 6.1: Simulationsmodell für die Simulationen der PV Anlagen.

Solarmodule. Die Paneeitemperatur kann dabei einerseits ein Messwert sein, sofern vorhanden, andererseits kann die Paneeitemperatur auch aus der Umgebungstemperatur und der Globalstrahlung auf die Paneele geschätzt werden. Weitere Parameter für die *Photovoltaik Energieumwandlung*, wie der Temperaturreduktionsfaktor und einerseits die Fläche und der Wirkungsgrad oder andererseits die kW-Peak Leistung, sind für die Berechnung der umgewandelten Energie notwendig.

Es folgt danach die Berechnung des Energieverlustes auf Grund der Kabel und Kabellängen. Dieser Wert ist in der Regel kleiner 1%, und wird hier für die Simulation mit 1% angenommen. Sollten höhere Kabelverluste bekannt sein, kann dies berücksichtigt werden.

Danach folgt der Teil der Simulation des Wechselrichters. Hierbei kann der Wirkleistungsfaktor $\cos \varphi$ und der Wechselrichterwirkungsgrad η angegeben werden. Für die Simulation wird in der Regel der Euro-Wirkungsgrad η_{EU} verwendet. Die Berücksichtigung der Auslastung des Wechselrichters kann nicht erfolgen, da keine gesonderte Messung der Leistung auf der DC Seite, P_{DC} , erfolgt.

Im letzten Simulationsteil wird der Eigenverbrauch berücksichtigt. Viele Wechselrichterschranke enthalten für die korrekte Umgebungstemperatur der Elektronik ein Heizelement. Dieses Heizelement hat einen Verbrauch von wenigen W . Wollte man diesen und andere diverse Eigenverbräuche berücksichtigen, so können in Form einer Aufzählung die einzelnen Verlustleistungen angegeben werden.

Das Endergebnis der Berechnung, stellt einen abgeschätzten Wert der Leistung einer PVA zu einem Zeitpunkt t dar. In der Simulation wird der AM-Wert (Luftmassenindex) auf Grund des Sonnenstandes nicht berücksichtigt. Ebenfalls werden weitere Effekte der Ath-

mosphäre nicht modelliert. Einerseits würde die Berücksichtigung dieser beiden Teile diese Arbeit überladen, andererseits hätte man dadurch kaum einen Qualitätsgewinn, da die Solarstrahlungsenergie auf der Erdoberfläche gemessen wird. Auch die Effekte von Abschattungsverluste werden in dieser Arbeit nicht berücksichtigt, da dies ebenfalls zu einer Überladung der Arbeit führen würde, und andererseits keine Messdaten und Bestandsaufnahmen der Anlagen vorhanden sind. Folgende Abschnitte umfassen dieses Kapitel:

- Evaluierung der Solarstrahlungsumrechnung von der horizontalen Fläche auf eine geneigte und gedrehte Fläche, über die Messdaten der nächstgelegenen Globalstrahlungsmessstation, welche die Solarstrahlung auf eine Horizontale misst, und der Messdaten der Globalstrahlung auf die geneigte und gedrehte Fläche der gemessenen PVA *St. Veit*.
- Evaluierung der Temperaturabschätzung der Solarpaneele, über die Messdaten der Umgebungstemperatur und der gemessenen Paneeltemperatur der PVA *St. Veit*.
- Verifizierung der Simulationskette mit Daten der gemessenen PVA *St. Veit*. Einerseits mit den gemessenen Daten der Paneeltemperatur und Globalstrahlung auf die geneigte und gedrehte Fläche, andererseits mit der Paneeltemperaturabschätzung und der Globalstrahlung auf die Horizontale.
- Anwendung des Modells für die Berechnung von Jahressummenwerten der Energieumwandlung von ungemessenen Anlagen, mit anschließendem Vergleich mit abgelesenen Jahressummenwerten von ungemessenen Anlagen.

Alle verwendeten Datenblätter technischer Einrichtungen, die für die Simulationen dieses Kapitels notwendig sind, sind im Anhang D zu finden.

6.1 Evaluierung der Berechnung der Solarstrahlung auf geneigte und gedrehte Flächen

Für Europa sind hauptsächlich das Modell nach *Erbs, Klein und Duffie*, siehe [Sch91], sowie jenes nach *Reindl, Beckman und Duffie*, siehe [Qua96] zur Abschätzung der diffusen Solarstrahlung bedeutend, da dafür unter anderem Messdaten von Messstationen aus Europa verwendet wurden. *Schumacher*, [Sch91], zeigt in Abbildung 6.2 den Diffusstrahlungsanteil in Abhängigkeit des Klarheitsfaktors k_T mit verschiedensten Korrelationen. Die Messdaten stammen von einer Messstation in Berlin aus dem Jahr 1960, und stellen Stundenmittelwerte dar. Die durchgezogene Linie zeigt die Korrelation nach *Erbs, Klein und Duffie*, wie sie in Kapitel 4.1.3 angeführt ist. Zur Evaluierung der Berechnung der Solarstrahlung auf geneigte Flächen, wird nun in Matlab ein Skript implementiert, das nach der Theorie von 4.1.1, 4.1.2, 4.1.3 und 4.1.4 die Umrechnung der gemessenen Solarstrahlung einer horizontalen Fläche auf eine geneigte und gedrehte Fläche umsetzt. Für die Anlage in *St. Veit* ist eine Solarstrahlungsmessung in der Paneelebene gegeben. Als Solarstrahlung auf eine horizontale Fläche werden die Messdaten der nächstgelegenen Messstation in *Klagenfurt* verwendet. Bei der Berechnung der diffusen Solarstrahlung sollen die zwei erwähnten Modelle verwendet werden. Es kann mit den Messwerten der Solarstrahlung in der Ebene ein direkter Vergleich mit den umgerechneten Werten gemacht werden. Notwendige Daten für die Simulation lauten wie folgt:

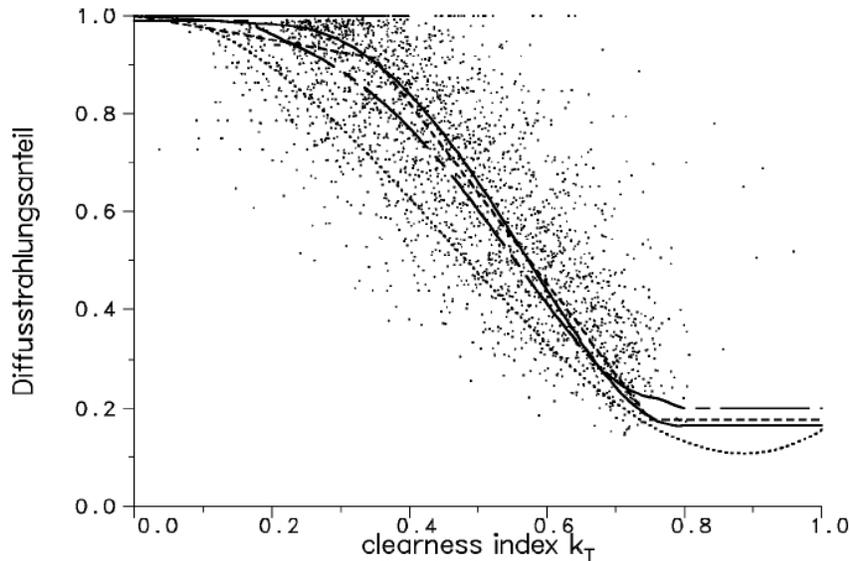


Abbildung 6.2: Diffusstrahlungsanteil in Abhängigkeit des Klarheitsfaktors einer Messstation in Berlin für das Jahr 1960, Stundenmittelwerte, [Sch91, Fig.35].

- $geo_laenge = 14.36$; Geographische Länge,
- $geo_breite = 46.76$; Geographische Breite,
- $zeitzone = 1$; Angabe über die Zeitzone,
- $start_is_summertime = 1$; Beginnt die Zeitreihe mit der Sommerzeit, 1=ja, 0=nein,
- $alpha_E_pv = -14$; Drehung der Paneelflächen / Abweichung von Süden,
- $gamma_E_pv = 33$; Neigung der Paneelfläche aus der Horizontalen,
- $refl_Faktor_A = 0.2$; Albedo-Konstante,
- Die Zeitreihen umfassen $\frac{1}{4}h$ Werte vom 03.05.2011, 00:15 Uhr, bis 24.05.2011, 00:00 Uhr.

Längere Zeitreihen der Anlage *St. Veit* sind nicht vorhanden, da die Anlage in *St. Veit* nicht früher in Betrieb genommen wurde, und keine bzw. keine korrekten Messwerte vor dem 03.05.2011 vorhanden sind. Dieser Zeitraum bietet aber verschiedenste Wettersituationen, von sehr klar bis stark bewölkt und regnerisch. Abbildung 6.3 zeigt einen Ausschnitt des Verlaufes der berechneten Solarstrahlung auf die geneigte und gedrehte Paneelfläche und die tatsächlich gemessene Solarstrahlung in der Paneelebene. Für das Modell nach *Erbs, Klein und Duffie*, siehe [Sch91], wird für die Solarstrahlung auf die geneigte und gedrehte Fläche ein mittlerer absoluter Error/Fehler (MAE) von $61.37 \frac{W}{m^2}$ und eine durchschnittliche Abweichung, bezogen auf den Maximalwert der gemessenen Solarstrahlung in der Paneelebene, mit 5.12% für diesen Zeitraum errechnet. Und für das Modell nach *Reindl, Beckman und Duffie*, siehe [Qua96], wird für die Solarstrahlung auf die geneigte

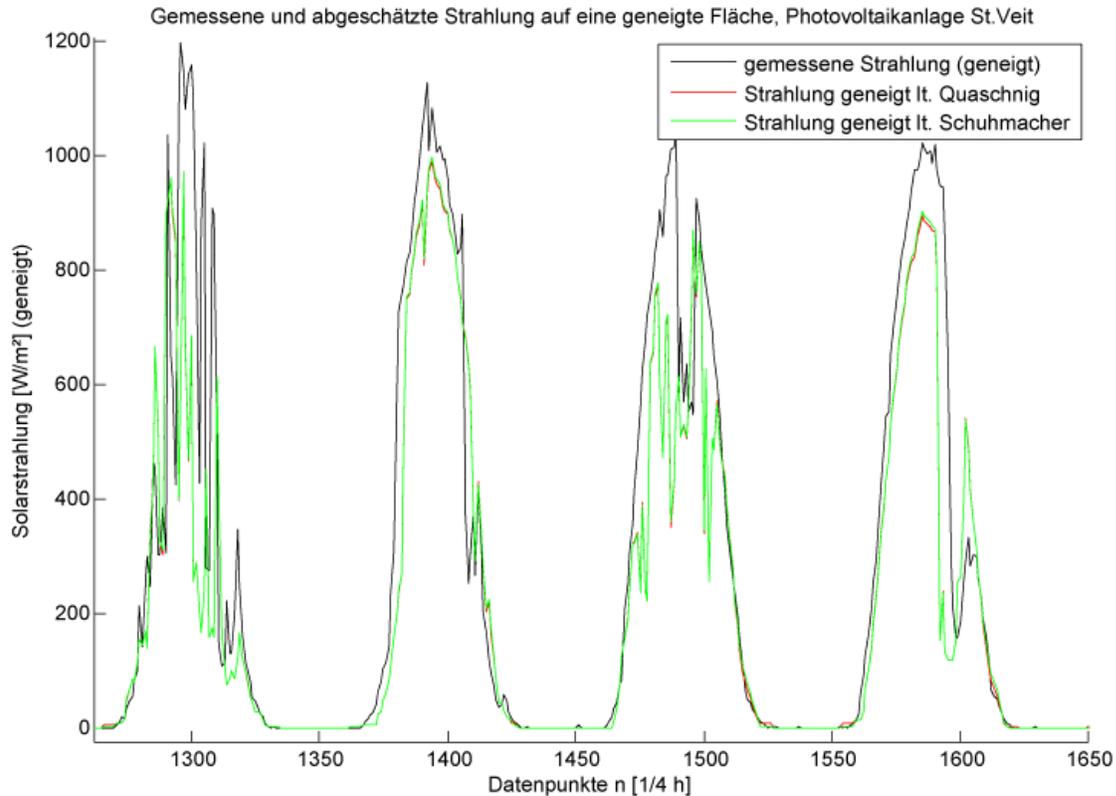


Abbildung 6.3: Ausschnitt der Verläufe der gemessenen Solarstrahlung in der Paneelebene der Anlage *St. Veit* und der abgeschätzten Solarstrahlung auf die geneigte und gedrehte Paneelebene, auf Basis der Umrechnungsmodelle sowie der Messwerte horizontal gemessener Solarstrahlung der nächstgelegenen Messstation in *Klagenfurt*.

und gedrehte Fläche ein mittlerer absoluter Fehler (MAE) von $62.03 \frac{W}{m^2}$ und eine durchschnittliche Abweichung, bezogen auf den Maximalwert der gemessenen Solarstrahlung in der Paneelebene, mit 5.18% für den angegebenen Zeitraum errechnet. Das Modell nach *Erbs, Klein und Duffie* schneidet für diesen Zeitraum geringfügig besser ab. Es ist bei den Ergebnissen zu beachten, dass die Messung der Solarstrahlung auf eine Horizontale einige *km* vom Standort *St. Veit* entfernt ist, und in *St. Veit* sicher öfters eine andere Bewölkungssituation vorherrscht, als am Standort *Klagenfurt*. Dieser Fehler kann aber auch bei einer Entfernung von wenigen 100m auftreten, jenachdem welche Bewölkungssituation vorliegt. Es ist an bewölkten Tagen, siehe Abbildung 6.3, dieser Fehler deutlich zu erkennen. Die starken *Ausreißer* in der berechneten Strahlungsleistung auf die geneigte und gedrehte Fläche, sind auf die Basis für deren Berechnung, die gemessene Horizontalstrahlung einige *km* entfernt, zurückzuführen. Außerdem wird nur gut ein Monat eines Jahres für die Evaluierung verwendet, da nicht mehr Daten vorhanden sind. Der Vergleich des Verlaufes eines ganzen Jahres wäre sicher aussagekräftiger.

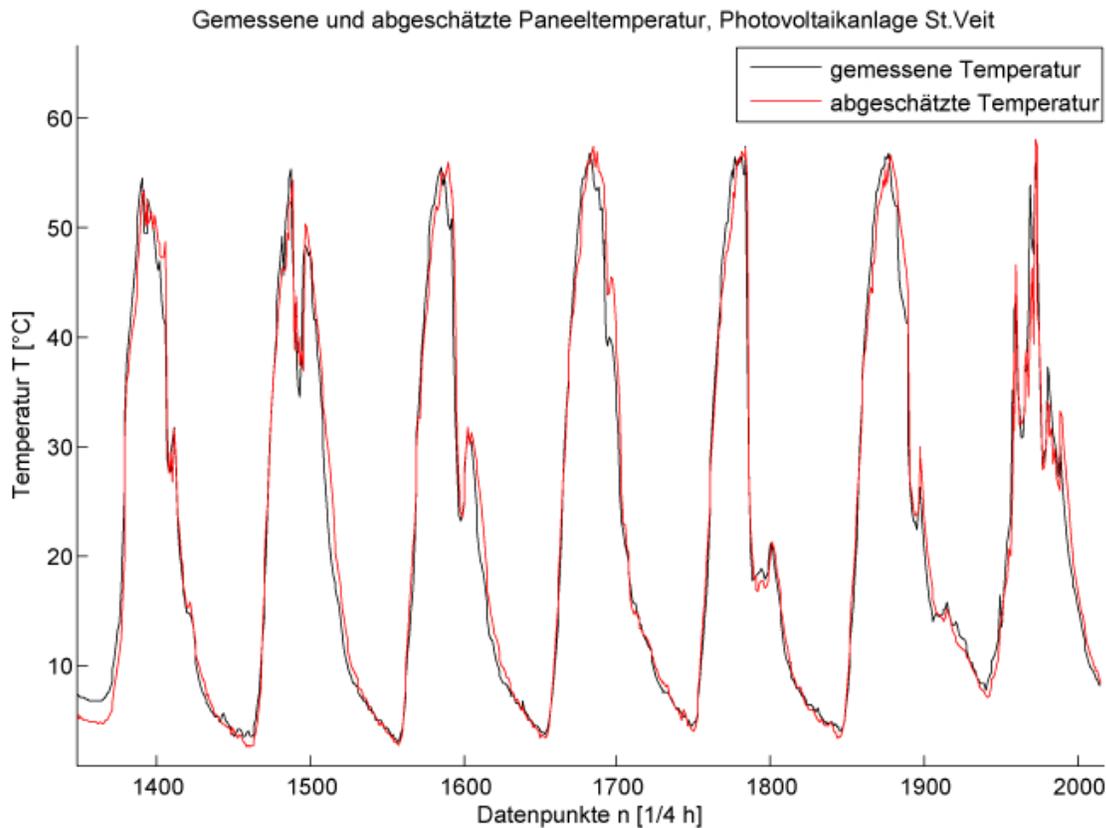


Abbildung 6.4: Ausschnitt der Verläufe der gemessenen Paneeltemperatur der Anlage *St. Veit* und der abgeschätzten Paneeltemperatur, auf Basis des Umrechnungsmodells sowie der Messwerte der Umgebungstemperatur und einfallenden Solarstrahlung auf die geneigte und gedrehte Paneelfläche in *St. Veit*.

6.2 Evaluierung der Temperaturabschätzung von Paneelen

Für die Evaluierung der Güte des Modells der Temperaturabschätzung, siehe 4.1.5, werden die Zeitreihen der Umgebungstemperatur und der gemessenen Paneeltemperatur, sowie die gemessene Solarstrahlung in der Paneelebene von der PVA in *St. Veit* verwendet. Über die Umgebungstemperatur und die Solarstrahlung auf die geneigte Fläche, wird eine Paneeltemperatur abgeschätzt. Die Abschätzung erfolgt mit einem Matlab Skript, und wird der gemessenen Paneeltemperatur gegenübergestellt. Abbildung 6.4 zeigt einen Ausschnitt der gemessenen Paneeltemperatur und der abgeschätzten Paneeltemperatur. Die Simulationsparameter sind dieselben wie in 6.1. Es wird mit diesem Modell ein mittlerer absoluter Fehler (MAE) von 1.98°C und eine durchschnittliche Abweichung, bezogen auf den Maximalwert der gemessenen Paneeltemperatur, mit 3.29% für den angegebenen Zeitraum errechnet. Auch hier gilt, dass die Evaluierung interessanter und sicherlich aussagekräftiger wäre, hätte man die Möglichkeit einen ganzen Jahresverlauf zu testen.

6.3 Verifizierung des Simulationsmodells

Aus den vorhergehenden Kapiteln 6.1 und 6.2 ist zu erkennen, dass die Solarstrahlungsabschätzung auf die geneigte und gedrehte Fläche, aus der Solarstrahlungsmessung auf eine horizontale Fläche, den größeren Fehler in der Abschätzung der Leistung einer PVA verursacht. Das Simulationsmodell, wie es in Abbildung 6.1 skizziert ist, soll nun auf vier Arten auf die selben Zeitreihen, wie bei den Simulationen in den Kapiteln 6.1 und 6.2, angewendet werden. Damit soll die Leistung der PVA abgeschätzt werden. Die Grundeinstellung der Simulationsparameter lautet:

- $geo_laenge = 14.36$,
- $geo_breite = 46.76$,
- $zeitzone = 1$,
- $start_is_summertime = 1$,
- $alpha_E_pv = -14$,
- $gamma_E_pv = 33$,
- $refl_Faktor_A = 0.2$,
- $wirkGr_PV = 0.1405$; Ist der Wirkungsgrad lt. Datenblatt der PVA *St. Veit*,
- $qm = 231.66$; Fläche der Paneele,
- $kabelverl = 0.01$; Maximaler Wert lt. Hersteller,
- $alpha_Pmpp = -0.00405$; Temperaturreduktionsfaktor lt. Datenblatt der PVA *St. Veit*,
- $EU_wirkGr_WR = 0.977$; Euro-Wirkungsgrad lt. Datenblatt des Wechselrichters der PVA *St. Veit*,
- $cos_Phi_WR = 0.47$. Aufgrund eines, für den Zeitraum der hier verwendeten Zeitreihen, für die PVA in *St. Veit* vorhandenen systematischen Messfehlers der Messumwandler, muss dieser Faktor konstant angenommen werden. Dies wurde in einer Messung vor Ort bestätigt, wird aber hier nicht weiter behandelt und soll so als gegeben gelten. Anmerkung: Der systematische Messfehler wurde inzwischen behoben, es würde somit für aktuelle Messwerte der PVA *St. Veit* wieder der standardmäßig eingestellte $cos\varphi = 1$ gelten.

Die vier Simulationsarten lauten wie folgt:

- Simulation mit gemessener Paneeltemperatur und gemessener Solarstrahlung in der Paneelebene.
- Simulation mit gemessener Paneeltemperatur, aber unbekannter Solarstrahlung in der Paneelebene und Umrechnung von der horizontal gemessenen Solarstrahlung.
 - Modell für die diffuse Solarstrahlung ist das Modell nach *Erbs, Klein und Duffie*, siehe [Sch91].

ϑ_P gemessen	$E_{0,gen}$ gemessen	MAE [$\frac{W}{m^2}$]	\emptyset Abweichung von max. Referenz[%]
✓	✓	337.3	1.9233
✓	X	1085.2	6.1878
X	✓	325.5	1.8563
X	X	1033.5	5.8927

Tabelle 6.1: Mittlerer absoluter Fehler und durchschnittliche Abweichung der Leistungsabschätzung vom maximalen Referenzwert, in Abhängigkeit unterschiedlicher Gegebenheiten für das Simulationsmodell.

ϑ_P gemessen	$E_{0,gen}$ gemessen	eg -Fehlermaß
✓	✓	1.4881
✓	X	22.8175
X	✓	1.5873
X	X	17.2123

Tabelle 6.2: Fehlermaß eg in Abhängigkeit unterschiedlicher Gegebenheiten für das Simulationsmodell.

- Simulation mit unbekannter Paneeltemperatur, aber bekannter Solarstrahlung in der Paneelebene und bekannter Umgebungstemperatur.
- Weder die Paneeltemperatur, noch die Solarstrahlung auf der Paneelebene sind gemessen. Nur die Umgebungstemperatur in *St. Veit* und die Solarstrahlung auf die Horizontale, gemessen bei einer Messstation in *Klagenfurt*, ist bekannt.
 - Modell für die diffuse Solarstrahlung ist das Modell nach *Erbs, Klein und Duffie*, siehe [Sch91].

Die Tabelle 6.1 zeigt die Simulationsergebnisse der vier Simulationen. In dieser Tabelle sind die Ergebnisse für den mittleren absoluten Fehler (MAE) und der durchschnittlichen Abweichung vom Maximalwert der Referenzzeitreihe angeführt. Auch hier ist wie erwartet zu erkennen, dass die Abschätzung der Solarstrahlung auf die geneigte und gedrehte Fläche den größten Fehler mit sich bringt. Ein Ausschnitt vom Verlauf der Leistungsabschätzung bei nicht gemessener Paneeltemperatur und nicht gemessener Solarstrahlung in der Paneelebene, im Vergleich zum Verlauf der gemessenen Leistung, ist in Abbildung 6.5 dargestellt. Der eg Fehlerwert, siehe Kapitel 5.1, wie er bei den Simulationen beim regressionsbasierten Modell der WLKW, siehe Kapitel 5, als Qualitätsmaß mit einer Margin von 10% des Referenzwertes zum Zeitpunkt t ermittelt wurde, hat die Werte für die vier unterschiedlichen Simulationen, wie sie in Tabelle 6.2 angeführt sind. Die Qualität des Modells ist sehr zufriedenstellend. Das Qualitätsmaß könnte durchaus um einige Prozent verbessert werden, würde man eine Solarstrahlungsmessung auf die Horizontale direkt bei der Anlage in *St. Veit* haben. Dies gilt im Übrigen für alle PVA die betrachtet werden sollen. Ideal wäre die Messung der Solarstrahlung in der Paneelebene. Dies würde das Fehlermaß nahe gegen 1 senken, da kein zusätzlicher Fehler durch die Abschätzung der Solarstrahlung auf die geneigte Fläche hinzukommt. Der Fehler durch die Temperaturabschätzung, sowie weitere systematische Fehler durch nicht behandelte Effekte, würden dann den Fehler

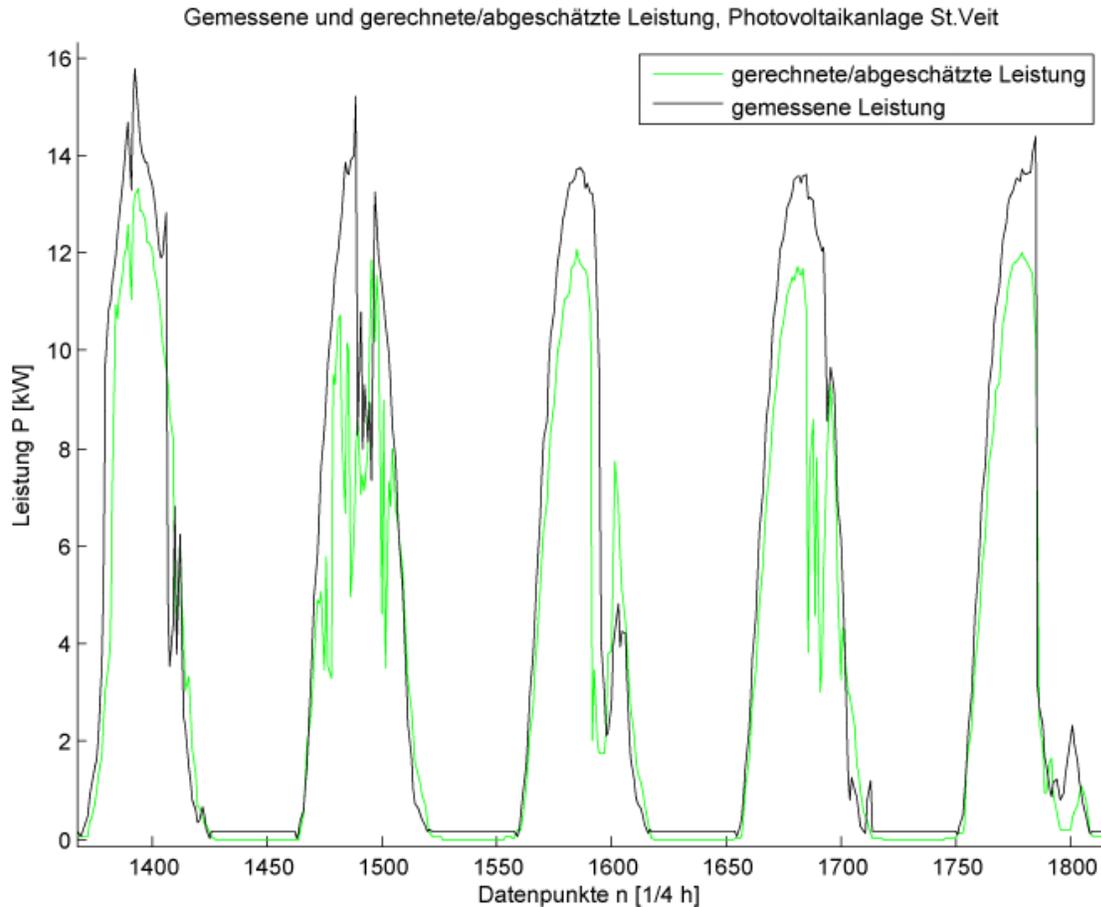


Abbildung 6.5: Ausschnitt der Verläufe der gemessenen Leistung der PVA *St.Veit* und der abgeschätzten Leistung, bei nicht gemessener Paneeltemperatur und nicht gemessener Solarstrahlung in der Paneelebene.

der Leistungsabschätzung ausmachen. Diese Fehler wären in Summe jedoch sehr gering. Abbildung 6.6 zeigt das Streudiagramm der Simulation mit ungemessener Paneeltemperatur und ungemessener Solarstrahlung in der Paneelebene, wobei die Leistungswerte auf den maximalen Leistungswert der Referenzreihe normiert wurden, und zwischen 0 und 1 liegen.

6.4 Anwendung vom Simulationsmodell auf ungemessene Photovoltaikanlagen

Von den ungemessenen PVA sind lediglich folgende Daten bekannt:

- Die $W - Peak$ Leistung der Anlage in kW .
- Die Jahressummen der umgewandelten Energie in kWh .
- Besitzer und Standort der Anlage.

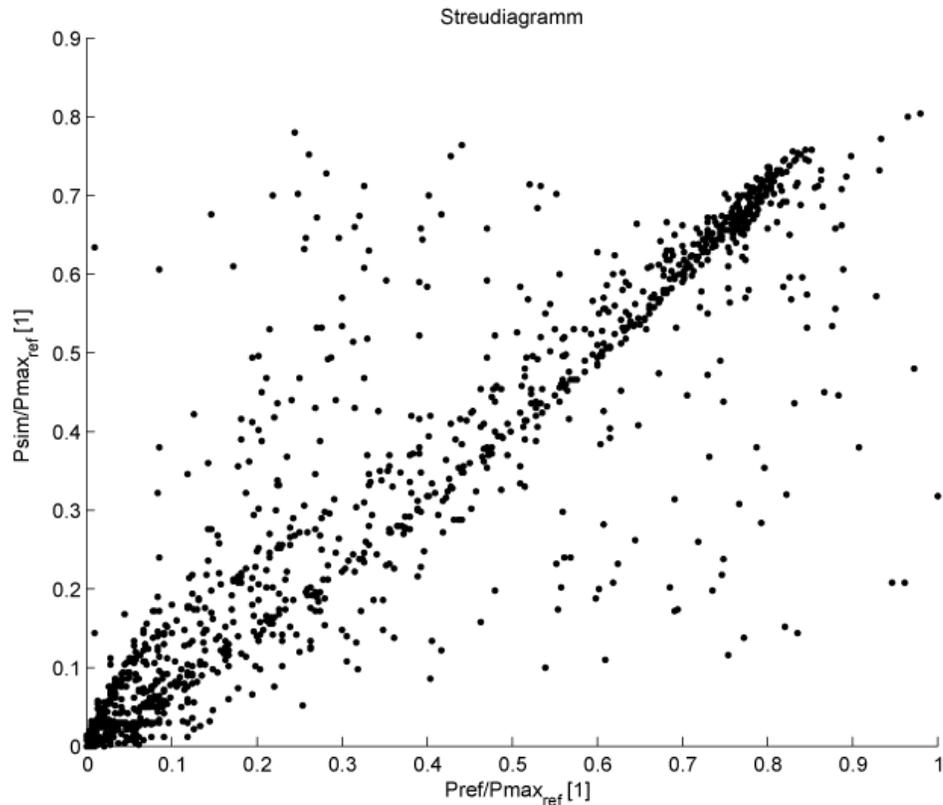


Abbildung 6.6: Streudiagramm der Simulation mit ungemessener Paneeltemperatur und ungemessener Solarstrahlung in der Paneelebene.

- Messdaten über Solarstrahlung und Umgebungstemperatur von den nächstgelegenen Messstationen.

Es sind also weder Daten über die technische Ausstattung, Ausrichtung und Neigung, oder andere für das Simulationsmodell notwendigen Daten vorhanden, wie Abbildung 6.7 veranschaulicht. Die nicht vorhandenen Daten müssen sinnvoll geschätzt werden:

- Die $kW - Peak$ Leistung beinhalten implizit die Information über die Fläche (m^2). Dieser Wert gibt an, welche Leistung bei STC von der gesamten Anlage abgegeben wird, nicht pro Flächeneinheit. Auch der Wirkungsgrad η der Paneele ist in dieser Größe implizit enthalten.
- $\cos\varphi$ und der Wirkungsgrad η des Wechselrichters sind nicht bekannt, und können mit durchschnittlichen Werten ersetzt werden, unter anderem mit dem Euro-Wirkungsgrad η_{EU} . Ein typischer Euro-Wirkungsgrad liegt im Bereich von 95%. Der $\cos\varphi$ kann bei korrekter Funktion mit 1 angenommen werden.
- Die Kabelverluste können, wie bereits bei der Simulation der PVA *St. Veit*, mit 1% angenommen werden.
- Die Zeit und das Datum sind wie bei der PVA *St. Veit* Laufvariablen, aus denen die Solarkonstante und der Sonnenstand berechnet werden können.

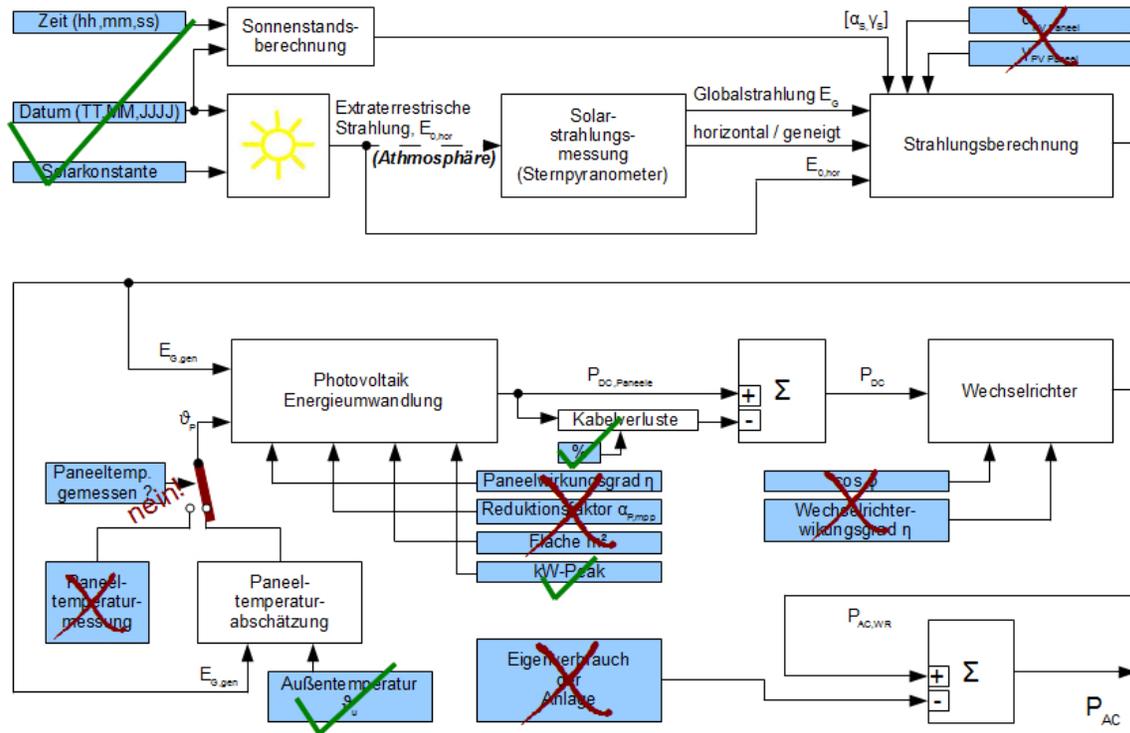


Abbildung 6.7: Skizze der Datensituation für die Simulation ungemessener Anlagen.

✓ ... vorhandene Parameter

X ... nicht vorhandene bzw. abzuschätzende Parameter

- Auch diese Simulation muss die Solarstrahlung auf die Paneelebene und die Paneelemp. abschätzen, wie es in den Kapiteln 6.1 und 6.2 beschrieben wurde.
- Aber die zwei wichtigsten Größen für die Abschätzung der Solarstrahlung auf die Paneelebene,
 - Ausrichtung der Paneele im Bezug auf Süden, α_{PV} ,
 - und die Neigung der Paneele, γ_{PV} ,

sind nicht bekannt.

α_{PV} und γ_{PV} müssen daher je ungemessene PVA passend gewählt, bzw. *gesucht* werden. Um einen Weg zur Abschätzung zu skizzieren, werden einige beispielhafte Anlagen aus der Datenbank aller ungemessenen PVA herausgenommen. Die Jahressummenwerte der umgewandelten Energie, die kW-Peak Leistungswerte und die Standorte sind in Tabelle 6.3 angeführt. Die Anlagen liegen alle in der Nähe von *Wolfsberg* in Kärnten/Österreich. Dort existiert auch eine Solarstrahlungsmessung über ein horizontal angebrachtes Sternpyranometer, sowie eine Umgebungstemperaturmessung. Die Jahresverläufe der Solarstrahlung auf eine horizontale Fläche für den Standort *Wolfsberg*, für die Jahre 2007, 2008, 2009 und 2010, sind in Abbildung 6.8 angegeben. Mitte des Jahres 2010 gab es offensichtlich einen Fehler bei der Messung der Solarstrahlung, der sich bei der Abschätzung der Jahressummen der umgewandelten Energie einer PVA auswirken wird, soll und kann hier aber nicht

Nr.	kW-Peak [kW]	$\Sigma_{2007}[kWh]$	$\Sigma_{2008}[kWh]$	$\Sigma_{2009}[kWh]$	$\Sigma_{2010}[kWh]$
660867	19.8	28381.7	27392.4	27342.8	25332.5
353243	49.5	70338.0	64038.0	65556.0	60296.0
246959	7.74	7951.0	7350.0	7478.0	7135.0
130425	3.2	3312.7	3057.0	3130.3	2726.8
143344	3.0	2619.1	2495.1	2538.1	2246.5

Tabelle 6.3: Daten beispielhafter ungemessener Photovoltaikanlagen.

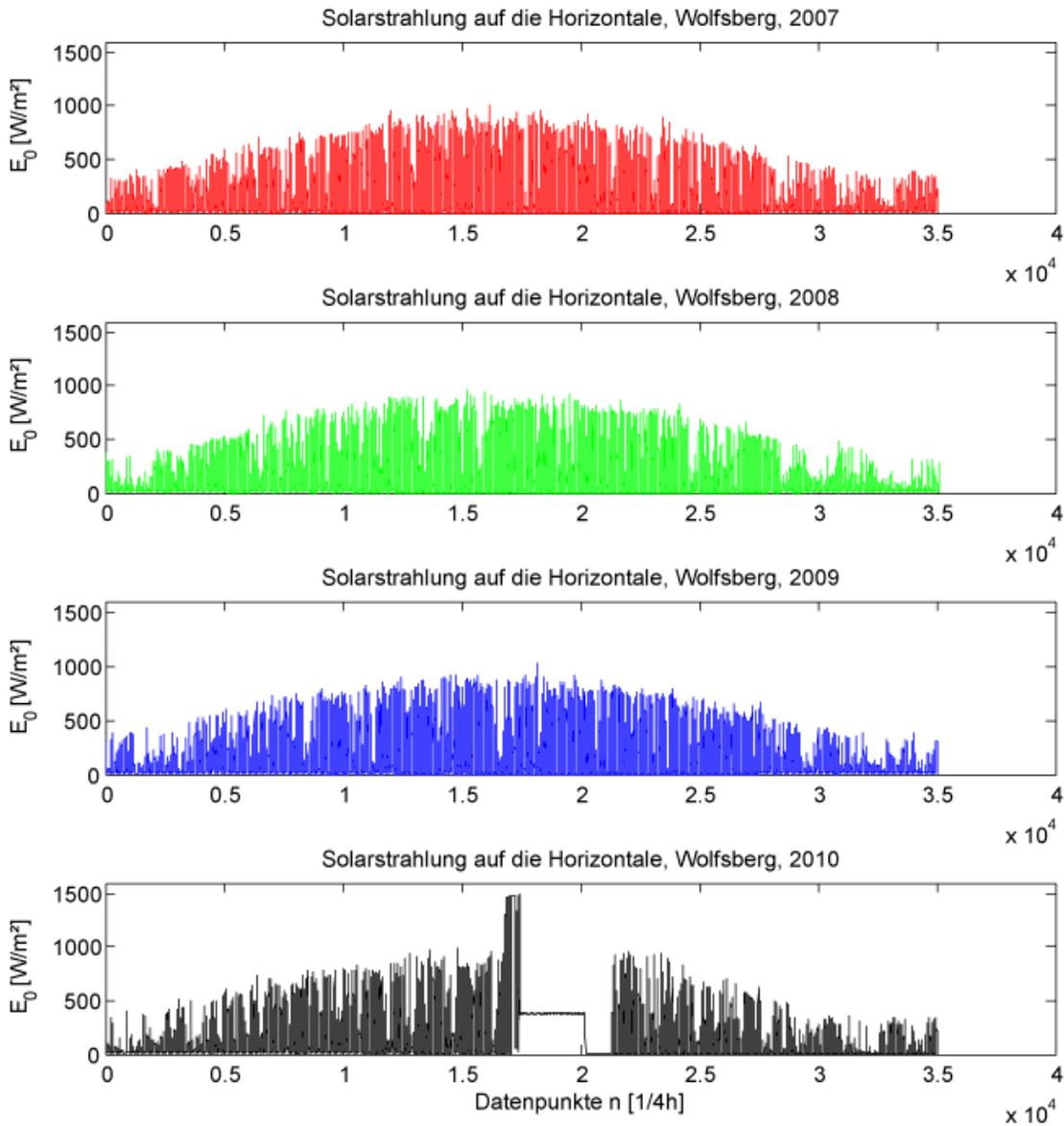


Abbildung 6.8: Jahresverläufe der Solarstrahlung auf eine horizontale Fläche für *Wolfsberg*. Jahre: 2007, 2008, 2009 und 2010.

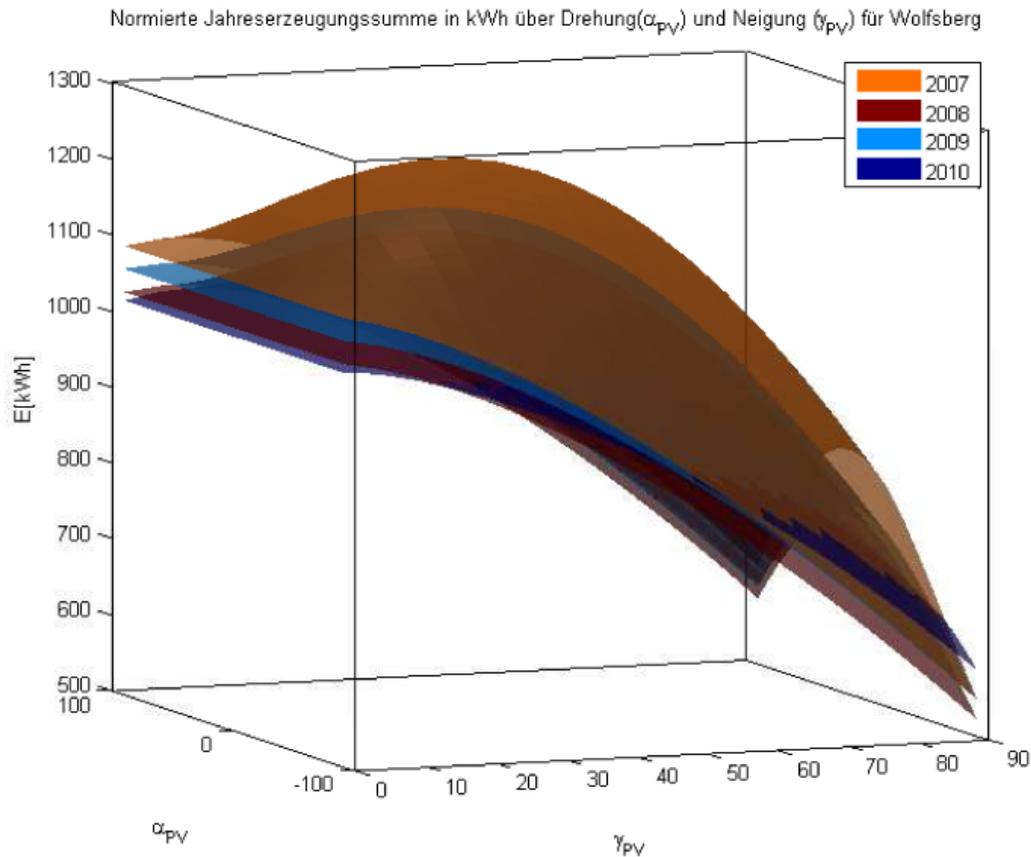


Abbildung 6.9: Jahressummen über α_{PV} und γ_{PV} einer $1kW - Peak$ Anlage am Standort *Wolfsberg*, für die Jahre 2007, 2008, 2009 und 2010.

weiter beachtet werden.

Zunächst soll ein normierter Verlauf der Jahressummen der umgewandelten Energie einer Anlage mit $1kW - Peak$ erstellt werden, wobei die Jahressummenwerte in Abhängigkeit unterschiedlichster α_{PV} und γ_{PV} gebildet werden sollen, jeweils für das Jahr 2007, 2008, 2009 und 2010. α_{PV} soll sich dabei in 5° -Schritten zwischen -90° (Osten) bis $+90^\circ$ (Westen) bewegen. γ_{PV} in 5° -Schritten zwischen 0° (horizontal) und 90° . Legt man für jedes Jahr über die Jahressummenwerte eine interpolierte Fläche und stellt die Flächen in einem 3-dim Plot dar, so ergibt sich Abbildung 6.9. Die zugrundeliegenden Werte der $1kW - Peak$ Anlage können nun mit dem jeweiligen $kW - Peak$ Wert einer bestimmten Anlage einfach skaliert werden. Damit erhält man die abgeschätzten Werte für die Jahressummen der Energieumwandlung in Abhängigkeit der Drehung und Neigung einer bestimmten Anlage mit einem bestimmten $kW - Peak$ Wert.

Um die Information über die mögliche Drehung und Neigung der Paneele zu erhalten, wird jede aufgespannte Fläche der simulierten Jahressummenwerte, mit jeweils einer aufgespannten Fläche der Höhe der zugehörigen gemessenen Jahressumme geschnitten. Zur Demonstration ist dies für die kleine Anlage mit der Nr. 130425 und die große Anlage mit der Nr. 353243 für das Jahr 2007 in Abbildung 6.10 und 6.11 dargestellt. Die Darstellun-

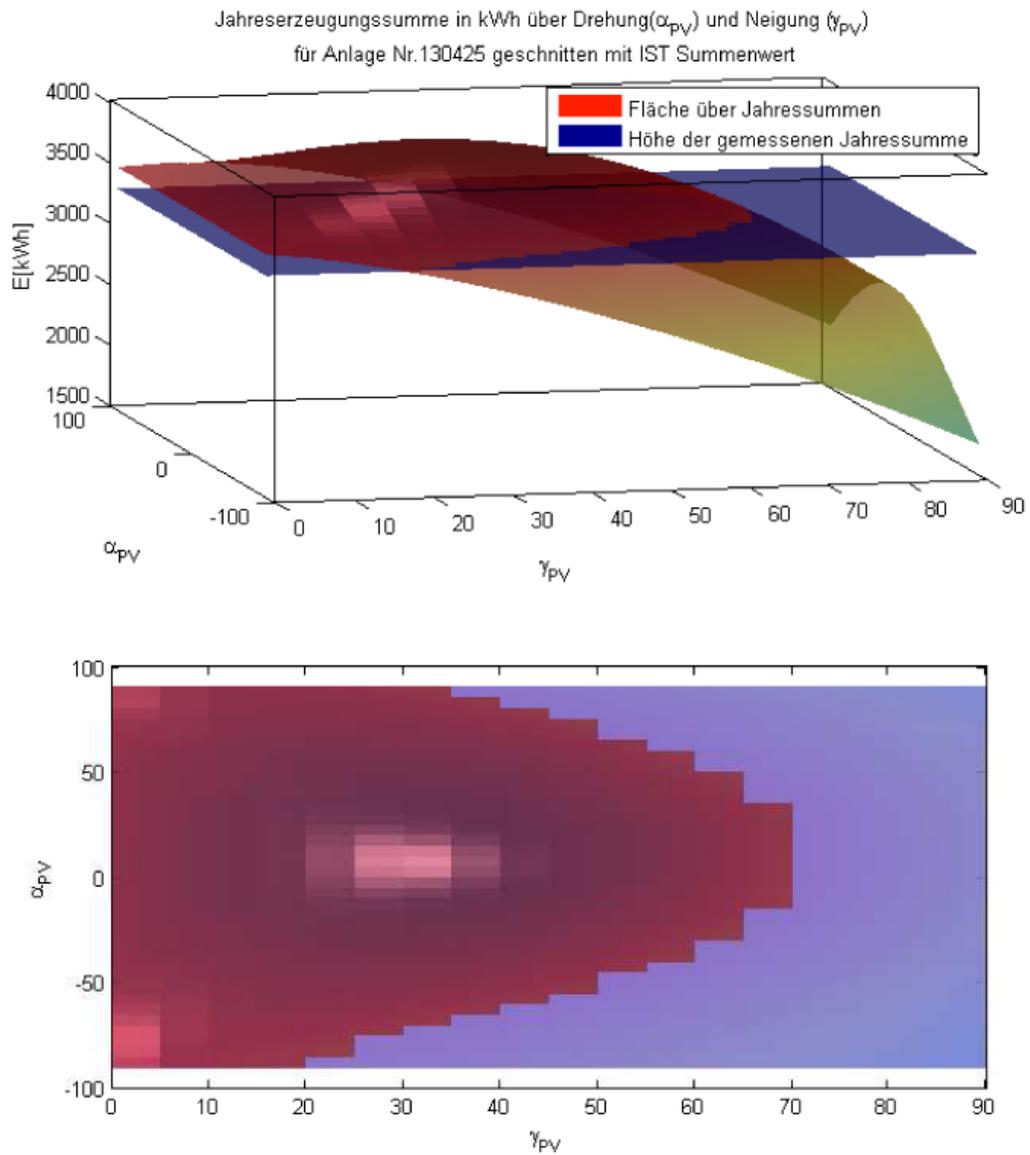


Abbildung 6.10: Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 130425 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

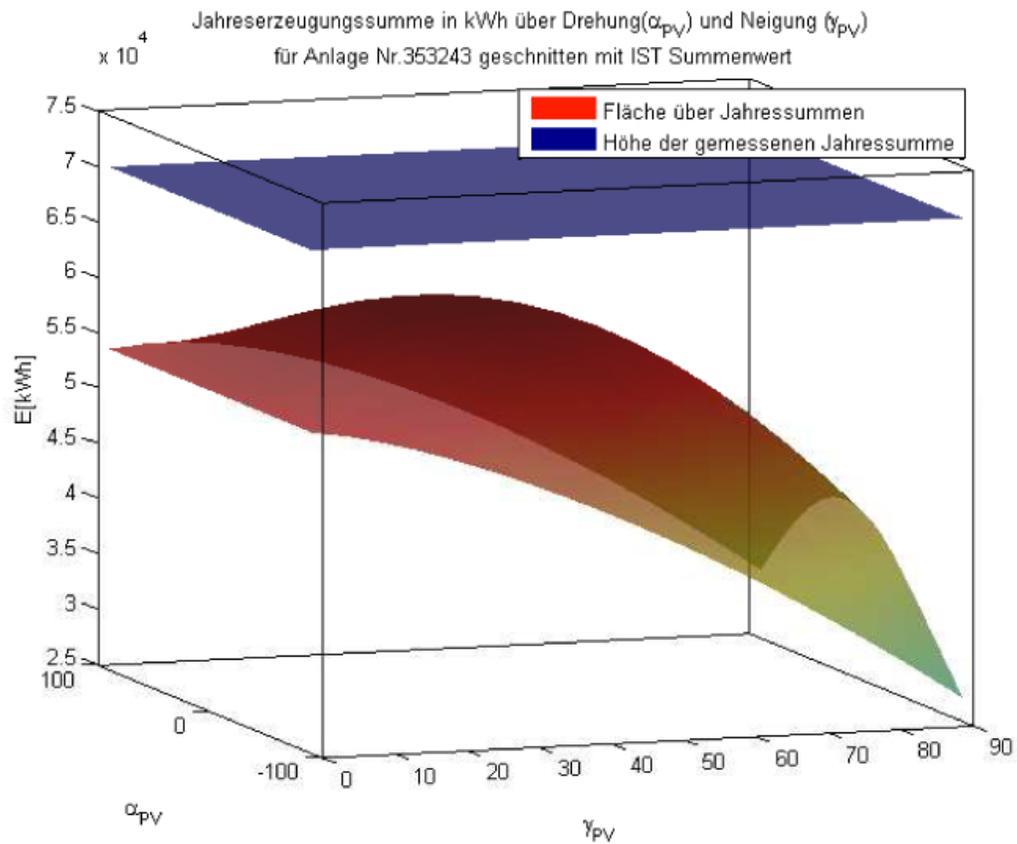


Abbildung 6.11: Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 353243 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

gen der Schnitte der anderen Anlagen sind im Anhang B zu finden: Nr. 246959 Abbildung B.3, Nr. 143344 Abbildung B.4 und Nr. 660867 Abbildung B.5. Bei den Anlagen mit der Nr. 130425, 246959 und 143344 kann anhand der Schnittlinie die Menge der Werte für die Kombination der Drehung und Neigung abgelesen werden, die zur gemessenen Jahressumme führen. Immer unter der Voraussetzung, dass die Annahmen für die Wirkungsgrade, Kabelverluste etc. stimmen. Um für gemessene Jahressummen in Frage kommende Werte für α_{PV} und γ_{PV} zu erhalten, müssten die Schnittlinien der betrachteten Jahre mathematisch ermittelt werden, und nach einer Vorschrift eine mittlere Schnittlinie errechnet werden, da jedes Jahr eine gering unterschiedliche Schnittlinie ergibt. Eine Berechnung der Schnittlinie und Ermittlung einer gemittelten Schnittlinie sei in dieser Arbeit nicht weiter behandelt. Warum nicht-nachgeführte Anlagen nicht optimal geneigt und ausgerichtet sind ($\alpha_{PV} = 0^\circ$ und $\gamma_{PV} = 33^\circ$), liegt meist daran, dass kleine Anlagen auf gegebene Dachflächen montiert werden, da eine Montage auf Gestellen zur optimalen Neigung und Ausrichtung auf eigens reservierten Flächen zu teuer ist.

Auffällig ist in Abbildung 6.11, dass die gemessene Jahressumme der Anlage Nr. 353243 deutlich über allen Werten der abgeschätzten Jahressummenwerte liegt. Dies gilt z.B. auch für die Anlage 660867, Abbildung B.5 im Anhang B. Dies könnte den Grund darin haben, dass diese Anlagen dem Sonnen-Azimut Winkel α_S nachgeführt sind. Die kW-Peak deuten auf flächenmäßig große Anlagen hin, und bei großen Anlagen zahlt sich eine Nachführung des Azimutwinkels aus. Um zu überprüfen, ob diese Anlagen nachgeführt werden, wird ein Normverlauf einer nachgeführten $1kW - Peak$ Anlage simuliert. Dabei handelt es sich nur mehr um einen 2-dim Plot, da $\alpha_S = \alpha_{PV}$ bzw. $\alpha_S = \alpha_{PV} - 180^\circ$ ist, je nachdem welchen Bezug α_S hat, Norden oder Süden. Abbildung 6.12 zeigt dazu die Normverläufe der simulierten Jahressummen für *Wolfsberg* bei Variierung der Neigung γ_{PV} der Anlage. Schneidet man zur Demonstration beispielsweise den Jahressummenverlauf 2007 mit einer Geraden der Höhe der gemessenen Jahressumme 2007 der Anlage mit der Nr. 353243, siehe Abbildung 6.13, so erkennt man deutlich, dass die Anlage nachgeführt sein muss. Anders würde die Jahressumme mit dieser Anlage nicht erreichbar sein. Optimal geneigt und nachgeführt ergibt aber relativ genau die gemessene Jahressumme der umgewandelten Energie.

6.5 Einsetzbarkeit des physikalischen Modells zur kurzfristigen Leistungsprognose von Photovoltaikanlagen

Nachdem im vorhergehenden Kapitel 6.4 die Methode zur Bestimmung der wichtigsten Unbekannten α_{PV} , γ_{PV} und Nachführung JA/NEIN für das Simulationsmodell nach Abbildung 6.7 für ungemessene Anlagen beschrieben wurde, kann man die Vorgehensweise zur kurzfristigen Prognose der Leistung von PVA zum Zeitpunkt $t + h$ wie folgt beschreiben:

- Alle PVA werden auf eine mögliche Nachführung hin untersucht. Erkennbar ist sie anhand der Höhe der Ebene der gemessenen Jahressumme der umgewandelten Energie, im Vergleich der Fläche über die skalierten Jahressummenwerte der umgewandelten Energie in Abhängigkeit von α_{PV} und γ_{PV} , wie im Kapitel 6.4 beschrieben.
- Wird die Anlage nachgeführt, muss nur γ_{PV} ermittelt werden und $\alpha_{PV} = \alpha_S$ bzw. $\alpha_{PV} = \alpha_S - 180^\circ$, je nach Bezug auf Norden oder Süden von α_S . Wird sie nicht

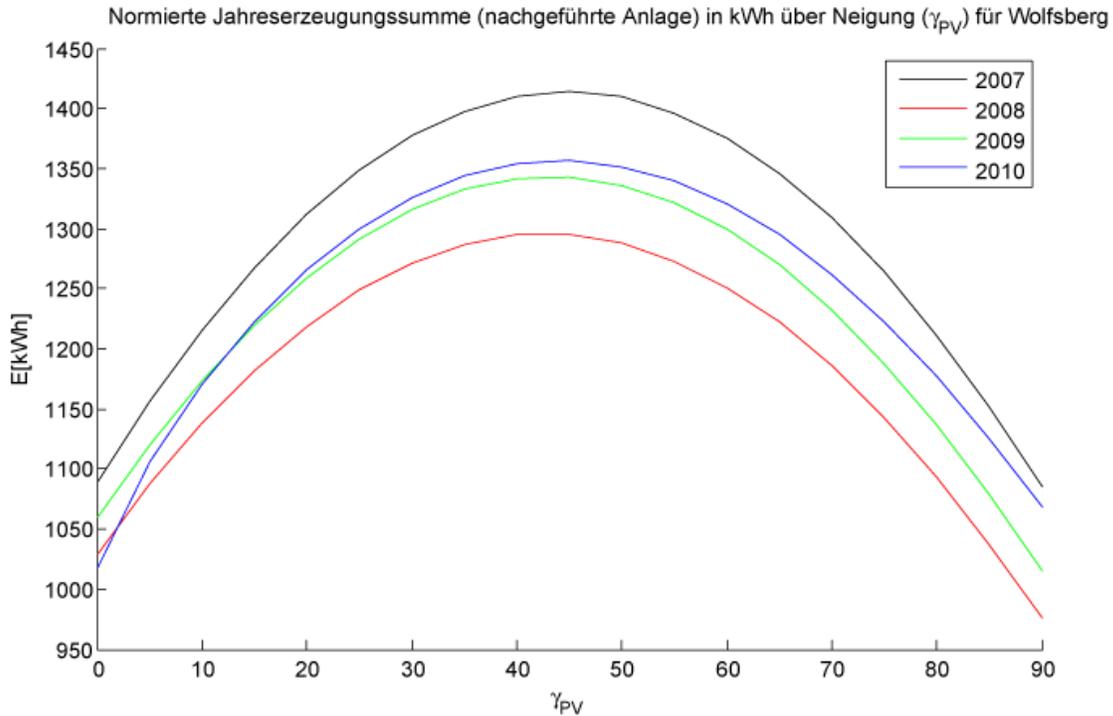


Abbildung 6.12: Jahressummenverlauf einer nachgeführten $1kW$ – *Peak* Anlage, in Abhängigkeit der Neigung γ_{PV} für den Standort *Wolfsberg*.

nachgeführt, muss zusätzlich α_{PV} bestimmt werden, wie im Kapitel 6.4 beschrieben.

- Als Solarstrahlungs- und Umgebungstemperaturwerte, werden für die Prognose die zukünftig zu erwartenden Solarstrahlungs- und Temperaturwerte benötigt. Bisher wurden bekannte Vergangenheitswerte zur Analyse verwendet. Für die Prognose wird jede Anlage einer Region zugeteilt, für die eine separate Prognose für Solarstrahlungs- und Temperaturwerte erstellt wird. Die Prognose kann z.B. vom Wetterdienst *ZAMG* oder anderen Wetterdiensten stammen. Liegt eine PVA genau an Grenzen solcher Regionen, können die zugehörigen Solarstrahlungs- und Temperaturprognosewerte aus einer gewichteten Summe aus den Werten der zutreffenden Regionen gebildet werden.
- Alle anderen Parameter können z.B. wie im Kapitel 6.4 beschrieben, gewählt werden.
- Die $\frac{1}{4}h$ Leistungswerte aus dem Simulationsmodell werden in einem Plot oder in einer Tabelle ausgegeben.

Die so ermittelten $\frac{1}{4}h$ Leistungswerte zu den Zeitpunkten t_i+h einer PVA, stellen dann die kurzfristigen Prognosewerte der Leistung dar. Die Prognosegüte ist, abgesehen von den Fehlern der verwendeten Modelle für einzelne Teile des Simulationsmodells, direkt von der Prognose der Solarstrahlung und Temperatur des Wetterdienstes abhängig. Wobei der Fehler der Prognose der Solarstrahlungswerte stärker in den Fehler der Leistungsprognose

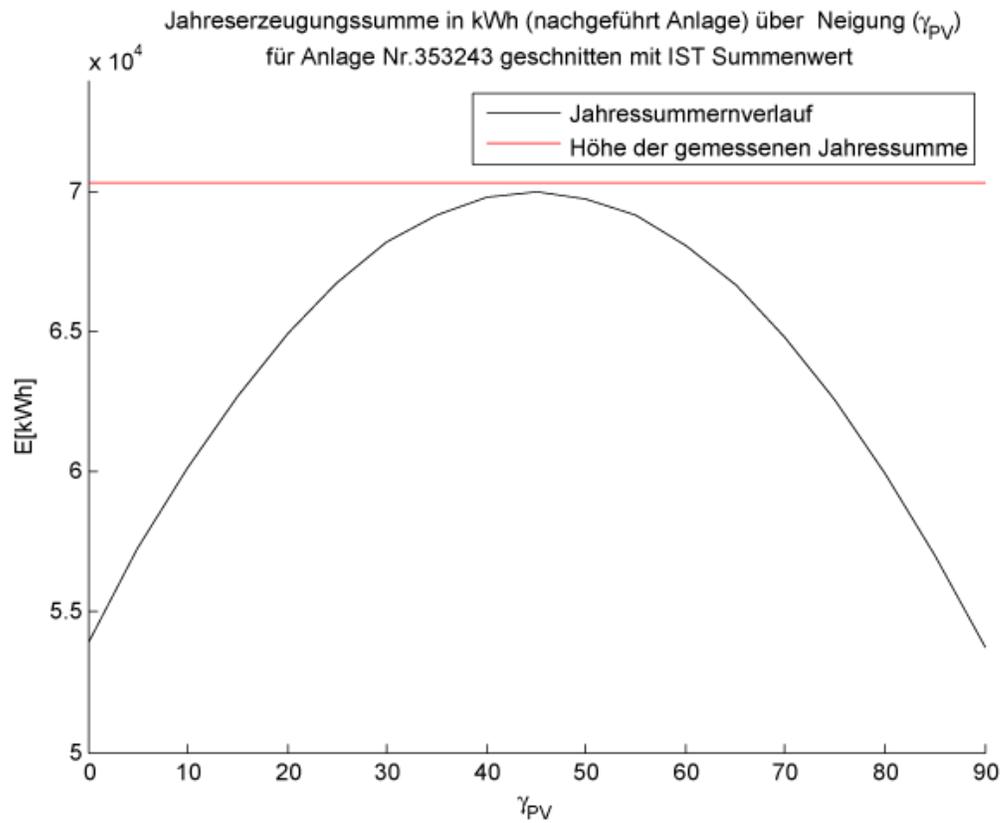


Abbildung 6.13: Sklierter Jahressummen-Normverlauf, in Abhängigkeit der Neigung der Anlage Nr. 353243 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

eingeht.

Kapitel 7

Schlussbemerkungen und Ausblick

Diese Arbeit beschreibt Ansätze zur Erstellung einer kurzfristigen Leistungsprognose von dezentralen Einspeisern und beschreibt neben theoretischen Grundlagen, auch ausführliche Simulationsergebnisse zu den Versuchen mit unterschiedlichen Modellen. Der Fokus wird dabei auf Wasserlaufkraftwerke und Photovoltaikanlagen gelegt. Die Prognose bei Wasserlaufkraftwerken wird über regressionsbasierte Methoden realisiert und unterscheidet zwischen zwei grundlegenden Arten. Einmal inklusive aktueller Leistungswerte eines Kraftwerks und einmal exklusive, dh. das Modell ist blind bezüglich des Verhaltens der nachzubildenden Strecke am Ausgang. Die Strecke einer Photovoltaikanlage wird über ein physikalisches Modell abgebildet und kann so für die Prognose der Leistung verwendet werden. Hierbei ist vor allem die Parameterschätzung bzw. Parameterfindung einzelner Modellteile interessant. Es gibt derzeit keine Information über den Eigenverbrauch einer Anlage bzw. jenem Teil an Energie einer Photovoltaikanlage oder eines Wasserlaufkraftwerks, die für private Zwecke vor der Netzeinspeisung abgezweigt wird. Solange die Energie aus erneuerbaren Quellen eine hohe staatliche Subvention hat, wird jeder Anlagenbesitzer bestrebt sein, die gesamte umgewandelte Energie aus solchen Anlagen in das Netz zu speisen und möglichst keine Energie aus diesen Anlagen selbst zu verwenden. Auch hier wäre es interessant, eine Strategie zur Erfassung oder Abschätzung des Eigenverbrauches eines Anlagenbesitzers zu entwickeln, andernfalls wird es nicht möglich sein, auf Dauer eine qualitative Prognose über die Einspeisung an Energie durch dezentrale Anlagen zu gewährleisten.

Als zukünftige Arbeit ist eine Ausarbeitung und Vorgehensweise für eine Implementierung der getesteten und beschriebenen Methoden für die Prognose der Wasserlaufkraftwerke und Photovoltaikanlagen in das bestehende System des Netzbetreibers *KELAG Netz GmbH* anzustreben. Da das momentan vorherrschende System beim Netzbetreiber *KELAG Netz GmbH* zur Lastprognose von Drittfirmen entwickelt und betreut wird, wird es in einer möglichen zukünftigen Arbeit zu diesem Thema vor allem nach einer genauen Studie des bestehenden Systems darum gehen, einer Drittfirma ein Modell zur Implementierung der kurzfristigen Prognose über die zu erwartende Leistung von Wasserlaufkraftwerken und Photovoltaikanlagen zum Zeitpunkt $t + h$ vorzuschlagen. Ebenfalls soll dabei die Implementierung der Erkennung der im Kapitel 1.5 beschriebenen Fehlerfälle im Stromversorgungsnetz untersucht und entwickelt werden. Eine direkte Fortsetzung dieser Arbeit könnte die Untersuchung der kurzfristigen Leistungsprognose von Windkraftanlagen darstellen. Zurzeit gibt es in Kärnten praktisch keine Windkraftanlagen, werden aber

mittelfristig im Zuge der Diskussion um den Atomenergieausstieg Europas, ein weiteres wichtiges Glied zur ökologischen Energiegewinnung auch in Kärnten/Österreich sein. Vor allem wäre hierzu eine Entwicklung eines physikalischen Modells interessant, da es dazu weitreichende und umfassende Literatur gibt. Ein Beispiel dazu könnte [HAB⁺10, Kap. 13] sein.

7.1 Ergänzendes zu Wasserlaufkraftwerke

Für die kurzfristige Leistungsprognose von Wasserlaufkraftwerken wäre es wichtig, wie im Kapitel 5.2.4 erwähnt, ein dichteres Netz an Niederschlagsmessstationen zu haben, um die Fehler durch lokal begrenzte, teils starke Niederschläge durch Gewitterschauer zu minimieren. Ideal wäre eine Niederschlagsmessung je Wasserlaufkraftwerk. Es gibt einige wenige Wasserlaufkraftwerke, die keine Messung bezüglich Ihrer Leistungswerte besitzen. Für sie kann das regressionsbasierte Modell ebenfalls verwendet werden. Dazu müsste ein im Verhalten ähnliches Wasserlaufkraftwerk zugeordnet und adaptiert werden, welches Bezug auf die selbe Prognoseregion für die Prognose der Niederschlagswerte nimmt. Die Umgebungstemperatur könnte dabei aus einer anderen Prognoseregion bezogen werden. Es wäre für die kurzfristige Prognose der Leistung von Wasserlaufkraftwerken empfehlenswert, die Datensituation der Prognose der Niederschlagsmengen durch den zuständigen Wetterdienst in der Art zu verbessern, dass die räumliche Auflösung feiner wird. Derzeit gibt es eine Niederschlagsprognose für 7 Regionen für Kärnten, im 6h Takt für 48h in die Zukunft, beginnend mit Mitternacht. Bei großflächigen Niederschlägen im Falle eines Weterntiefdrucks, ist dieses räumliche Prognoseraster sicherlich ausreichend. Vor allem wäre eine feinere Auflösung der Niederschlagsprognosen bei Gewitterperioden interessant, da Gewitter meist lokal und auf gewisse Talschaften begrenzt auftreten.

Für weiterführende Arbeiten könnte es interessant sein, zu untersuchen, welche Umweltparameter und Messdaten zur physikalischen Beschreibung der Strecke von Wasserlaufkraftwerken hauptsächlich notwendig wären, inwiefern damit ein physikalisches Modell erstellbar wäre, welchen Aufwand dies bedeuten würde und wie gut die Vorhersagequalität damit sein würde. So ein physikalisches Modell könnte man z.B. für das in dieser Arbeit verwendete Wasserlaufkraftwerk *Kirchbach* bilden, um einen Vergleich mit der Qualität des hier beschriebenen regressionsbasierten Modells zu haben.

7.2 Ergänzendes zu Photovoltaikanlagen

Im Bezug auf die Photovoltaikanlagen, stehen dem Netzbetreiber *KELAG Netz GmbH* für Ober- und Unterkärnten (Ost- und Westkärnten) für 48h im 6h Takt, beginnend mit Mitternacht, Prognosewerte des Bedeckungsgrades zur Verfügung. Der Bedeckungsgrad ist in 8 Stufen unterteilt, 1–8, und skaliert die 6h Intervalle der extraterrestrischen Solarstrahlung auf eine Horizontale, siehe Kapitel 4.1.2 und 4.1.3, logarithmisch. Dh., der Tagesverlauf der theoretischen Solarstrahlung auf eine Horizontale auf der Erdoberfläche, bei absolut klarem Himmel und keinen Streuungseffekten durch die Atmosphäre, wird mit diesen Bedeckungsgraden skaliert, wobei ein Wert des Bedeckungsgrades 6h lang gültig ist. Die Prognose der Bedeckung bzw. Solarstrahlung auf der Erdoberfläche kann nur mittlere Werte beschreiben, da es nicht möglich ist, einzelne lokal begrenzte Wolkenzüge und daraus resul-

tierende lokal begrenzte Schwankungen der Solarstrahlung zu prognostizieren. Das Modell ist direkt von der Eingangsgröße *Solarstrahlung zum Zeitpunkt t* abhängig. Daher ist die Qualität der Prognose des Modells für Photovoltaikanlagen direkt von der Qualität der Prognosedaten von Solarstrahlung und Temperatur abhängig. Die Umgebungstemperatur wird vom Wetterdienst ZAMG mit 7 Regionen für Kärnten unterteilt. Diese räumliche Auflösung sollte für die Umgebungstemperatur ausreichend sein, da sich vor allem die Umgebungstemperatur innerhalb der Regionen und des Zeitrasters nur wenig bzw. langsam ändert. Für die Paneeltemperatur und die Leistung einer Photovoltaikanlage ist aber die Solarstrahlung das Um und Auf. Und gerade hier fehlt eine feine räumliche und zeitliche Auflösung der Prognose. Wie in den Simulationen zu den Photovoltaikanlagen im Kapitel 6 gezeigt wurde, wären für eine gute Qualität der Abschätzung der vergangenen Leistungswerte einer Photovoltaikanlage je Ort oder Gemeinde gültige Solarstrahlungsmesswerte notwendig. Dies gilt auch für die Prognose der Leistung der Photovoltaikanlagen. Hier wäre eine örtlich und zeitlich fein aufgelöste Prognose der Solarstrahlung notwendig. Es gilt, je feiner und genauer die Auflösung, desto feiner und genauer die Prognose der Leistung der Photovoltaikanlagen, da ein direkter Zusammenhang vorliegt. Ebenfalls ist zu empfehlen, dass anstatt der Bedeckungsgrade, Solarstrahlungswerte auf eine Horizontale auf der Erdoberfläche vom zuständigen Wetterdienst an den Netzbetreiber übergeben werden. Sie sind für die Berechnung des Klarheitsfaktors k_T notwendig.

Zukünftig könnte man versuchen, im Simulationsmodell Abschattungsverluste mit zu modellieren. Als Grundlage dazu kann die Arbeit von *Quaschnig* dienen, [Qua96]. Außerdem könnten Effekte der Atmosphäre auf das Solarstrahlungsspektrum betrachtet werden, vor allem deshalb, weil Solarmodule auf bestimmte Solarstrahlungsspektren abgestimmt sind, um einen guten Wirkungsgrad zu erreichen. Ebenfalls könnten die Modelle für die Berechnung der Solarstrahlung auf die Paneelebene aus der gemessenen Solarstrahlung auf die Horizontale genauer für Kärnten/Österreich untersucht werden. Vor allem wäre es interessant, einen eigenen statistischen Zusammenhang für die Umrechnung der wirksamen Solarstrahlung auf eine geneigte Paneelebene aus der horizontal eintreffenden Solarstrahlung zu bilden. Damit wäre ein für Kärnten/Österreich maßgeschneidertes wichtiges Modell für die Simulation vorhanden. Es gibt eine gemessene und gut dokumentierte Photovoltaikanlage in *St. Veit*, bei der, bis auf die Messung der Solarstrahlung auf eine horizontale Fläche, alle Messwerte vorhanden sind. Für die Evaluierung des physikalischen Modells für Photovoltaikanlagen ist zu empfehlen, dass eine Messung der Solarstrahlung über ein horizontal montiertes Solarstrahlungsmessgerät vor Ort bei der gemessenen Photovoltaikanlage in *St. Veit* ergänzend hinzukommt. Hätte man vor Ort eine Solarstrahlungsmessung auf eine Horizontale, so könnte die Evaluierung der Strahlungsumrechnung genauer und aussagekräftiger erfolgen. Die Abschätzung der Paneeltemperatur kann auf die Art wie im Kapitel 4.1.5 beschrieben ist, durchgeführt werden. Aus einem Fehler in der Umrechnung der Solarstrahlung folgt aber auch eine fehlerhafte Temperaturabschätzung der Paneele, da die Solarstrahlung auf die Paneelebene eine Eingangsgröße der Paneeltemperaturabschätzung ist.

Literaturverzeichnis

- [ABIBM06] E. Agirre-Basurko, G. Ibarra-Berastegi, and I. Madariaga. Regression and multilayer perceptron-based models to forecast hourly O_3 and NO_2 levels in Bilbao area. In *Environmental Modelling & Software (21)*, pages 430 – 446. University of the Basque Country, Elsevier, Science Direct, 2006.
- [Bla06] Tobias Blank. Modellierung und Verifikation von Photovoltaikanlagen im Verteilnetz. Master’s thesis, Technische Universität München, Fachgebiet für Elektrische Energieversorgungsnetze, 2006.
- [Bur98] Christopher J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. In Usama Fayyad, editor, *Data Mining and Knowledge Discovery (2)*, pages 121–167. Bell Laboratories, Lucent Technologies, Kluwer Academic Publishers, 1998.
- [BWB⁺04] Hans Joachim Bodenhöfer, Norbert Wohlgemuth, Markus Bliem, Anneliese Michael, and Klaus Weyerstraß. Bewertung der volkswirtschaftlichen Auswirkungen der Unterstützung von Ökostrom in Österreich; Endbericht. Technical report, Institut für Höhere Studien Kärnten, Domgasse 3, 9020 Klagenfurt, Austria, Juli 2004.
- [CAB01] Paulin Coulibaly, François Anctil, and Bernard Bobée. Multivariate Reservoir Inflow Forecasting Using Temporal Neural Networks. *Journal of Hydrologic Engineering*, 367-376, September/October 2001.
- [CKI] G. Cramer, W. Kleinkauf, and M. Ibrahim. PV-System-Technik; Entwicklungsstand und Trends in der dezentralen Elektrifizierung. Technical report, Institut für Elektrische Energietechnik IEE-EVS der Universität Kassel, SMA Regelsysteme GmbH Niestetal, Wilhelmshöher Allee 73, 34121 Kassel, Germany und Hannoversche Straße 1-5, 34266 Niestetal, Germany.
- [CLG] Rich Caruana, Steve Lawrence, and Lee Giles. Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping. Technical report, CALD CMU Pittsburgh, NEC Research Institute Princeton, Information Sciences Penn State University.
- [Elm90] Jeffrey L. Elman. Finding Structure in Time. *Cognitive Science*, 14:179–211, 1990. University of California, San Diego.

- [FKL09] Ludwig Fahrmeir, Thomas Kneib, and Stefan Lang. *Regression; Modelle, Methoden und Anwendungen*. Zweite Auflage. Springer-Verlag Berlin Heidelberg, 2007, 2009.
- [HAB⁺10] Richard Zahoransky (Hrsg.), Hans-Josef Allelein, Elmar Bollin, Helmut Oehler, and Udo Schelling. *Energietechnik; Systeme zur Energieumwandlung. Kompaktwissen für Studium und Beruf*. 5., überarbeitete und erweiterte Auflage. Vieweg+Teubner Verlag, 2010.
- [HAIM07] Nikos Hatziargyriou, Hiroshi Asano, Reza Iravani, and Chris Marnay. Microgrids; An Overview of Ongoing Research, Development, and Demonstration Projects. *IEEE power & energy magazine, IEEE 1540-7977/07*, July/August 2007.
- [Jae] Herbert Jaeger. Adaptive Nonlinear System Identification with Echo State Networks. Technical report, International University Bremen, D-28759 Bremen, Germany.
- [Jae10] Herbert Jaeger. The “echo state” approach to analysing and training recurrent neural networks. Technical report, Fraunhofer Institute for Autonomous Intelligent Systems, January 2010. Corrected version.
- [Kal03] Richard Kales. Die Eignung mathematischer Ansätze zur Lastprognose. Master’s thesis, Universität Klagenfurt, Fakultät für Wirtschaftswissenschaften und Informatik, März 2003.
- [KH95] Anders Krogh and John A. Hertz. A Simple Weight Decay Can Improve Generalization. In *Advances in Neural Information Processing Systems*, number 4, pages 950–957. Connect, The Niels Bohr Institute, Denmark, and Nordita, Denmark, Morgan Kaufmann Publishers, San Mateo CA, 1995.
- [KKR⁺11] Koen Kok, Stamatis Karnouskos, Jan Ringelstein, Aris Dimeas, Anke Weidlich, Cor Warmer, Stefan Drenkard, Nikos Hatziargyriou, and Valy Lioliou. Field-Testing Smart Houses for a Smart Grid. In *21st International Conference on Electricity Distribution, Frankfurt*. ECN The Netherlands, SAP Germany, Fraunhofer IWES Germany, NTUA ICCS Greece and MVV Germany, PPC Greece, CIRED, June 2011. Paper No. 1291.
- [KLSK] Timo Koskela, Mikko Lehtokangas, Jukka Saarinen, and Kimmo Kaski. Time Series Prediction with Multilayer Perceptron, FIR and Elman Neural Networks. Technical report, Tampere University of Technology, Electronics Laboratory, 33101 Tampere, Finland.
- [LXZC] Da-Ying Li, Wei Xu, Hong Zhao, and Rong-Quy Chen. A SVR Based Forecasting Approach for Real Estate Price Prediction. In *Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding*. Huazhong University of Science and Technology, Wuhan, and Renmin University of China, Beijing, and Graduate University of Chinese Academy of Science, Beijing.

- [Maa07] Wolfgang Maass. Eigenhändige Mitschrift aus der Vorlesung *Computational Intelligence*, Vorlesungsteil Prof. Wolfgang Maass, verfasst durch Aichholzer Thomas, 2007. Institut für Grundlagen der Informationsverarbeitung, Technische Universität Graz, Austria.
- [Nic10] Daniel Nicklaß. *Kurzzeit-Prognose von PM₁₀-Konzentrationen in der Luft unter Verwendung maschineller Lernverfahren*. PhD thesis, Universität Stuttgart, Germany, 2010. Institut für Geographie.
- [Pfa01] Jochen Pfalzgraf. Künstliche neuronale Netze; Grundlagen und Anwendungen, 2001. Nach der Vorlesung an der Paris-Lodron-Universität 5020 Salzburg, Austria; Dokument erstellt von *Eckschlager Manfred, Pfligersdorffer Christian, Rettenbacher Christopher und Vogl Christopher*.
- [Qua96] Volker Quaschnig. *Simulation der Abschattungsverluste bei solarelektrischen Systemen; Beiträge zur Solarforschung - Solarstrahlung, Solarenergie, Photovoltaik; 1.Auflage; Band 6*. Verlag Dr. Köster, Berlin, 1996.
- [REG10] Sebnem Rusitschka, Kolja Eger, and Christoph Gerdes. Smart Grid Data Cloud: A Model for Utilizing Cloud Computing in the Smart Grid Domain. Technical report, Corporate Technology Information and Communications, Siemens AG Munich, Otto-Hahn-Ring 6, 81739 Munich, Germany, 2010. IEEE 978-1-4244-6511-8/10.
- [Sch91] Jürgen Schumacher. *Digitale Simulation regenerativer elektrischer Energieversorgungssysteme*. PhD thesis, Universität Oldenburg; Fachbereich Physik, 1991.
- [Sch02] Jörg Scheffler. *Bestimmung der maximal zulässigen Netzanschlussleistung photovoltaischer Energiewandlungsanlagen in Wohnsiedlungsgebieten*. PhD thesis, Technische Universität Chemnitz; Fakultät für Elektrotechnik und Informationstechnik, 2002.
- [SH01] Rainer Schlittgen and Bernd H.J.Streitberg. *Zeitreihenanalyse*. 9., unwesentlich veränderte Auflage. R. Oldenbourg Verlag München Wien, 2001.
- [SHP08] Ernst Stadlober, Siegfried Hörmann, and Brigitte Pfeiler. Quality and performance of a PM10 daily forecasting model. In *Atmospheric Environment (42)*, pages 1098 – 1109. Institut of Statistics and Department of Mathematics, Graz University of Technology, Austria, Elsevier, Science Direct, October 2008.
- [SOP⁺07] Rodrigo Sacchi, Mustafa C. Ozturk, José C. Príncipe, Adriano A.F.M. Carneiro, and Ivan N. da Silva. Water Inflow Forecasting using the Echo State Network: a Brazilian Case Study. In *Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA*, pages 12–17, August 2007.
- [SS04] Alex J. Smola and Bernhard Schölkopf. A tutorial on support vector regression. In *Statistics and Computing (14)*, pages 199 – 222. RSISE, Australian

- National University, Canberra, Australia, and Max-Planck-Institut für biologische Kybernetik, Tübingen, Germany, Kluwer Academic Publishers, 2004.
- [VC] Vladimir Vapnik and Corinna Cortes. Support-Vector Networks. Technical report, AT&T Labs-Research, USA.
- [WCY08] C.L. Wu, K.W. Chau, and Y.S.Li. River Stage Prediction Based on a Distributed Support Vector Regression. *Journal of Hydrology*, 358(No.1-2, pp96-111), 2008. pre-published version.
- [WP90] Ronald J. Williams and Jing Peng. An Efficient Gradient-Based Algorithm for On-Line Training of Recurrent Network Trajectories. *Neural Computation*, 2:490–501, 1990. College of Computer Science, Northeastern University Boston.
- [WRV09] Richard L. Welch, Stephen M. Ruffing, and Ganesh K. Venayagamoorthy. Comparison of Feedforward and Feedback Neural Network Architectures for Short Term Wind Speed Prediction. In *Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA*, number 978-1-4244-3553-1/09, pages 3335 – 3340. IEEE, June 2009.
- [WW] Dominik Wieland and Franz Wotawa. Local Maximum Ozone Concentration Prediction Using Neural Networks. Workshop on Environmental Decision Support Systems and Artificial Intelligence, Technische Universität Wien, Institut für Informationssysteme and Ludwig Wittgenstein Laboratory for Information Systems, Wien.
- [ZXDH] Pan Zhao, Junrong Xia, Yiping Dai, and Jiaying He. Wind Speed Prediction Using Support Vector Regression. In *5th IEEE Conference on Industrial Electronics and Applications*, pages 882–886. IEEE 978-1-4244-5046-6/10.

Anhang A

Implementierung der Simulationen, README

In der Abhandlung dieser Diplomarbeit wurden die Hintergründe und Vorgehensweisen ausreichend geschildert. In diesem Readme sollen Hinweise zur Ausführung der praktischen Simulation angeführt werden. Es werden nur die Ordner und Dateien, die für die Erstellung des Dokumentes notwendig waren, aufgezählt.

- Alle Simulationen sind mit dem Matlab Programmpaket 2009b erstellt worden.
- Für die regressionsbasierten Simulationsmodelle (*/_MApractice_regMod*) gilt:
 - */CSV* enthält die Zeitreihen für die Simulationen für Niederschlag, Temperatur sowie Leistung der WLKW für unterschiedliche Zeiträume in Form von *.csv Dateien. Die erste Spalte beschreibt die Laufvariable Datum und Uhrzeit, die zweite Spalte enthält den jeweiligen Wert zur jeweiligen Größe.
 - */FF_nn* ist die Umgebung für die Simulationen für die feedforward neuronalen Netze.
 - */esn* ist die Umgebung für die Simulationen für die Echo State Netze.
 - */rnn_elman* ist die Umgebung für die Simulationen mit den Elman Netzen.
 - */best* ist die Umgebung für die Simulationen mit dem besten gefundenen neuronalen Netz.
- Für das physikalische Simulationsmodell (*/_MApractice_phyMod*) gilt:
 - */CSV* enthält die Zeitreihen für die Simulationen für Niederschlag, Temperatur sowie Leistung der WLKW für unterschiedliche Zeiträume in Form von *.csv Dateien. Die erste Spalte beschreibt die Laufvariable Datum und Uhrzeit, die zweite Spalte enthält den jeweiligen Wert zur jeweiligen Größe.
 - */Photovoltaic* ist die Umgebung für die Simulationen des physikalischen Modells für die PVA.
 - */stVeit* enthält relevante Daten zur PVA *St. Veit*, wie Wechselrichterdatenblätter, Verschaltungsplan, Paneeldatenblatt, Datenblatt des Solarkabels, Sternpyranometerdatenblatt und Fotos der Anlage.

- Generell gilt, dass die Matlabdateien mit der Bezeichnung *getDataKonfigs<X>.m* und *getNetKonfigs<X>.m*, X ist optional eine Zahl, die Konfigurationen für die Inputvektoren bzw. des Netzaufbaues für die einzelnen Simulationen definieren. Hier kann also eine Simulation eingestellt werden. Die zugehörige Simulation lädt diese beiden Dateien.

(Regressionsbasierte Modelle)

/FF_nn enthält:

- */_1_ersteTestImpl* ist die Umgebung für den ersten Versuch mit feedforward neuronalen Netzen, zum Kapitel 5.2.3. *FF_nn1.m* startet dazu die Simulation. *AuswertungKonfigs1_inkl2007.m* macht die Auswertung der Simulation, dazu muss aber die *ergebnisse1.mat* Datei geladen sein.
- */_2_verfeinerung* ist die Umgebung für die verfeinerte Simulation mit feedforward neuronalen Netzen, zum Kapitel 5.2.4. *FF_nn2.m* startet dazu die Simulation. *AuswertungKonfigs2_inkl2007.m* macht die Auswertung der Simulation, dazu muss aber die *ergebnisse2.mat* Datei geladen sein.
- */_3_inklRand* ist die Umgebung für die Simulation mit randomisierter Abfolge der Inputvektoren für die feedforward neuronalen Netze, zum Kapitel 5.2.4. *FF_nn3.m* startet dazu die Simulation. *AuswertungKonfigs3_inkl2007.m* macht die Auswertung der Simulation, dazu muss aber die *ergebnisse3.mat* Datei geladen sein.
- */_4_inklAktErz* ist die Umgebung für die Simulation der Inputvektoren mit aktuellem Leistungswert eines WLKW für die feedforward neuronalen Netze zum Kapitel 5.2.5. *FF_nn4.m* startet dazu die Simulation. *AuswertungKonfigs4_inkl2007.m* macht die Auswertung der Simulation, dazu muss aber die *ergebnisse4.mat* Datei geladen sein.
- */_5_inklErzReihe* ist die Umgebung für die Simulation der Inputvektoren inklusive der vergangenen Leistungszeitsreihe eines WLKW für die feedforward neuronalen Netze zum Kapitel 5.2.5. *FF_nn5.m* startet dazu die Simulation. *AuswertungKonfigs5_inkl2007.m* macht die Auswertung der Simulation, dazu muss aber die *ergebnisse5.mat* Datei geladen sein.

/rnn_elman enthält:

- */elman* ist die Umgebung für die Simulation zum reinen online Training mit Elman Netze, ohne Windowing, zum Kapitel 5.3.1. *elman.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.
- */elman_windowing* ist die Umgebung für die Simulation zum online Training mit Elman Netze, mit Windowing, zum Kapitel 5.3.2. *elman.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.

- */elman_windowing_inklErzReihe* ist die Umgebung für die Simulation zum online Training mit Elman Netze, inklusive der Information über vergangene Leistungswerte eines WLKW und mit Windowing, zum Kapitel 5.3.3. *elman.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.

/esn enthält:

- */single1_pinv_setuu* ist die Umgebung für die Simulation des Echo State Netzes mit einem linearen Readout, zum Kapitel 5.4.1. *esn.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.
- */single2_pinv_setuu_avgTemp* ist die Umgebung für die Simulation des Echo State Netzes mit einem linearen Readout und geglätteter Temperaturzeitreihe, zum Kapitel 5.4.1. *esn.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.
- */single3_perc_setuu* ist die Umgebung für die Simulation des Echo State Netzes mit einem Perzeptron als Readout, zum Kapitel 5.4.1. *esn.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.
- */tripple1_perc_slidingWin_setuu_asym* ist die Umgebung für die Simulation des erweiterten Echo State Netzes mit einem Perzeptron als Readout, zum Kapitel 5.4.2. *esn.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.
- */tripple2_perc_slidingWin_setuu_asym_inklErzReihe* ist die Umgebung für die Simulation des erweiterten Echo State Netzes mit einem Perzeptron als Readout, inklusive Information über vergangene Leistungswerte eines WLKW, zum Kapitel 5.4.3. *esn.m* führt die Simulation durch, *AuswertungKonfigs.m* macht die Auswertung, wobei zuvor die */ergebnisse/ergebnisse.mat* Datei geladen sein muss.

/best enthält die Simulationen für das beste ausgewählte Modell nach Kapitel 5.5:

- */ffnn_inklErz_gaussRauschenWettervorhersage*: Überlagertes additives Gaußsches Rauschen auf den Wetterdaten, Kapitel 5.6.1. Gestartet wird die Simulation mit *FF_nn.m*.
- */ffnn_inklErz_scaleRauschenWettervorhersage*: Skalieren der Wetterdaten mit einem Gauss Rauschen, Kapitel 5.6.1. Gestartet wird die Simulation mit *FF_nn.m*.
- */ffnn_inklErz_verschiedeneWLKWs*: Simulationen weiterer WLKW, Kapitel 5.6.2. Gestartet wird die Simulation mit *FF_nn.m*.

(Physikalisches Modell)

/Photovoltaic enthält die Simulationen für das physikalische Modell:

- *A_eval_diffStrahlungModell.m* führt das Skript zur Evaluierung der Umrechnung der Strahlung von einer Horizontalen auf die Paneelebene der gemessenen PVA *St. Veit* aus, Kapitel 6.1.

- *A_eval_tempAbschaetzung.m* führt das Skript zur Evaluierung der Paneeltemperaturabschätzung aus, Kapitel 6.2.
- *A_pv_simmodel_verifikation_alleParam* führt die Simulation zur Verifikation des physikalischen Simulationsmodells aus, wobei alle Parameter als bekannt bzw. gemessen angenommen werden, Kapitel 6.3.
- *A_pv_simmodel_verifikation_GenStrahlungAbsch.m* führt die Simulation zur Verifizierung des physikalischen Simulationsmodells aus, wobei die Abschätzung der Strahlung auf die Peaneelebene gemacht wird, Kapitel 6.3.
- *A_pv_simmodel_verifikation_PTempAbsch.m* führt die Simulation zur Verifizierung des physikalischen Simulationsmodells aus, wobei die Abschätzung der Paneeltemperatur gemacht wird, Kapitel 6.3.
- *A_pv_simmodel_verifikation_PTempGenStrahlungAbsch.m* führt die Simulation zur Verifizierung des physikalischen Simulationsmodells aus, wobei die Abschätzung der Paneeltemperatur und der Strahlung auf die Peaneelebene gemacht wird, Kapitel 6.3.
- *A_ungemesseneAnlage_normjahressummen_villach.m* führt die Simulation zur Bildung der Normjahressummen einer nicht-nachgeführten Anlage für den Standort *Villach* aus, Kapitel 6.4.
- *A_ungemesseneAnlage_normjahressummen_villach_nachgefuehrt.m* führt die Simulation zur Bildung der Normjahressummen einer nachgeführten Anlage für den Standort *Villach* aus, Kapitel 6.4.
- *A_ungemesseneAnlage_normjahressummen_wolfsberg.m* führt die Simulation zur Bildung der Normjahressummen einer nicht-nachgeführten Anlage für den Standort *Wolfsberg* aus, Kapitel 6.4.
- *A_ungemesseneAnlage_normjahressummen_wolfsberg_nachgefuehrt.m* führt die Simulation zur Bildung der Normjahressummen einer nachgeführten Anlage für den Standort *Wolfsberg* aus, Kapitel 6.4.
- *A_ungemesseneAnlagen.m* führt die Simulation zur Bildung der Jahressummen und Suche der Drehung und Neigung bestimmter ausgesuchter Anlagen aus (Schnitt der Ebenen...), Kapitel 6.4.
- *A_ungemesseneAnlagen_nachgefuehrt.m* führt die Simulation zur Bildung der Jahressummen und Suche der Neigung bestimmter ausgesuchter nachgeführter Anlagen aus (Schnitt der Kurven...), Kapitel 6.4.
- *solarstrahlung_wolfsberg.m* plottet die Solarstrahlungsverläufe der Jahre 2007, 2008, 2009 und 2010 der Solarstrahlungsmessstation *Wolfsberg*, Kapitel 6.4.

Anhang B

Anhang, physikalische Modellierung

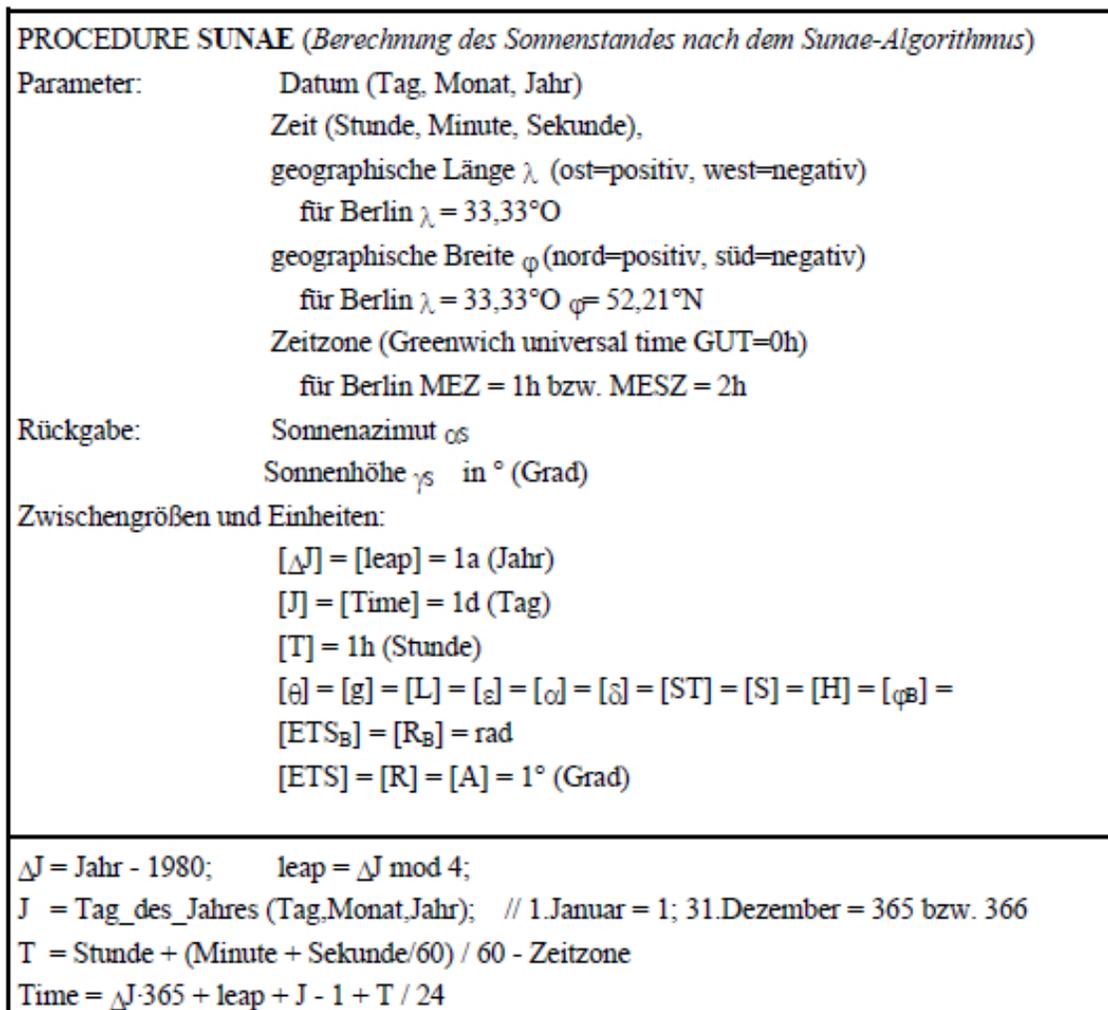


Abbildung B.1: Strukturgramm des SUNAE Algorithmus, [Qua96, Anhang], Teil 1.

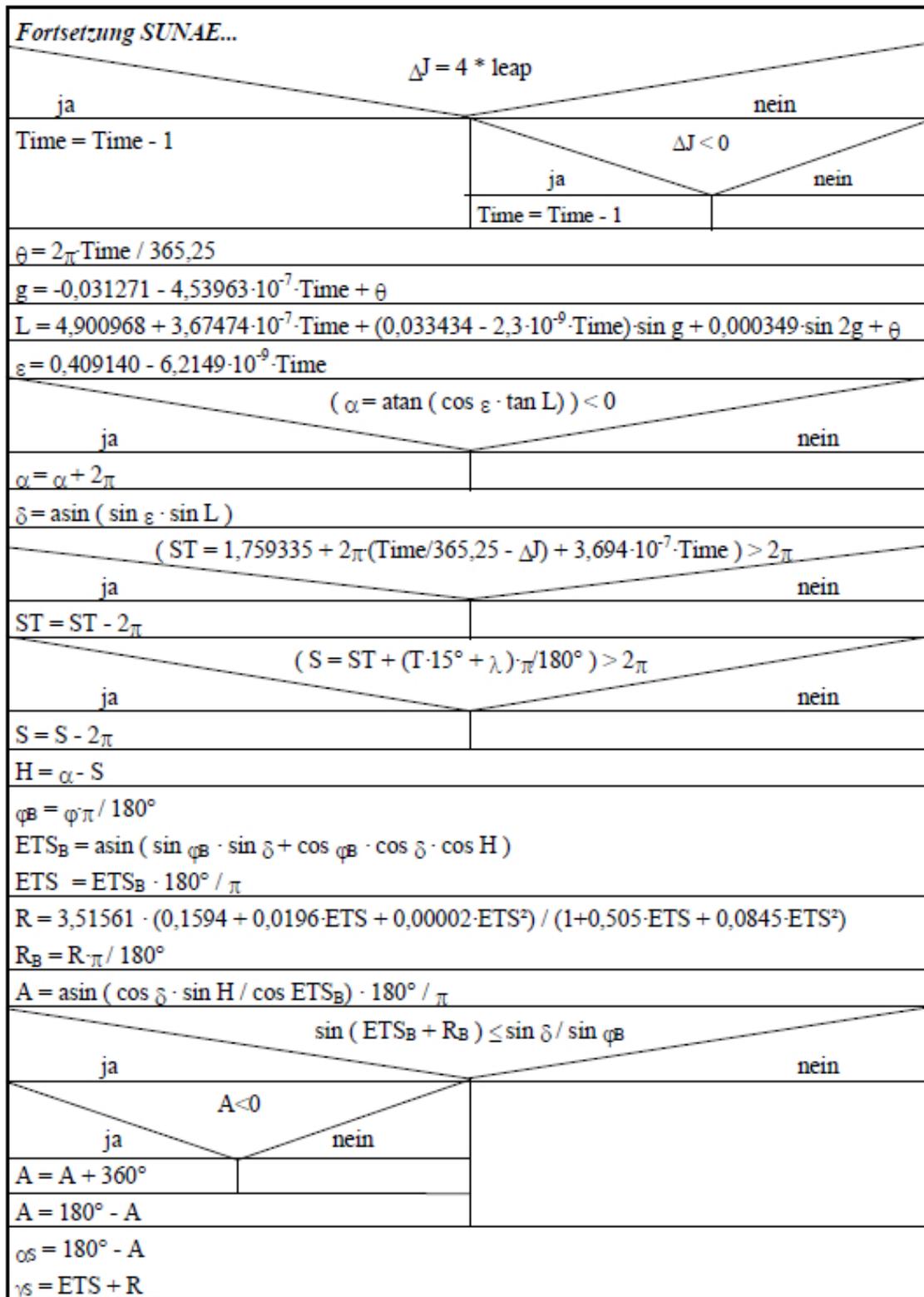


Abbildung B.2: Strukturgramm des SUNAE Algorithmus, [Qua96, Anhang], Teil 2.

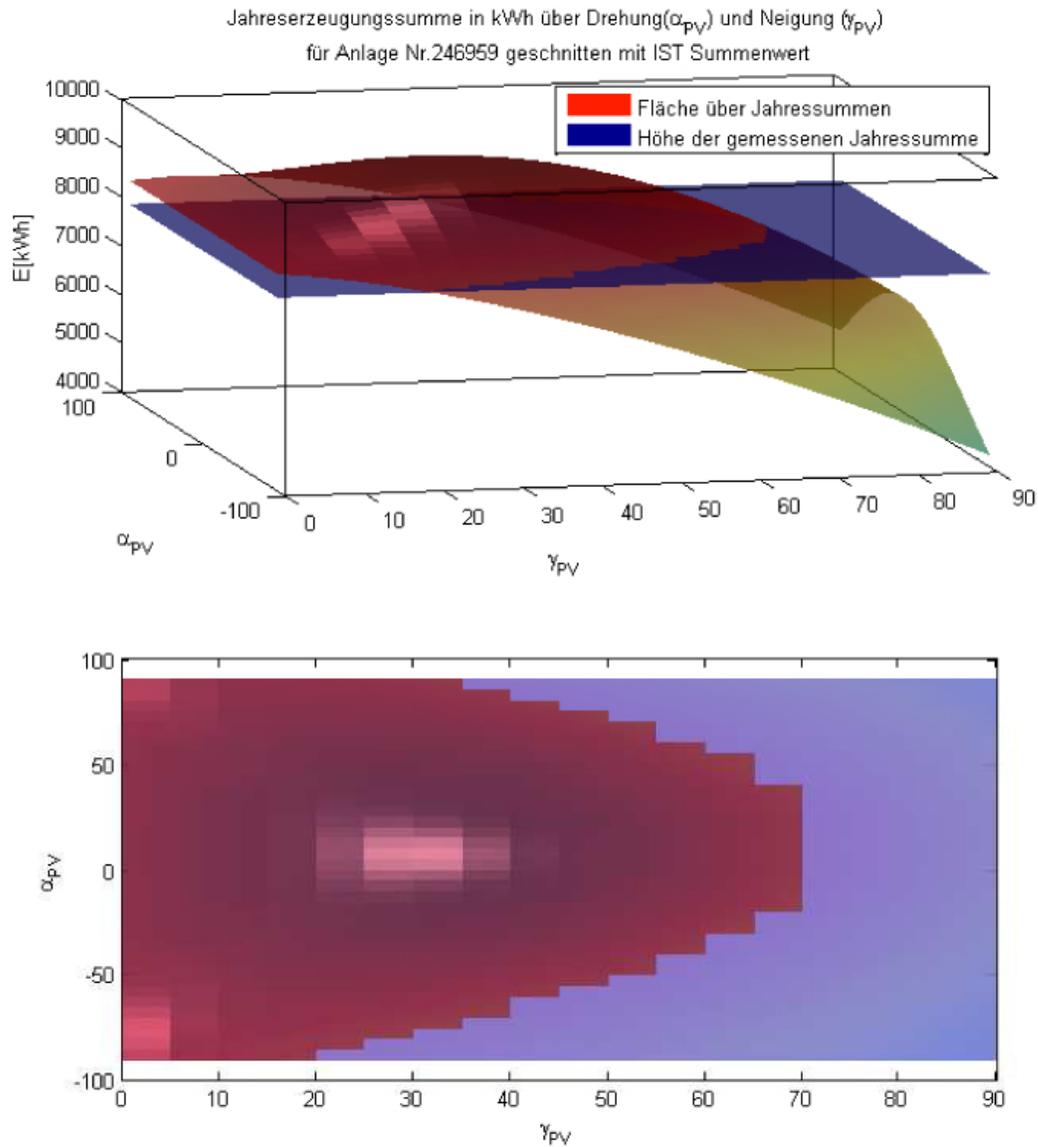


Abbildung B.3: Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 246959 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

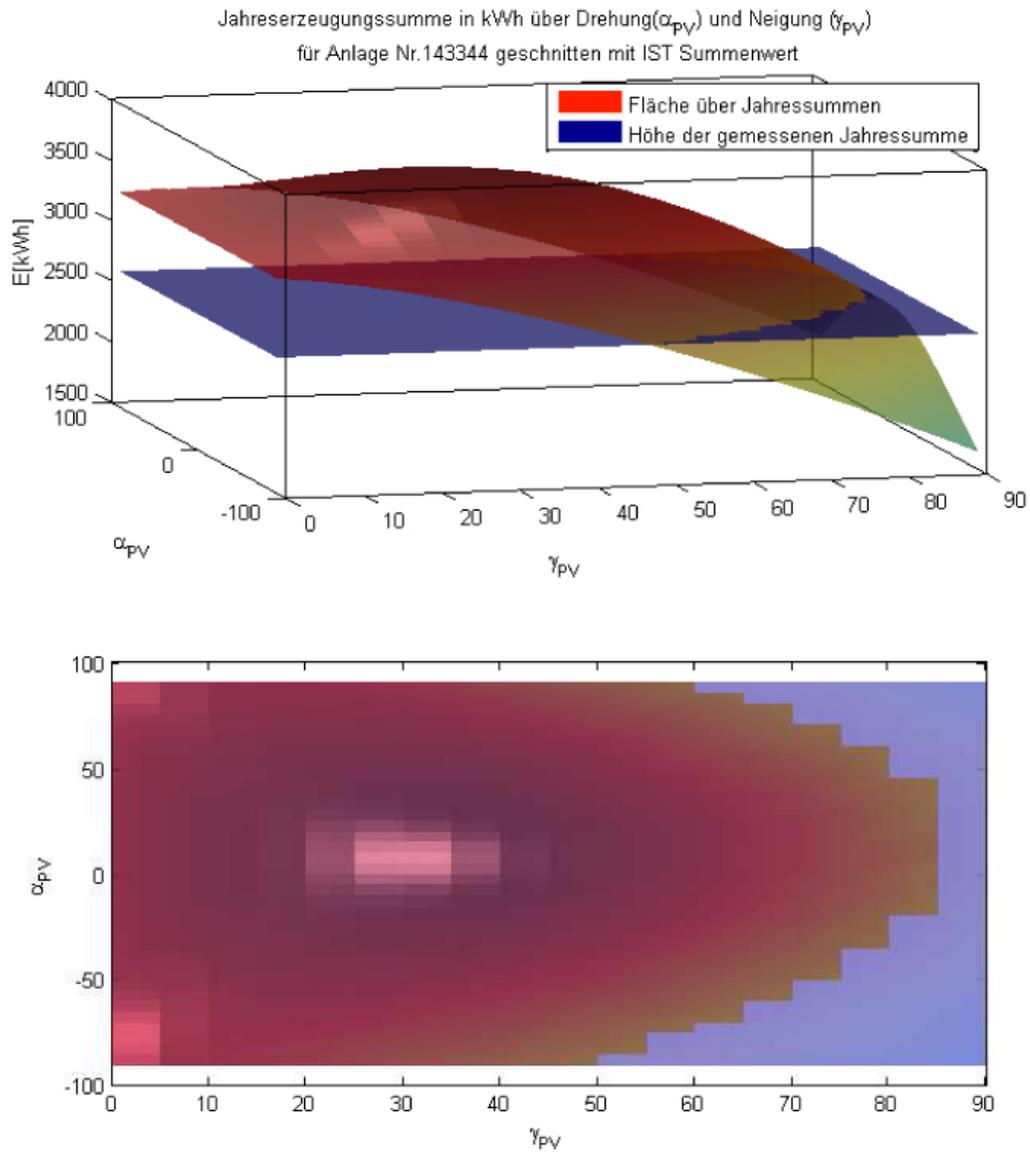


Abbildung B.4: Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 143344 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

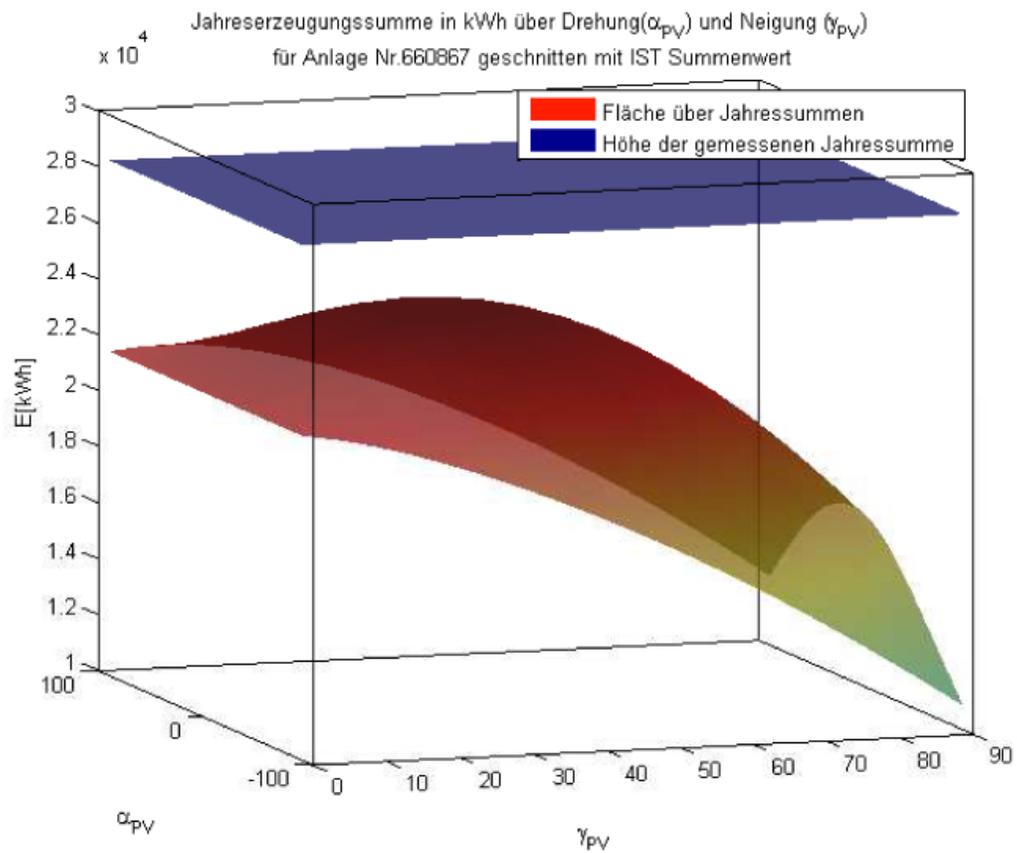


Abbildung B.5: Sklierte $1kW - Peak$ Jahressummenwerte mit dem $kW - Peak$ Wert, in Abhängigkeit der Drehung und der Neigung der Anlage Nr. 660867 für 2007, geschnitten mit der Fläche der Höhe des Jahressummenwertes 2007.

Anhang C

Anhang, regressionsbasierte Modellierung

Nr	1. Vorher sagewert	fein aufgelöst	Langzeitfenster	Verg.-fester	α	Hidden-layeranzahl	min(MSE)	Zeit Training[s]	<i>eg</i>
1	1	1	150	50	0.2	1	0.020924	27.75	55.061
2	0	1	100	50	0.2	1	0.021629	31.793	54.321
3	1	0	150	50	0.2	3	0.021901	65.869	70.296
4	0	1	50	50	0.2	3	0.02223	69.144	62.464
5	0	1	100	50	0.2	2	0.022265	46.757	62.473
6	1	1	150	50	0.2	2	0.022406	48.297	60.914
7	0	1	150	50	0.2	3	0.022471	74.811	72.541
8	1	1	50	50	0.2	2	0.022518	48.415	68.371
9	0	1	200	50	0.2	2	0.022528	52.501	65.93
10	1	1	50	50	0.2	3	0.022549	66.998	65.084
11	1	0	150	40	0.2	3	0.02257	66.373	73.452
12	1	1	100	50	0.2	1	0.022746	27.151	63.337
13	0	1	100	40	0.2	3	0.022781	64.997	71.346
14	0	1	50	50	0.2	1	0.022823	26.934	58.019
15	0	0	50	50	0.2	2	0.022839	49.452	68.532
16	0	0	100	40	0.2	3	0.022851	73.738	68.011
17	0	1	150	40	0.2	2	0.022945	47.938	70.904
18	1	0	50	50	0.2	3	0.022949	67.521	64.549
19	0	0	200	50	0.2	2	0.022949	51.739	61.217
20	1	0	100	50	0.2	2	0.022964	48.101	62.821
21	1	0	150	50	0.2	2	0.022995	47.723	65.511
22	1	1	200	50	0.2	3	0.02302	66.893	71.579
23	0	0	200	40	0.2	3	0.023027	70.2	65.685
24	1	1	100	50	0.2	2	0.023055	48.044	67.436
25	0	1	200	50	0.2	1	0.023081	26.041	62.928

26	1	1	200	50	0.2	2	0.023202	43.671	62.028
27	1	1	200	40	0.2	3	0.023407	67.219	63.314
28	0	1	50	50	0.2	2	0.023427	43.864	57.529
29	0	1	100	50	0.2	3	0.023465	66.796	67.596
30	0	1	100	40	0.2	1	0.02351	28.604	58.873
31	0	1	100	40	0.2	2	0.02359	46.753	68.533
32	0	1	200	40	0.2	1	0.023601	28.308	60.085
33	0	1	150	50	0.2	1	0.023628	29.853	56.21
34	1	0	200	40	0.2	3	0.023777	65.725	71.921
35	0	1	200	50	0.2	3	0.023783	62.355	64.068
36	1	1	150	50	0.2	3	0.023818	60.201	68.345
37	1	0	200	50	0.2	3	0.023836	62.044	67.846
38	1	0	100	30	0.2	3	0.023837	73.553	65.311
39	1	0	50	40	0.2	3	0.023872	67.76	62.58
40	0	0	200	50	0.2	3	0.024024	65.062	62.589
41	1	0	50	40	0.2	2	0.024033	47.019	61.978
42	0	1	150	50	0.2	2	0.02406	44.284	63.685
43	0	1	100	30	0.2	3	0.024098	70.214	64.424
44	1	1	100	40	0.2	3	0.024115	61.678	68.905
45	1	1	100	30	0.2	3	0.024149	67.143	72.98
46	0	0	150	40	0.2	3	0.024169	62.04	66.366
47	1	1	50	50	0.2	1	0.024174	27.905	56.112
48	1	1	50	30	0.2	3	0.024242	68.929	70.986
49	0	1	200	40	0.2	2	0.024246	50.609	70.471
50	1	0	50	50	0.2	2	0.024255	47.71	69.61
51	0	0	150	50	0.2	3	0.024305	62.26	62.687
52	0	0	150	50	0.2	1	0.024403	28.937	57.938
53	1	1	200	40	0.2	1	0.024417	28.858	59.74
54	0	1	50	40	0.2	3	0.024457	65.426	73.16
55	1	1	200	50	0.2	1	0.024508	27.247	59.738
56	0	1	150	40	0.2	3	0.024515	69.581	69.471
57	1	0	200	30	0.2	3	0.024533	65.967	72.119
58	1	1	50	30	0.2	2	0.02454	46.789	67.498
59	1	0	200	50	0.2	2	0.024576	42.376	61.529
60	0	0	50	50	0.2	3	0.024592	58.761	69.423
61	0	0	200	30	0.2	3	0.024654	68.917	74.341
62	1	0	150	40	0.2	2	0.024675	44.918	64.968
63	1	1	50	40	0.2	3	0.024723	57.945	62.925
64	1	1	100	40	0.2	2	0.024729	45.969	64.137
65	0	0	150	50	0.2	2	0.024739	43.7	61.885
66	0	1	50	40	0.2	1	0.024748	29.344	59.519
67	0	0	200	50	0.2	1	0.024759	27.538	61.698
68	0	1	50	30	0.2	3	0.024803	60.404	68.025

69	1	0	200	40	0.2	2	0.024867	47.405	67.569
70	0	0	100	40	0.2	2	0.024924	48.339	66.826
71	1	1	150	40	0.2	3	0.02493	60.789	72.735
72	0	0	150	40	0.2	2	0.025023	42.037	59.696
73	0	0	50	30	0.2	3	0.025031	65.784	69.387
74	1	1	50	40	0.2	2	0.025085	40.565	69.551
75	1	1	150	40	0.2	2	0.025105	45.912	61.093
76	0	0	100	50	0.2	1	0.025153	26.207	59.613
77	0	0	50	50	0.2	1	0.025205	28.125	59.48
78	1	1	100	20	0.2	3	0.025213	65.946	70.743
79	1	0	100	40	0.2	2	0.025256	43.597	71.621
80	0	1	100	20	0.2	3	0.025279	69.839	71.432
81	0	0	200	40	0.2	2	0.025294	47.853	63.402
82	1	0	150	50	0.2	1	0.025311	26.077	54.615
83	1	1	200	30	0.2	3	0.025393	59.756	68.93
84	1	1	200	20	0.2	3	0.025422	66.777	69.531
85	0	1	150	30	0.2	3	0.025498	58.773	69.791
86	1	1	150	40	0.2	1	0.025518	27.983	60.111
87	1	0	200	20	0.2	3	0.025557	65.982	67.69
88	0	0	100	30	0.2	2	0.025612	49.936	70.959
89	1	0	200	30	0.2	2	0.025634	46.21	66.506
90	1	0	50	30	0.2	2	0.02565	46.342	64.134
91	1	0	100	40	0.2	3	0.025671	59.36	67.604
92	1	0	50	20	0.2	3	0.025733	66.093	65.675
93	0	1	50	40	0.2	2	0.025751	43.046	65.747
94	1	1	150	30	0.2	2	0.025766	43.912	62.289
95	1	1	50	40	0.2	1	0.025773	26.664	60.156
96	1	0	100	50	0.2	1	0.025786	23.612	60.789
97	1	1	50	20	0.2	3	0.025808	66.785	68.379
98	0	1	150	30	0.2	2	0.02586	44.841	63.783
99	1	1	150	30	0.2	3	0.0259	58.893	69.624
100	0	0	150	30	0.2	3	0.025919	70.271	62.79

Tabelle C.1: Eigenschaften der vermeindlich besten 100 regulierten Netze nach dem ersten Simulationsdurchgang.

Nr	1. Vorhersagewert	fein aufgelöst	Langzeitfenster	Vergleichsfenster	Hidden layer Anz.	Hidden n	min (MSE)	Zeit Training [s]	<i>eg</i>
1	1	1	150	50	0.2	1	0.020924	27.75	55.061
2	0	1	100	50	0.2	1	0.021629	31.793	54.321
3	1	0	150	50	0.2	3	0.021901	65.869	70.296
4	0	1	50	50	0.2	3	0.02223	69.144	62.464

5	0	1	100	50	0.2	2	0.022265	46.757	62.473
6	1	1	150	50	0.2	2	0.022406	48.297	60.914
7	0	1	150	50	0.2	3	0.022471	74.811	72.541
8	1	1	50	50	0.2	2	0.022518	48.415	68.371
9	0	1	200	50	0.2	2	0.022528	52.501	65.93
10	1	1	50	50	0.2	3	0.022549	66.998	65.084
11	1	0	150	40	0.2	3	0.02257	66.373	73.452
12	1	1	100	50	0.2	1	0.022746	27.151	63.337
13	0	1	100	40	0.2	3	0.022781	64.997	71.346
14	0	1	50	50	0.2	1	0.022823	26.934	58.019
15	0	0	50	50	0.2	2	0.022839	49.452	68.532
16	0	0	100	40	0.2	3	0.022851	73.738	68.011
17	0	1	150	40	0.2	2	0.022945	47.938	70.904
18	1	0	50	50	0.2	3	0.022949	67.521	64.549
19	0	0	200	50	0.2	2	0.022949	51.739	61.217
20	1	0	100	50	0.2	2	0.022964	48.101	62.821
21	1	0	150	50	0.2	2	0.022995	47.723	65.511
22	1	1	200	50	0.2	3	0.02302	66.893	71.579
23	0	0	200	40	0.2	3	0.023027	70.2	65.685
24	1	1	100	50	0.2	2	0.023055	48.044	67.436
25	0	1	200	50	0.2	1	0.023081	26.041	62.928
26	1	1	200	50	0.2	2	0.023202	43.671	62.028
27	1	1	200	40	0.2	3	0.023407	67.219	63.314
28	0	1	50	50	0.2	2	0.023427	43.864	57.529
29	0	1	100	50	0.2	3	0.023465	66.796	67.596
30	0	1	100	40	0.2	1	0.02351	28.604	58.873
31	0	1	100	40	0.2	2	0.02359	46.753	68.533
32	0	1	200	40	0.2	1	0.023601	28.308	60.085
33	0	1	150	50	0.2	1	0.023628	29.853	56.21
34	1	0	200	40	0.2	3	0.023777	65.725	71.921
35	0	1	200	50	0.2	3	0.023783	62.355	64.068
36	1	1	150	50	0.2	3	0.023818	60.201	68.345
37	1	0	200	50	0.2	3	0.023836	62.044	67.846
38	1	0	100	30	0.2	3	0.023837	73.553	65.311
39	1	0	50	40	0.2	3	0.023872	67.76	62.58
40	0	0	200	50	0.2	3	0.024024	65.062	62.589
41	1	0	50	40	0.2	2	0.024033	47.019	61.978
42	0	1	150	50	0.2	2	0.02406	44.284	63.685
43	0	1	100	30	0.2	3	0.024098	70.214	64.424
44	1	1	100	40	0.2	3	0.024115	61.678	68.905
45	1	1	100	30	0.2	3	0.024149	67.143	72.98
46	0	0	150	40	0.2	3	0.024169	62.04	66.366
47	1	1	50	50	0.2	1	0.024174	27.905	56.112

48	1	1	50	30	0.2	3	0.024242	68.929	70.986
49	0	1	200	40	0.2	2	0.024246	50.609	70.471
50	1	0	50	50	0.2	2	0.024255	47.71	69.61
51	0	0	150	50	0.2	3	0.024305	62.26	62.687
52	0	0	150	50	0.2	1	0.024403	28.937	57.938
53	1	1	200	40	0.2	1	0.024417	28.858	59.74
54	0	1	50	40	0.2	3	0.024457	65.426	73.16
55	1	1	200	50	0.2	1	0.024508	27.247	59.738
56	0	1	150	40	0.2	3	0.024515	69.581	69.471
57	1	0	200	30	0.2	3	0.024533	65.967	72.119
58	1	1	50	30	0.2	2	0.02454	46.789	67.498
59	1	0	200	50	0.2	2	0.024576	42.376	61.529
60	0	0	50	50	0.2	3	0.024592	58.761	69.423
61	0	0	200	30	0.2	3	0.024654	68.917	74.341
62	1	0	150	40	0.2	2	0.024675	44.918	64.968
63	1	1	50	40	0.2	3	0.024723	57.945	62.925
64	1	1	100	40	0.2	2	0.024729	45.969	64.137
65	0	0	150	50	0.2	2	0.024739	43.7	61.885
66	0	1	50	40	0.2	1	0.024748	29.344	59.519
67	0	0	200	50	0.2	1	0.024759	27.538	61.698
68	0	1	50	30	0.2	3	0.024803	60.404	68.025
69	1	0	200	40	0.2	2	0.024867	47.405	67.569
70	0	0	100	40	0.2	2	0.024924	48.339	66.826
71	1	1	150	40	0.2	3	0.02493	60.789	72.735
72	0	0	150	40	0.2	2	0.025023	42.037	59.696
73	0	0	50	30	0.2	3	0.025031	65.784	69.387
74	1	1	50	40	0.2	2	0.025085	40.565	69.551
75	1	1	150	40	0.2	2	0.025105	45.912	61.093
76	0	0	100	50	0.2	1	0.025153	26.207	59.613
77	0	0	50	50	0.2	1	0.025205	28.125	59.48
78	1	1	100	20	0.2	3	0.025213	65.946	70.743
79	1	0	100	40	0.2	2	0.025256	43.597	71.621
80	0	1	100	20	0.2	3	0.025279	69.839	71.432
81	0	0	200	40	0.2	2	0.025294	47.853	63.402
82	1	0	150	50	0.2	1	0.025311	26.077	54.615
83	1	1	200	30	0.2	3	0.025393	59.756	68.93
84	1	1	200	20	0.2	3	0.025422	66.777	69.531
85	0	1	150	30	0.2	3	0.025498	58.773	69.791
86	1	1	150	40	0.2	1	0.025518	27.983	60.111
87	1	0	200	20	0.2	3	0.025557	65.982	67.69
88	0	0	100	30	0.2	2	0.025612	49.936	70.959
89	1	0	200	30	0.2	2	0.025634	46.21	66.506
90	1	0	50	30	0.2	2	0.02565	46.342	64.134

91	1	0	100	40	0.2	3	0.025671	59.36	67.604
92	1	0	50	20	0.2	3	0.025733	66.093	65.675
93	0	1	50	40	0.2	2	0.025751	43.046	65.747
94	1	1	150	30	0.2	2	0.025766	43.912	62.289
95	1	1	50	40	0.2	1	0.025773	26.664	60.156
96	1	0	100	50	0.2	1	0.025786	23.612	60.789
97	1	1	50	20	0.2	3	0.025808	66.785	68.379
98	0	1	150	30	0.2	2	0.02586	44.841	63.783
99	1	1	150	30	0.2	3	0.0259	58.893	69.624
100	0	0	150	30	0.2	3	0.025919	70.271	62.79

Tabelle C.2: Eigenschaften der vermeindlich besten 100 nicht regularisierten Netze nach dem ersten Simulationsdurchgang.

Nr	Langzeitfenster	Vergleichsfenster	Hidden-layer Anz.	Hidden n	Akt.-fkt.	min (MSE)	Zeit Training [s]	<i>eg</i>
1	225	40	1	155	1	0.03766	26.111	54.512
2	150	55	1	200	1	0.037698	32.854	59.075
3	225	55	1	170	1	0.037729	29.943	63.081
4	175	50	1	185	1	0.0379	31.288	54.927
5	150	55	1	230	1	0.038227	38.008	58.521
6	225	65	1	185	1	0.038412	31.983	54.98
7	150	40	1	215	1	0.038629	33.842	54.618
8	225	50	1	215	1	0.038753	34.692	49.216
9	175	40	1	155	1	0.03876	25.323	49.947
10	200	65	1	170	1	0.038761	27.975	57.043
11	200	50	1	185	1	0.038989	31.126	54.196
12	150	65	1	140	1	0.039065	21.307	57.511
13	225	50	1	185	1	0.039065	28.399	54.517
14	175	60	1	215	1	0.039133	27.915	55.052
15	225	50	1	230	1	0.039168	34.782	55.934
16	225	45	1	215	1	0.03917	34.973	52.725
17	150	60	1	200	1	0.039182	29.368	55.097
18	200	45	1	230	1	0.039191	46.15	58.718
19	175	60	1	170	1	0.039213	28.681	54.424
20	200	55	1	140	1	0.039339	21.531	58.673
21	175	40	1	185	1	0.039391	32.781	51.716
22	150	60	1	140	1	0.039439	19.83	55.142
23	175	65	1	185	1	0.039505	27.94	56.007
24	150	60	1	230	1	0.039509	31.737	50.503
25	175	50	1	215	1	0.039657	33.457	54.41
26	150	40	1	140	1	0.03966	21.983	52.309

27	225	50	1	155	1	0.039713	22.82	52.753
28	150	55	1	140	1	0.039716	21.157	57.591
29	150	45	1	230	1	0.039719	31.916	48.935
30	175	40	1	230	1	0.03973	35.792	53.194
31	200	60	1	230	1	0.039736	33.072	52.207
32	150	60	1	215	1	0.039752	32.335	51.606
33	200	40	1	200	1	0.039817	39.708	50.425
34	150	50	1	185	1	0.039827	28.023	57.484
35	175	55	1	200	1	0.039863	33.294	59.353
36	200	60	1	215	1	0.039888	32.883	63.128
37	200	55	1	215	1	0.039889	36.561	55.624
38	200	45	1	140	1	0.03993	25.375	51.651
39	225	40	1	215	1	0.039948	39.122	48.001
40	150	40	1	170	1	0.040038	24.839	51.221
41	225	65	1	215	1	0.040055	30.817	54.296
42	150	60	1	185	1	0.040089	31.476	59.539
43	175	50	1	140	1	0.040191	21.046	52.94
44	200	50	1	215	1	0.040236	32.259	58.713
45	150	40	1	185	1	0.040272	27.013	48.77
46	175	45	1	185	1	0.040288	31.171	51.021
47	200	65	1	185	1	0.040292	26.422	50.504
48	175	45	1	140	1	0.040308	20.207	56.472
49	200	45	1	170	1	0.040384	31.518	53.986
50	175	40	1	200	1	0.040436	34.573	58.493
51	200	40	1	230	1	0.040457	40.591	52.831
52	150	45	1	155	1	0.040495	22.38	51.793
53	200	40	1	155	1	0.040511	24.427	54.37
54	225	55	1	230	1	0.04052	32.132	50.742
55	150	50	1	155	1	0.04054	23.686	52.272
56	225	45	1	200	1	0.040544	29.31	57.306
57	175	55	1	230	1	0.040617	34.422	53.585
58	175	60	1	200	1	0.040724	27.097	53.527
59	150	50	1	230	1	0.040732	34.747	45.634
60	225	60	1	170	1	0.040745	27.604	56.299
61	175	55	1	170	1	0.040748	27.265	53.505
62	150	55	1	185	1	0.040763	28.99	48.793
63	225	45	1	140	1	0.040787	20.539	50.364
64	225	45	1	230	1	0.040794	42.282	52.175
65	175	45	1	155	1	0.040842	22.176	52.566
66	175	50	1	155	1	0.040879	21.697	56.949
67	200	40	1	170	1	0.040899	30.841	50.195
68	175	40	1	140	1	0.040904	19.72	58.643
69	225	55	1	185	1	0.040975	32.035	54.533

70	200	45	1	200	1	0.041008	28.329	52.184
71	225	60	1	200	1	0.041021	29.493	57.349
72	200	45	1	185	1	0.041034	32.023	55.318
73	150	65	1	170	1	0.041045	22.642	54.161
74	150	45	1	200	1	0.041108	31.439	47.408
75	200	60	1	200	1	0.041125	27.684	57.87
76	200	60	1	185	1	0.041239	30.892	52.728
77	200	50	1	230	1	0.041338	31.985	54.571
78	150	65	1	230	1	0.041496	31.409	51.909
79	225	40	1	185	1	0.041555	32.733	52.592
80	225	40	1	230	1	0.041603	35.06	48.938
81	200	45	1	215	1	0.041628	30.811	48.322
82	225	45	1	185	1	0.04163	27.381	60.937
83	200	65	1	140	1	0.04164	18.694	51.981
84	175	65	1	215	1	0.04167	27.67	49.901
85	200	50	1	200	1	0.041671	31.576	61.19
86	175	55	1	215	1	0.041672	29.416	49.025
87	150	50	1	170	1	0.041699	26.272	56.762
88	175	45	1	200	1	0.041719	33.415	64.462
89	200	60	1	140	1	0.041861	20.219	56.228
90	200	65	1	215	1	0.041947	33.015	55.728
91	200	50	1	155	1	0.041981	21.183	57.742
92	225	45	1	170	1	0.041997	24.169	51.767
93	175	50	1	170	1	0.042055	23.467	54.579
94	225	40	1	140	1	0.042072	19.695	53.229
95	225	65	1	200	1	0.042098	29.399	55.737
96	200	65	1	155	1	0.042103	21.235	59.924
97	200	50	1	140	1	0.042136	20.455	61.484
98	225	50	1	200	1	0.042163	28.538	49.599
99	200	45	1	155	1	0.042222	22.502	56.729
100	200	40	1	140	1	0.04224	20.353	51.929

Tabelle C.3: Eigenschaften der besten 100 nicht regularisierten Netze aus der verfeinerten Simulation. Aktivierungsfunktion am Ausgang: 1=logsig, 2=purelin.

Nr	Langzeitfenster	Vergleichsfenster	Hiddenlayer Anz.	Hidden n	Akt.fkt.	min (MSE)	Zeit Training [s]	eg
1	150	20	1	140	1	0.015573	43.333	19.06
2	50	20	1	100	1	0.015591	32.247	17.158
3	200	20	1	120	1	0.01588	38.763	25.846
4	100	20	1	80	1	0.015991	29.733	27.111

5	100	20	1	100	1	0.016025	31.625	17.359
6	50	20	1	140	1	0.016068	40.468	17.106
7	50	20	1	160	1	0.016111	45.421	23.883
8	50	40	1	160	1	0.016182	48.982	19.498
9	50	40	2	60	1	0.016191	48.974	19.179
10	200	20	1	100	1	0.01626	31.351	24.52
11	50	50	1	60	1	0.016267	32.618	30.025
12	50	50	3	40	1	0.016281	43.7	21.935
13	50	50	1	180	1	0.01637	69.467	25.739
14	150	40	2	40	1	0.016375	32.962	23.089
15	150	20	1	80	1	0.016397	32.166	21.153
16	50	20	1	200	1	0.016407	55.717	21.773
17	200	20	1	160	1	0.016467	47.423	18.772
18	50	20	1	120	1	0.016477	35.806	17.498
19	150	20	1	160	1	0.016512	49.944	19.26
20	100	20	1	160	1	0.016523	45.642	18.763
21	100	40	2	60	1	0.016565	36.976	17.764
22	100	20	1	120	1	0.016577	38.02	27.251
23	150	30	1	180	1	0.016583	53.028	18.157
24	50	50	2	60	1	0.016657	35.487	31.94
25	150	40	1	180	1	0.016662	70.79	31.157
26	150	30	1	160	1	0.016694	55.101	18.913
27	200	20	1	80	1	0.016711	25.616	24.424
28	100	30	1	120	1	0.016732	44.409	26.924
29	100	20	1	180	1	0.016746	49.953	18.562
30	200	30	2	40	1	0.016757	30.95	21.688
31	50	50	2	80	1	0.016759	44.89	20.18
32	100	40	1	160	1	0.016809	47.832	22.355
33	150	30	1	140	1	0.016849	43.2	33.143
34	50	30	1	140	1	0.01685	40.596	19.624
35	150	20	1	60	1	0.016893	19.472	37.849
36	100	40	3	60	1	0.016917	57.976	26.194
37	100	40	1	200	1	0.01692	58.569	18.409
38	100	30	1	160	1	0.016927	48.71	22.883
39	150	30	1	200	1	0.016933	59.732	23.656
40	150	30	2	40	1	0.016949	23.512	43.631
41	200	40	1	80	1	0.016967	28.09	35.324
42	200	40	2	60	1	0.016976	48.073	24.23
43	50	30	1	180	1	0.016981	53.795	18.052
44	50	20	2	80	1	0.016983	50.002	33.845
45	100	30	1	80	1	0.016984	25.681	17.085
46	150	50	1	200	1	0.017114	67.459	24.18
47	200	20	1	200	1	0.017123	54.692	25.061

48	200	30	1	180	1	0.017133	52.612	21.741
49	150	40	1	160	1	0.017154	57.114	18.56
50	50	30	1	200	1	0.017173	66.025	17.7
51	100	60	3	60	1	0.017177	67.073	44.042
52	200	30	1	120	1	0.017182	38.798	29.058
53	100	30	1	100	1	0.017182	30.242	20.617
54	50	60	1	140	1	0.017197	43.005	21.482
55	150	30	2	60	1	0.017236	39.019	22.33
56	100	20	2	40	1	0.017264	26.779	22.776
57	50	40	2	80	1	0.017278	43.928	27.964
58	50	70	2	100	1	0.0173	73.527	25
59	200	40	1	120	1	0.017321	37.99	35.456
60	100	70	2	80	1	0.017339	52.612	19.523
61	100	60	2	40	1	0.017343	31.92	25.341
62	100	20	1	140	1	0.017349	39.395	18.44
63	200	40	2	40	1	0.017375	23.081	24.69
64	50	30	2	20	1	0.017389	18.166	38.291
65	200	30	1	140	1	0.017417	42.165	21.082
66	200	50	2	40	1	0.01744	33.523	31.522
67	100	50	1	180	1	0.017456	53.773	18.639
68	50	60	1	180	1	0.017501	63.669	18.719
69	100	40	1	60	1	0.017502	23.676	23.859
70	50	60	1	120	1	0.017502	36.007	28.455
71	150	50	1	160	1	0.01751	48.567	22.684
72	50	40	1	120	1	0.017513	38.624	33.431
73	150	30	1	60	1	0.017513	26.947	40.276
74	200	30	2	60	1	0.017525	32.294	23.164
75	50	20	1	180	1	0.017535	46.165	19.784
76	50	20	3	40	1	0.017557	28.982	29.946
77	50	60	3	40	1	0.017576	40.584	34.359
78	200	40	1	100	1	0.017592	35.293	20.86
79	150	50	1	80	1	0.0176	33.52	29.259
80	50	40	2	40	1	0.017606	26.986	21.691
81	100	70	1	100	1	0.017612	47.427	46.593
82	50	30	3	20	1	0.017618	20.002	28.514
83	200	70	1	160	1	0.017649	52.119	38.142
84	200	60	3	60	1	0.017662	64.165	27.773
85	100	70	2	100	1	0.017682	56.732	22.189
86	200	70	2	120	1	0.017707	80.114	20.752
87	50	50	1	160	1	0.017722	49.662	21.026
88	50	30	1	60	1	0.017731	21.727	27.372
89	50	40	1	100	1	0.017745	31.074	24.673
90	200	20	1	180	1	0.017763	53.911	23.726

91	100	40	2	40	1	0.017769	23.833	23.275
92	50	40	1	80	1	0.017775	24.974	22.532
93	100	60	1	180	1	0.017792	51.7	18.288
94	200	20	2	40	1	0.017793	22.426	19.452
95	150	20	1	180	1	0.017805	53.047	23.648
96	150	30	2	80	1	0.017806	45.361	23.129
97	200	20	2	80	1	0.017826	41.732	47.889
98	150	50	2	80	1	0.017841	45.353	28.715
99	150	70	2	40	1	0.017849	32.743	26.726
100	100	30	3	80	1	0.01785	62.787	25.764

Tabelle C.4: Eigenschaften der besten 100 nicht regularisierten Netze aus der Simulation mit dem aktuellen Leistungswert als Input. Aktivierungsfunktion am Ausgang ist in dieser Simulation konstant 1=logsig.

Nr	Langzeitfenster	Vergleichsfenster	Hiddenlayer Anz.	Hidden n	Akt.fkt.	min (MSE)	Zeit Training [s]	<i>eg</i>
1	150	20	1	120	1	0.015555	34.187	23.84
2	50	20	1	90	1	0.015735	27.351	22.924
3	50	20	1	90	1	0.015774	26.27	18.972
4	200	20	1	150	1	0.015809	42.474	20.97
5	50	20	1	120	1	0.015869	33.566	18.519
6	50	20	1	60	1	0.016078	21.017	21.031
7	100	20	1	150	1	0.016086	42.135	18.397
8	50	20	1	180	1	0.016172	50.025	19.487
9	200	20	1	90	1	0.016234	26.233	19.522
10	50	20	1	60	1	0.016255	18.548	21.938
11	100	20	1	180	1	0.016346	47.18	18.981
12	150	20	1	90	1	0.016376	26.315	19.33
13	50	30	1	180	1	0.01651	49.352	19.844
14	200	20	1	150	1	0.01657	42.702	21.223
15	100	20	1	210	1	0.016617	54.548	19.627
16	50	20	1	240	1	0.016672	59.952	19.644
17	50	20	1	210	1	0.016683	53.47	19.33
18	50	20	1	270	1	0.016687	69.762	18.528
19	150	20	1	210	1	0.016698	57.214	19.356
20	50	20	1	150	1	0.016707	39.438	22.584
21	50	30	1	210	1	0.016717	53.401	20.784
22	50	50	1	180	1	0.016723	51.311	23.129
23	150	40	1	210	1	0.016774	59.414	20.966
24	50	40	2	60	1	0.016829	32.129	22.09

25	50	20	1	240	1	0.016843	59.048	20.316
26	50	20	1	180	1	0.016847	45.905	18.929
27	150	40	1	180	1	0.016936	50.801	22.992
28	50	50	1	240	1	0.016996	65.834	22.131
29	200	20	2	60	1	0.017002	31.763	24.52
30	150	30	1	90	1	0.017004	26.609	22.391
31	50	30	1	300	1	0.017011	74.732	20.968
32	50	30	1	240	1	0.017016	63.631	20.239
33	100	20	1	210	1	0.017049	56.694	23.002
34	100	40	1	120	1	0.017057	34.963	23.531
35	50	20	1	300	1	0.017096	71.331	19.775
36	150	20	1	120	1	0.017111	34.176	19.566
37	150	40	1	150	1	0.017117	43.284	21.877
38	100	20	1	60	1	0.01712	19.477	26.431
39	50	30	1	150	1	0.017126	40.52	19.721
40	100	20	1	240	1	0.017136	59.232	19.182
41	200	30	1	180	1	0.017136	49.765	27.952
42	200	30	1	150	1	0.017139	43.505	23.05
43	150	20	1	180	1	0.017151	56.577	24.538
44	100	20	1	150	1	0.017153	44.795	19.426
45	50	20	2	90	1	0.017159	43.009	23.866
46	50	30	1	240	1	0.017162	61.862	19.703
47	100	70	2	60	1	0.017181	41.541	27.052
48	100	20	1	300	1	0.017207	71.434	20.124
49	100	30	1	270	1	0.017218	71.264	21.934
50	150	30	1	150	1	0.017226	41.729	19.773
51	200	20	1	240	1	0.017254	60.293	20.822
52	200	40	2	60	1	0.017257	40.891	25.168
53	200	20	1	90	1	0.017258	25.701	21.642
54	200	30	1	270	1	0.017263	70.87	20.16
55	50	30	1	270	1	0.017271	68.571	22.646
56	100	30	1	180	1	0.017276	47.928	19.984
57	200	20	1	210	1	0.017291	53.603	20.298
58	200	20	1	240	1	0.017305	62.079	19.871
59	150	30	1	150	1	0.017319	42.749	21.917
60	50	30	1	210	1	0.017323	54.394	19.624
61	150	20	2	60	1	0.017327	31.488	20.316
62	100	20	1	180	1	0.017331	45.814	21.537
63	100	40	1	300	1	0.017356	79.33	21.054
64	50	30	1	150	1	0.017356	40.312	18.693
65	200	30	1	150	1	0.017359	41.309	27.012
66	50	20	1	210	1	0.017359	56.182	23.064
67	150	40	1	270	1	0.017365	72.684	20.789

68	50	40	1	210	1	0.017371	55.928	19.878
69	200	40	1	90	1	0.01739	28.196	28.937
70	150	20	1	150	1	0.017405	41.305	21.31
71	50	30	1	60	1	0.01741	18.05	23.963
72	50	30	2	90	1	0.017413	45.493	23.559
73	50	40	1	150	1	0.017418	41.455	19.453
74	100	40	1	210	1	0.01743	56.335	24.248
75	50	40	1	210	1	0.017439	57.822	19.878
76	150	30	1	60	1	0.01744	18.524	24.956
77	150	20	1	60	1	0.017449	17.638	25.148
78	150	30	1	300	1	0.017484	75.986	19.238
79	150	30	2	60	1	0.017485	31.334	23.621
80	200	40	1	210	1	0.017486	57.552	20.241
81	100	20	1	240	1	0.017494	60.776	19.67
82	100	30	2	90	1	0.0175	43.805	32.528
83	100	40	1	180	1	0.0175	47.775	19.913
84	100	20	1	90	1	0.017502	25.368	22.845
85	100	20	1	270	1	0.017504	66.516	19.435
86	100	40	1	240	1	0.017511	63.737	19.471
87	200	20	1	270	1	0.017531	69.793	20.883
88	100	20	1	270	1	0.017534	67.3	19.138
89	50	40	1	270	1	0.017545	71.461	21.638
90	150	20	1	300	1	0.017576	72.68	19.644
91	100	50	1	120	1	0.017577	34.77	30.836
92	100	50	1	210	1	0.01759	61.275	20.599
93	100	30	2	60	1	0.017601	30.847	23.067
94	200	30	1	210	1	0.017604	54.011	20.195
95	100	30	1	240	1	0.017606	61.571	19.44
96	50	30	1	90	1	0.017637	25.77	21.1
97	200	40	1	150	1	0.017647	41.623	20.948
98	150	40	1	240	1	0.017661	63.976	20.931
99	150	30	1	270	1	0.017672	68.262	19.896
100	200	40	2	120	1	0.017676	60.034	22.939

Tabelle C.5: Eigenschaften der besten 100 nicht regularisierten Netze aus der Simulation mit dem aktuellen und vergangenen Leistungswerten als Input. Aktivierungsfunktion am Ausgang ist in dieser Simulation konstant 1=logsig.

Nr	Hidden-layer Anz.	Hidden n	Akt.-fkt.	Inkl. Wettervorhersage	Zeit Training [s]	eg	MSE
1	1	40	1	1	2391.5	57.576	0.04432

2	2	12	1	1	2729.9	61.286	0.045911
3	2	20	1	0	2761.7	65.591	0.046339
4	2	40	1	1	2768.7	56.715	0.047218
5	1	20	1	1	2394.3	57.653	0.047802
6	2	12	1	0	2694	65.281	0.049635
7	2	20	1	1	2720.1	58.824	0.050749
8	1	12	2	1	2354.6	62.181	0.050895
9	2	12	2	1	2699.6	60.356	0.051432
10	1	12	1	1	2367.4	60.305	0.051435
11	1	20	2	1	2343.8	61.338	0.051782
12	1	40	2	1	2358.3	60.503	0.051946
13	1	40	1	0	2385.5	63.542	0.052984
14	2	40	2	1	2731.4	61.501	0.053536
15	2	20	2	1	2709.7	63.111	0.053646
16	2	20	2	0	2742.3	68.586	0.057135
17	1	20	2	0	2354	69.31	0.058481
18	1	20	1	0	2376.2	69.017	0.058977
19	1	12	1	0	2354.3	68.449	0.059674
20	1	12	2	0	2360.8	69.688	0.060647
21	1	40	2	0	2336	70.05	0.06199
22	2	40	2	0	2744.7	69.809	0.062236
23	2	40	1	0	2826.7	66.667	0.062611
24	2	12	2	0	2699	73.003	0.062742
25	1	40	3	0	2053.2	87.25	0.083705
26	1	40	3	1	2057.9	87.25	0.083705
27	2	40	3	1	2395.7	87.25	0.083705
28	2	40	3	0	2381.5	87.25	0.083705
29	1	20	3	0	2044.8	87.25	0.083705
30	2	20	3	0	2374.1	87.25	0.083705
31	2	20	3	1	2377.1	87.25	0.083705
32	1	12	3	0	2065.3	87.25	0.083705
33	2	12	3	0	2378.5	87.25	0.083705
34	1	12	3	1	2085	87.25	0.083705
35	2	12	3	1	2417	87.25	0.083705
36	1	20	3	1	2087.1	87.259	0.083903

Tabelle C.6: Eigenschaften der 36 Elman Netze, ohne Windowing mit reinem online Training, aufsteigend sortiert nach dem mean squared Error. Aktivierungsfunktion am Ausgang: 1=tansig, 2=purelin, 3=logsig.

Nr	Hidden-layer Anz.	Hidden n	Inkl. Wettervorhersage	Verg.-heitsfenster	Langzeitfenster	Zeit Training [s]	<i>eg</i>	MSE
----	-------------------	----------	------------------------	--------------------	-----------------	-------------------	-----------	-----

1	1	200	1	30	140	2653.2	48.537	0.036855
2	2	200	1	50	140	3614.7	54.236	0.037302
3	2	300	1	70	140	5642.4	52.049	0.03735
4	1	300	1	70	100	3082.1	52.462	0.037455
5	2	200	1	70	140	3676.4	52.118	0.03757
6	2	300	1	30	140	5438.2	51.524	0.037765
7	1	200	1	70	140	2690.4	51.171	0.037863
8	1	300	1	70	140	3089.6	54.873	0.037885
9	2	200	1	70	180	3654.1	53.874	0.03796
10	1	300	1	50	100	3025.1	53.254	0.038541
11	1	200	1	70	180	2742.8	55.045	0.038708
12	1	200	1	50	100	2679.7	52.436	0.038881
13	1	200	1	70	100	2685	52.841	0.038904
14	2	200	1	30	100	3577.6	55.544	0.039001
15	2	300	1	50	100	5521.4	55.312	0.03901
16	2	200	1	70	100	3634	56.009	0.039023
17	1	300	1	30	140	2985.5	53.633	0.039238
18	2	300	1	70	100	5612.7	56.405	0.03927
19	1	300	1	70	180	3123.2	56.551	0.039398
20	1	200	1	30	100	2732.6	53.349	0.039519
21	1	300	1	30	100	2982.5	55.785	0.039614
22	2	200	1	50	100	3610.2	56.534	0.03963
23	2	200	1	30	140	3616.2	52.247	0.0398
24	1	200	1	50	140	2678.8	51.98	0.039816
25	2	200	1	50	180	3603.7	57.748	0.039877
26	1	300	1	50	180	3019.1	56.293	0.039887
27	1	200	1	50	180	2668.8	58.023	0.039994
28	2	300	1	50	140	5543.8	54.296	0.039999
29	2	300	1	30	100	5398.7	56.009	0.040012
30	1	300	1	50	140	3038.1	53.228	0.040102
31	2	200	1	30	180	3578	55.398	0.040492
32	1	300	1	30	180	2985	56.947	0.040598
33	2	300	1	70	180	5573.3	58.867	0.040699
34	1	200	1	30	180	2699	59.229	0.041216
35	2	300	1	50	180	5532.2	56.81	0.041339
36	2	300	1	30	180	5423.1	57.447	0.041711

Tabelle C.7: Eigenschaften der 36 Elman Netze, mit Windowing und online Training, ohne Information über aktuelle und vergangene Leistungswerte, aufsteigend sortiert nach dem mean squared Error. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.

Nr	Hidden-	Hidden	Inkl.	Verg.-	Lang-	Zeit	<i>eg</i>	MSE
----	---------	--------	-------	--------	-------	------	-----------	-----

	layer Anz.	n	Wetter- vorhersage	heits- fenster	zeit- fenster	Training [s]		
1	1	200	1	30	180	2662.7	19.611	0.017753
2	1	200	1	50	180	2700.3	21.358	0.017844
3	1	200	1	30	140	2659.6	19.792	0.017918
4	2	300	1	30	180	5525	21.092	0.018332
5	1	200	1	70	100	2778	22.254	0.018368
6	1	200	1	50	140	2689	20.876	0.018385
7	2	300	1	30	140	5491.4	21.358	0.018526
8	1	300	1	50	180	3104.4	22.366	0.018541
9	2	200	1	30	100	3663.5	20.808	0.018615
10	1	200	1	50	100	2707.4	21.04	0.018773
11	1	200	1	30	100	2671.9	20.911	0.018779
12	2	200	1	30	140	3599.1	23.295	0.018811
13	1	300	1	30	140	2990.4	21.212	0.018831
14	2	200	1	50	100	3702.9	21.97	0.018868
15	2	200	1	30	180	3578.4	21.453	0.018923
16	2	200	1	50	140	3656.3	22.409	0.019095
17	1	300	1	50	100	3100.2	20.687	0.019124
18	1	300	1	30	100	2985	21.763	0.019182
19	1	300	1	50	140	3129.2	23.33	0.019203
20	1	300	1	70	100	3228.5	23.227	0.019287
21	1	300	1	70	180	3192.5	24.372	0.019364
22	1	300	1	30	180	3000.2	23.252	0.019398
23	2	300	1	50	140	5608.7	23.287	0.019415
24	2	300	1	30	100	5451.2	22.228	0.01946
25	2	300	1	50	100	5882.8	22.77	0.019594
26	1	200	1	70	180	2714.9	23.7	0.019626
27	2	200	1	70	180	3719.8	24.638	0.019664
28	1	300	1	70	140	3195.3	23.855	0.019708
29	2	200	1	50	180	3633.9	22.813	0.019739
30	2	300	1	70	180	5775.5	24.423	0.019812
31	2	200	1	70	100	3737.1	23.537	0.020007
32	1	200	1	70	140	2711.5	23.631	0.020026
33	2	300	1	50	180	5619.2	22.977	0.020035
34	2	300	1	70	100	5742.8	24.905	0.020197
35	2	200	1	70	140	3752.1	25.568	0.020668
36	2	300	1	70	140	5756.4	25.611	0.021153

Tabelle C.8: Eigenschaften der 36 Elman Netze, mit Windowing, online Training und mit Information über aktuelle und vergangene Leistungswerte, aufsteigend sortiert nach dem mean squared Error. Aktivierungsfunktion am Ausgang ist die tansig-Funktion.

Nr	Netz Dimension	σ	c	i_c	input scale	inkl. Wetter Vorhersage	Zeit Training [s]	eg	MSE
1	200	0.1	0.2	0.1	2	1	25.624	62.156	0.057002
2	200	0.1	0.1	0.1	2	1	25.645	61.484	0.059218
3	200	0.3	0.1	0.1	2	1	25.614	60.623	0.059271
4	200	0.3	0.2	0.1	2	1	25.583	61.854	0.059577
5	500	0.1	0.5	0.1	2	1	65.32	62.603	0.059632
6	500	0.1	0.15	0.1	2	1	65.33	61.166	0.059715
7	100	0.3	0.5	0.1	1	0	12.866	65.565	0.059903
8	200	0.3	0.5	0.1	2	1	25.582	61.82	0.060576
9	200	0.7	0.1	0.1	2	0	25.04	64.42	0.060612
10	200	0.1	0.5	0.1	2	1	25.609	62.242	0.060751
11	50	0.7	0.1	0.1	2	0	6.6626	63.963	0.060846
12	200	0.99	0.5	0.5	2	0	25.101	64.687	0.061047
13	200	0.5	0.15	0.1	2	1	25.596	60.176	0.06107
14	100	0.5	0.2	0.1	2	0	12.952	65.651	0.061076
15	100	0.3	0.1	0.1	2	0	12.845	63.877	0.061144
16	500	0.7	0.1	0.1	2	1	65.783	61.897	0.061217
17	200	0.3	0.1	0.1	1	1	25.591	61.123	0.061312
18	100	0.1	0.15	0.1	2	0	12.861	65.832	0.061335
19	200	0.5	0.2	0.1	2	1	25.597	62.423	0.061358
20	200	0.1	0.15	0.1	2	1	25.617	61.14	0.06141
21	100	0.5	0.1	0.1	2	0	12.86	66.546	0.06144
22	500	0.9	0.15	0.1	1	0	64.94	63.826	0.061472
23	200	0.5	0.1	0.1	2	1	25.566	60.176	0.061484
24	100	0.7	0.15	0.1	1	0	12.877	66.727	0.061541
25	100	0.1	0.2	0.1	2	1	13.38	60.365	0.061557
26	200	0.1	0.15	0.1	2	0	24.907	64.618	0.061588
27	200	0.3	0.2	0.4	2	0	24.983	63.938	0.061664
28	100	0.1	0.1	0.1	1	0	12.89	65.651	0.061664
29	100	0.5	0.1	0.1	1	0	12.864	65.711	0.061761
30	200	0.7	0.1	0.4	2	0	25.063	67.045	0.061773
31	200	0.7	0.15	0.1	1	0	25.076	65.892	0.061776
32	200	0.5	0.15	0.1	2	0	25.034	63.275	0.06184
33	500	0.5	0.1	0.1	2	1	65.487	60.856	0.061862
34	200	0.1	0.1	0.1	1	1	25.627	61.235	0.061987
35	100	0.7	0.2	0.1	1	0	12.846	66.744	0.062002
36	200	0.9	0.1	0.1	1	0	25.138	63.378	0.062186
37	200	0.9	0.15	0.1	1	0	25.118	64.928	0.062262
38	500	0.99	0.15	0.1	1	0	65.088	61.553	0.062291
39	500	0.5	0.15	0.1	2	1	65.783	62.715	0.062311
40	500	0.99	0.15	0.1	2	1	65.779	63.447	0.062321

41	100	0.5	0.15	0.4	2	0	12.873	65.393	0.062327
42	100	0.3	0.5	0.1	2	1	13.402	62.405	0.062335
43	500	0.99	0.15	0.1	0.5	0	64.998	59.005	0.062354
44	500	0.1	0.1	0.1	1	1	65.339	59.874	0.06242
45	200	0.9	0.15	0.1	2	1	25.59	61.467	0.062437
46	200	0.3	0.2	0.1	1	1	25.577	61.174	0.062465
47	100	0.5	0.15	0.5	2	0	12.89	62.965	0.062528
48	200	0.5	0.1	0.1	2	0	25.019	63.361	0.06254
49	100	0.7	0.5	0.5	2	0	12.885	63.223	0.062677
50	100	0.5	0.5	0.4	2	0	12.884	66.012	0.06272
51	100	0.3	0.1	0.1	1	1	13.367	64.248	0.062763
52	100	0.1	0.2	0.1	1	0	12.851	66.563	0.062786
53	500	0.3	0.1	0.1	2	1	66.035	61.579	0.062786
54	200	0.7	0.5	0.4	1	0	25.109	63.654	0.06288
55	500	0.9	0.1	0.1	2	1	65.644	62.836	0.062907
56	200	0.5	0.15	0.4	2	0	25.053	63.826	0.06296
57	100	0.1	0.15	0.1	1	0	12.851	64.885	0.062987
58	100	0.3	0.1	0.1	2	1	13.38	63.774	0.063025
59	200	0.1	0.1	0.1	2	0	24.901	65.823	0.063028
60	200	0.9	0.1	0.1	1	1	25.584	63.111	0.063047
61	200	0.1	0.2	0.1	1	1	25.606	61.966	0.063188
62	200	0.99	0.5	0.5	1	0	25.109	67.459	0.063189
63	200	0.1	0.5	0.1	2	0	25.023	63.369	0.063211
64	200	0.3	0.15	0.1	2	0	24.987	65.169	0.063221
65	200	0.9	0.2	0.1	2	0	25.098	65.384	0.063222
66	500	0.5	0.2	0.1	2	1	65.243	62.44	0.063236
67	200	0.1	0.15	0.4	2	1	25.637	62.044	0.063245
68	200	0.99	0.15	0.4	1	0	25.134	65.453	0.063324
69	100	0.3	0.15	0.1	2	0	12.951	64.704	0.063325
70	200	0.1	0.5	0.1	1	1	25.595	62.483	0.063336
71	200	0.7	0.15	0.1	2	1	25.598	63.809	0.063338
72	500	0.3	0.15	0.1	2	1	65.105	60.838	0.063355
73	500	0.7	0.2	0.1	2	1	64.678	63.301	0.063409
74	200	0.7	0.2	0.1	2	1	25.601	63.998	0.063421
75	100	0.1	0.2	0.4	2	0	12.864	64.644	0.063462
76	500	0.9	0.2	0.1	1	0	64.846	64.687	0.063498
77	500	0.7	0.5	0.1	2	0	64.638	63.654	0.063502
78	200	0.5	0.5	0.5	2	0	25.026	66.305	0.06357
79	200	0.3	0.15	0.4	2	0	24.963	63.938	0.063585
80	200	0.5	0.5	0.4	1	0	25.064	64.867	0.063591
81	200	0.3	0.15	0.1	1	1	25.59	60.098	0.063601
82	500	0.99	0.2	0.1	1	1	65.655	62.56	0.063607
83	100	0.9	0.1	0.5	2	0	12.881	65.246	0.063614

84	100	0.1	0.5	0.4	1	0	12.858	64.153	0.063632
85	200	0.7	0.15	0.1	2	0	25.042	64.756	0.063642
86	200	0.7	0.5	0.1	0.5	0	25.071	65.272	0.063652
87	500	0.1	0.5	0.1	1	1	65.383	62.07	0.063697
88	200	0.3	0.15	0.1	2	1	25.621	60.838	0.063697
89	100	0.3	0.2	0.1	2	0	12.869	68.543	0.063738
90	50	0.1	0.15	0.1	2	0	7.1221	62.569	0.063746
91	200	0.9	0.5	0.1	1	0	25.074	60.735	0.063759
92	500	0.9	0.5	0.1	2	0	64.572	65.246	0.063774
93	200	0.7	0.1	0.5	2	0	25.046	64.971	0.063868
94	200	0.7	0.15	0.4	2	0	25.108	65.281	0.063926
95	100	0.3	0.1	0.1	1	0	12.854	64.205	0.06398
96	200	0.3	0.5	0.4	2	0	25.053	64.514	0.064055
97	50	0.5	0.15	0.1	2	0	6.7362	67.321	0.064056
98	200	0.1	0.15	0.1	1	1	25.598	62.827	0.064062
99	200	0.99	0.1	0.1	1	0	25.117	66.253	0.064072
100	500	0.3	0.15	0.1	1	1	65.008	60.563	0.064084

Tabelle C.9: Eigenschaften der nach dem mean squared Error gefundenen besten Echo State Netzen, ohne Windowing mit reinem online Training und linearem Readout. Aktivierungsfunktion am Ausgang: purelin.

Nr	N_D	σ_1	σ_2	σ_3	c	i_c	in_s	i_W	pS	mT	Zeit Tr. [s]	eg	MSE
1	250	0.05	0.15	0.3	0.1	0.1	2	1	75	150	67.932	64.437	0.037674
2	250	0.5	0.75	0.99	0.1	0.1	1	1	75	150	68.899	68.767	0.03783
3	250	0.05	0.1	0.9	0.1	0.6	0.8	1	75	150	69.817	63.292	0.038169
4	250	0.05	0.15	0.3	0.1	0.4	1	1	75	200	69.231	66.124	0.038356
5	250	0.05	0.1	0.9	0.2	0.1	1	1	75	200	69.278	62.698	0.039218
6	100	0.1	0.9	0.99	0.1	0.1	2	1	75	200	27.785	63.266	0.039325
7	250	0.1	0.9	0.99	0.1	0.1	0.8	1	75	150	69.068	59.978	0.039334
8	250	0.1	0.9	0.99	0.2	0.1	1	1	75	150	68.627	63.447	0.039511
9	250	0.1	0.9	0.99	0.2	0.6	0.8	1	40	150	69.173	61.183	0.039632
10	250	0.5	0.75	0.99	0.2	0.1	0.8	1	75	200	68.851	69.112	0.039659
11	250	0.05	0.15	0.3	0.1	0.1	0.8	1	75	150	68.086	70.377	0.039707
12	100	0.05	0.1	0.9	0.2	0.1	2	1	75	150	27.511	58.772	0.039955
13	250	0.1	0.9	0.99	0.2	0.1	2	1	75	200	68.71	69.508	0.040043
14	250	0.1	0.9	0.99	0.1	0.4	0.8	1	75	150	69.712	66.374	0.040097
15	250	0.05	0.15	0.3	0.1	0.1	1	1	75	200	68.454	71.591	0.040141
16	250	0.5	0.75	0.99	0.2	0.1	1	1	75	150	68.505	70.859	0.040387
17	250	0.05	0.5	0.99	0.1	0.4	0.8	1	75	200	69.602	57.223	0.040505

18	250	0.05	0.5	0.99	0.1	0.4	1	1	75	150	69.558	66.116	0.040611
19	250	0.1	0.9	0.99	0.1	0.4	0.8	1	75	200	69.557	66.787	0.040637
20	250	0.05	0.15	0.3	0.2	0.4	1	1	75	200	69.53	63.834	0.040688
21	250	0.05	0.5	0.99	0.2	0.1	2	1	40	150	84.059	66.012	0.040728
22	250	0.05	0.15	0.3	0.2	0.6	2	1	75	150	69.813	64.997	0.040748
23	250	0.05	0.5	0.99	0.2	0.1	1	1	75	150	68.953	71.97	0.040754
24	250	0.05	0.5	0.99	0.2	0.6	0.8	1	75	200	70.517	65.651	0.040754
25	250	0.5	0.75	0.99	0.1	0.4	1	1	75	150	69.551	68.001	0.040912
26	250	0.1	0.9	0.99	0.2	0.4	0.8	1	40	150	93.755	67.949	0.040913
27	250	0.1	0.9	0.99	0.1	0.1	1	1	75	150	68.49	70.764	0.041036
28	250	0.05	0.15	0.3	0.1	0.6	0.8	1	40	200	104.18	64.325	0.041147
29	250	0.05	0.1	0.9	0.2	0.1	2	1	75	200	69.362	71.798	0.041171
30	100	0.05	0.5	0.99	0.1	0.1	1	1	40	200	30.137	66.167	0.04124
31	100	0.5	0.75	0.99	0.2	0.1	0.8	1	75	150	27.689	70.007	0.041295
32	250	0.5	0.75	0.99	0.1	0.6	1	1	75	150	71.761	61.303	0.0414
33	250	0.1	0.9	0.99	0.2	0.4	1	1	40	150	87.538	63.464	0.04143
34	250	0.5	0.75	0.99	0.1	0.6	1	1	40	150	85.168	66.288	0.04144
35	250	0.1	0.9	0.99	0.2	0.1	0.8	1	75	150	68.674	61.226	0.041463
36	250	0.05	0.1	0.9	0.2	0.4	0.8	1	75	200	69.383	69.43	0.041491
37	250	0.05	0.5	0.99	0.2	0.6	2	1	75	150	69.688	69.499	0.041531
38	250	0.5	0.75	0.99	0.2	0.1	2	1	75	200	68.906	70.67	0.041567
39	250	0.5	0.75	0.99	0.1	0.1	0.8	1	75	200	69.989	69.258	0.041573
40	250	0.05	0.1	0.9	0.2	0.6	2	1	75	150	69.832	66.314	0.041722
41	250	0.05	0.15	0.3	0.2	0.4	1	1	75	150	69.057	70.291	0.041774
42	250	0.05	0.15	0.3	0.1	0.4	0.8	1	75	200	69.076	72.004	0.041793
43	250	0.1	0.9	0.99	0.2	0.1	1	1	40	150	73.641	67.175	0.041798
44	100	0.5	0.75	0.99	0.2	0.1	0.8	1	75	200	28.089	68.982	0.041807
45	250	0.5	0.75	0.99	0.2	0.6	1	1	75	200	69.704	69.723	0.041812
46	250	0.05	0.1	0.9	0.1	0.1	0.8	1	75	200	69.805	72.788	0.041873
47	100	0.05	0.1	0.9	0.2	0.1	2	1	40	150	29.391	66.994	0.041889
48	250	0.5	0.75	0.99	0.2	0.4	0.8	1	40	150	84.018	67.82	0.041913
49	250	0.1	0.9	0.99	0.2	0.4	2	1	40	200	70.281	65.117	0.041932
50	250	0.05	0.5	0.99	0.2	0.4	2	1	75	150	69.174	67.493	0.041966
51	250	0.1	0.9	0.99	0.1	0.1	1	1	75	200	68.713	69.396	0.042017
52	250	0.05	0.1	0.9	0.2	0.6	2	1	40	150	71.615	66.512	0.04203
53	100	0.05	0.5	0.99	0.1	0.1	0.8	1	75	150	27.652	63.025	0.042069
54	250	0.05	0.5	0.99	0.1	0.6	1	1	75	150	70.278	68.604	0.042104
55	250	0.05	0.15	0.3	0.2	0.4	1	1	40	200	69.479	67.889	0.042107
56	250	0.5	0.75	0.99	0.1	0.1	1	1	75	200	68.758	63.335	0.042117
57	250	0.05	0.1	0.9	0.1	0.6	0.8	1	40	200	69.666	64.187	0.042124
58	100	0.5	0.75	0.99	0.2	0.1	1	1	40	150	28.363	61.217	0.042169
59	250	0.05	0.5	0.99	0.2	0.6	0.8	1	40	200	69.721	63.628	0.042176
60	250	0.05	0.15	0.3	0.1	0.6	1	1	75	150	69.211	62.56	0.042195

61	250	0.5	0.75	0.99	0.2	0.4	1	1	75	150	69.222	63.567	0.042222
62	250	0.5	0.75	0.99	0.2	0.6	0.8	1	75	150	70.108	67.657	0.042231
63	250	0.05	0.15	0.3	0.1	0.6	2	1	75	150	69.893	72.116	0.042264
64	250	0.5	0.75	0.99	0.1	0.1	2	1	40	150	68.397	73.158	0.042337
65	250	0.05	0.1	0.9	0.1	0.6	1	1	40	200	69.673	67.459	0.042393
66	100	0.05	0.15	0.3	0.2	0.1	1	1	75	150	27.942	67.975	0.042416
67	250	0.05	0.15	0.3	0.1	0.1	2	1	75	200	68.451	72.908	0.042427
68	250	0.05	0.15	0.3	0.2	0.1	1	1	75	150	68.64	73.683	0.042436
69	50	0.05	0.1	0.9	0.1	0.1	0.8	1	40	150	14.398	58.867	0.042437
70	250	0.5	0.75	0.99	0.1	0.6	0.8	1	75	150	69.764	66.383	0.042443
71	250	0.05	0.5	0.99	0.1	0.4	0.8	1	40	200	69.665	65.212	0.042465
72	100	0.05	0.15	0.3	0.1	0.1	0.8	1	75	200	27.888	59.995	0.042471
73	250	0.05	0.5	0.99	0.1	0.1	1	1	40	200	69.624	70.351	0.042473
74	250	0.05	0.5	0.99	0.2	0.6	0.8	1	75	150	70.091	68.862	0.042484
75	250	0.5	0.75	0.99	0.1	0.6	1	1	75	200	70.504	71.677	0.042512
76	100	0.5	0.75	0.99	0.1	0.1	1	1	40	200	29.543	64.118	0.042528
77	100	0.05	0.15	0.3	0.2	0.1	2	1	75	150	27.512	62.388	0.042617
78	250	0.1	0.9	0.99	0.2	0.4	1	1	75	150	69.615	71.074	0.042621
79	250	0.05	0.5	0.99	0.1	0.1	2	1	75	150	68.767	75.611	0.042661
80	250	0.05	0.1	0.9	0.2	0.6	1	1	75	200	69.936	70.498	0.042663
81	250	0.05	0.5	0.99	0.2	0.4	0.8	1	75	200	69.277	73.7	0.042671
82	250	0.05	0.1	0.9	0.2	0.6	0.8	1	40	200	69.711	67.812	0.042686
83	250	0.05	0.5	0.99	0.1	0.1	0.8	1	75	200	68.686	75.68	0.042704
84	250	0.05	0.15	0.3	0.1	0.4	2	1	75	150	68.876	65.358	0.04272
85	250	0.05	0.1	0.9	0.1	0.4	0.8	1	75	200	69.749	66.959	0.042749
86	100	0.5	0.75	0.99	0.1	0.4	1	1	40	150	28.862	64.041	0.042832
87	50	0.05	0.5	0.99	0.1	0.1	1	1	40	150	14.603	62.268	0.042838
88	250	0.5	0.75	0.99	0.2	0.6	0.8	1	40	150	91.585	66.374	0.042865
89	250	0.05	0.1	0.9	0.1	0.1	1	1	75	150	68.868	61.846	0.042872
90	250	0.05	0.15	0.3	0.1	0.4	2	1	40	150	69.137	68.431	0.042896
91	250	0.05	0.15	0.3	0.2	0.1	2	1	40	200	68.824	68.457	0.042904
92	250	0.05	0.1	0.9	0.2	0.1	2	1	75	150	68.68	67.209	0.042913
93	250	0.5	0.75	0.99	0.2	0.1	2	1	40	150	94.104	67.803	0.042924
94	250	0.5	0.75	0.99	0.1	0.1	2	1	75	150	68.698	67.984	0.042971
95	250	0.05	0.15	0.3	0.1	0.1	1	1	40	150	70.378	74.191	0.042979
96	100	0.05	0.5	0.99	0.1	0.6	0.8	1	40	150	29.464	62.276	0.043062
97	250	0.1	0.9	0.99	0.2	0.1	0.8	1	75	200	68.893	67.2	0.043064
98	250	0.05	0.1	0.9	0.2	0.4	0.8	1	40	200	69.556	69.12	0.043064
99	250	0.05	0.1	0.9	0.1	0.4	2	1	40	200	69.295	65.427	0.043071
100	250	0.05	0.5	0.99	0.2	0.1	2	1	75	150	68.495	72.185	0.043134

Tabelle C.10: Eigenschaften der besten nach dem mean squared Error gefundenen Echo State Netzen unter Verwendung des sliding Windows als Input, ohne Information über die Leistungswerte des Kraftwerks.

N_D ... Netz/Reservoir Dimension

σ_n ... Spektralradius des Reservoir n

c ... Connectivity der Reservoirs

i_c ... Inputconnectivity zu den Reservoirs

in_s ... Inputskalierung

i_W ... inklusive Wettervorhersagewerte (ja 1 / nein 0)

pS ... Länge Vergangenheitsfenster

mT ... Langzeittemperaturfenster.

Anhang D

Relevante Datenblätter, gemessene PVA *St. Veit*

SUNNY TRIPower
10000TL / 12000TL / 15000TL / 17000TL



STP 10000TL-10 / STP 12000TL-10 / STP 15000TL-10 / STP 17000TL-10

Wirtschaftlich	Sicher	Flexibel	Einfach
<ul style="list-style-type: none"> • Maximaler Wirkungsgrad von 98,1 % • Bester Anpassungswirkungsgrad durch OptiTrac Global Peak* • Bluetooth® Kommunikation 	<ul style="list-style-type: none"> • Elektronische String-Sicherung und -ausfallerkennung • Integrierbare DC-Überspannungsableiter (Typ II) • String-Stromüberwachung 	<ul style="list-style-type: none"> • DC-Eingangsspannung bis 1.000 V • Integrierte Netzmanagement-funktionen • Passgenaue Anlagenauslegung durch Optiflex 	<ul style="list-style-type: none"> • Dreiphasige Einspeisung • Werkzeugloser Kabelanschluss • DC-Stecksystem SUNCLIX • Leicht zugänglicher Anschlussbereich

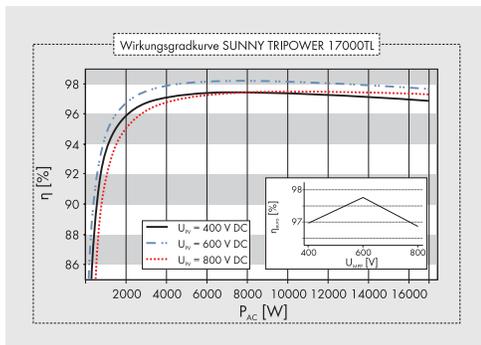
SUNNY TRIPower
10000TL / 12000TL / 15000TL / 17000TL

Der Dreiphasige für einfache Systemplanung

Vollgepackt mit zukunftsweisender Technologie: Der dreiphasige Sunny Tripower eignet sich durch die neue Optiflex-Technologie mit zwei MPP-Eingängen in Verbindung mit einem sehr weiten Eingangsspannungsbereich für nahezu alle Modulkonfigurationen. Und ist dabei hochflexibel bei der Anlagenauslegung - bis in den Megawattbereich. Der Sunny Tripower erfüllt alle Anforderungen an Blindleistungsbereitstellung, Einspeisemanagement und Netzstützung und beteiligt sich somit zuverlässig am Netzmanagement. Das umfassende Sicherheitskonzept Optiprotect mit selbstlernender String-Ausfallerkennung, elektronischer String-Sicherung und integrierbarem DC-Überspannungsableiter Typ II sorgt für höchste Verfügbarkeit.

Abbildung D.1: Datenblatt des SMA Sunny Tripower 10000TL/12000TL/15000TL/17000TL Wechselrichters, 1 von 2.

Technische Daten	Sunny Tripower 10000TL	Sunny Tripower 12000TL	Sunny Tripower 15000TL	Sunny Tripower 17000TL
Eingang (DC)				
Max. DC-Leistung (@ cos φ=1)	10200 W	12250 W	15340 W	17410 W
Max. DC-Spannung	1000 V	1000 V	1000 V	1000 V
MPP-Spannungsbereich	320 V - 800 V	380 V - 800 V	360 V - 800 V	400 V - 800 V
DC-Nennspannung	600 V	600 V	600 V	600V
Min. DC-Spannung / Startspannung	150 V / 188 V	150 V / 188 V	150 V / 188 V	150 V / 188 V
Max. Eingangsstrom / pro String	A: 22 A, B: 11 A / 33 A	A: 22 A, B: 11 A / 33 A	A: 33 A, B: 11 A / 33 A	A: 33 A, B: 11 A / 33 A
Anzahl MPP-Tracker / Strings pro MPP-Tracker	2 / A: 4, B: 1	2 / A: 4, B: 1	2 / A: 5, B: 1	2 / A: 5, B: 1
Ausgang (AC)				
AC-Nennleistung (@ 230 V, 50 Hz)	10000 W	12000 W	15000 W	17000 W
Max. AC-Scheinleistung	10000 VA	12000 VA	15000 VA	17000 VA
AC-Nennspannung; Bereich	3 / N / PE, 230 V / 400 V; 160 V - 280 V			
AC-Netzfrequenz; Bereich	50, 60 Hz; -6 Hz, +5 Hz	50, 60 Hz; -6 Hz, +5 Hz	50, 60 Hz; -6 Hz, +5 Hz	50, 60 Hz; -6 Hz, +5 Hz
Max. Ausgangsstrom	16 A	19,2 A	24 A	24,6 A
Leistungsfaktor (cos φ)	0,8 übererregt ... 0,8 untererregt			
Einspeisephase / Anschlussphasen / Power Balancing	3 / 3 / -	3 / 3 / -	3 / 3 / -	3 / 3 / -
Wirkungsgrad				
Max. Wirkungsgrad / Euro-eta	98,1 % / 97,7 %	98,1 % / 97,7 %	98,2 % / 97,8 %	98,2 % / 97,8 %
Schutzrichtungen				
DC-Verpolungsschutz / Rückstromschutz	●/elektronisch	●/elektronisch	●/elektronisch	●/elektronisch
DC-Lasttrennschalter ESS	●	●	●	●
AC-Kurzschlussfestigkeit	●	●	●	●
Erdschlussüberwachung	●	●	●	●
Netzüberwachung (SMA Grid Guard)	●	●	●	●
Galvanisch getrennt / Allstromsensitive Fehlerstromüberwachungseinheit	-/●	-/●	-/●	-/●
DC-Überspannungsableiter Typ II	○	○	○	○
String-Ausfallerkennung	●	●	●	●
Schutzklasse / Überspannungskategorie	I / III	I / III	I / III	I / III
Allgemeine Daten				
Maße (B / H / T) in mm	665 / 690 / 265	665 / 690 / 265	665 / 690 / 265	665 / 690 / 265
Gewicht	65 kg	65 kg	65 kg	65 kg
Betriebstemperaturbereich	-25 °C ... +60 °C	-25 °C ... +60 °C	-25 °C ... +60 °C	-25 °C ... +60 °C
Geräuschemission, typisch	www.SMA-Solar.com	www.SMA-Solar.com	www.SMA-Solar.com	www.SMA-Solar.com
Eigenverbrauch (Nacht)	1 W	1 W	1 W	1 W
Topologie	transformatorlos	transformatorlos	transformatorlos	transformatorlos
Kühlkonzept	OptiCool	OptiCool	OptiCool	OptiCool
Schutzart Elektronik / Anschlussbereich (nach IEC 60529)	IP65 / IP54	IP65 / IP54	IP65 / IP54	IP65 / IP54
Klimaklasse (nach IEC 60721-3-4)	4K4H	4K4H	4K4H	4K4H
Ausstattung				
DC-Anschluss: SUNCLIX	●	●	●	●
AC-Anschluss: Schraubklemme / Federzugklemme	-/●	-/●	-/●	-/●
Display: Textzeile / Grafik	-/●	-/●	-/●	-/●
Schnittstellen: RS485 / Bluetooth®	○/●	○/●	○/●	○/●
Garantie: 5 / 10 / 15 / 20 / 25 Jahre	●/○/○/○/○	●/○/○/○/○	●/○/○/○/○	●/○/○/○/○
Zertifikate und Zulassungen (weitere auf Anfrage)	CE, VDE 0126-1-1, EnelGUIDA*, G83/1-1, PPC, AS4777, EN 50438**, C10/C11, IEC 61727*			
* In Planung, ** gilt nicht für alle nationalen Anhänge der EN 50438				
● Serienausstattung ○ Optional – nicht verfügbar				
Angaben Stand: Februar 2011 – Angaben bei Nennbedingungen				
Typenbezeichnung	STP 10000TL-10	STP 12000TL-10	STP 15000TL-10	STP 17000TL-10



Zubehör

- RS485-Schnittstelle DM-485CB-10
- DC-Überspannungsableiter Typ II, Eingang A DCSPD KIT1-10
- DC-Überspannungsableiter Typ II, Eingang A und B DCSPD KIT2-10

Abbildung D.2: Datenblatt des SMA Sunny Tripower 10000TL/12000TL/15000TL/17000TL Wechselrichters, 2 von 2.

TRIPOWER001 10323 SMA und Sunny Tripower sind eingetragene Warenzeichen der SMA Solar Technology AG. RS485, SUNCLIX und OptiCool sind eingetragene Marken der SUNCLIX GmbH & Co. KG. Alle in den Abbildungen und Tabellen verwendeten Symbole sind die üblichen Symbole der IEC. Technische Änderungen vorbehalten. Keine Haftung für Druckfehler. Gedruckt auf recyclingfähigen Papier.



KIOTO
PHOTOVOLTAICS

KPV PE NEC
190/195/200/205/210 Wp

quality: **first**



Abbildung D.3: Datenblatt des *Kioto Photovoltaics KPV PE NEC 190/195/200/205/210 Wp* Photovoltaikmoduls, 1 von 2.

Die KIOTO Photovoltaics GmbH produziert bereits seit mehreren Jahren hochwertige Photovoltaik-Module, welche alle Qualitätsanforderungen der europäischen Märkte übertreffen. Die hocheffizienten multikristallinen Solarmodule auf der Basis von 6" Wafern werden mit modernstem Equipment ausschließlich in Österreich produziert. Ein innovatives Messverfahren garantiert die sehr genaue Spezifikation. Die Solarmodule mit Leistungen von 190 bis 210 Wp werden mit Aluminiumrahmen und Eckverbindern ausgeliefert. Serienmäßig ist eine Tyco Anschlussdose angebracht. Es verlassen nur erstklassige Solarmodule die Fertigungsstraße der KIOTO PV. Diese leistungs- und ertragsoptimierten Solarmodule werden vorwiegend in netzgekoppelten Anlagen eingesetzt.



Leistungsklassen:

KPV PE NEC

Type	PmpP _(Wp)	UmpP _(V)	ImpP _(A)	Uoc _(V)	Isc _(A)	Wirkungsgrad	Flächenbedarf pro kWp
KPV 190 PE	190 Wp	25,91 V	7,38 A	32,13 V	7,68 A	12,71 %	7,84 m ²
KPV 195 PE	195 Wp	26,08 V	7,49 A	32,71 V	8,08 A	13,04 %	7,64 m ²
KPV 200 PE	200 Wp	26,34 V	7,63 A	32,91 V	8,30 A	13,38 %	7,45 m ²
KPV 205 PE	205 Wp	26,39 V	7,80 A	33,08 V	8,33 A	13,71 %	7,26 m ²
KPV 210 PE	210 Wp	26,47 V	7,96 A	33,26 V	8,59 A	14,05 %	7,09 m ²

54 polykristalline Zellen, 156mm x 156mm			
Tyco-Solarlok®, max. Systemspannung 1000 V DC,			
Leistungstoleranz: (+ 3% / 0%)*			
Temperaturkoeffizienten:	PmpP= -0,405%/K	Uoc= -102,6mV/K	Isc= +4,1mA/K
Standard Test Bedingungen:	AM1,5 / 1000W pro m ² / 25°C		
Umgebungstemperatur:	+ 85°C bis - 40°C		
Kabellänge:	2000mm		
Bypassdioden:	3 Stk. Tyco SL1515		
Leistungsgarantie:	90% auf 10 Jahre	80% auf 25 Jahre	
Produktgarantie:	5 Jahre		

* Messung nach Standard Test Bedingungen

Technische Daten:

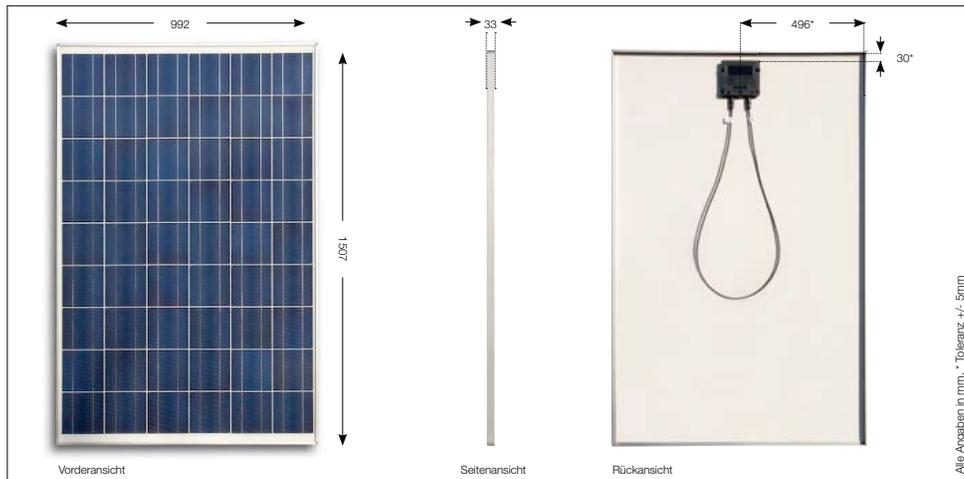
Abmessungen:

inkl. Alurahmen:	1507mm x 992mm x 33mm (+/-2 mm)
Lamine:	1500mm x 985mm x 4,5mm (Dosenhöhe 22,5mm)
Gewicht inkl. Alurahmen:	16,50kg
Glasspezifikationen:	Solarglas ESG 3,2mm
Verkapselungsmaterial:	Etimex
Rückseitenmaterial:	Isovolta, Krempel
Prüfzertifikat:	Schutzklasse II*, IP 65, IEC 61215, Ed. 2 inkl. erweitertem mechanischen Belastungstest bis 5400 Pa



Abmessungen:

** bis 1000 V



Alle Angaben in mm. * Toleranz +/- 5mm

Stand: Jänner 2010 Technische Änderungen und Designanpassungen sind vorbehalten!

KIOTO Photovoltaics GmbH
Solarstraße 1
A-9300 St. Veit/Glan, AUSTRIA

T: +43 (0) 4212 28 300-0
F: +43 (0) 4212 28 300-799
office@kioto-pv.com

www.kioto-pv.com

Abbildung D.4: Datenblatt des Kioto Photovoltaics KPV PE NEC 190/195/200/205/210 Wp Photovoltaikmoduls, 2 von 2.



LSI SpA - Via Ex SP. 161, 9 - Loc. Dosso
 20090 Settala - Milano - Italy
 Tel. (+39) 02 95 41 41
 Fax (+39) 02 95 77 05 94
 E-mail: info@lsi-lastem.it
 http://www.lsi-lastem.it



MW8003.1-10/05

Radiometri globali - Piranometri **Global radiation sensors Pyranometers**



I radiometri globali sono sensori per la misura di intensità di radiazione elettromagnetica, emessa direttamente dal sole o riflessa da altri agenti ambientali (atmosfera, nuvole). La LSI Lastem propone una linea di strumentazione altamente affidabile, capace di rispondere con i suoi modelli alle diverse esigenze.

Caratteristiche principali:

- * Sensore a termopila o ad elemento fotovoltaico.
- * Riferibilità al WRC (World Radiation Center).
- * Uscita diretta o amplificata (0/4-20 mA, 0/1-5 V).
- * Stabilità.
- * Correzione della risposta al coseno.
- * Alta sensibilità.
- * Protezione da disturbi elettromagnetici.

Global radiometers are sensors for measuring electromagnetic radiation intensity, directly emitted from sunshine disc as well as reflected by the atmosphere and cloud layers.

LSI LASTEM supplies a highly reliable set of instruments, capable to satisfy all the requests with different models.

Main features:

- * Thermopile or fotovoltaic element sensors.
- * WRC (World Radiation Center) traceability.
- * Direct or analogue output (0/4-20 mA, 0/1-5 V).
- * Stability.
- * Cosine correction.
- * High sensibility
- * Protection from electromagnetic disturbances.

VERSIONI **VERSIONS**

Modelli	Descrizione	Cod.		
		Alimentazione / Power supply	24 V [Ⓢ] Analog*	12 V= Analog*
C101R	Radiometro globale a termopila, First class ISO9060 <i>ISO9060 first class thermopile global radiometer</i>			DPA253
C100R	Radiometro globale a termopila, First class ISO9060 <i>ISO9060 first class thermopile global radiometer</i>			DPA153
C510R	Radiometro globale a termopila, First class ISO9060 con uscita normalizzata <i>ISO9060 first class thermopile global radiometer Normalized output</i>		DPA554	DPA559
C511R	Radiometro globale a termopila, First class ISO9060 con uscita normalizzata <i>ISO9060 first class thermopile global radiometer Normalized output</i>		DPA553	DPA558
C102R	Radiometro globale a fotodiode <i>Photodiode global radiometer</i>			DPA047
C502R	Radiometro globale a fotodiode con uscita normalizzata <i>Photodiode global radiometer. Normalized output</i>		DPA560	DPA568

* uscita programmabile localmente / locally programmable output: 0/4-20 mA, 0/1-5 V (default 4-20 mA)

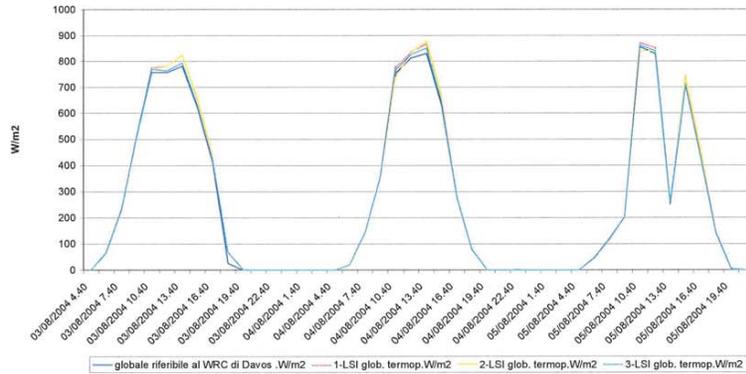
Abbildung D.5: Datenblatt der LSI DPAxxx-xxx Sternpyranometer, 1 von 3.

CARATTERISTICHE TECNICHE				TECHNICAL SPECIFICATIONS			
	C101R DPA253	C511R DPA553-558	C100R DPA153	C510R DPA554-559	C102R DPA047	C502R DPA560-568	
Elemento sensibile	Termopila / Thermopile				Fotodiodo / Photodiode		Sensitive element
Campo spettrale	305-2800 nm				300-1100 nm		Spectral range
Campo di misura	<2000 W/m ²	0-1500 W/m ²	<2000 W/m ²	0-1500 W/m ²	<2000 W/m ²	0-1500 W/m ²	Range
Uscita (sensibilità)	9-15 V/Wm ²	Normalizzata	25-40 V/Wm ²	Normal.	10 V/Wm ²	Normal.	Output (Sensitivity)
Incertezza totale (su 1 gg di misura)	<5%				<10%	<10%	Total achievable daily uncertainty
Tempo di risposta (90%)	18 sec		27 sec		50 ms		Response time (90%)
Zero off-set dovuto al cambiamento di temperatura (5°C/ora)	<± 4 W/m ²				NA		Zero off-set due to temperature changes (5°C/hr)
Errore direzionale (a 80° con 800 W/m ²)	<± 20 W/m ²				NA		Directional error (80°, 800 W/m ²)
Non linearità (1000 W/m ²)	<± 1,2 W/m ²				NA		Non-linearity (1000 W/m ²)
Non stabilità (deriva/anno)	± 1% sensitività/ sensitivity		± 1,5% sensitività/sensitivity		NA		Non-stability (change per year)
Deriva termica sulla sensitività	<± 4% sensitività/sensitivity				0,05% sensitività/sensitivity/°C		Temperature dependance of sensitivity
Errore di tilt	± 1% (1000 W/m ²)		<± 2% (1000 W/m ²)		NA		Tilt error
Consumo	NA	0,7 W	NA	0,7 W	NA	0,7 W	Consumption
Resistenza di carico (Uscita mA)	12V=: 300 Ω 24V=: 500 Ω		12V=: 300 Ω 24V=: 500 Ω		12V=: 300 Ω 24V=: 500 Ω		R Load Max (mA output)
Cavo uscita	10 m incluso / attached	DWA... Non incluso / not attached	5 m incluso / attached	DWA... Non incluso / not attached	10 m incluso / attached	DWA... Non incluso / not attached	Cable
Attacco a palo diam 50 mm.	DYA034+ DYA051	DYA051	DYA034+ DYA051	DYA051	DYA032+ DYA051	DYA051	Arms for pole diam 50 mm
Peso (senza cavo)	0,86 Kg	1,9 Kg	0,9 Kg	1,9 Kg	80 gr	1,1 Kg	Weight
Limiti ambiental	-40...+80°C						Environ. limits

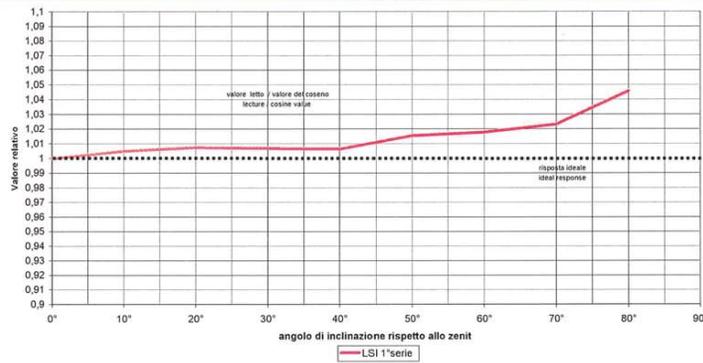
ACCESSORI E RICAMBI		SPARES AND ACCESSORIES	
Cod	Descrizione	Description	
Elementi di connessione elettrica		Electrical connection units	
MG2251	Connettore volante stagno IP65 per sensori ad uscita normalizzata	IP65 waterproof free connector	
DWA510	Cavo schermato L=10 m con connettore per sensori C511R, C510R e C502R	10 m shielded cable with connector for C511R, C510R and C502R sensors	
DWA525	Cavo schermato L=25 m con connettore per sensori C511R, C510R e C502R	25 m shielded cable with connector for C511R, C510R and C502R sensors	
DWA526	Cavo schermato L=50 m con connettore per sensori C511R, C510R e C502R	50 m shielded cable with connector for C511R, C510R and C502R sensors	
DWA527	Cavo schermato L=100 m con connettore per sensori C511R, C510R e C502R	100 m shielded cable with connector for C511R, C510R and C502R sensors	
Elementi di connessione meccanica		Mechanical connection units	
DYA032	Supporto per fissaggio sensori C102R a collare DYAO51	Arm for mounting C102R sensor to DYAO51 collar	
DYA034	Supporto per fissaggio sensori C100R e C101R a collare DYAO51	Arm for mounting C100R and C101R sensors to DYAO51 collar	
DYA051	Collare per attacco sensori e supporti a palo diam. 50 mm	Collar for fixing sensors and arm to diam.50 mm poles	
Banda di occultazione per radiazione diffusa		Occultation band for diffuse radiation	
DPA245	Adatta per radiometri globali C101R e C100R - Regolazione della latitudine: da 0 a 60° - Regolazione della declinazione: ±25° - Materiale: alluminio opaco nero - Peso: 11 Kg - Dimensioni: 500x600x900 mm	Suitable for C101R and C100R global radiometers - Latitude adjustment from 0 to 60° - Declination adjustment: ±25° - Material: matt black aluminium - Weight: 11 Kg - Dimensions: 500x600x900 mm	

RIFERIBILITÀ WCR **WCR TRACEABILITY**

Confronto radiometro globale LSI - radiometro globale riferibile al WCR di Davos.
LSI global radiometer vs WCR global radiometer
 Premenugo di Settala-Milano, ITALY. 3 - 5 agosto 2004



RISPOSTA AL COSENO PER RADIOMETRI GLOBALI **GLOBAL RADIOMETERS COSINE RESPONSE**



MONTAGGIO E DIMENSIONI **MOUNTING & DIMENSIONS**

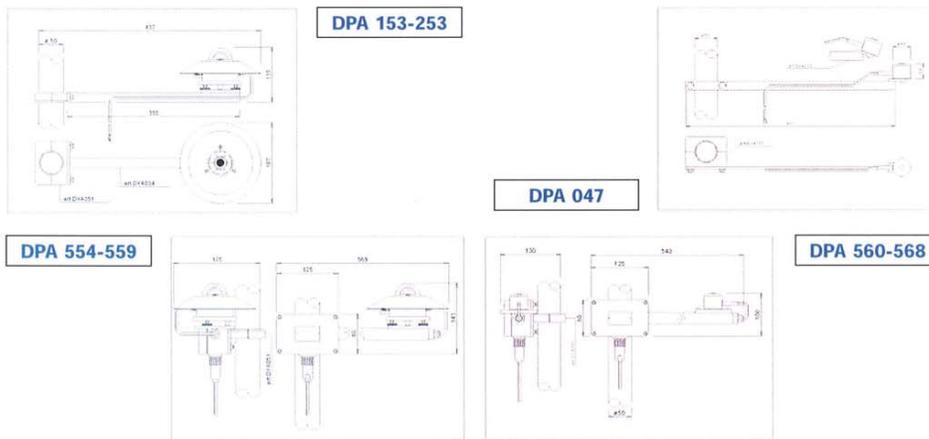


Abbildung D.7: Datenblatt der LSI DPAnnn-xxx Sternpyranometer, 3 von 3.

Anhang E

Abkürzungen

- DEA - Dezentrale Erzeugungsanlagen
- *eg* - Globales Fehlermaß zur Bewertung der Prognosegüte eines Modells
- IDNN - Input delayed neuronale Netze
- MLP - Multilayer Perzeptron
- MSE - Mean Squared Error
- PM_{10} - Particulate matter (Feinstaub) mit aerodynamischen Durchmesser $< 10\mu m$
- PVA - Photovoltaikanlage
- RNN - Rekurrente neuronale Netze
- STC - Standard Test Conditions / Standard Test Bedingungen
- SVM - Support Vector Machine
- SVR - Support Vector Regression
- WLKW - Wasserlaufkraftwerk
- ZAMG - Zentralanstalt für Meteorologie und Geodynamik