



Fakultät Life Sciences

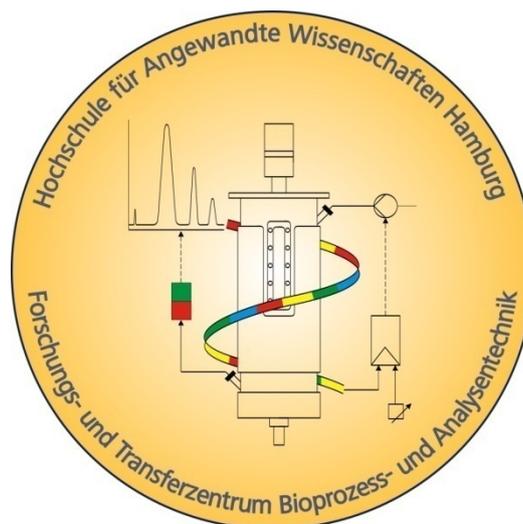
Hochschule für Angewandte Wissenschaften Hamburg

Hamburg University of Applied Sciences

Masterarbeit

im Studiengang Pharmaceutical Biotechnology

Entwicklung eines soft sensors/smart sensors zur Überwachung des Kultivierungsprozesses von *Pichia pastoris* zur Herstellung virusähnlicher Partikel (VLPs)



Stefanie Kaul

27.09.2016

1. Gutachter/in

Prof. Dr. Gesine Cornelissen
(HAW Hamburg)

2. Gutachter/in

Prof. Dr. Reiner Luttmann
(HAW Hamburg)

Inhaltsverzeichnis

Abbildungsverzeichnis.....	1
1 Einleitung.....	5
2 Theoretische Grundlagen.....	6
2.1 Smart/Soft Sensors.....	6
2.2 „White Box“-Modelle.....	6
2.2.1 „White Box“-Modell für den Sauerstoffpartialdruck p_{O_2}	7
2.2.2 „White Box“-Modell für die Methanolkonzentration $c_{S_{2M}}$	10
2.3 „Black Box“ und „Grey Box“-Modelle.....	13
2.3.1 „Grey Box“-Mathematik.....	16
2.3.1.1 Die Methanolkonzentration $c_{S_{2M}}$ über das Konzentrationsgleichgewicht .	16
2.3.1.2 p_{O_2} – quasi-stationäres Verhältnis von Sauerstofftransferrate OTR und Sauerstoffzufuhr rate Q_{O_2}	17
2.3.1.3 p_{O_2} – quasi-stationäres Verhältnis von Sauerstofftransferrate OTR und Sauerstoffaufnahme rate OUR.....	18
2.3.2 Neuronale Netze.....	19
2.3.3 Hauptkomponentenregression.....	24
2.3.4 Multivariate lineare Regression.....	26
2.3.5 Extended Kalman-Filter (EKF).....	26
2.4 Fehlerberechnung.....	28
2.5 Alkoholmessung.....	28
2.6 Sauerstoffpartialdruckmessung.....	28
3 Material und Methoden.....	29
3.1 Datenpräparation.....	29
3.2 Festlegen von Trainings-, Validierungs- und Testdatensätzen spezifisch zu den verschiedenen Modellierungswegen und die Festlegung der Hyperparametergrenzen	29

3.3	Einpflegen der Modelle in SIPAT	31
3.4	Fermentation	31
3.4.1	Mikroorganismus.....	31
3.4.2	Medienzusammensetzung	32
3.4.3	Ansetzen der Vorkultur	33
3.4.4	Vorbereitung der Bioreaktoren.....	33
3.4.5	Fermentationsverlauf im ED10	36
3.4.6	Fermentationsverlauf im C30.....	36
3.4.7	Probenahme und Probenvorbereitung.....	37
3.4.8	Analytische Methoden	37
3.4.8.1	Bestimmung der Biotrockenmasse	37
3.4.8.2	Bestimmung der optischen Dichte (OD)	38
4	Ergebnis und Diskussion.....	39
4.1	Verlauf von Kultivierung SK2716 und die Abweichung vom Normalprozess.....	39
4.2	Darstellung der Zellkonzentration aus der Basentitration	43
4.3	Vergleich der verschiedenen Modellierungen	45
4.3.1	„White Box“	45
4.3.2	Extended Kalman-Filter, Hauptkomponentenregression und multivariate lineare Regression	52
4.3.3	Neuronale Netzwerke – „Black Box“	57
4.3.4	Neuronale Netzwerke – „Grey Box“	67
4.3.5	Vergleich der Modelle	74
5	Zusammenfassung.....	75
6	Literaturverzeichnis.....	77
	Anhang.....	79

Abbildungsverzeichnis

Abbildung 1: Schematische Darstellung der "Black Box" Vorgehensweise. Bei dieser Art der Modellfindung liegt der Fokus auf den empirischen Daten und nicht auf den mathematischen Zusammenhängen. x_i bezeichnet dabei das Inputarray, das zum Zeitpunkt i in das Modell eingeht. Solche Inputwerte können zum Beispiel Rührerdrehzahl, Molenbrüche und andere Messwerte sein. \hat{y} bezeichnet den durch das Modell vorhergesagten Zielwert, wie zum Beispiel pO_2 oder c_{S2M}	13
Abbildung 2: Schematische Darstellung der verwendeten "Grey Box". Die Messwerte fließen dort in ein "Black Box"-Modell ein, berechnen dort aber nicht direkt die zu messende Größe, sondern einen Parameter der bekannten Gleichungssysteme.....	14
Abbildung 3: Schematische Darstellung eines einfachen Netzwerkes mit drei versteckten Neuronen. Die Inputs x_j werden vorwärts durch das Netz übertragen, daher wird dieses Netzwerk auch als Feedforward-Netzwerk bezeichnet. Die Aktivierung a_i bezeichnet den Ausgabewert des versteckten Neurons i . „Versteckt“ bedeutet, dass dieses Neuron keine Eingangs- beziehungsweise Ausgangsneuron ist. Bei der Übertragung des Inputs x wird dieser mit dem sogenannten Gewicht der Verbindung ω zwischen des Input- und des versteckten Neurons multipliziert. Die versteckten Neuronen haben einen variablen Schwellenwert b , auch Bias genannt, der die Aktivierung der Neurone modifiziert. Die Aktivierung wird mithilfe der Aktivierungsfunktion σ berechnet.	19
Abbildung 4: Schematische Darstellung eines nichtlinearen regressiven closed loop neuronalen Netzwerkes. Der Output \hat{y} des Netzes geht zusammen mit x als Input mit ein. Zudem wird eine Zeitverzögerung t_p miteingebracht. n_H ist die Anzahl an versteckten Neuronen.....	23
Abbildung 5: Instrumentierung Biostat ED10 und Geräteperipherie.	34
Abbildung 6: Instrumentierung Biostat C30 und Geräteperipherie.	34
Abbildung 7: Verlauf der Parameter pO_2 , N_{St} , c_{S2M} und F_{R2rel} der Kultivierung SK2716 und SK2216 im ED10.	40
Abbildung 8: Verlauf der Parameter pO_2 , N_{St} , c_{S2M} und F_{R2rel} der Kultivierung SK2716 und SK2216 im C30.....	42

Abbildung 9: Darstellung der Korrelation des Basenverbrauches zu der Zellkonzentration im ED10 und im C30. Der Basenverbrauch wurde linear über die Formel $c_{XL_T2}(t) = p(1) \cdot \Delta m_{T2}(t) + p(2)$ mit der Zellkonzentration korreliert. Schwarz ist die aus der Trübung berechnete Zellkonzentration zu erkennen. Es ist am sprunghaften Verlauf von c_{XL_turb} im C30 erkennbar, dass eine die starke Abhängigkeit des Trübungssignals zur Rührerdrehzahl besteht. 43

Abbildung 10: Verlauf der Reaktionsraten $q_{S2/X}(t)$ und $q_{X/X}(t)$ in beiden Reaktoren während der Kultivierung SK2716. Nur die Methanol-Feed-Phase ist dargestellt. Die rote Linie markiert den Bereich über den die Konstanten $q_{I/X}$ bestimmt wurden. 45

Abbildung 11: Vergleich der mit Gleichung (12) berechneten mit der tatsächlich gemessenen Methanolkonzentration der verschiedenen Kultivierungen im ED10. 46

Abbildung 12: Vergleich der mit Gleichung (12) berechneten mit der tatsächlich gemessenen Methanolkonzentration der verschiedenen Kultivierungen im C30. 48

Abbildung 13: Vergleich der aus den Ladekurven und der mit Hilfe der $k_L a$ -Korrelation berechneten $k_L a$ -Werte. Es wurden die Werte für $N_{St} = 200, 400$ und 600 rpm aufgenommen. 49

Abbildung 14: Vergleich des mit Gleichung (4) berechneten pO_2 mit dem tatsächlich gemessenen der verschiedenen Kultivierungen im ED10. 50

Abbildung 15: Verlauf der mit Hilfe des Kalman-Filters bestimmten Methanolkonzentration mit der gemessenen aller Kultivierungen im C30. 52

Abbildung 16: Gemessener und durch Hauptkomponentenregression bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im C30. 54

Abbildung 17: Gemessener und durch multivariate lineare Regression bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im C30. 55

Abbildung 18: Die RMSE der durch das Feedforward Netzwerk bestimmten Methanolkonzentration im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen nH und Epochenanzahlen $epoch$ 58

Abbildung 19: Die RMSE der durch das Feedforward Netzwerk bestimmten Methanolkonzentration im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) bei verschiedener Inputkonfigurationen. 59

Abbildung 20: Gemessene und durch Feedforward-"Black Box" bestimmte Methanolkonzentration für alle vier Kultivierungen im ED10 und die berechneten RMSE....	60
Abbildung 21: Die RMSE des durch das Feedforward Netzwerk bestimmten Sauerstoffpartialdrucks im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen nH und Epochenanzahlen epoch.	61
Abbildung 22: Gemessener und durch Feedforward-"Black Box" bestimmter Sauerstoffpartialdruck p_{O_2} für alle vier Kultivierungen im ED10 und die berechneten RMSE.	62
Abbildung 23: Die RMSE der durch das regressive closed loop Netzwerk bestimmten Methanolkonzentration im C30 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen nH, Verzögerungsschritte t_p und Epochenanzahlen epoch.	64
Abbildung 24: Gemessene und durch regressive "Black Box" bestimmte Methanolkonzentration $c_{S_{2M}}$ für alle vier Kultivierungen im C30 und die berechneten RMSE.	65
Abbildung 25: Gemessene und durch „Grey Box" bestimmte Methanolkonzentration $c_{S_{2M}}$ für alle vier Kultivierungen im C30. Die Substrataufnahmerate $q_{S_{2/X}}(t)$ wurde über den „Black Box“-Anteil bestimmt.....	67
Abbildung 26: Gemessene und durch „Grey Box" bestimmte Methanolkonzentration $c_{S_{2M}}$ für alle vier Kultivierungen im C30. $y_{gC_2}(t) = V_L(t) \cdot V_M(t)^{-1} \cdot q_{S_{2X}}(t) \cdot c_{XL}(t)$ wurde über den „Black Box“-Anteil bestimmt.....	68
Abbildung 27: Gemessener und durch „Grey Box" bestimmter Sauerstoffpartialdruck p_{O_2} für alle vier Kultivierungen im ED10. Der volumenbezogene Sauerstoffübergangskoeffizient $k_L a(t)$ wurde über den „Black Box“-Anteil bestimmt.....	69
Abbildung 28: Gemessener und durch „Grey Box" bestimmter Sauerstoffpartialdruck p_{O_2} für alle vier Kultivierungen im ED10. $y_{g_OQ_2}(t) = Q_{O_2}(t) \cdot (k_L a(t) \cdot c_{OLmax}(t))^{-1}$ wurde über den „Black Box“-Anteil bestimmt.....	70
Abbildung 29: Gemessener und durch „Grey Box" bestimmter Sauerstoffpartialdruck p_{O_2} für alle vier Kultivierungen im ED10. $y_{g_OO_1}(t) = q_{O/X}(t) \cdot OTR_{max}(t)^{-1}$ wurde über den „Black Box“-Anteil bestimmt.....	71

Abbildung 30: Gemessener und durch „Grey Box" bestimmter Sauerstoffpartialdruck $pO_2\%$ für alle vier Kultivierungen im ED10. $y_{g_OO_1}(t) = q_{O/X}(t) \cdot OTR_{max}(t)^{-1}$ wurde über den „Black Box“-Anteil bestimmt..... 72

Tabellenverzeichnis

Tabelle 1: Grenzen in denen die Hyperparameter für das Feedforward-Netzwerk variiert wurden.	30
Tabelle 2: Medienzusammensetzung.....	32
Tabelle 3: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der "White Box"-Modelle.	51
Tabelle 4: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe von Kalman-Filter, Hauptkomponentenregression (PCR) und multivariater linearer Regression.	56
Tabelle 5: RMSE des Testes zum Festlegen des einzusetzenden Lernalgorithmus zum Trainieren aller weiteren Netzwerke	57
Tabelle 6: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der „Black Box“ Neuronalen Netzwerke.	66
Tabelle 7: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der „Grey Box“. ..	73
Tabelle 8: Durchschnittliche RMSE der Zielvariablen aller Kultivierungen und aller Modelle	74

1 Einleitung

Bei industriellen Bioprozessen existiert eine sehr große Menge an Parametern von denen gewünscht ist, dass sie, aus verschiedenen Gründen, gemessen werden. Diese werden dazu genutzt um die Prozesse zu beschreiben, zu charakterisieren und zu überwachen. Die Herausforderung dabei ist, dass die benötigten Hardwaresensoren oft teuer, wie zum Beispiel die at-line HPLC, oder unzuverlässig sind oder schlicht und einfach nicht existieren. Für solche Fälle kommen die sogenannten „Soft Sensors“ in Frage, die diese Nachteile der Hardwaresensoren auffangen können. Bei diesen wird versucht aus den aufgenommenen Daten die gewünschten Zielvariablen darzustellen. Diese Zielvariablen können zum Beispiel die Wachstumsrate, Substratkonzentration, Sauerstoffpartialdruck, und viele weitere mehr sein. Wenn Zielvariablen gewählt werden, die ein Hardwaresensoräquivalent im Reaktor haben, können diese Hardwaresensoren überwacht, ergänzt oder sogar ersetzt werden. Dabei existieren, je nach Prozess, verschiedene Anforderungen. Dazu gehören zum Beispiel die Kosten, von denen gewünscht ist, dass sie möglichst minimal sind, eine Automatisierung der Sensoren, so dass sie kontinuierlich und mit minimalem manuellen Aufwand betrieben werden können, und dass sie stabil und gut validiert werden können. Dieser letzte Punkt wird insbesondere in dieser Arbeit betrachtet, da dieser für die industriellen und besonders GMP-konformen Prozesse von äußerster Wichtigkeit ist.

Ziel dieser Arbeit war die Entwicklung, Validierung und das Testen über verschiedene Modellierungswege eines „Soft Sensor“ zur Überwachung der Methanol- und Sauerstoffsonde der Reaktoren BIOSTAT® ED10 und BIOSTAT® C30.

2 Theoretische Grundlagen

2.1 Smart/Soft Sensors

Der Kunstbegriff „Soft Sensor“ setzt sich zusammen aus „Software“ und „Sensor“ und beschreibt eine Art der Ermittlung von Messgrößen über andere gemessene Daten des Prozesses. Zum Beispiel gehören dazu die Berechnung der Zellkonzentration aus der Basentitration, der Substratkonzentrationen über NIR-Spektren oder von Wachstumsraten über die Sauerstoffaufnahme. Die Vorteile der Soft Sensors sind, dass komplexe Messwerte, für die keine Hardwaremesssysteme existieren, sich so ermitteln lassen können, dass vorhandene Hardwaressysteme überwacht werden können um driftende Werte, Messfehler, Ausfälle und weiteres feststellen zu können, und dass die Installation von teilweise teuren und ungenauen Messsystemen umgangen werden könnte. Einige Nachteile der Soft Sensors sind, dass insbesondere die Datenaufbereitung bei Entwicklung und Einsatz dieser sehr wichtig ist und großes Vertrauen in die Aufbereitung bestehen muss. Zudem kann je nach eingesetztem Modellierungsweg eine häufige Neuvalidierung der Modelle beziehungsweise Parameter vonnöten sein. Ein weiterer Nachteil ist zudem, dass eine intensive Validierung der Soft Sensor Ausgaben vonnöten sind. [Kadlec et al., 2009]

Im weitesten Sinne können Soft Sensors in drei Kategorien eingeteilt werden: „White Box“-Modelle die auf bekannten grundlegenden Prozessgleichungen beruhen, „Black Box“ die rein datengetrieben sind und „Grey Box“-Modelle, die eine Mischung aus diesen beiden Extrempunkten darstellen. Jede Kategorie hat eigenen Stärken und Schwächen auf die später eingegangen wird. [Mandenius und Gustavsson, 2015]

2.2 „White Box“-Modelle

Die sogenannten „White Box“-Modelle basieren darauf, die zu ermittelnde Größe mit Hilfe bekannter Zustandsgleichungen zu ermitteln. Die Vorteile dieser Modelle sind, dass sie relativ stabil sind gegenüber Prozessschwankungen und aus dem Verständnis der grundlegenden Vorgänge entstanden sind. Zudem kann bei diesen Modellen bei gleichbleibendem Ablauf auf häufiges Neutrainieren beziehungsweise Neuanfertigen verzichtet werden. Einer der größten Nachteile dieser Modellarten ist, dass ein tiefes Verständnis der grundlegenden Geschehnisse und Mathematik vonnöten ist. Zudem ist die Bestimmung der notwendigen Pa-

parameter der Zustandsgleichungen teilweise sehr arbeitsaufwendig, sodass es wünschenswert sein kann diese zu umgehen. [Mandenius, Gustavsson, 2015; Lin et al., 2007]

2.2.1 „White Box“-Modell für den Sauerstoffpartialdruck p_{O_2}

Das „White Box“-Modell für den p_{O_2} wird aus dem quasi-stationären Zusammenhang der Sauerstoffzufuhr rate Q_{O_2} und der Sauerstofftransferrate OTR ermittelt. Die Sauerstoffzufuhr rate Q_{O_2} ,

$$Q_{O_2}(t) = Q_{O_2\max}(t) \cdot \frac{x_{OGin}(t) \cdot (1 - x_{CG}(t)) - x_{OG}(t) \cdot (1 - x_{CGin}(t))}{1 - x_{OG}(t) - x_{CG}(t)} \quad (1)$$

mit	Q_{O_2}	:= Sauerstoffzufuhr rate	$[g l^{-1} h^{-1}]$
	$Q_{O_2\max}$:= maximale Sauerstoffeintrags rate	$[g l^{-1} h^{-1}]$
	x_{OGin}	:= Sauerstoffmolenbruch der Zuluft	[-]
	x_{CG}	:= Kohlenstoffdioxidmolenbruch der Abluft	[-]
	x_{OG}	:= Sauerstoffmolenbruch der Abluft	[-]
	x_{CGin}	:= Kohlenstoffdioxidmolenbruch der Zuluft	[-] ,

mit der maximalen Sauerstoffzufuhr rate $Q_{O_2\max}$,

$$Q_{O_2\max}(t) = \frac{M_{O_2} \cdot F_{nG}(t)}{V_{nM} \cdot V_L(t)} \quad (2)$$

mit	M_{O_2}	:= Sauerstoffmolmasse	$[g mol^{-1}]$
	F_{nG}	:= Begasungs rate	$[l h^{-1}]$
	V_{nM}	:= molares Normvolumen	[l]
	V_L	:= Flüssigphasenvolumen	[l] ,

wird mit der Sauerstofftransferrate OTR,

$$\text{OTR}(t) = \text{OTR}_{\max}(t) \cdot \left(x_{\text{OG}}(t) - \frac{p_{\text{Gcal}} \cdot x_{\text{OGcal}}}{p_{\text{G}}(t)} \cdot \frac{p_{\text{O}_2}(t)}{100 \%} \right) \quad (3)$$

mit	OTR	:= Sauerstofftransfertrate	[g l ⁻¹ h ⁻¹]
	OTR _{max}	:= maximale Sauerstoffeintragsrate	[g l ⁻¹ h ⁻¹]
	p _{Gcal}	:= absoluter Druck der Gasphase bei pO ₂ -Sondenkalibrierung	[Pa]
	x _{OGcal}	:= Sauerstoffmolenbruch der Zuluft bei pO ₂ -Sondenkalibrierung	[-]
	p _G	:= absoluter Druck der Gasphase im Bioreaktor	[Pa]
	pO ₂	:= Sauerstoffpartialdruck	[%] ,

gleichgesetzt und die resultierende Gleichung nach pO₂,

$$p_{\text{O}_2}(t) = \left(x_{\text{OG}}(t) - \frac{Q_{\text{O}_2}(t)}{\text{OTR}_{\max}(t)} \right) \cdot \frac{p_{\text{G}}(t)}{p_{\text{Gcal}} \cdot x_{\text{OGcal}}} \cdot 100 \% \quad (4)$$

aufgelöst.

Um die maximal mögliche Sauerstofftransfertrate OTR_{max},

$$\text{OTR}_{\max}(t) = k_{\text{L}}a(t) \cdot \frac{p_{\text{G}}(t) \cdot \left(1 + \sum_{i=1}^4 K_{i\text{O}_2} \cdot \vartheta_{\text{L}}^i \right)}{H_{\text{nO}_2}} \quad (5)$$

mit	k _L a	:= volumenbezogener Stoffübergangskoeffizient	[h ⁻¹]
	K _{iO₂}	:= Bunsenkonstanten	[°C ⁻¹]
	ϑ _L	:= Temperatur der Flüssigphase	[°C]
	H _{nO₂}	:= theor. Henrykoeffizient für O ₂ unter Normalbedingungen	[Nmkg ⁻¹] ,

zu bestimmen, wird der volumenbezogener Stoffübergangskoeffizient k_La,

$$k_{\text{L}}a(t) = k_{\text{L}}a_{\max} \cdot \left(\frac{N_{\text{St}}(t)}{N_{\text{Stmax}}} \right)^{3\alpha} \cdot \left(\frac{V_{\text{L}}(t)}{V_{\text{Lmax}}} \right)^{-\alpha} \cdot \left(\frac{F_{\text{nG}}(t)}{F_{\text{nGmax}}} \right)^{\beta} \quad (6)$$

mit k_La_{max} := größter volumenbezogener Stoffübergangskoeffizient bei dem α und β bestimmt wurden [h⁻¹]

	N _{St}	:= Rührerdrehzahl	[min ⁻¹]
	N _{Stmax}	:= größte Rührerdrehzahl bei der α und β bestimmt wurden	[min ⁻¹]
	V _{Lmax}	:= größtes Flüssigphasenvolumen bei dem α und β bestimmt wurden	[l]
	F _{nGmax}	:= größte Begasungsrate bei der α und β bestimmt wurden	[lh ⁻¹] ,

aus der Korrelation nach van't Riet ermittelt. Dafür werden die Rührerdrehzahl, das Flüssigvolumen und die Begasungsrate variiert und die pO_2 -Ladekurve durch das Umstellen von Stickstoff- auf Sauerstoffbegasung aufgenommen. Aus der pO_2 -Ladekurve wird dann der k_La berechnet. Dafür wird eine Abstraktion des Sauerstoffsystems im Reaktor über ein elektrisches System benutzt. Die Übergangsgrößen Q_{O_2max} und OTR_{max} werden dabei als Widerstände interpretiert, der OTR und Q_{O_2} als fließender Strom, x_{OGin} , x_{OG} und Sauerstoffmolenbruch der Flüssigphase x_{OL} als Spannungen und die Kapazität der einzelnen Phasen als Kondensatorkapazitäten. Daraus ergibt sich ein System von Differentialgleichungen, mit denen dann der k_La als eine Zeitkonstante des Systems errechnet werden kann. Die verschiedenen bestimmten k_La -Werte werden benutzt, um nach der Methode der kleinsten Fehlerquadrate die van't Riet Parameter α und β zu bestimmen. Bei der Methode der kleinsten Quadrate wird versucht, den quadratischen Abstand zwischen der durch die Korrelation nach van't Riet bestimmten k_La -Werten und den berechneten k_La -Werten durch Anpassen der Modellparameter zu minimieren.

2.2.2 „White Box“-Modell für die Methanolkonzentration c_{S2M}

Für das „White Box“-Modell wird vom generellen Konzentrationsgleichgewicht $\dot{c}_{IM}(t)$,

$$\dot{c}_{IM}(t) = \frac{F_R(t) \cdot c_{IR} - (F_{in}(t) - F_Z(t)) \cdot c_{IM}(t)}{V_M(t)} + \frac{V_L(t)}{V_M(t)} \cdot (ITR(t) \pm q_{I/X}(t) \cdot c_{XL}(t)) \quad (7)$$

mit	\dot{c}_{IM}	:= Konzentrationsänderung der Komponente I in der Phase M	$[g l^{-1} h^{-1}]$
	F_R	:= Zuflussrate aus Reservoir R	$[l h^{-1}]$
	c_{IR}	:= Konzentration der Komponente I im Reservoir	$[g l^{-1}]$
	F_{in}	:= Summe aller einfließenden Volumenströme	$[l h^{-1}]$
	F_Z	:= Volumenverlustrate der Medienphase M zur Zellphase Z	$[l h^{-1}]$
	c_{IM}	:= Konzentration der Komponente I in der Phase M	$[g l^{-1}]$
	V_M	:= Volumen der Medienphase M	$[l]$
	ITR	:= Transferrate der Komponente I über die Phasengrenze	$[g l^{-1} h^{-1}]$
	$q_{I/X}$:= zellspezifische Reaktionsrate der Komponente I	$[h^{-1}]$
	c_{XL}	:= Zelldichte in L	$[g l^{-1}]$,

ausgegangen. Die Transferrate von Methanol $S2TR(t)$ zwischen der Medienphase und der Gasphase wird als Null angenommen.

Die Zuflussraten F_K ,

$$F_K(t_i) = \frac{m_K(t_{i-1}) - m_K(t_{i+1})}{(t_{i+1} - t_{i-1}) \cdot \rho_K} \quad (8)$$

mit	F_K	:= Zufütterraterate aus Reservoir K	$[g l^{-1} h^{-1}]$
	m_K	:= Gewicht des Reservoirs K	$[g]$
	t	:= Zeitpunkt der Messung	$[h]$
	ρ_K	:= Dichte im Reservoir K	$[g l^{-1}]$,

mit $K = R1, R2, T1, T2, AF$, wird über die Gewichtsänderung der jeweiligen Waagensignale der Reservoirs bestimmt.

Die Zufütterraterate aller Komponenten F_{in} ,

$$F_{in}(t) = \sum F_K(t) \quad (9)$$

bestimmt sich aus der Summe aller einfließenden Flüssigvolumenströme.

Für die Volumenverlustrate F_Z ,

$$F_Z(t) = q_{X/X}(t) \cdot (V_L(t) - V_M(t)) \quad (10)$$

mit $q_{X/X} :=$ Wachstumsrate $[h^{-1}]$,

wird das Volumen der Medienphase V_M ,

$$V_M(t) = V_L(t) \cdot \left(1 - \frac{\alpha_{Z/X} \cdot c_{XL}(t)}{\rho_Z}\right) \quad (11)$$

mit $\alpha_{Z/X} :=$ Biotrockenmasse zu Biofeuchtmasse Verhältnis $[-]$

$\rho_Z :=$ Dichte der Biofeuchtmassephase Z $[g^{-1}]$,

benötigt.

Die Konzentrationsänderung $\dot{c}_{S2M}(t)$,

$$\dot{c}_{S2M}(t) = \frac{c_{S2M}(t_{i+1}) - c_{S2M}(t_{i-1})}{t_{i+1} - t_{i-1}} \quad (12)$$

mit $\dot{c}_{S2M} :=$ Änderung der Methanolkonzentration in der Medienphase M $[g^{-1}h^{-1}]$

$c_{S2M} :=$ Methanolkonzentration in der Medienphase M $[g^{-1}]$,

kann annäherungsweise über den mittleren Differenzquotienten bestimmt werden.

Aus Gleichung (7) bis (12) kann dann die Methanolkonzentration $c_{S2M}(t_{i+1})$,

$$c_{S2M}(t_{i+1}) = \left(\frac{(F_{R2}(t_i) \cdot c_{S2R2} - (F_{in}(t_i) - F_Z(t_i)) \cdot c_{S2M}(t_i))}{V_M(t_i)} + \frac{V_L(t_i)}{V_M(t_i)} \cdot q_{S2/X}(t_i) \cdot c_{XL}(t_i) \right) \cdot (t_{i+1} - t_{i-1}) + c_{S2M}(t_{i-1}) \quad (13)$$

mit $F_{R2} :=$ Methanolzufüterraterate $[lh^{-1}]$

$c_{S2R2} :=$ Methanolkonzentration im Reservoir $[g^{-1}]$

$q_{S2/X} :=$ zellspezifische Methanolaufnahmerate $[h^{-1}]$,

berechnet werden. Dafür müssen die ersten zwei Methanolkonzentrationswerte vorgegeben werden.

Zudem muss die Methanolreaktionsrate bestimmt werden. Dafür wird $q_{S2/X}(t)$,

$$q_{S2/X}(t) = \frac{F_{R2}(t) \cdot c_{S2R2}}{V_L(t) \cdot c_{XL}(t)} - \left(\frac{F_{in}(t)}{V_L(t)} - \frac{\alpha_{Z/X} \cdot q_{X/X}(t)}{\rho_Z} \cdot c_{XL}(t) \right) \cdot \frac{c_{S2M}(t)}{c_{XL}(t)} - \left(\frac{\rho_Z - \alpha_{Z/X} \cdot c_{XL}(t)}{\rho_Z} \right) \cdot \frac{\dot{c}_{S2M}(t)}{c_{XL}(t)} \quad (14)$$

bestimmt. Die dafür benötigte zellspezifische Wachstumsrate $q_{X/X}(t)$,

$$q_{X/X}(t) = \frac{F_{in}(t)}{V_L(t)} + \frac{\dot{c}_{XL}(t)}{c_{XL}(t)} \quad (15)$$

mit $\dot{c}_{XL} :=$ Änderung der Zellkonzentration $[lh^{-1}]$,

wird ebenfalls berechnet und eingesetzt.

Die als zeitunabhängig angenommenen Reaktionsraten $q_{I/X}$ mit $I = S2, X$ können dann aus Gleichung (14) und (15) mit einer linearen Regressionsanalyse bestimmt werden. Zwischen den Messdaten von $c_{XL}(t)$ und den berechneten Daten $q_{I/X}(t)$ wurde mit Hilfe der Methode der kleinsten Quadrate eine lineare Beziehung hergestellt, sodass sich $q_{I/X}$,

$$q_{I/X} = \frac{(q_{I/X}(t) \cdot c_{XL}(t))}{c_{XL}(t)} \quad (16)$$

ergibt.

2.3 „Black Box“ und „Grey Box“-Modelle

Häufig ist das Verwenden eines „White Box“-Modells nicht ausreichend um die Prozessdaten genügend zu beschreiben; weil zum Beispiel die biologischen, chemischen und physikalischen Zusammenhänge zu komplex sind, um sie durch Parametrisierung ausreichend darstellen zu können, die Ermittlung der Parameter zu arbeitsaufwendig oder die Bestimmung der benötigten Zustandsgleichungen zu komplex ist.

Für solche Fälle kann ein „Black Box“-Modell von Vorteil sein. Bei diesem werden die Inputs und Outputs nicht über bekannte Prozessgleichungen miteinander verknüpft, sondern stattdessen werden die Struktur und Parameter des Modells durch die Daten selbst bestimmt. Dadurch können die Prozesse auch bei einem geringeren Verständnis der zugrundeliegenden mathematischen Zusammenhänge beschrieben werden [Nelles, 2001].

Die Bezeichnung „Black Box“ rührt daher, dass das Modell selbst keine Beziehung zu den zugrundeliegenden Prozessmechanismen hat und nur auf empirischen Daten beruht, daher werden auch als datengetrieben bezeichnet. [Kadlec et al., 2009] Das Training, beziehungsweise Erstellen, eines „Black Box“-Modelles findet mittels bekannter Messwerte statt. Diese Art der Modellfindung hat einige Vorteile, zum einen, dass kein tiefes Wissen über den Pro-

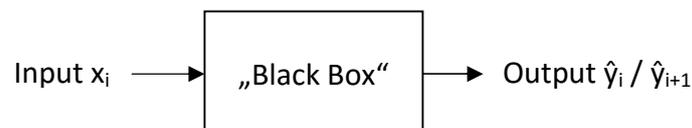


Abbildung 1: Schematische Darstellung der "Black Box" Vorgehensweise. Bei dieser Art der Modellfindung liegt der Fokus auf den empirischen Daten und nicht auf den mathematischen Zusammenhängen. x_i bezeichnet dabei das Inputarray, das zum Zeitpunkt i in das Modell eingeht. Solche Inputwerte können zum Beispiel Rührerdrehzahl, Molenbrüche und andere Messwerte sein. \hat{y} bezeichnet den durch das Modell vorhergesagten Zielwert, wie zum Beispiel pO_2 oder c_{S2M} .

zess vonnöten ist, und zum anderen, dass die Entwicklung dieser Modelle zügig von statten gehen kann. Nachteile sind, dass diese Modelle intensiv validiert werden müssen, um die Generalisierung, also die Verallgemeinerungsfähigkeit und somit die Fähigkeit des Modells spätere Prozesse zu beschreiben, zu gewährleisten. Es besteht, besonders bei komplexen Modellierungswegen, wenn viele Parameter den Zusammenhang zwischen wenig Variablen erklären, eine starke Gefahr des Overfittings, und somit eine Verringerung der Generalisierungsfähigkeit. Beim Overfitting wird, statt dem zugrundeliegenden Prozess, das Signalrauschen der Daten beschrieben, die genutzt wurden um das Modell zu entwickeln. Des Weiteren ist häufiges Neukalibrieren, beziehungsweise Neutrainieren, vonnöten und die Genauigkeit der Vorhersage ist stark an die Güte der Daten gebunden. Bekannte „Black Box“-Verfahren sind zum Beispiel die Hauptkomponentenregression und künstliche neuronale Netzwerke. [Nelles, 2001; Kadlec et al., 2009]

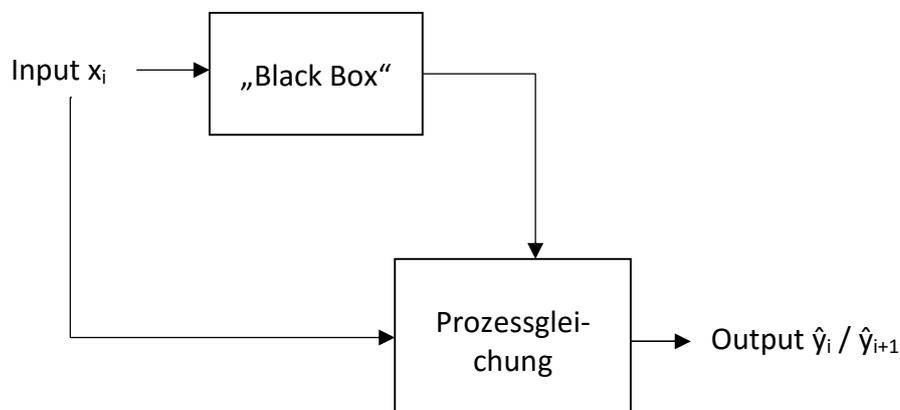


Abbildung 2: Schematische Darstellung der verwendeten "Grey Box". Die Messwerte fließen dort in ein "Black Box"-Modell ein, berechnen dort aber nicht direkt die zu messende Größe, sondern einen Parameter der bekannten Gleichungssysteme.

Die „Grey Box“ ist eine Mischung aus „White“ und „Black Box“. Diese können einen Kompromiss bieten um die Nachteile der jeweiligen Extreme auszugleichen. Das Mischungsverhältnis der „Black Box“ und „White Box“-Anteile kann beliebig groß sein. Zum Beispiel kön-

nen die komplex zu bestimmenden Parameter der „White Box“-Zusammenhänge mit Hilfe der „Black Box“ bestimmt werden, um dann die gewünschte Messgröße bestimmen zu können. [Nellens, 2001]

2.3.1 „Grey Box“-Mathematik

Es werden verschiedene Parameter ausgewählt, die durch den „Black Box“-Anteil des Modells bestimmt werden sollen, der hier aus einem neuronalen Netz besteht. Die Ausgabe des Netzes wird dann in die Formeln eingesetzt, um so den Sauerstoffpartialdruck p_{O_2} , beziehungsweise die Methanolkonzentration c_{S2M} , zu berechnen.

2.3.1.1 Die Methanolkonzentration c_{S2M} über das Konzentrationsgleichgewicht

Die Methanolkonzentration $c_{S2M}(t_{i+1})$ kann mit Hilfe von Gleichung (13) über das Konzentrationsgleichgewicht bestimmt werden. Dafür wird zum einen die Substrataufnahmerate $q_{S2/X}$ als y_{gC_1} ,

$$y_{gC_1}(t) = q_{S2/X}(t) \quad (17)$$

durch die „Black Box“ bestimmt, wodurch dann die erste „Grey Box“-Ausgabe für Methanolkonzentration $c_{S2M_g_1}(t_{i+1})$,

$$c_{S2M_g_1}(t_{i+1}) = \left(\frac{(F_{R2}(t_i) \cdot c_{S2R} - (F_{in}(t_i) - F_Z(t_i)) \cdot c_{S2M}(t_i))}{V_M(t_i)} + \frac{V_L(t_i)}{V_M(t_i)} \cdot y_{gC_1}(t_i) \cdot c_{XL}(t_i) \right) \cdot (t_{i+1} - t_{i-1}) + c_{S2M}(t_{i-1}) \quad (18)$$

ermittelt wird.

Zudem wird, aufgrund des starken Rauschens, das die berechnete Substrataufnahmerate $q_{S2/X}$ aufwies, y_{gC_2} ,

$$y_{gC_2}(t) = \frac{V_L(t)}{V_M(t)} \cdot q_{S2/X}(t) \cdot c_{XL}(t) \quad (19)$$

benutzt, wodurch sich die zweite „Grey Box“-Ausgabe für Methanolkonzentration $c_{S2M_g_2}(t)$,

$$c_{S2M_g_2}(t_{i+1}) = \left(\frac{(F_{R2}(t_i) \cdot c_{S2R} - (F_{in}(t_i) - F_Z(t_i)) \cdot c_{S2M}(t_i))}{V_M(t_i)} + y_{gC_2}(t_i) \right) \cdot (t_{i+1} - t_{i-1}) + c_{S2M}(t_{i-1}) \quad (20)$$

ergibt.

2.3.1.2 pO_2 – quasi-stationäres Verhältnis von Sauerstofftransferrate OTR und Sauerstoffzufuhr rate Q_{O_2}

Der p_{O_2} kann über den quasi-stationären Zusammenhang von Sauerstoffzufuhr rate Q_{O_2} und Sauerstofftransferrate OTR errechnet werden. Dafür wird der Q_{O_2} gleich dem OTR gesetzt und nach p_{O_2} aufgelöst, wie in Gleichung (4) gezeigt. Es werden zwei verschiedene „Black Box“-Outputs betrachtet. Zum einen wird der volumetrische Stoffübergangskoeffizient $k_L a$ als $y_{g_OQ_1}$,

$$y_{g_OQ_1}(t) = k_L a(t) \quad (21)$$

Betrachtet, wodurch dann die erste „Grey Box“-Ausgabe für den Sauerstoffpartialdruck $p_{O_{2g_OQ_1}}$,

$$p_{O_{2g_OQ_1}}(t) = \left(x_{OG}(t) - \frac{Q_{O_2}(t)}{y_{g_OQ_1}(t) \cdot c_{OLmax}(t)} \right) \cdot \frac{p_G(t)}{p_{Gcal} \cdot x_{OGcal}} \cdot 100 \% \quad (22)$$

errechnet werden kann und, als Versuch der weiteren Vereinfachung der p_{O_2} -Gleichung, $y_{g_OQ_2}$,

$$y_{g_OQ_2}(t) = \frac{Q_{O_2}(t)}{k_L a(t) \cdot c_{OLmax}(t)} \quad (23)$$

wodurch die zweite „Grey Box“-Ausgabe für den Sauerstoffpartialdruck $p_{O_{2g_OQ_2}}$,

$$p_{O_{2g_OQ_2}}(t) = \left(x_{OG}(t) - y_{g_OQ_2}(t) \right) \cdot \frac{p_G(t)}{p_{Gcal} \cdot x_{OGcal}} \cdot 100 \% \quad (24)$$

errechnet werden konnte.

2.3.1.3 pO_2 – quasi-stationäres Verhältnis von Sauerstofftransferrate OTR und Sauerstoffaufnahme rate OUR

Der pO_2 kann auch über den quasi-stationären Zusammenhang von OTR und OUR berechnet werden. Die Sauerstoffaufnahme rate OUR,

$$OUR(t) = q_{O/X}(t) \cdot c_{XL}(t) \quad (25)$$

mit OUR := volumetrische Sauerstoffaufnahme rate $[gl^{-1}h^{-1}]$
 $q_{O/X}$:= zellspezifische Sauerstoffaufnahme rate $[gl^{-1}h^{-1}]$,

wird gleich Gleichung (3), der Sauerstofftransferrate OTR, gesetzt und nach pO_2 ,

$$pO_2(t) = \left(x_{OG}(t) - \frac{q_{O/X}(t) \cdot c_{XL}(t)}{OTR_{max}(t)} \right) \cdot \frac{p_G(t)}{p_{Gcal} \cdot x_{OGcal}} \cdot 100 \% \quad (26)$$

aufgelöst.

Es werden zwei verschiedene „Black Box“-Anteile untersucht. Zum einen das Verhältnis der beiden komplex zu bestimmenden Parameter $q_{O/X}$ und OTR_{max} als $y_{g_OO_1}$,

$$y_{g_OO_1}(t) = \frac{q_{O/X}(t)}{OTR_{max}(t)} \quad (27)$$

wodurch sich dann die erste „Grey Box“-Ausgabe für den Sauerstoffpartialdruck $pO_{2g_OO_1}$,

$$pO_{2g_OO_1}(t) = \left(x_{OG}(t) - y_{g_OO_1}(t) \cdot c_{XL}(t) \right) \cdot \frac{p_G(t)}{p_{Gcal} \cdot x_{OGcal}} \cdot 100 \% \quad (28)$$

ergibt.

Zum anderen, als versuchte Vereinfachung der pO_2 -Gleichung, wird $y_{g_OO_2}$,

$$y_{g_OO_2}(t) = \frac{q_{O/X}(t) \cdot c_{XL}(t)}{OTR_{max}(t)} \quad (29)$$

durch die „Black Box“ bestimmt, wodurch sich die zweite „Grey Box“-Ausgabe für den Sauerstoffpartialdruck $pO_{2g_OO_2}$,

$$pO_{2g_OO_2}(t) = \left(x_{OG}(t) - y_{g_OO_2}(t) \right) \cdot \frac{p_G(t)}{p_{Gcal} \cdot x_{OGcal}} \cdot 100 \% \quad (30)$$

ergibt.

2.3.2 Neuronale Netze

Als ein „Black Box“-Verfahren werden künstliche neuronale Netzwerke betrachtet. Bei diesen handelt es sich, vom biologischen Vorbild inspiriert, um ein vermaschtes Netzwerk aus verschiedenen Knotenpunkten, die als „Neuronen“ bezeichnet werden, und den Verbindungen zwischen diesen.

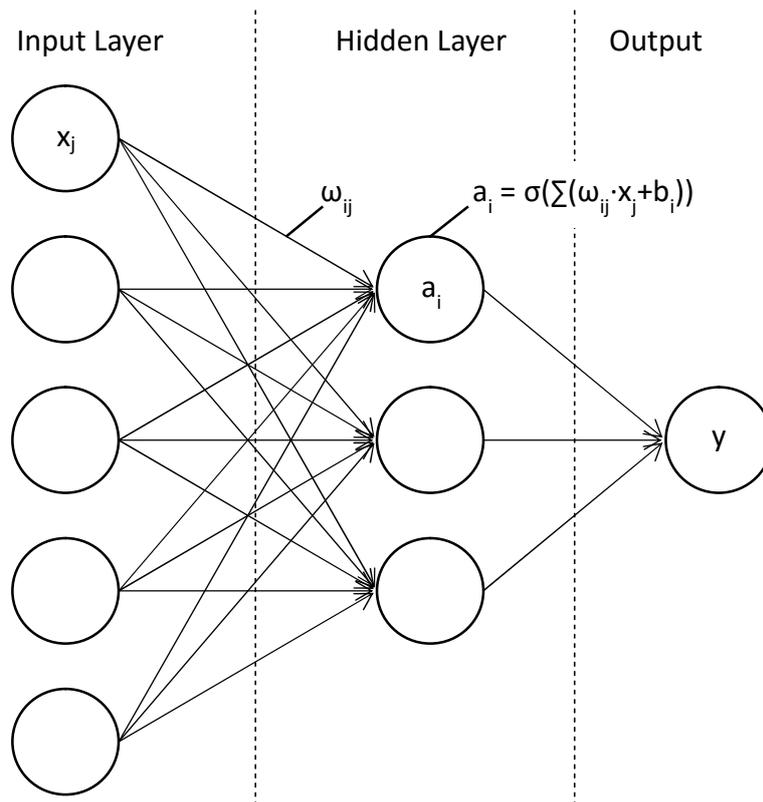


Abbildung 3: Schematische Darstellung eines einfachen Netzwerkes mit drei versteckten Neuronen. Die Inputs x_j werden vorwärts durch das Netz übertragen, daher wird dieses Netzwerk auch als Feedforward-Netzwerk bezeichnet. Die Aktivierung a_i bezeichnet den Ausgabewert des versteckten Neurons i . „Versteckt“ bedeutet, dass dieses Neuron keine Eingangs- beziehungsweise Ausgangsneuron ist. Bei der Übertragung des Inputs x wird dieser mit dem sogenannten Gewicht der Verbindung ω zwischen des Input- und des versteckten Neurons multipliziert. Die versteckten Neuronen haben einen variablen Schwellenwert b , auch Bias genannt, der die Aktivierung der Neurone modifiziert. Die Aktivierung wird mithilfe der Aktivierungsfunktion σ berechnet.

In Abbildung 3 ist ein einfaches Netzwerk dargestellt. Da die Weiterleitung des Signals nur in eine Richtung, von Input in Richtung Output stattfindet, wird dies auch als „Feedforward“-Netzwerk bezeichnet. In die Inputebene gehen die vorbereiteten Messwerte ein. Alle Inputs müssen, damit das Trainieren des Netzwerkes korrekt funktioniert, sich zwischen 0 und 1

befinden, daher ist eine Normierung der Messwerte erforderlich. Die Weitergabe des Inputs x an die nächste Ebene erfolgt über die Verbindung zwischen des Inputneurons j und des versteckten Neurons i . Diese Verbindungen haben ein Gewicht w_{ij} und der Input x_j wird mit diesem Gewicht multipliziert. Jedes Inputneuron ist mit jedem versteckten Neuron verbunden, und alle diese Verbindungen haben ihr eigenes Gewicht. Die versteckten Neuronen haben zudem alle einen variablen Schwellenwert, der auch als Bias b bezeichnet wird. Dieser modifiziert zusätzlich die Aktivität des Neurons. In dem versteckten Neuron i findet nun ein Aufsummieren aller eingehenden Werte $x_j \cdot w_{ij} + b_i$ statt und der Zahlenwert des Neurons, der auch als die Aktivierung a_i ,

$$a_i = \sigma \left(\sum_{j=1}^{n_{in}} (\omega_{ij} \cdot a_j + b_i) \right) \quad (31)$$

mit a_i := Aktivierung des Neurons i

n_{in} := Anzahl der Neuronen/Inputs der vorhergehenden Ebene

ω_{ij} := Gewicht der Verbindung des vorhergehenden Neurons j mit dem Neuron i

a_j := Aktivierung des vorhergehenden Neurons j

b_i := Bias des Neurons i ,

bezeichnet wird, wird berechnet. Die Sigmoidfunktion $\sigma(x)$,

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (32)$$

dient dabei als Aktivierungsfunktion. Diese dient dazu, eine Beziehung zwischen dem Input in das Neuron und zu dem Aktivitätslevel des Neurons herzustellen. Es existieren verschiedene Aktivierungsfunktionen, die je nach Ziel des Netzwerkes ausgewählt werden. Zum Beispiel wird bei Netzwerken, die nur 0 oder 1 ausgeben sollen, die Stufenfunktion verwendet. Für die meisten Anwendungen wird die Sigmoidfunktion verwendet, da sie gut differenzierbar ist und dies für die Lernalgorithmen vonnöten ist. [Hecht-Nielsen, 1989]

Die Anpassung der Gewichte ω_{ij} der Neuronenverbindungen und des Bias b_i der Neuronen wird mit Hilfe von Fehlerrückführung durchgeführt. Dafür wird der mittlere quadratische Fehler (MSE) von den Zielvariablen, also pO_2 oder c_{S2M} , und dem Ausgabewert des Netzes ermittelt, und dieser Fehler rückwärts durch das Netz übertragen, also rückpropagiert. Es

existieren verschiedene Fehlerrückführungsalgorithmen, die unterschiedliche Vor- und Nachteile haben. Der grundlegendste Algorithmus ist die Methode des Gradientenverfahrens, bei dem die partielle Ableitung der MSE-Funktion nach den Gewichten und Biases gebildet wird und diese genutzt wird um das Minimum der Funktion, also den kleinsten Fehler, zu finden. [Rumelhart et al., 1986] Aus diesem wurden weitere Algorithmen entwickelt. Besonders betrachtet wurden hier der Levenberg-Marquardt-Algorithmus und der Bayesian Regulation Algorithmus. Der Levenberg-Marquardt-Algorithmus ist eine Mischung des Gradientenverfahrens und des Newtonverfahrens. Ein großer Nachteil des Gradientenverfahrens an sich ist, dass bei Nähe zum Minimum der MSE-Funktion die Schrittweite sehr klein wird und somit die Geschwindigkeit mit der es in Richtung Minimum läuft sehr gering wird. Um dem entgegen zu wirken dient die Kombination mit dem Newtonverfahren. Das Newtonverfahren, auch bekannt als Tangentenverfahren, nähert sich dem Minimum über eine lokale Linearisierung der Funktion. Der Nullwert dieser Tangente wird als verbesserte Näherung zum Minimum angenommen und die Funktion an dieser Stelle wieder linearisiert. So kann iterativ das Minimum erreicht werden. Das Newtonverfahren ist rechenintensiver, aber schneller im Bereich nahe des Minimums. Dadurch konvergiert dieser Algorithmus mit höherer Geschwindigkeit als andere Algorithmen, aber ist dafür auch ressourcenlastiger. [Hagan, Menhaj, 1986]

Der Bayesian Regulation Algorithmus ist eine Erweiterung des Levenberg-Marquardt Algorithmus. Für diesen wird die MSE-Funktion um die Quadratsumme der Netzwerkgewichte und zwei Gewichtungparameter erweitert. Dies hat zur Folge, dass eine bessere Generalisierung des Netzwerkes erreicht wird, aber unter dem Nachteil, dass dies die Trainingsdauer erhöht. [Foresee, Hagan, 1997]

Eine weitere Einflussgröße auf die Generalisierung und Güte des Netzes sind die verschiedenen Hyperparameter, die angepasst werden können. Dazu gehören zum Beispiel die Lernraten, die die Schrittweite in Richtung Minimum bestimmen, die Epochenanzahl, die die Menge an Vorwärts- und nachfolgender Fehlerrückpropagierung des gesamten Trainingsdatensatz bezeichnet, Anzahl der Neuronen in der versteckten Ebene, Transferfunktion, Fehlerfunktion, und viele mehr. Schlechte Hyperparameter können dabei dafür sorgen dass niemals Konvergenz erreicht wird, können das Overfitting vergrößern oder eine schlechte An-

passung erzeugen. Insbesondere die Epochenanzahl und die Anzahl der versteckten Neuronen haben einen großen Einfluss. Bei zu wenigen Epochen, beziehungsweise Neuronen, kann es dazu führen, dass die Daten nicht ausreichend gut beschrieben werden. Bei zu vielen Epochen, beziehungsweise versteckten Neuronen, kann es zu einem Overfitting und somit zu einer schlechten Generalisierung führen. Daher wurde insbesondere auf diese zwei Hyperparameter der Fokus gelegt zusammen mit der Auswahl der Inputs. Die Optimierung der Hyperparameter wurde in vorher festgelegten Grenzen durchgeführt.

Das sich ein neuronales Netzwerk vermutlich zur Beschreibung von pO_2 und c_{S2M} eignen liegt daran, dass diese, bei ausreichend hoher Neuronenzahl, theoretisch jede kontinuierliche Funktion beliebig gut approximieren kann. Das bedeutet, für eine ausreichend große Menge an versteckten Neuronen N wird

$$|\hat{y}-y|<\varepsilon \tag{33}$$

mit \hat{y} := Aus dem Netzwerk ausgegebener Zahlenwert

y := Zahlenwert der normierten Zielvariablen, pO_2 oder c_{S2M}

ε := Genauigkeit die erfüllt werden soll,

erfüllt. Dabei ist die zu erfüllende Genauigkeit $\varepsilon > 0$, es findet daher eine Approximation der darzustellenden Funktion statt, keine absolute Abbildung. [Hornik et al., 1989; Hornik, 1991]

Nichtlineare regressive closed loop neuronale Netzwerke wurden ebenfalls betrachtet. Diese unterscheiden sich insofern von den normalen Feedforward-Netzen, als dass zum einen eine Zeitverzögerung Teil des Netzes ist, was bedeutet, dass zu einem Zeitpunkt i alle Inputs von $t = i - t_p, i - t_p + 1, \dots, i$, mit $t_p = 0, 1, \dots$ mit eingehen. t_p kann dabei, je nach vorhandener Rechenleistung und gewünschter maximaler Trainingsdauer sehr groß werden und kann ebenfalls optimiert werden. t_p sorgt dafür, dass nicht nur der aktuelle Zeitpunkt i bei der Bestimmung des Ergebnisses \hat{y}_i mit eingeht, sondern auch alle $i - t_p$ Zeitpunkte davor. Dies kann helfen, die Trägheit des Messsystems zu simulieren. Zum anderen geht bei diesem Netz der Output \hat{y} nach Ausgabe durch das Netz wieder als Input mit ein, wodurch eine Selbstabhängigkeit simuliert werden kann.

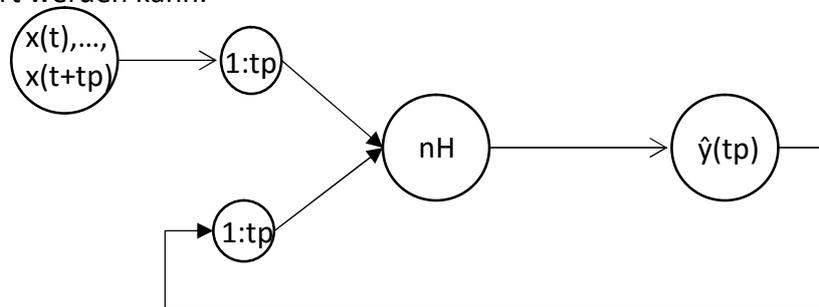


Abbildung 4: Schematische Darstellung eines nichtlinearen regressive closed loop neuronalen Netzwerkes. Der Output \hat{y} des Netzes geht zusammen mit x als Input mit ein. Zudem wird eine Zeitverzögerung t_p miteingebracht. nH ist die Anzahl an versteckten Neuronen.

2.3.3 Hauptkomponentenregression

Bei der Hauptkomponentenregression wird die Hauptkomponentenanalyse (PCA) mit einer multiplen linearen Regression verbunden. Die Hauptkomponentenanalyse ist ein Verfahren zur Reduktion des Datenvolumens bei möglichst kleinem Informationsverlust. Die eingehenden Messdaten werden gewichtet und aus diesen die sogenannten Hauptkomponenten durch Linearkombination bestimmt. Die Hauptkomponenten stellen ein neues Koordinatensystem für die Eingangsdaten dar. Auf den neuen Koordinatenachsen sollen diese Eingangsdaten die größte Varianz aufzeigen. Bei diesem neuen Koordinatensystem können dann höher dimensionale Achsen wegfallen, da die Daten ausreichend in den geringeren Dimensionen erklärt werden können. Für die PCA werden die sogenannten Score- und Loadingwerte benötigt. Die Loadings sind die Hauptkomponenten und bilden den neuen Koordinatenraum in dem die Daten beschrieben werden sollen. Um nun die Ausgangsdaten mit diesem neuen Raum beschreiben zu können, wird die Score-Matrix berechnet, welche die Daten in das neue Koordinatensystem transformiert.

Um die benötigte Anzahl an Hauptkomponenten zu beurteilen wird die erklärte Gesamtvarianz der Daten auf den Hauptkomponenten verwendet. Diese Varianz nimmt für höhere Komponenten immer weniger zu, und höheren Komponenten können dementsprechend weggelassen werden. Dabei werden drei Regeln für die Signifikanz einer Komponente aufgestellt:

(1) Die totale Varianz der Eingangsvariablenmatrix X , beziehungsweise Zielvariablenmatrix Y , die durch die Komponente vorhergesagt werden, ist größer als 0,

(2) die Varianz einer Variable x_j , beziehungsweise y , der durch die Komponente vorhergesagt wird ist größer als $\sqrt{\text{Anzahl } x\text{-Variablen} \cdot \log_{10}(\max(10, \text{Anzahl } x\text{-Variablen} - 20))}$ und

(3) die Varianz einer Variable x_j , beziehungsweise y , der durch die Komponente vorhergesagt wird ist größer als $\sqrt{\text{Anzahl } y\text{-Variablen}}$. Wenn diese drei Regeln erfüllt sind, gilt die Komponente als signifikant und geht in die Berechnung mit ein.

Bei der Hauptkomponentenregression wird zuerst eine PCA der Messdatenmatrix X ohne den zu berechnenden y -Wert durchgeführt. Die Matrix X ,

$$X = T \cdot P^T \quad (34)$$

mit X := mittenzentrierte Messdatenmatrix

T := Scores

P := Loadings,

wird durch die Scores und Loadings beschrieben. Die Zielvariable y ,

$$y = T \cdot q + f \quad (35)$$

mit y := Zielvariable

q, f := Regressionskoeffizienten,

kann dann mit Hilfe der Scorematrix berechnet werden. Zum Schluss wird die Gleichung (34) nach der Scorematrix T umgestellt und in Gleichung (35) eingesetzt um die direkte Verbindung zwischen Zielvariable y und Messdatenmatrix X zu erhalten. Dadurch wird die neue Regressionsgleichung für y ,

$$y = X \cdot P \cdot q + f = X \cdot b + f \quad (36)$$

mit b := Regressionskoeffizient,

erstellt. Die Regressionskoeffizienten werden durch die Methode der kleinsten Quadrate bestimmt.

Für die PCA wird eine etwas andere Datenpräparation verwendet. Die Daten werden zuerst mittenzentriert, also der Mittelwert abgezogen, und dann durch die Standardabweichung geteilt. [Kessler, W. 2007]

2.3.4 Multivariate lineare Regression

In der multivariaten linearen Regression werden die Regressionsparameter β des linearen Regressionsmodells für y_i ,

$$y = X \cdot \beta \tag{37}$$

mit X := Messdatenmatrix

β := Regressionskoeffizientenmatrix,

mittels der Maximum-Likelihood-Schätzung bestimmt. Bei dieser wird versucht die sogenannte Loglikelihood-Funktion zu maximieren. Dafür wird iterativ ein zweistufiger Algorithmus zum Abschätzen von β durchlaufen, bis die Änderung in Loglikelihood-Funktion und in den Regressionsparametern kleiner ist als das vorgegebene Toleranzkriterium von $1,49 \cdot 10^{-8}$.

2.3.5 Extended Kalman-Filter (EKF)

Das Extended Kalman-Filter ist eine Erweiterung des Kalman-Filters, um auch nichtlineare Probleme behandeln zu können. Das Kalman-Filter ist ein Set von Gleichungen, das eine Prädiktion eines zukünftigen Zustandes durchführt und diese Vorhersage dann mit vorhandenen Messwerten versucht zu korrigieren. Um nichtlineare Probleme behandeln zu können, wird lokal differenziert, wodurch eine Linearisierung stattfindet.

Die Prädiktion der Messwerte zum nächsten Zeitpunkt $i+1$ findet mittels der bekannten Grundgleichungen aus Kapitel 2.2.1 und 2.2.2 statt. Um den Korrekturschritt später durchführen zu können, wird die aktuelle Kovarianz benötigt, die ein Maß für den Zusammenhang zwischen den Variablen ist. Die Kovarianzmatrix P ,

$$P_{i+1} = J_f \cdot P_i \cdot J_f^T + Q \tag{38}$$

mit P := Zustandskovarianzmatrix

J_f := Matrix aller partiellen Ableitungen der Grundgleichungen

Q := Prozessrauschenkovarianzmatrix,

wird in dem Prädiktionsschritt bestimmt. Danach erfolgt der Korrekturschritt. Dafür wird der sogenannte Kalman Gain benötigt. Dieser beschreibt den Zusammenhang zwischen dem vorherigen Zustand x und den Messwerten z . Der Kalman Gain K ,

$$K = P \cdot J_h^T \cdot (J_h \cdot P \cdot J_h^T + R)^{-1} \quad (39)$$

mit K := Kalman Gain

J_h := Matrix aller partiellen Ableitungen der Messgleichungen

R := Messrauschenkovarianzmatrix,

wird berechnet und damit der korrigierte Zustandsmatrix x ,

$$x = x + K \cdot (z - h(z)) \quad (40)$$

mit z := Messsignal

$h(z)$:= Messgleichungen,

ermittelt. Zum Schluss wird die Kovarianzmatrix P ,

$$P = P - K \cdot P \cdot J_h^T \quad (41)$$

aktualisiert. In dieser Arbeit wurden die Messgleichungen als $h(z) = z$ angenommen. [Chui, Chen, 2009]

2.4 Fehlerberechnung

Um die Modellausgabe bewerten zu können wird die Wurzel aus dem mittleren quadratischen Fehler, genannt RMSE,

$$\text{RMSE} = \sqrt{\frac{\sum_i^n (y_i - \hat{y}_i)^2}{n}} \quad (42)$$

mit RMSE := Wurzel des mittleren quadratischen Fehler

n := Anzahl der y-Werte

y_i := gemessener y-Wert zum Zeitpunkt i

\hat{y}_i := modellierter y-Wert zum Zeitpunkt i,

von Modellausgabe und Messwert berechnet. Der RMSE gibt den Simulationsfehler in der gleichen Einheit wie die Messwerte aus. Bei einem RMSE von unter 5 % des Set-Points ist das Modell ausreichend zur Überwachung der Hardwaresonden.

2.5 Alkoholmessung

Zur Methanolmessung kann eine Methanolsonde eingesetzt werden. Das zu messende Methanol diffundiert durch die Membran die den Sondenraum von der Flüssigphase trennt und wird dort von einem Trägergasstrom an die Zinndioxidkeramikschiicht der Elektrode herangebracht. Dort verdrängt es an der Korngrenze den vorhandenen Sauerstoff und sorgt so für einen Widerstandabfall. Diese Senkung des Widerstandes wird gemessen und daraus kann die Methanolkonzentration berechnet werden. Zur Regeneration der Zinnoxidkeramikschiicht ist diese geheizt und der Trägergasstrom stetig trägt neuen Sauerstoff heran. [Browne et al., 1990]

2.6 Sauerstoffpartialdruckmessung

Zur Messung des Sauerstoffpartialdruckes kann eine Clark-Elektrode eingesetzt werden. Der in der Flüssigphase gelöste Sauerstoff diffundiert, entsprechend seines Partialdruckes, durch die sauerstoffdurchlässige Membran. Der Sauerstoff nimmt Elektronen von der Kathode auf und wird so zu OH^- reduziert. An der Anode findet eine Oxidation statt, und es entsteht so ein Stromfluss, der proportional zum Sauerstoffpartialdruck ist. [Clark et al., 1953]

3 Material und Methoden

3.1 Datenpräparation

Für die Modellierung und die Entwicklung von Software Sensoren ist die vorhergehende Datenpräparation sehr wichtig. Insbesondere wurde der Fokus auf die Anpassung der Zeitachsen, die Exklusion von Ausreißern und die Auswahl der passenden Variablen gelegt. Zudem musste je nach Modellierungsweg noch eine eventuelle Messwertwertvorbereitung stattfinden, wie zum Beispiel eine Normierung. Die Anpassung der Zeitachsen erfolgte über eine lineare Interpolation auf eine Schrittweite von 1,2 Minuten. Dies war notwendig, da eine Aufnahme eines Messwertes unter der genutzten Prozessleitsoftware nur bei Änderung des Messwertes oder bei konstantem Verlauf einer Prozessgröße innerhalb des vorgesehenen Totbandes alle 30 min stattfindet, und daher unterschiedlich viele Messwerte, zu unterschiedlichen Messzeitpunkten, je nach Variable, vorhanden waren.

Die Transferphase, Startphase im C30 vor dem Transfer und gegebenenfalls die Endphase nach Ernte wurden als nicht modellierbar behandelt und aus den Daten rausgenommen. Dies wurde über einen Druckabfall identifiziert und entfernt. Spitzen im Waagensignal der Reaktoren, die durch die Probenahme entstanden, wurden als Ausreißer behandelt und ebenfalls entfernt.

Die Variablenauswahl erfolgte nach folgenden Kriterien: (1) Nicht dauerhaft konstant, (2) Einfluss auf oder durch die vorherzusagende Größe und (3) online erfassbar.

3.2 Festlegen von Trainings-, Validierungs- und Testdatensätzen spezifisch zu den verschiedenen Modellierungswegen und die Festlegung der Hyperparametergrenzen

Die Bestimmung der Regressionsparameter von Hauptkomponentenregression und multivariate lineare Regression und die Parameter k_{La} , $q_{X/X}$ und $q_{S2/X}$ für die „White Box“-Modelle wurde mit Hilfe der Kultivierung SK2716 bestimmt und die anderen drei Kultivierungen zum Testen der Modellausgabe verwendet.

Die verschiedenen neuronalen Netzwerke für „Grey Box“ und „Black Box“ wurden mit Hilfe der Kultivierung SK2716 trainiert. Die Daten von Kultivierung SK1616 wurden als Validierungsdatensatz verwendet. Während des Trainings wird der Validierungsdatensatz durch das

Netzwerk geschickt und der Ausgabefehler berechnet, um die Genrealisierungsfähigkeit zu erhöhen. Kultivierung SK2116 und SK2216 dienen als Testdatensätze.

Für die „Black Box“ neuronalen Netzwerke wurden die Hyperparameter Epochenanzahl epoch und Anzahl an versteckten Neuronen nH in den Grenzen die in Tabelle 1 variiert. Als Inputarray wurde dafür S_{turb} , N_{st} , x_{OG} , Δm_{R2} , x_{CG} und $c_{\text{XL_T2}}$ verwendet.

Tabelle 1: Grenzen in denen die Hyperparameter für das Feedforward-Netzwerk variiert wurden.

Parameter	Untere Grenze	Schrittweite	Obere Grenze
Anzahl versteckter Neuronen nH	1	1	30
Epochenanzahl epoch	1	10	500

Die beiden Hyperparameter wurden nacheinander optimiert. Dafür wurde bei jedem Schritt ein Netzwerk dreimal trainiert, die Wurzel des mittleren quadratischen Fehlers für Trainingsset, was die Kombination von Trainingsdatensatz und Validierungsdatensatz war, und Testset für jedes Training bestimmt und dann der Mittelwert von diesen Fehlern gebildet. Der Parameterwert mit der geringsten Fehlerausgabe wurde dann ausgewählt. Mit diesen Parametern wurden dann verschiedene Inputkonfigurationen getestet und die, die das beste Ergebnis zeigten, ausgewählt.

Für das regressive neuronale Netzwerk kam zudem noch die Zeitverzögerung t_p hinzu. Diese wurde von 1 bis 5 gleichzeitig mit der versteckten Neuronenzahl nH variiert. Auch dort wurde das Netzwerk mit dem kleinsten Fehler ausgewählt. Die Epochenanzahl epoch für die regressive closed loop Netzwerke wurde wie in Tabelle 1 variiert. Dafür wurde das Inputarray x_{CG} , m_B , Δm_{R2} und $c_{\text{XL_T2}}$ verwendet. Auch für dieses Netzwerk wurde nach Festlegen der Hyperparameter wurden verschiedene Inputkonfigurationen getestet und die mit dem besten Ergebnis ausgewählt.

3.3 Einpflegen der Modelle in SIPAT

SIMATIC SIPAT von Siemens ist ein Prozessautomatisierungstool zur Überwachung und Kontrolle von Bioprozessen. Dieses kann MATLAB®-Skripte auslesen und die Ausgaben aus diesen an das Prozesskommunikationstool OPC-CAN weiterleiten. Das Auslesen von Inputs und Outputs erfolgte nach dem korrekten Formatieren des MATLAB®-Skriptes automatisch. Bei Einbinden des Modelles in eine SIPAT-Methode wurden manuell die Inputs korrekt vergeben. OPC-CAN leitete die Outputs und Inputs an das Datenerfassung und –Regelungstool mfcs/win weiter, wo sie dann verwendet werden konnten.

3.4 Fermentation

3.4.1 Mikroorganismus

Für die Fermentationen wurde ein rekombinanter *Pichia pastoris* KM71H Stamm eingesetzt. Als methylotrophe Hefe kann dieser Methanol als Kohlenstoffquelle verwenden. Zu diesem Zweck besitzt *Pichia* zwei AOX (Alkoholoxidasen) codierende Gene, AOX1 und AOX2. Die Promotoren dieser beiden Gene, von denen der für AOX1 der stärkere ist, stehen unter Katabolitrepression. Dies bedeutet, dass nur bei Anwesenheit von Methanol und Abwesenheit eines energetisch günstigeren Substrates Glycerin diese Gene abgelesen werden.

Um diesen rekombinante Stämme herzustellen wird das Gen des Zielproteins hinter den AOX1 Promotor gestellt. Dadurch kann die Zielproteinproduktion durch Methanol induziert werden. Wenn das alkoholoxidasencodierende Gen durch das Zielgen substituiert wird, entsteht der Mut^S-Phänotyp, der aufgrund des fehlenden alkoholoxidasencodierenden Gens langsamer wächst.

3.4.2 Medienzusammensetzung

Titriert wurden die Reaktoren mit 2 M H_2PO_4 und 12,5 %-igem Ammoniak. Die Base diente zugleich auch als Stickstoffquelle.

Tabelle 2: Medienzusammensetzung

Bestandteil	Konzentration [g l^{-1}]				
	Vorkultur	ED10 Start-up	ED10 Zyklus 1	C30	Vorlagenmedium
Glycerin	20	25	15	-	630
KH_2PO_4	6,43				-
$(\text{NH}_4)_2\text{SO}_4$	1,25				-
K_2SO_4	2,15				-
$\text{CaSO}_4 \cdot 2 \text{H}_2\text{O}$	0,35				-
$\text{MgSO}_4 \cdot 7 \text{H}_2\text{O}$	4,10				-
$\text{Na}_3\text{-citrat} \cdot 2 \text{H}_2\text{O}$	1,47				-
Biotin	0,0016				-
$\text{CuSO}_4 \cdot 5 \text{H}_2\text{O}$	0,008				-
Nal	0,00032				-
$\text{MnSO}_4 \cdot \text{H}_2\text{O}$	0,012				-
$\text{Na}_2\text{MoO}_4 \cdot 2 \text{H}_2\text{O}$	0,0008				-
H_3BO_3	0,0008				-
$\text{CaSO}_4 \cdot 2\text{H}_2\text{O}$	0,002				-
$\text{CoCl}_2 \cdot 6 \text{H}_2\text{O}$	0,002				-
$\text{ZnSO}_4 \cdot 7 \text{H}_2\text{O}$	0,028				-
$\text{FeSO}_4 \cdot 7 \text{H}_2\text{O}$	0,088				-
H_2SO_4	$4,0 \mu\text{l l}^{-1}$				-

3.4.3 Ansetzen der Vorkultur

Unter der Sterilwerkbank wurden zum erkalteten, sterilen Vorkulturmedium die sterilfiltrierte PTM4- und Biotin-Stocklösung hinzugegeben. 800 ml Vorkulturmedium wurden dann gleichmäßig in vier 1-Liter-Schüttelkolben überführt und mit je 700 µl Kryokultur angeimpft. Es wurde über Nacht bei 30 °C und 150 rpm inkubiert. Der Inhalt wurde nach Inkubation in die Inokulumflasche überführt, geschwenkt und daraus 1 ml Probe für die Bestimmung der optischen Dichte entnommen. Mit dem Rest wurde der Reaktor angeimpft.

3.4.4 Vorbereitung der Bioreaktoren

Verwendet wurden zwei Reaktoren der Firma Satorius Stedim Biotech Göttingen, der BIOSTAT® ED10 mit einem Totalvolumen von 15 l und BIOSTAT® C30 mit einem Totalvolumen von 42 l.

Als Hardware-Sonden wurden eine pH-, eine pO₂-, eine Trübungs-, eine Temperatur-, eine Druck- und eine Methanolsonde eingesetzt. Für die -Schaumkontrolle wurde zudem im ED10 eine Levelsonde und im C30 eine Level-Sonde und High-Foam-Sonde eingebaut. Die Brücken für die Säure-, Base- und Antischaumzugabe beider Reaktoren befanden sich in Deckelstützen, die Glycerin- und Methanolzugabe des ED10 ebenfalls. Die Methanolbrücke des C30 befand sich in einem Seitenstützen.

Zur Kalibrierung der Methanolsonde wurden in dem ED10 9 l, und in dem C30 15 l VE-Wasser vorgelegt und die Rührerdrehzahl auf 500 rpm, der Überdruck auf 500 mbar und die Temperatur auf 22 °C eingestellt. Über einen Seitenstützen wurden dann ausgewogene Mengen Methanol eingespritzt, sodass die Konzentration im Reaktor 0,5 g l⁻¹, 1,5 g l⁻¹ und 2,5 g l⁻¹ entsprach. Nach jeder Zugabe wurde gewartet bis die Methanolsonde einen konstanten Widerstandswert für mindestens 5 Minuten ausgab und dieser notiert. Im Kalibrieremenü des Messverstärkers der Methanolsonde wurden diese drei Werte dann eingetragen.

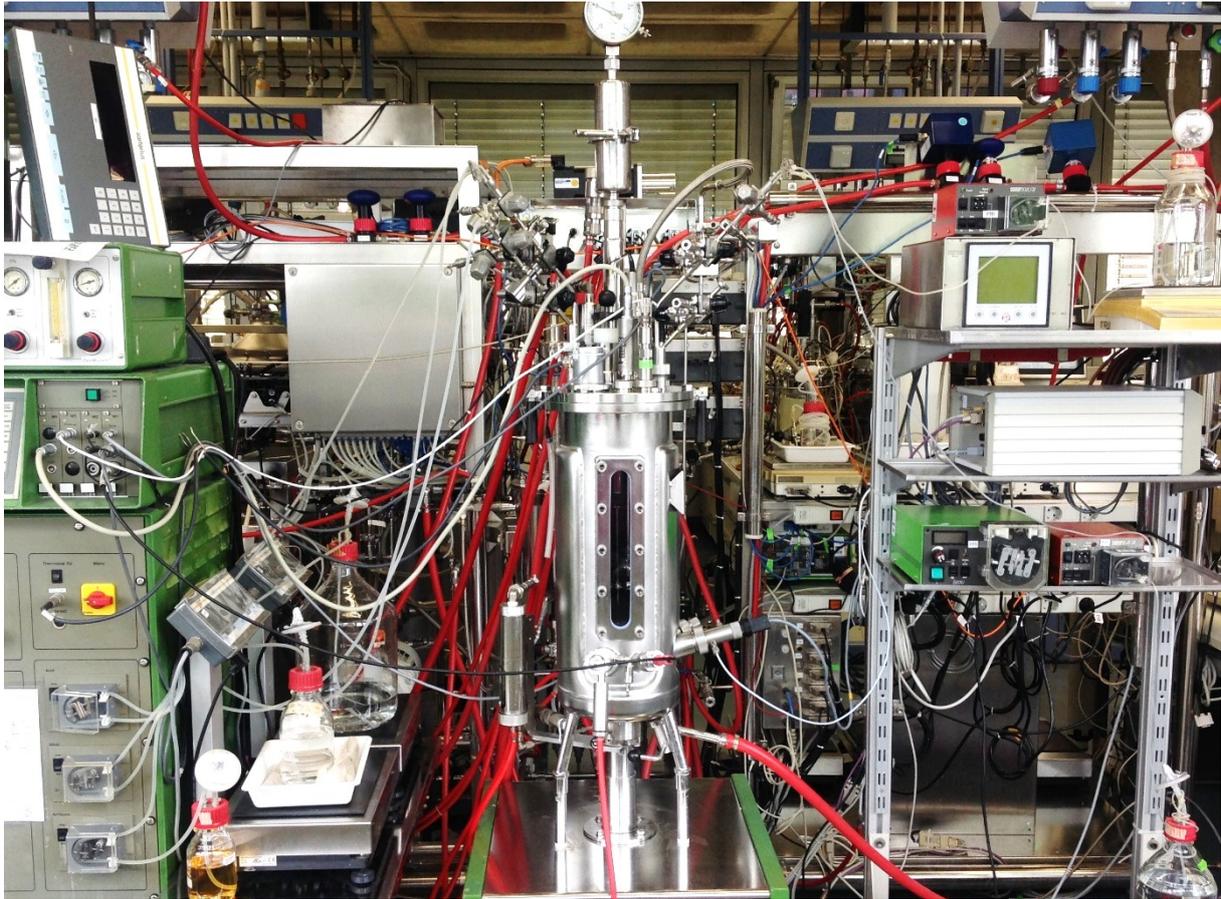


Abbildung 5: Instrumentierung Biostat ED10 und Geräteperipherie.



Abbildung 6: Instrumentierung Biostat C30 und Geräteperipherie.

Der Nullpunkt der pO₂-Sonde wurde bei Begasung mit 1,5 vvm Stickstoff eingestellt. Es wurde gewartet bis ein konstanter Ausgangsstrom ausgegeben wurde und dieser als Nullpunkt festgelegt. Dann wurde auf Luftbegasung umgestellt und nach Erreichen des konstanten Elektrodenstroms dieser als 100 % festgelegt.

Die pH-Sonde wurde bei pH 7,00 als Nullpunkt und pH 4,01 als Slope kalibriert. Die pH-Sonde wurde in die gerührte Kalibrierlösung getaucht bis die Elektrodenspannung konstant war und dies als Kalibrierpunkt verwendet.

Die sterilen Vorlagen Glycerin am ED10 und Base, Säure, Antischaum, Methanol an beiden Reaktoren wurden angeschlossen, die Ventile aber bis nach der fertigen Sterilisation des Reaktors und der Brücken geschlossen gehalten.

Nach Kalibrierung wurde das Wasser aus dem ED10 abgelassen und 7,89 l des Batchmediums eingefüllt. In den C30 wurden 10 l VE-Wasser gefüllt. Beide Reaktoren wurden bei 121 °C für 60 Minuten sterilisiert. Während der Haltephase wurde das Überdruckventil für 5 Minuten durchdämpft. Die Brücken wurden mit Hilfe eines Dampferzeugers in den Reaktor hinein sterilisiert. Ebenso wurde mit den beiden Verbindungsstücken der Transferstrecke mit den Reaktoren verfahren. Das durch den Dampfverlust veränderte Reaktorvolumen des ED10 wurde notiert und der Volumenverlust durch steriles Wasser ausgeglichen, sodass das Endvolumen mit Biotin- und PTM4-Lösung und Vorkultur 9 l betrug. Nach Sterilisation wurde das Trübungssondenkabel eingesteckt und die Schläuche von Glycerin-, Säure- und Antischaumzugabe durch Aktivieren der Pumpen gefüllt. Base- und Methanolschlauch wurden kurz vor Inokulation vorgepumpt, da durch Ausgasen durch den Silikonschlauch Gasblasen entstehen können, die die Regelung stören.

Während der Kultivierung wurde der pH auf $p_{H_w} = 5,00$, die Begasung auf $F_{nGW} = 1,5$ vvm, und der Druck auf $p_{GW} = 500$ mbar Überdruck geregelt. Die Temperatur wurde im C30 und im ED10 während der Methanol-Feed-Phase auf 22 °C und im ED10 während Batch- und Glycerin-Feed-Phase auf 30 °C geregelt.

Datenerfassung und Reglereinstellungen erfolgten über die Software MFCS/win.

3.4.5 Fermentationsverlauf im ED10

800 ml Vorkultur mit einer optischen Dichte von circa 4,5 wurden zur Inokulation vom ED10 eingesetzt. Nach Entnahme der Nullprobe erfolgte eine Probenahme alle zwei Stunden, und auch am Anfang und Ende jeder Phase. Im ED10 wurden bei jedem Zyklus drei Phasen durchlaufen: Glycerin-Batch, Glycerin-Fed-Batch und Vorinduktion mit Methanol für mindestens 3 h. Der pO_2 wurde mit Hilfe der Rührerdrehzahl auf 30 % geregelt. Das Batchende wurde über einen Anstieg des pO_2 auf über 40 % und einen Rührerdrehzahlabfall von 40 rpm detektiert über 4 Messzyklen, was 12 s entsprach. Die Dauer des fed-Batches wurde für den Start-up, also der Zyklus nach Animpfen mit der Vorkultur, auf 3 h und für den ersten Zyklus, der auf den Start-up folgte, auf 8 h festgelegt. Im Glycerin-fed-Batch wurde ein exponentielles Fütterungsprofil gefahren mit einem $\mu_w = 0,15 \text{ h}^{-1}$. In der Methanol-Feed-Phase wurde c_{S2M} auf $1,5 \text{ g l}^{-1}$ geregelt. Für die Kultivierung SK2716 wurde eine Variation der Sollwerte Rührerdrehzahl N_{Stw} , Sauerstoffpartialdruck pO_{2w} und Methanolkonzentration c_{S2Mw} durchgeführt um eine möglichst große Datenvariabilität zum Erstellen der Modelle zu erhalten.

3.4.6 Fermentationsverlauf im C30

Nach Durchlauf eines Zyklus (Start-up oder erster Zyklus) wurde der Transfer eingeleitet. Dafür wurde der C30 zuerst über ein Steigrohr geerntet. c_{XL_turb} des ED10 wurde an c_{XL_OD} angepasst damit die Berechnungen mit der tatsächlichen Zellkonzentration stattfinden, und anschließend wurde der Transfer eingeleitet. Über eine Druckdifferenz zwischen ED10 und C30 wurde dann ein automatisch berechnetes Transfervolumen V_T vom ED10 in den C30 gedrückt. Der verbleibende Rest im ED10 wurde, bis auf ein Volumen von 0,7 l verworfen. Danach erfolgt die Medienzugabe in ED10 und C30, sodass das Endvolumen im ED10 7 l betrug und im C30 eine Zellkonzentration von 15 g l^{-1} bei einem Endvolumen von 15 l erreicht wurde. Im C30 erfolgte die Produktion durch Methanolzugabe. Dabei wurde auf einen Set-Point von $c_{S2Mw} = 1,5 \text{ g l}^{-1}$ geregelt. Der pO_2 wurde durch die Rührerdrehzahl auf 30 % gehalten. Für die Kultivierung SK2716 wurde eine Variation der Sollwerte N_{Stw} , pO_{2w} , F_{R2relw} und c_{S2Mw} durchgeführt um eine möglichst große Datenvariabilität zu erhalten.

3.4.7 Probenahme und Probenvorbereitung

Aus dem Probenahmeventil wurden zu jeder Probenahme zuerst 10 ml Vorlauf abgelassen und dann circa 10 – 20 ml Probe genommen. Daraus wurde zur Überwachung der Zelldichte in Doppelbestimmung die optische Dichte und die Biotrockenmasse bestimmt.

3.4.8 Analytische Methoden

3.4.8.1 Bestimmung der Biotrockenmasse

Die Biotrockenmasse BTM wurde gravimetrisch in 1,5 ml Reaktionsgefäßen bestimmt. Die beschrifteten Gefäße wurden über Nacht bei 70 °C getrocknet, abkühlen gelassen und auf der Analysenwaage ausgewogen. 1 ml Probe wurde in ein Reaktionsgefäß gegeben, 10 min bei 20,160 g zentrifugiert und der Überstand abgenommen. Dieses Pellet wurde ebenfalls über Nacht getrocknet und nach Abkühlen gewogen. Aus der Differenz des leeren und vollen Reaktionsgefäßes wurde dann die Biotrockenmassekonzentration c_{XL_BTM} ,

$$c_{XL_BTM} = \frac{m_{total} - m_{leer}}{V_{Probe}} \quad (43)$$

Mit c_{XL_BTM} := Biotrockenmassekonzentration [g]

m_{total} := Gewicht des trockenen Reaktionsgefäßes mit Pellet [g]

m_{leer} := Gewicht des getrockneten, leeren Reaktionsgefäßes [g]

V_{Probe} := Probenvolumen, hier $1 \cdot 10^{-3}$ l [l] ,

berechnet.

3.4.8.2 Bestimmung der optischen Dichte (OD)

Die optische Dichte wurde bei einer Wellenlänge von 600 nm gegenüber VE-Wasser bestimmt. Die Probe wurde nach der erwarteten Zellkonzentration mit Wasser verdünnt, so dass die Absorption zwischen 0,1 bis 0,5 lag. Es wurde 1:5, 1:25 1:123 oder 1:625 verdünnt, wobei der letzte Verdünnungsschritt in der Küvette stattfand. Die Zellkonzentration c_{XL_OD} ,

$$c_{XL_OD} = K_{X/OD} \cdot F \cdot OD_{Probe} \quad (44)$$

mit c_{XL_OD} := Zellkonzentration aus der optischen Dichte $[g\ l^{-1}]$

$K_{X/OD}$:= Faktor von optischer Dichte zu Zellkonzentration, hier $0,29\ g\ l^{-1}$ $[g\ l^{-1}]$

F := Verdünnungsfaktor $[-]$

OD_{Probe} := optische Dichte der verdünnten Probe $[-]$,

kann dann aus der optischen Dichte errechnet werden.

4 Ergebnis und Diskussion

4.1 Verlauf von Kultivierung SK2716 und die Abweichung vom Normalprozess

Der Verlauf der Kultivierung SK2716 wurde abweichend vom Normalprozess durchgeführt, um eine möglichst große Datenvariabilität zur Modellerstellung zu erhalten. In Abbildung 7 auf der nächsten Seite ist der Verlauf der vier variierten Parameter in Kultivierung SK2716 und SK2216 zu erkennen. SK2216 wurde hier stellvertretend für den Normalprozess gewählt. Nach dem anfänglichen pO_2 -Abfall auf den Set-Point pO_{2w} von 30 % wurde dieser über die Rührerdrehzahl geregelt. Die Rührerdrehzahl stieg an bis Batch-Ende, da aufgrund des Zellwachstums mehr Sauerstoff im Verlauf der Batchphase benötigt wurde. Nach Detektion des Batch-Endes fiel N_{St} ab, da die Zellen in der Glycerin-Feed-Phase auf eine Wachstumsrate $\mu_w = 1,5 \text{ gl}^{-1}$ geregelt wurden und daher einen geringeren Sauerstoffbedarf aufwiesen. Ein weiterer Abfall der Rührerdrehzahl ist zu Start der Methanol-Feed-Phase zu erkennen, da dort sich der Zellstoffwechsel zuerst umstellen musste, bevor das Methanol verstoffwechselt werden konnte. Es ist auch die Regelung der Methanolkonzentration auf $1,5 \text{ gl}^{-1}$, nach einem anfänglichen Überschwingen, zu erkennen. In SK2716 war der Verlauf dieser vier Parameter abweichend. Im Start-Up von SK2716 im ED10 löste die Batch-End-Detektion aufgrund eines Fehlers bereits während der Glycerin-Batch-Phase aus, wodurch zuerst 3 h lang Glycerin zugefüttert wurde, und dann die Methanol-Zufütterung zu früh gestartet wurde. Die Regelung sollte dabei verschiedene jeweils immer höhere F_{R2relw} durchlaufen, sobald c_{S2M} unter je 1 gl^{-1} fiel. Die Alkoholoxidasen der Zellen standen aber unter Katabolitenrepression, wodurch die Zellen bevorzugt das Glycerin anstatt dem Methanol aufnahmen, und es daher zu einer Akkumulation des Methanols in der Medienphase bis auf fast 25 gl^{-1} kam. Da die Zellen trotzdem noch Wachstumsfähigkeit zeigten, wurde nicht abgebrochen, sondern der nächste Zyklus, Zyklus 1, gestartet. Aufgrund des fehlerhaft ausgelösten Batchendes wurden im Start-Up auch nicht alle geplanten N_{Stw} durchlaufen, sondern nur die ersten zwei. Im Zyklus 1 wurde pO_{2w} stark variiert. Für c_{S2M} wurde ein kleiner Methanolbatch von 4 gl^{-1} Spitzenkonzentration durchgeführt. Nach Abfall auf $0,7 \text{ gl}^{-1}$ folgte ein zweiter kleinerer Methanolbatch und dann eine Einstellung auf zwei Set-Points, $c_{S2Mw} = 2 \text{ gl}^{-1}$ und $1,5 \text{ gl}^{-1}$.

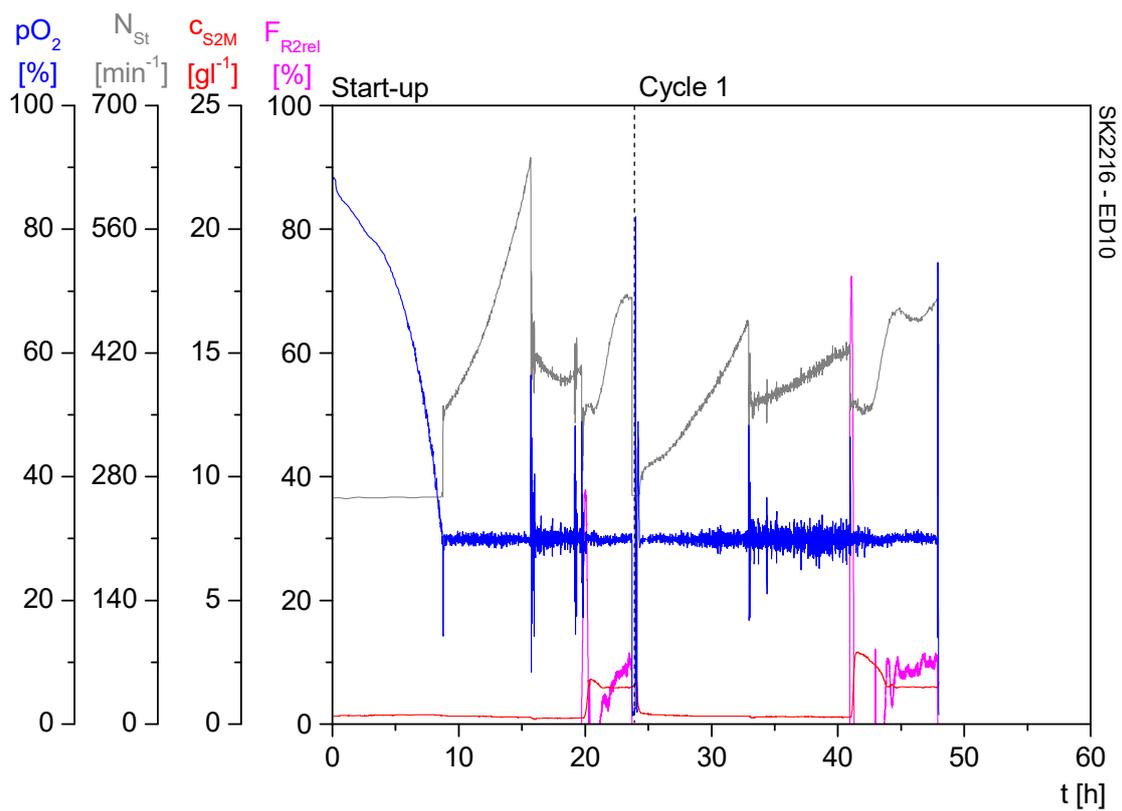
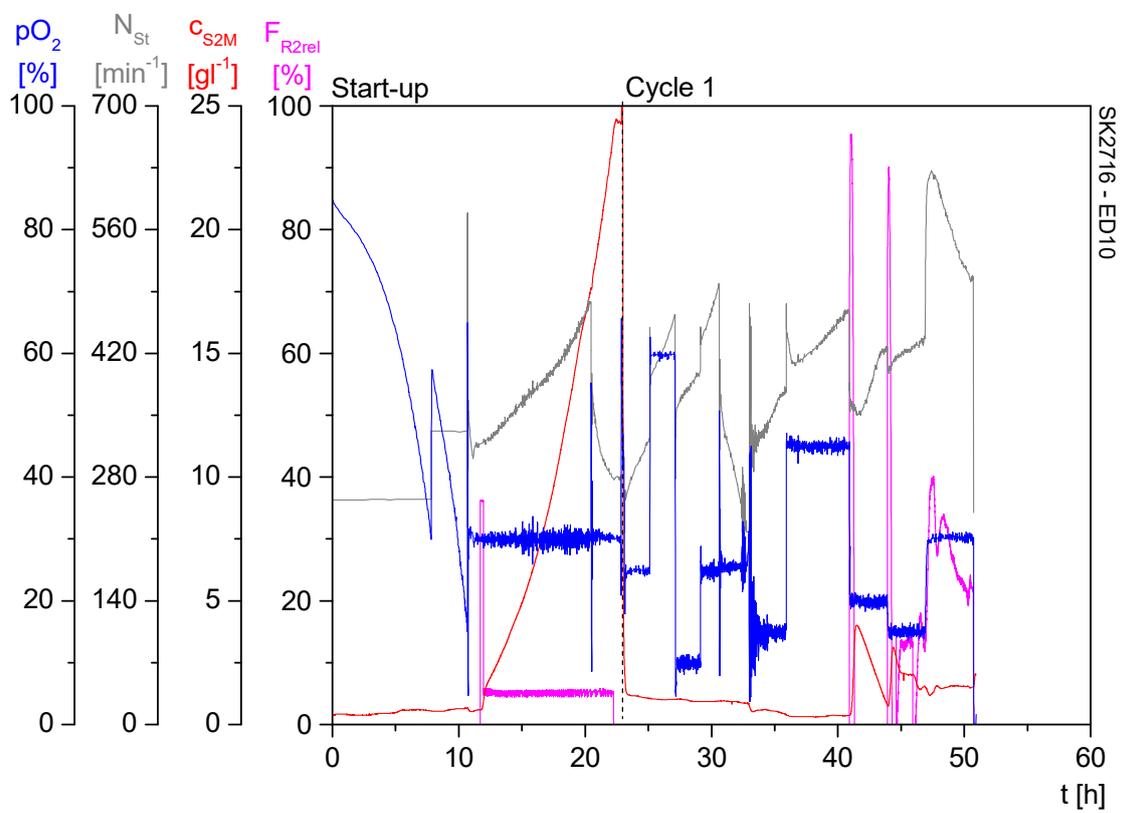


Abbildung 7: Verlauf der Parameter pO_2 , N_{St} , c_{S2M} und F_{R2rel} der Kultivierung SK2716 und SK2216 im ED10.

In Abbildung 8 auf der nächsten Seite sind Kultivierung SK2216, stellvertretend für den Normalprozess, und SK2716 im C30 dargestellt. Im Normalprozess wurde der pO_2 über die Rührerdrehzahl auf 30 % geregelt, und die Methanolkonzentration auf $1,5 \text{ g l}^{-1}$. Für SK2716 ist nur der Zyklus 1 dargestellt, nicht der Start-Up, da die Basenpumpe defekt war und erst am nächsten Tag repariert werden konnte. Daher übersäuerten die Zellen und mussten verworfen werden. Zudem mussten die Daten der ersten drei Stunden des Zyklus 1 ebenfalls verworfen werden, da die Abgasmessung für Sauerstoff ausgefallen war. In Kultivierung SK2716 wurden zuerst verschiedene N_{Stw} eingestellt, dann verschiedene pO_{2w} , wie in der Abbildung 8 zu erkennen ist. Zuerst wurde die Methanolkonzentration abfallen gelassen, um dann zwei kleine Methanolbatches aufeinander folgen zu lassen. Danach wurden zwei verschiedene F_{R2relw} eingesetzt, und dann verschiedene c_{S2Mw} . Diese Unterschiede zum Normalprozess führten zu einer Datenvariabilität um die Modelle zu trainieren.

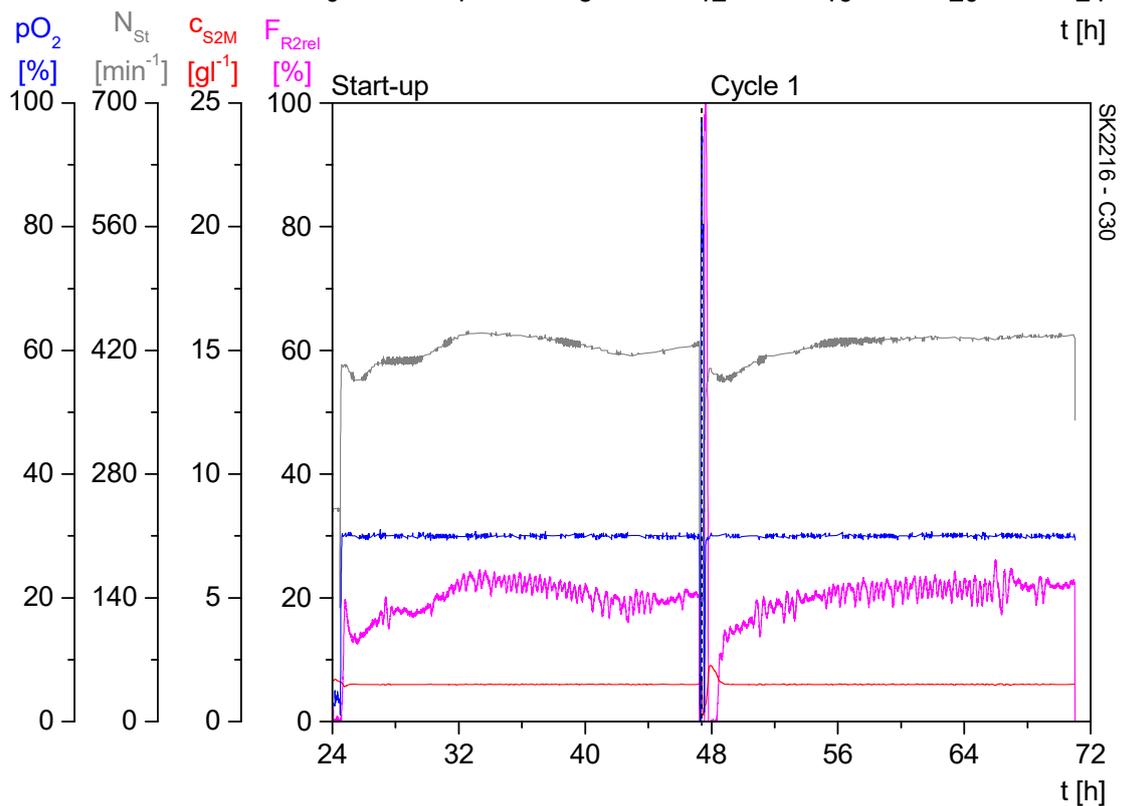
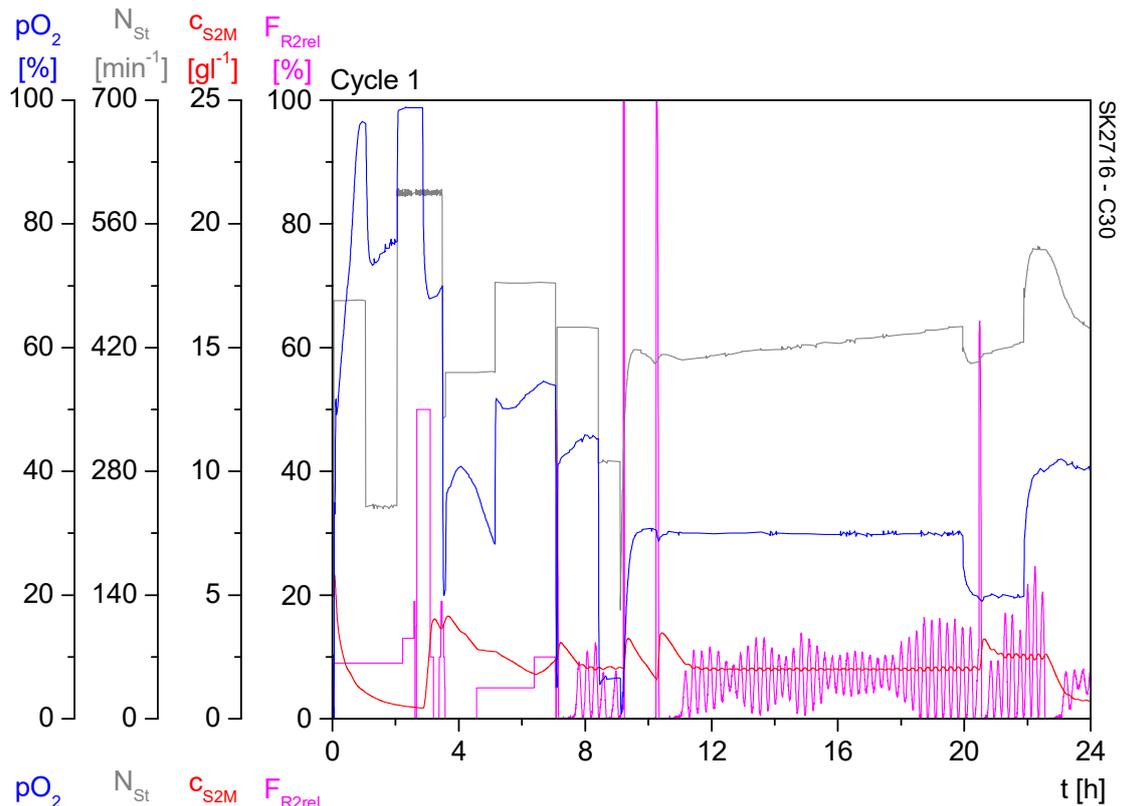


Abbildung 8: Verlauf der Parameter pO_2 , N_{St} , c_{S2M} und F_{R2rel} der Kultivierung SK2716 und SK2216 im C30.

4.2 Darstellung der Zellkonzentration aus der Basentitration

Zur Bestimmung der Zellkonzentration wurde eine Korrelation mit der Basentitration herausgearbeitet. Dies ist möglich, da die zugegebene Base Ammoniak nicht nur zur Titration der durch die Zellen freigesetzten Säure dient, sondern auch als Stickstoffquelle für das Zellwachstum.

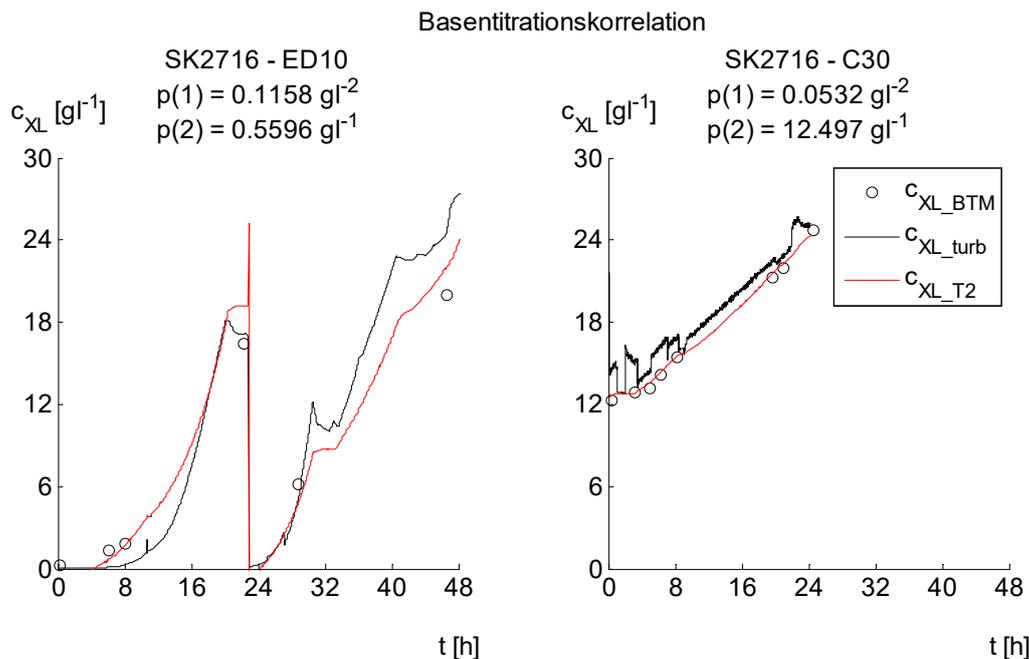


Abbildung 9: Darstellung der Korrelation des Basenverbrauches zu der Zellkonzentration im ED10 und im C30. Der Basenverbrauch wurde linear über die Formel $c_{XL_T2}(t) = p(1) \cdot \Delta m_{T2}(t) + p(2)$ mit der Zellkonzentration korreliert. Schwarz ist die aus der Trübung berechnete Zellkonzentration zu erkennen. Es ist am sprunghaften Verlauf von c_{XL_turb} im C30 erkennbar, dass eine die starke Abhängigkeit des Trübungssignals zur Rührerdrehzahl besteht.

Für den ED10 konnte die Zellkonzentration $c_{XL_T2_ED10}$,

$$c_{XL_T2_ED10}(t) = 0,1158 \text{ gl}^{-2} \cdot \Delta m_{T2}(t) + 0,5596 \text{ gl}^{-1}$$

und für den C30 $c_{XL_T2_C30}$,

$$c_{XL_T2_C30}(t) = 0,0532 \text{ gl}^{-2} \cdot \Delta m_{T2}(t) + 12,497 \text{ gl}^{-1}$$

mit $\Delta m_{T2}(t) = m_{T2}(t) - m_{T2}(t_0)$ berechnet werden. Diese Korrelation wurde für alle folgenden Schritte eingesetzt.

Normalerweise wurde die Zellkonzentration zum einen online aus den Messdaten der Trübungssonde bestimmt und zum anderen offline aus der optischen Dichte und der Biotrockenmasse. Das Problem daran war aber, dass zum einen die Trübungssonde eine starke Abhängigkeit zur Rührerdrehzahl aufwies, wie insbesondere in der rechten Hälfte von Abbildung 9 zu erkennen. Die Rührerdrehzahl wurde dort sprunghaft variiert, was ebenfalls zu Sprüngen im Trübungssondenmesssignal, und somit in c_{XL_turb} führte. Zum anderen werden die offline-Messungen für OD und BTM zu selten durchgeführt und, im Falle von BTM, erst mit einer großen Zeitverzögerung verfügbar, womit diese sich für die online-Bestimmung von Variablen nicht eignen. Die RMSE der Ausgabe aus der Korrelation betrug für beide Reaktoren und für alle Kultivierungen zwischen 1 bis 3 gl^{-1} , also war diese Art der Zellkonzentrationsbestimmung ausreichend genau.

4.3 Vergleich der verschiedenen Modellierungen

4.3.1 „White Box“

Für die „White Box“-Verfahren wurden zuerst die Wachstumsrate $q_{X/X}$ und Substrataufnahmerate $q_{S2/X}$ in der Methanol-Feed-Phase und die Parameter α und β für die $k_L a$ -Korrelation nach van't Riet bestimmt.

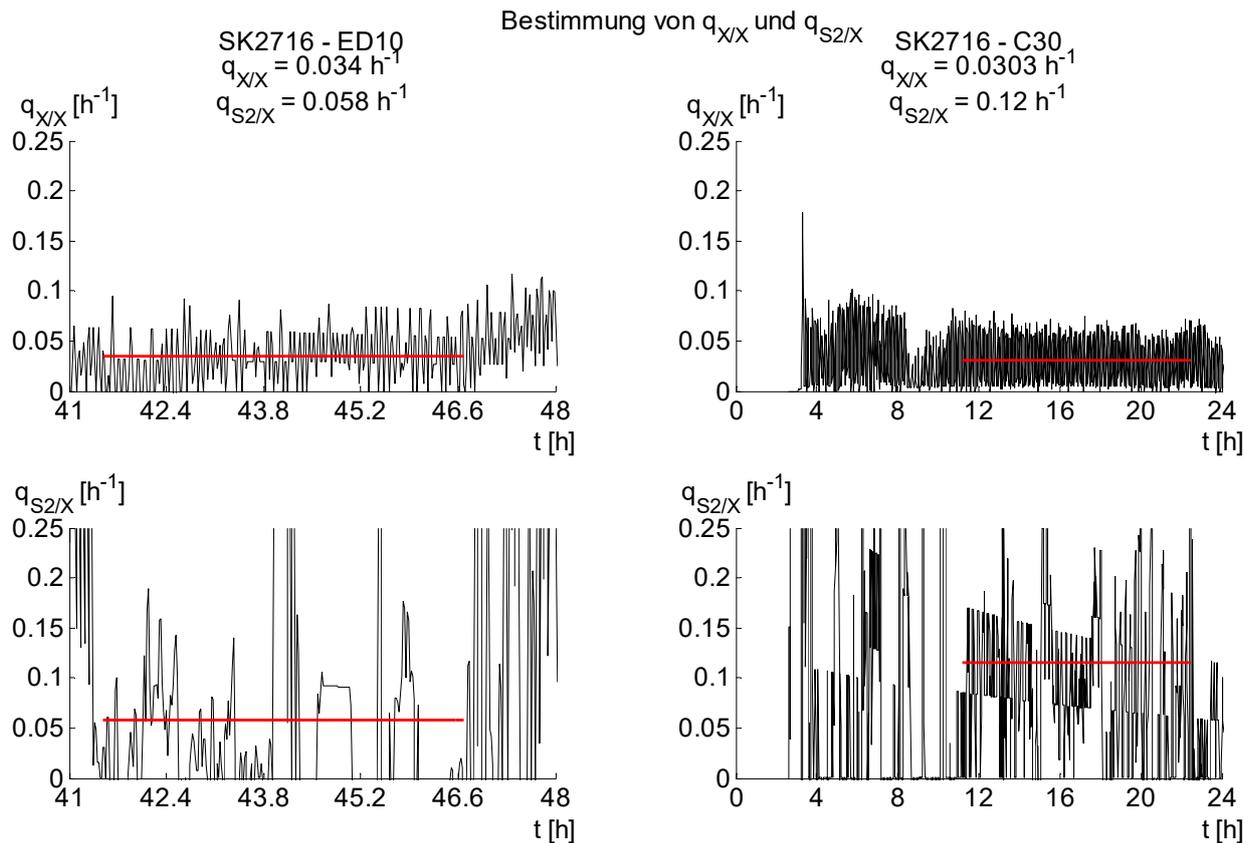


Abbildung 10: Verlauf der Reaktionsraten $q_{S2/X}(t)$ und $q_{X/X}(t)$ in beiden Reaktoren während der Kultivierung SK2716. Nur die Methanol-Feed-Phase ist dargestellt. Die rote Linie markiert den Bereich über den die Konstanten $q_{i/X}$ bestimmt wurden.

In Abbildung 10 ist der zeitliche Verlauf der nach Gleichung (14) und (15) berechneten Reaktionsraten $q_{X/X}(t)$ und $q_{S2/X}(t)$ während der Methanol-Feed-Phase dargestellt. Als rote Linie sind die mit Hilfe von Gleichung (17) ermittelten Konstanten eingezeichnet. Die Länge dieser roten Linie markiert den Bereich, über den die Konstante berechnet wurde. Die beiden Parameter $q_{X/X}$ und $q_{S2/X}$ wurden für die Methanol-Feed-Phase aller Kultivierungen als konstant angenommen. Die Wachstumsrate $q_{X/X}$ war wie erwartet, da es die gleichen Zellen waren, in beiden Reaktoren auf Methanol als Substrat fast gleich mit circa $0,03 \text{ h}^{-1}$. Die Substratauf-

nahmerate dagegen wurde im C30 als fast doppelt so hoch ermittelt. Dies könnte daran liegen, dass die Zellen zuerst ihren Stoffwechsel umstellen müssen, um das Methanol verwenden zu können, und daher die Aufnahmerate geringer war. Dagegen sprach, dass die Wachstumsrate auf der anderen Seite in beiden $0,03 \text{ h}^{-1}$ betrug. Daher mussten diese Werte mit Vorsicht betrachtet werden.

Als Wachstumsraten in der Methanol-Feed-Phase wurden $q_{X/X_ED10} = 0,034 \text{ h}^{-1}$ und $q_{X/X_C30} = 0,03 \text{ h}^{-1}$ ermittelt, als Substrataufnahmeraten $q_{S2/X_ED10} = 0,058 \text{ h}^{-1}$ und $q_{S2/X_C30} = 0,12 \text{ h}^{-1}$.

Mit diesen Werten wurde die Methanolkonzentration c_{S2M} nach Gleichung (13) berechnet.

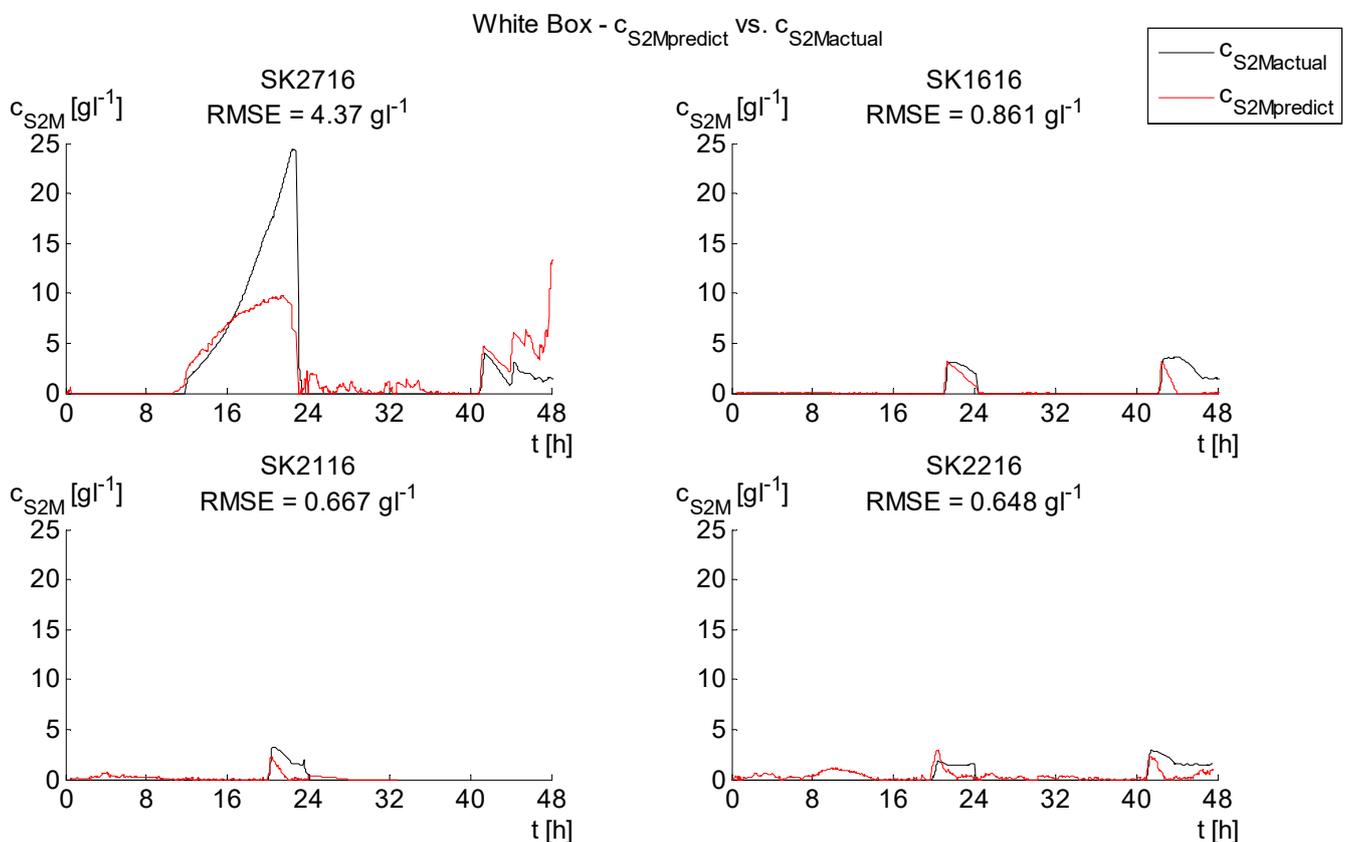


Abbildung 11: Vergleich der mit Gleichung (13) berechneten mit der tatsächlich gemessenen Methanolkonzentration der verschiedenen Kultivierungen im ED10.

In Abbildung 11 ist der Vergleich von gemessener und durch die „White Box“ berechneter Methanolkonzentration zu erkennen. Es ist deutlich an den großen Fehlern, die immer über $0,6 \text{ g l}^{-1}$ lagen, zu erkennen, dass der Fit unzulänglich war. Bei einer Soll-

Methanolkonzentration von $1,5 \text{ g l}^{-1}$ entspräche das einem durchschnittlichen Fehler von über 40 %, was nicht tragbar ist. Der Start der Methanolzugabe wurde zwar immer korrekt dargestellt und auch die tatsächlich erreichte Spitzenkonzentration war passend dargestellt, aber nach diesem initialen Punkt liefen die tatsächliche und berechnete Konzentration übermäßig auseinander. Zumeist fiel die berechnete Konzentration schnell wieder auf 0 g l^{-1} ab, wie zum Beispiel deutlich im zweiten Zyklus von SK1616 zu sehen, oder sie nahm unkontrolliert zu wie gegen Ende der Kultivierung SK2716. Der Grund für das Auseinanderlaufen rührte wahrscheinlich daher, dass die Substrataufnahmerate als konstant über die gesamte Produktionsphase angenommen wurde. Tatsächlich nahm diese aber im Verlauf zu, da die intrazelluläre Proteinzusammensetzung, bei Einsatz von Methanol als einziger Kohlenstoffquelle, sich zuerst umstellen muss. Ein weiterer Nachteil dieses Modellierungsweges war, dass außerplanmäßige Verläufe, wie im ersten Zyklus von SK2716, nicht korrekt beschrieben wurden.

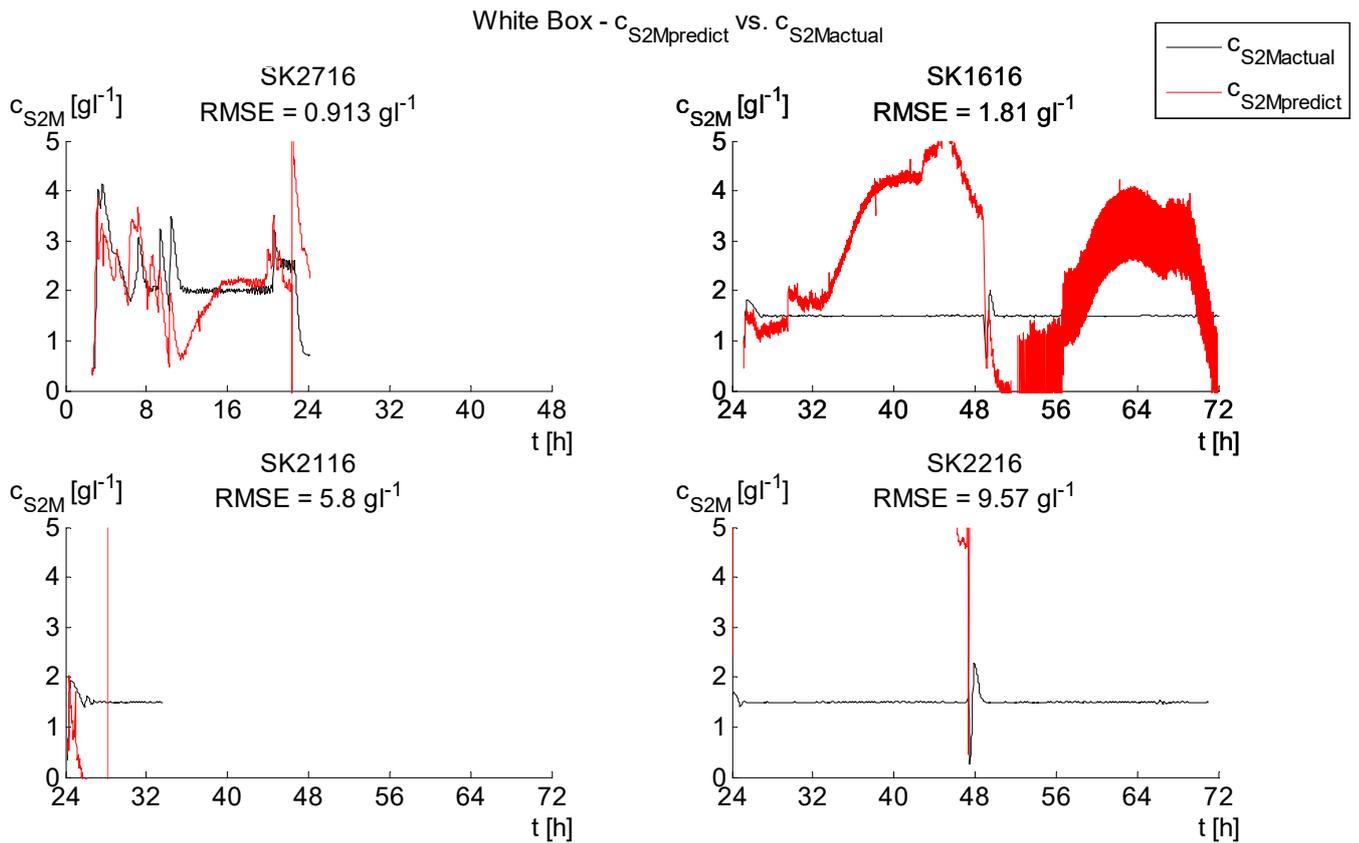


Abbildung 12: Vergleich der mit Gleichung (13) berechneten mit der tatsächlich gemessenen Methanolkonzentration der verschiedenen Kultivierungen im C30.

Noch auffälliger war dies für den C30, wie in Abbildung 12 zu erkennen. Lediglich die Kultivierung SK2716 wies teilweise eine Wiedergabe des Messsignals auf, aber selbst dort war der Fit sehr schlecht mit einem großen RMSE von fast $1\ g\ l^{-1}$. Bei den anderen Kultivierungen waren die Fehler noch größer, daher ist diese Art der Berechnung für c_{S2M} definitiv ungeeignet. Es stellt sich wieder die Frage warum dem so war. Zum einen konnte dies an der Variabilität des biologischen Systems selbst liegen – die Zellen sind ein lebendes System und wiesen daher eine gewisse Batch zu Batch Variabilität auf. Zudem besaß die Gleichung (13) zur Berechnung der Methanolkonzentration eine Rekursivität, wodurch sich kleine Fehler immer größer fortpflanzen und so zu großen Fehlern werden konnten. Des Weiteren war die bestimmte Substrataufnahmerate in beiden Reaktoren stark unterschiedlich, wodurch sich ebenfalls der Fehler erklären ließe. Zusammenfassend lässt sich sagen, dass das „White Box“-Verfahren nicht geeignet für die Berechnung der Methanolkonzentration c_{S2M} war, da die Ausgabe der Gleichungen die Messwerte nicht ausreichend genau beschrieben.

Um den $k_L a$ für die Berechnung des pO_2 zu bestimmen zu können, wurden die Parameter α und β für die Korrelation nach van't Riet ermittelt.

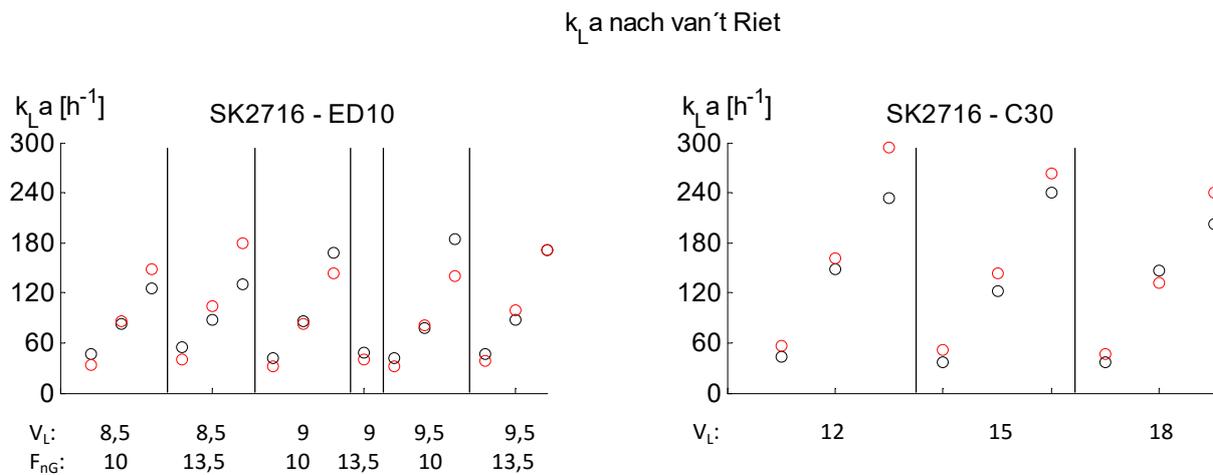


Abbildung 13: Vergleich der aus den Ladekurven und der mit Hilfe der $k_L a$ -Korrelation berechneten $k_L a$ -Werte. Es wurden die Werte für $N_{St} = 200, 400$ und 600 rpm aufgenommen.

Für die Ermittlung der Parameter für den ED10 wurden zwei Werte, bei $V_L = 9$ l, $F_{nG} = 13,5$ l min^{-1} , $N_{St} = 400$ rpm und 600 rpm, entfernt, da diese zu weit außerhalb der Erwartungen lagen. Aus den restlichen wurde dann die Formel für die k_{LaED10} ,

$$k_{LaED10}(t) = 170,8 \text{ h}^{-1} \cdot \left(\frac{N_{St}}{600 \text{ rpm}}\right)^{3 \cdot 0,451} \cdot \left(\frac{V_L}{9,5 \text{ l}}\right)^{-0,451} \cdot \left(\frac{F_{nG}}{13,5 \text{ l min}^{-1}}\right)^{0,652}$$

und für den C30 k_{LaC30} ,

$$k_{LaC30}(t) = 240,6 \text{ h}^{-1} \cdot \left(\frac{N_{St}}{600 \text{ rpm}}\right)^{3 \cdot 0,497} \cdot \left(\frac{V_L}{18 \text{ l}}\right)^{-0,497}$$

Bestimmt. Diese Gleichungen wurden verwendet um nach Formel (4) den pO_2 zu berechnen.

Der pO_2 wurde dann mithilfe von Gleichung in beiden Reaktoren (4) berechnet.

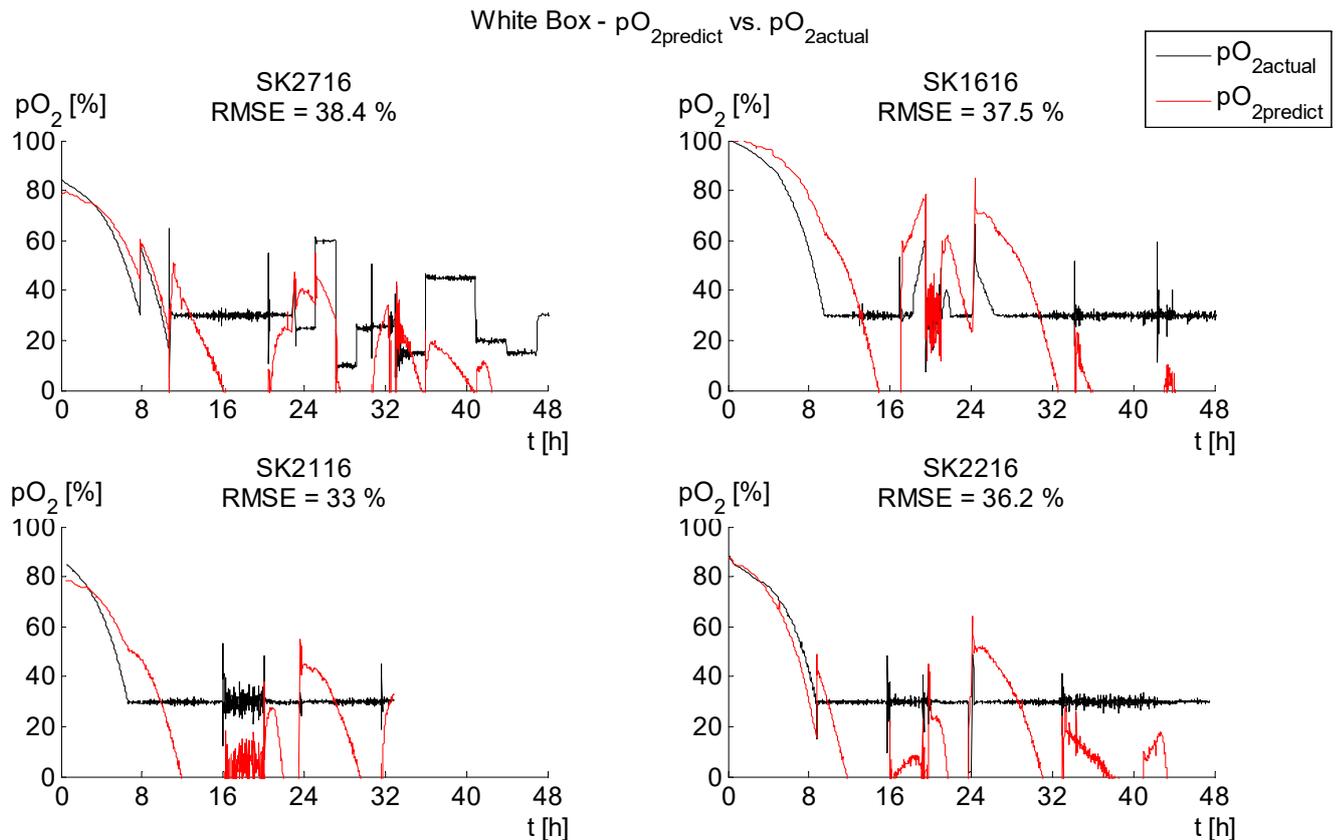


Abbildung 14: Vergleich des mit Gleichung (4) berechneten pO_2 mit dem tatsächlich gemessenen der verschiedenen Kultivierungen im ED10.

In Abbildung 14 ist deutlich zu erkennen, dass mit diesem Modell keine gute Beschreibung der Messwerte darstellt. Diese Berechnung war ungeeignet zur Bestimmung des pO_2 , die Fehler waren für alle Kultivierungen sehr groß mit immer über 30 %. Auch für den C30 war der Fit ähnlich schlecht, mit Fehlern die von 20 % bis 75 % reichten. Zu klären wäre, warum die bekannten Grundlagengleichungen so ungeeignet sind die Realität zu beschreiben. Einer der Gründe könnte sein, dass die bestimmten Parameter für die Korrelation nach van't Riet nicht passend waren. In Abbildung 13 ist erkennbar, dass die aus der Korrelation bestimmten $k_L a$ -Werte teilweise stark, insbesondere bei hohen Rührerdrehzahlen, von denen aus den Ladekurven bestimmten abwichen.

Tabelle 3: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der "White Box"-Modelle.

Zielvariable	Reaktor	Durchschnittlicher RMSE aller Kultivierungen
c_{S2M}	ED10	1,6 $g\ l^{-1}$
$c_{S2Mw} = 1,5\ g\ l^{-1}$	C30	4,5 $g\ l^{-1}$
pO_2	ED10	36,3 %
$pO_{2w} = 30\ %$	C30	35,8 %

In Tabelle 3 sind die durchschnittlichen Fehler aller „White Box“-Modelle aufgelistet. Es ist deutlich zu erkennen, dass die Ausgaben nicht genau genug waren um den Sauerstoffpartialdruck und die Methanolkonzentration ausreichend gut vorherzusagen. Einer der Gründe war wahrscheinlich, dass die „White Box“-Modelle sehr rigide waren und nicht gut auf die Batch- zu Batch-Variabilität reagieren konnten. Zudem spielte auch die Güte der Messsysteme eine sehr große Rolle. Zum Beispiel war das Waagsignal m_{R2} des C30 oft verrauscht durch Luftbewegungen oder Bodenschwingungen bei der normalen Arbeit um den Reaktor. Diese Messschwankungen hatten dann starken Einfluss auf die „White Box“-Ausgabe. Des Weiteren wiesen die Zellen als ein biologisches System eine Batch zu Batch Variabilität auf, die ebenfalls dafür sorgte, dass die Parameter, die an einer Kultivierung bestimmt wurden, nicht optimal die anderen beschrieb. Daher eignete sich die „White Box“ für das Ziel, die Überwachung der vorhandenen Hardwaresonden, nicht.

4.3.2 Extended Kalman-Filter, Hauptkomponentenregression und multivariate lineare Regression

In der Einteilung in „White“, „Grey“ und „Black Box“ fallen die Hauptkomponentenregression und multivariate lineare Regression unter die „Black Box“, der Extended Kalman-Filter dagegen unter „Grey Box“. Als Kalman-Filter wurde das Skript von Yi Cao der Cranfield University verwendet. Das Skript wurde im Anhang 1 beigelegt.

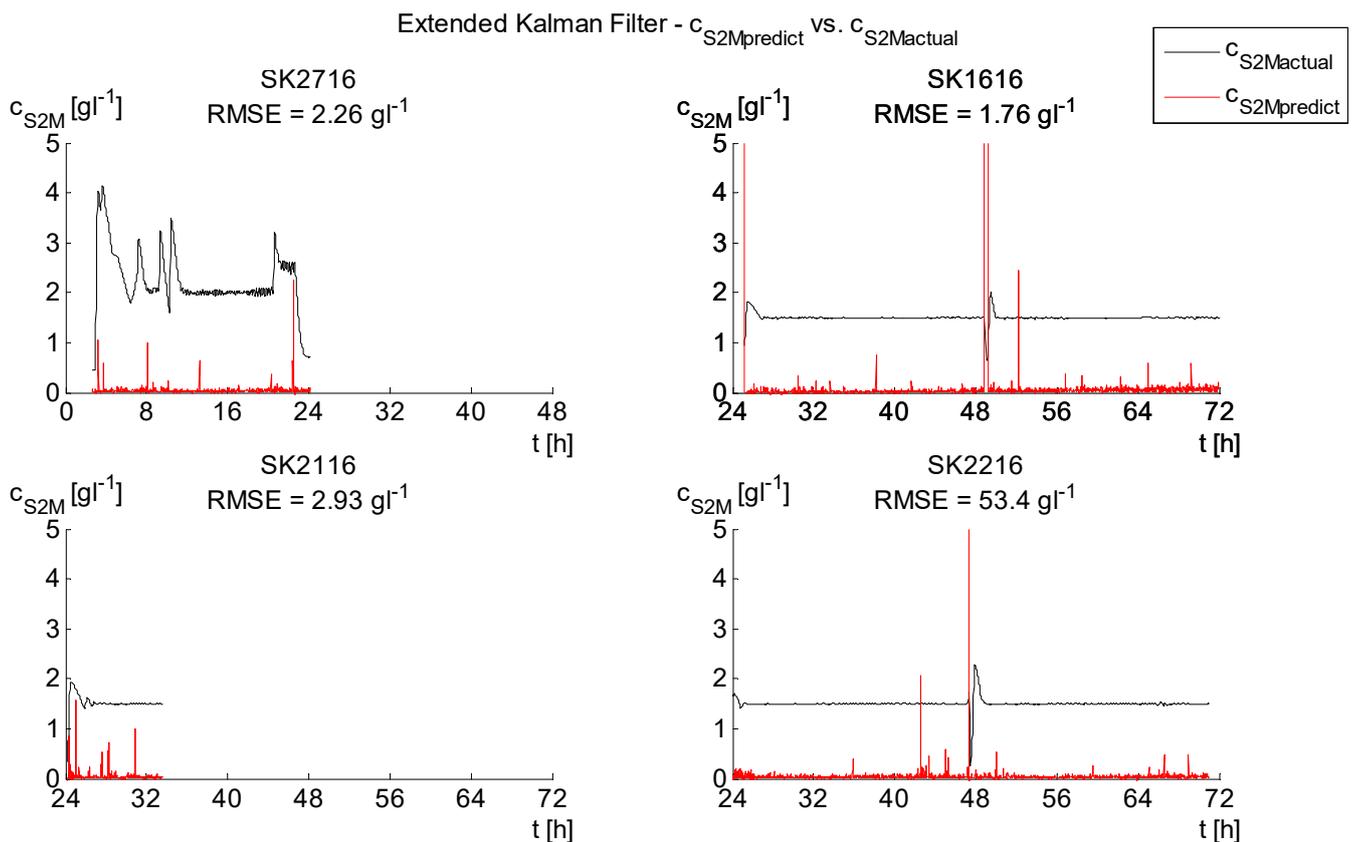


Abbildung 15: Verlauf der mit Hilfe des Kalman-Filters bestimmten Methanolkonzentration mit der gemessenen aller Kultivierungen im C30.

Wie in Abbildung 15 zu sehen ist, war der Kalman-Filter für diese Anwendung nicht geeignet. Es fand absolut keine Prädiktion von Werten statt, und große Fehler wurden erreicht. Bei SK2216 war der Fehler sogar bis zu 53,4 gl⁻¹ groß, da Spitzen bis zu -200.000 gl⁻¹ ausgegeben wurden. Auch für die anderen Zielvariablen, c_{S2M} im ED10 und pO_2 für ED10 und C30, war der Kalman-Filter nicht geeignet. Dies ließ sich durch mehrere Dinge erklären. Zum einen gingen die Grundlagengleichungen, wie sie schon in der „White Box“ verwendet wurden, in den

Extended Kalman-Filter ein, aber, wie im vorherigen Kapitel beschrieben, waren diese schon ungenügend um den Prozess zu darzustellen. Zudem wurde mit Kalman-Filter auch versucht, das unbekannte Prozessrauschen auszudrücken, was in diesem Fall aber zu einer Erhöhung des Fehlers führte, anstatt zu einer besseren Prozesscharakterisierung. Des Weiteren erklärte sich der Fehler bei Betrachtung der benutzte Mathematik hinter dem Kalman-Filter. Bei dem Kalman-Filter wurde zuerst der nächste Schritt voraus und die Fehlerkovarianz bestimmt. Diese wurden dann in einem Folgeschritt versucht zu korrigieren, indem der Kalman Gain mit Hilfe der partiellen Ableitung der eingehenden Gleichungssysteme bestimmt wurde. Diese waren aber, wie bei der „White Box“ gesehen, an sich schon ungeeignet. Mit dem Kalman Gain wurde versucht, die Schätzung und die Fehlerkovarianz zu korrigieren, aber stattdessen wurden nur die vorhandenen Fehler vergrößert.

c_{S2M} im ED10 und pO_2 im ED10 und C30 wurden hier nicht dargestellt, da die Ergebnisse ähnlich wie in Abbildung 15 waren. Zusammenfassend war der Kalman-Filter nicht für die Modellbildung geeignet.

Es wurde eine Hauptkomponentenregression durchgeführt. Die Hauptkomponenten wurden anhand der Kultivierung SK2716 bestimmt und dann die Vorhersagefähigkeit an SK1616, SK2116 und SK2216 getestet.

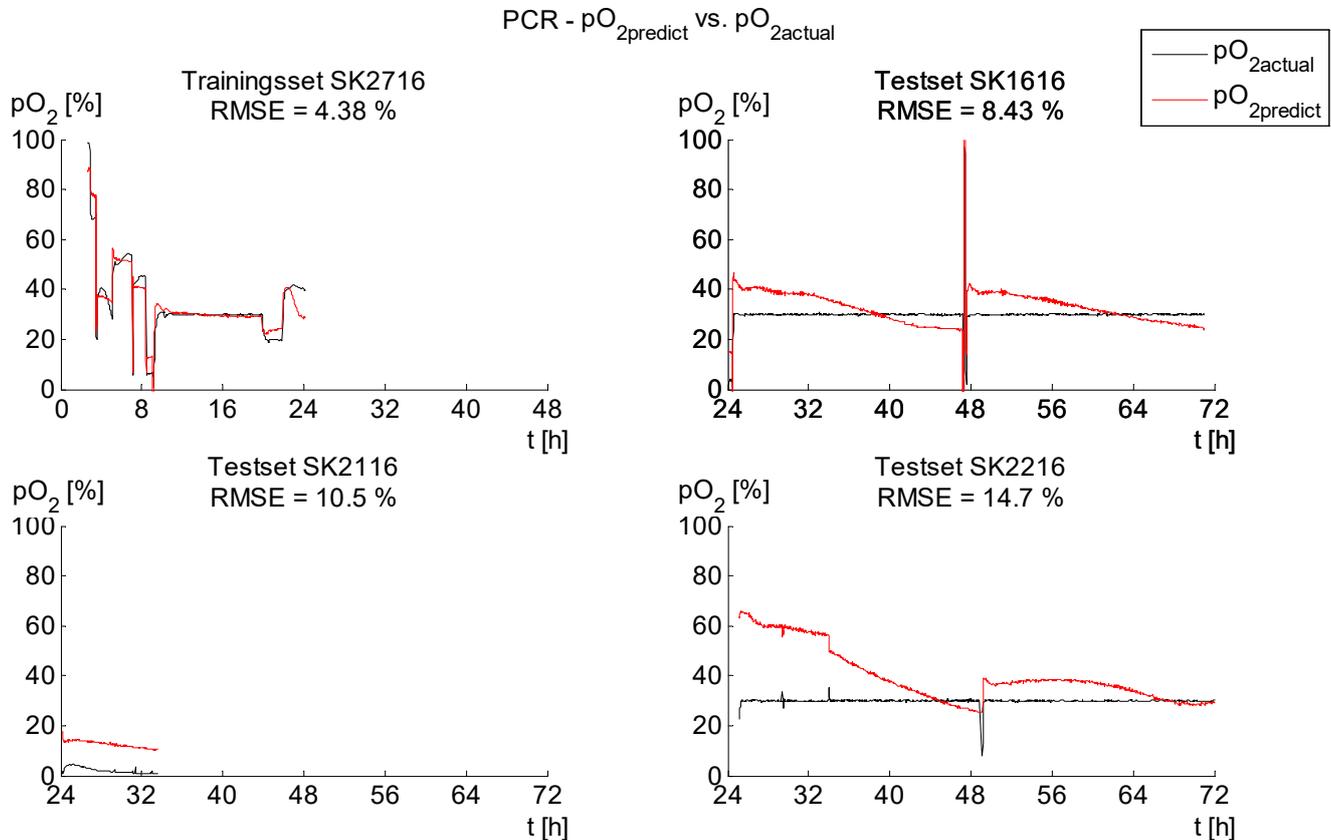


Abbildung 16: Gemessener und durch Hauptkomponentenregression bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im C30.

In Abbildung 16 ist zu erkennen, dass die Hauptkomponentenregression zwar den pO_2 einigermaßen beschreiben konnte, aber dennoch große Fehler aufwies. Dies wird besonders deutlich bei Betrachtung der Testsets. Dafür gab es verschiedene Gründe. Zum einen ist die PCA ein Verfahren, das unterliegende Linearitäten aus den Daten herausarbeitet – was bei nicht-linearen Datensätzen nachteilig ist. Zudem beschreibt die PCA die Richtung mit der höchsten Datenvarianz, was ebenfalls dazu führen konnte, dass die Daten nicht optimal beschrieben werden konnten. Da ähnlich große Fehler als PCR-Ergebnis auch für c_{S2M} in C30 und ED10 und für pO_2 im ED10 beobachtet wurden, wurden diese hier nicht dargestellt. Aufgrund der relativ großen Fehler eignete sich die Hauptkomponentenregression nicht zur Beschreibung der Zielvariablen und daher nicht zur Überwachung der Hardwaresonden.

Es wurden die Regressionsparameter für eine multivariate lineare Regression aus den Kultivierungen SK2716 und SK1616 bestimmt und das entstehende Regressionsmodell an SK2116 und SK2216 getestet.

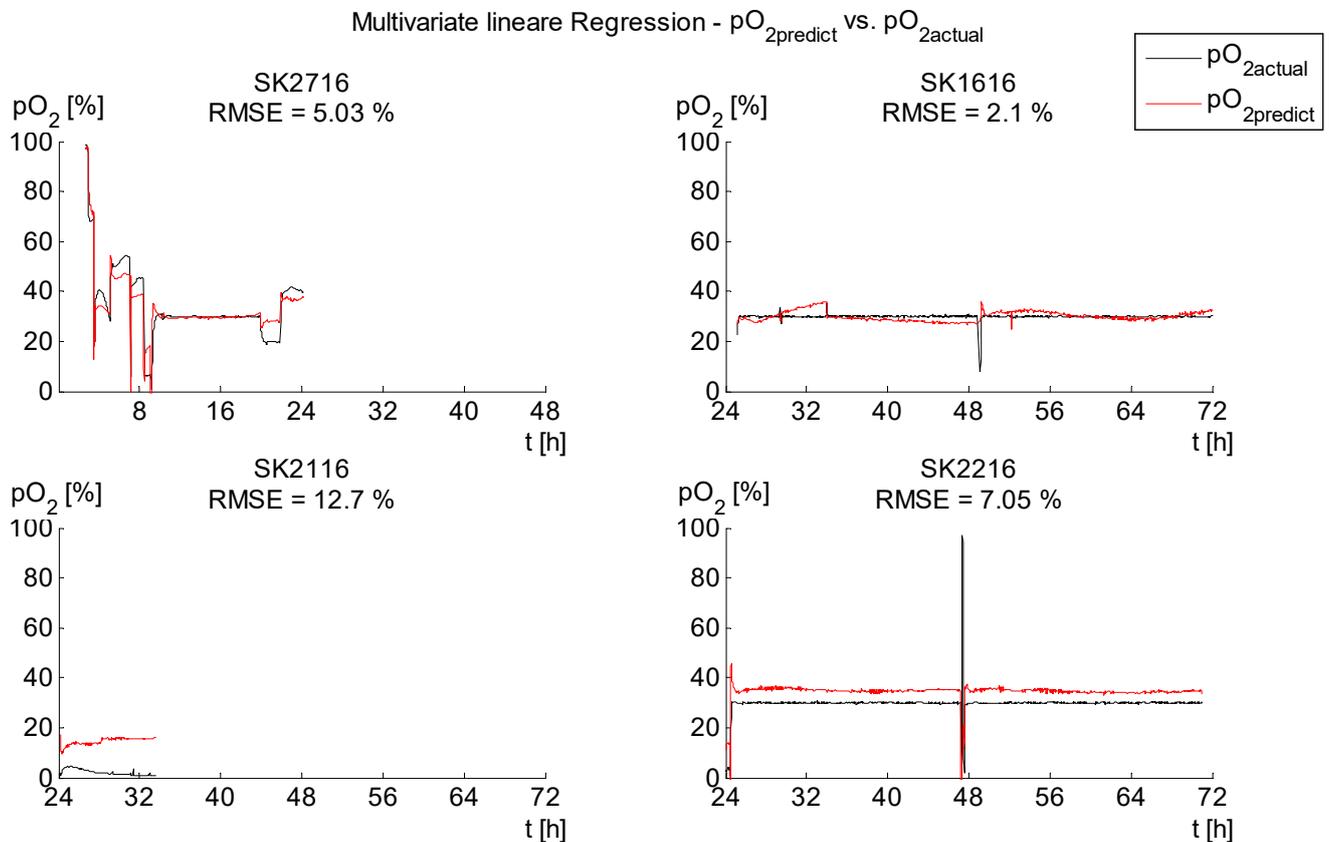


Abbildung 17: Gemessener und durch multivariate lineare Regression bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im C30.

In Abbildung 17 ist das Ergebnis aus der multivariaten linearen Regression für den pO_2 im C30 dargestellt. Es ist zu sehen, dass der Messwert einigermaßen mit Hilfe dieses Modells beschrieben werden konnte, aber nicht genau genug, um es in Kultivierungen einzusetzen. Dies lag vor allem daran, dass die Zusammenhänge zwischen den einfließenden Messwerten und den Zielvariablen nicht linear waren, die multivariate lineare Regression, wie der Name es schon aussagt, aber ein lineares Verfahren ist. Dies wurde besonders bei Betrachtung der drei anderen Zielvariablen klar. Für den c_{S2M} im C30 ergab sich eine große Fehlerbreite von $0,12 \text{ g l}^{-1}$ bis $0,69 \text{ g l}^{-1}$, für den pO_2 im ED10 ein RMSE von immer über 9 % und für c_{S2M} im ED10 sogar bis zu $3,8 \text{ g l}^{-1}$. Daher war dieses Regressionsmodell nicht für die Überwachung der Sonde geeignet.

Tabelle 4: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe von Kalman-Filter, Hauptkomponentenregression (PCR) und multivariater linearer Regression.

Zielvariable	Reaktor	durchschnittlicher RMSE aller Kultivierungen		
		Kalman-Filter	PCR	Multivariate lineare Regression
$c_{S_{2M}}$	ED10	2,2 $g\ l^{-1}$	5,4 $g\ l^{-1}$	2,4 $g\ l^{-1}$
$c_{S_{2Mw}} = 1,5\ g\ l^{-1}$	C30	6,5 $g\ l^{-1}$	4,2 $g\ l^{-1}$	0,4 $g\ l^{-1}$
pO_2	ED10	551 %	17,0 %	10,6 %
$pO_{2w} = 30\ %$	C30	1982 %	9,5 %	6,7 %

In Tabelle 4 sind die verschiedenen durchschnittlichen Fehler aller Modelle und Zielvariablen aufgelistet. Es ist deutlich, dass diese drei Modellierungswege nicht geeignet waren, um die beiden Zielvariablen ausreichend zu beschreiben und somit das eigentliche Ziel – die Überwachung der vorhandenen Hardwaresensoren – zu erreichen. Die Fehler waren zum Teil außerordentlich groß, und selbst in den Fällen, in denen die RMSE etwas geringer waren, beschrieben die Modelle die Prozessverläufe nicht akkurat. Daher waren der Extended Kalman-Filter, die Hauptkomponentenregression und multivariate lineare Regression ungeeignet.

4.3.3 Neuronale Netzwerke – „Black Box“

Es wurde versucht ein neuronales Netzwerk als „Black Box“ zur Beschreibung der Zielvariablen zu entwickeln und zu trainieren.

Dafür wurden zuerst die beiden Lernalgorithmen Levenberg-Marquardt und Bayesian Regulation überprüft und einer von beiden ausgewählt. Als Lernbedingungen wurde eine Anzahl an versteckten Neuronen von 10 und 100 Epochen gewählt. Der pO_2 wurde im ED10 als Testvariable zum Training benutzt.

Tabelle 5: RMSE des Testes zum Festlegen des einzusetzenden Lernalgorithmus zum Trainieren aller weiteren Netzwerke

	Levenberg-Marquardt	Bayesian Regulation
RMSE des Trainingssets	13,8 %	3,3 %
RMSE des Testsets	13 %	9,5 %
Trainingsdauer	0,05 Min	0,32 min

Anhand der Tabelle 5 ist zu erkennen, dass die Abweichung des Levenberg-Marquardt-Algorithmus sowohl im Trainingsset als auch im Testset größer war. Auf der anderen Seite benötigte dieser im Vergleich zum Bayesian Regulation Algorithmus nur ein Siebtel an Trainingszeit. Dies liegt vor allem daran, dass der Levenberg-Marquardt-Algorithmus sogenannte „Validation Checks“ durchführt, bei denen die Generalisierung des Netzwerkes überprüft wird, und das Training bei erfolgreichen Checks abbricht. Eine Änderung der Nummer an „Validation Checks“ verbesserte die Netzwerk-Performance dennoch nicht. Und auch das Ausschließen von „Validation Checks“ führte nicht zu einer besseren Performance als beim Bayesian Regulation Algorithmus. Aufgrund der kleineren Ausgabefehler und dadurch, dass die Trainingsdauer zwar höher, aber dennoch akzeptabel ist, wurde der Bayesian Regulation Algorithmus für alle weiteren Versuche gewählt.

Danach wurden die zwei Hyperparameter des Netzwerkes, Neuronenanzahl n_H und Epochenanzahl $epoch$ des Netzwerkes versucht zu optimieren.

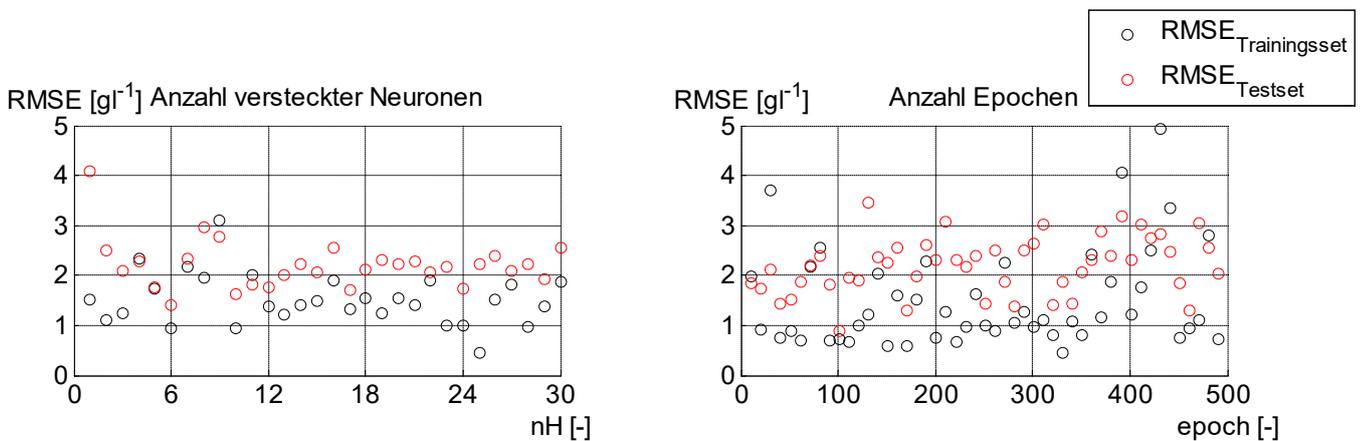


Abbildung 18: Die RMSE der durch das Feedforward Netzwerk bestimmten Methanolkonzentration im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen n_H und Epochenanzahlen $epoch$.

Aus Abbildung 18 ist erkennbar, dass eine zu kleine Neuronenanzahl sich negativ auf die Beschreibungsfähigkeit des Netzwerkes auswirkt. Es ist ein genereller absteigender Trend in den RMSE von Testset und Trainingsset bis circa $n_H = 12$ zu erkennen, danach streut der Fehler nur noch in einem Bereich von 1 bis 3 $g\ l^{-1}$. Eine kleinere Neuronenanzahl sollte sich theoretisch positiv auf die Generalisierungsfähigkeit des Netzwerkes auswirken, was sich aber anhand des Testsets nicht wiedergab. Dort schien der RMSE nur um 2 $g\ l^{-1}$ zu streuen. Da höhere Neuronenanzahlen die Generalisierungsfähigkeit nicht verbesserten und die Trainingsdauer signifikant erhöhten, wurde für weitere Tests $n_H = 6$ als Neuronenanzahl ausgewählt. Im zweiten Teil der Abbildung 18 ist der Einfluss der Epochenanzahl $epoch$ zu erkennen. Es ist zu sehen, dass die Streuung der RMSE der Trainingssets mit höherer Epochenanzahl leicht zunimmt. Generell können die Algorithmen, unabhängig von der Epochenanzahl, in lokale Minima hineinlaufen. Bei hohen Epochenanzahlen werden dann diese lokalen Minima sehr gut beschrieben – wodurch dann aber die Beschreibungsfähigkeit des Netzwerkes leidet. Zudem wird auch hier ebenfalls die Trainingsdauer signifikant durch mehr Epochen erhöht. Es ist daher eher sinnvoll geringere Epochenanzahlen zu wählen. Stattdessen werden mehrere Trainingsdurchläufe durchgeführt um das absolute, und nicht nur lokale, Minimum der Fehlerfunktion zu finden, und somit das bestbeschreibende Netzwerk zu finden. Für weitere Tests wurde $epoch = 50$ gewählt.

Es wurden verschiedene Inputkonfigurationen überprüft bei den vorher ermittelten Netzwerk-Hyperparametern.

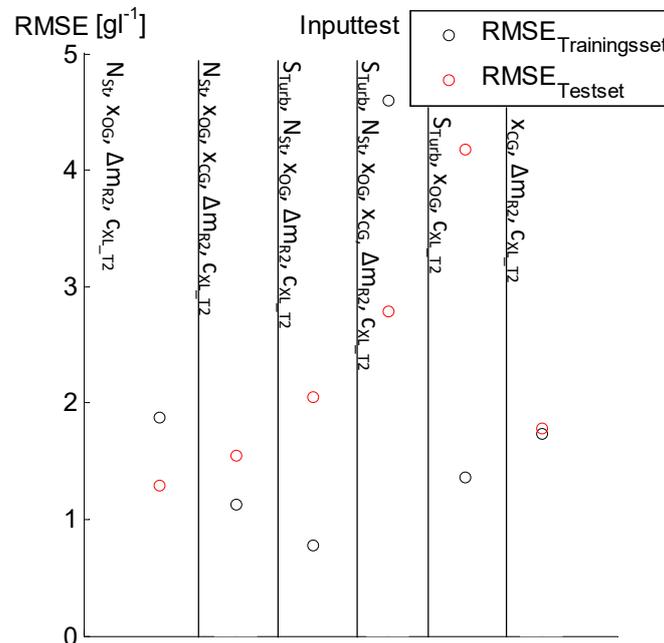


Abbildung 19: Die RMSE der durch das Feedforward Netzwerk bestimmten Methanolkonzentration im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) bei verschiedener Inputkonfigurationen.

Verschiedene Inputkonfigurationen wurden auf den RMSE hin überprüft, wie in Abbildung 19 zu sehen ist. Den geringsten Fehler für Test- und Trainingsset zeigte das Inputarray N_{st} , X_{OG} , X_{CG} , Δm_{R2} und C_{XL_T2} , welches dann auch verwendet wurde um das endgültige Netzwerk für c_{S2M} im ED10 zu trainieren. Diese drei Schritte, Ermitteln der Hyperparameter nH und $epoch$, und des Inputarrays wurden auch für den pO_2 als Zielvariable für beide Reaktoren durchgeführt.

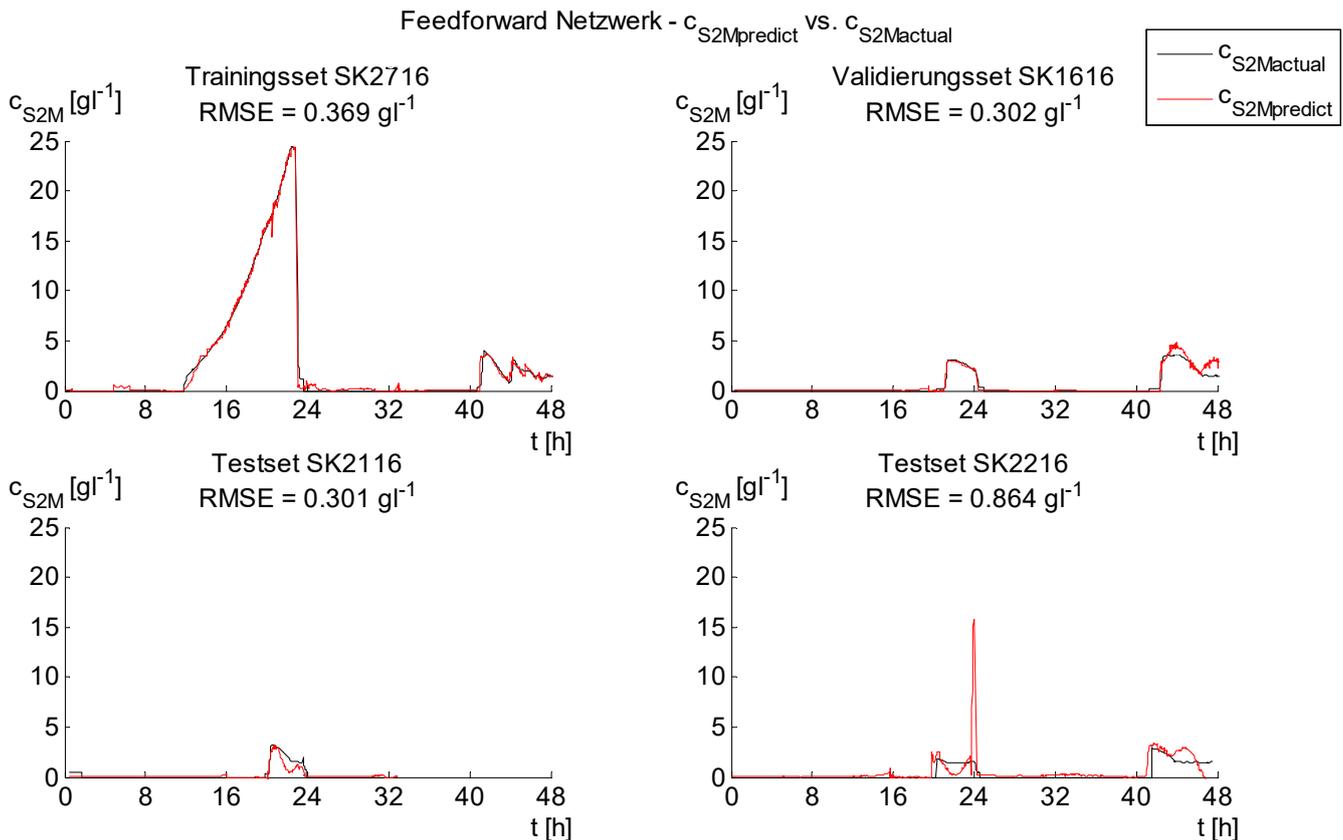


Abbildung 20: Gemessene und durch Feedforward-"Black Box" bestimmte Methanolkonzentration für alle vier Kultivierungen im ED10 und die berechneten RMSE.

In Abbildung 20 ist das Ergebnis des endgültigen, trainierten Feedforward neuronalen Netzwerkes, das als „Black Box“ mit $n_H = 6$, epoch = 50 und N_{St} , x_{OG} , x_{CG} , Δm_{R2} und c_{XL_T2} als Inputs, die Methanolkonzentration c_{S2M} im ED10 berechnete. Es sind deutlich geringere Fehler als bei den vorherigen Modellen, bei denen nie RMSE unter $1\ g\ l^{-1}$ erreicht wurde, zu erkennen. Das Trainings- und Validierungsset konnten einigermaßen gut beschrieben werden – mit einem durchschnittlichen RMSE von circa $0,3\ g\ l^{-1}$. Andererseits war die Beschreibung von c_{S2M} noch nicht optimal, insbesondere bei Betrachtung des Testsets SK2216. Dort lag der Fehler bei über $0,86\ g\ l^{-1}$. Dementsprechend war auch dieses Modell für die Erfüllung des Ziels – die Überwachung der Methanolsonde – wahrscheinlich nicht geeignet.

Für die Berechnung der Methanolkonzentration im C30 war die „Black Box“ nicht geeignet, dort wurden unabhängig von Inputkonfiguration, Neuronenanzahl, Epochenanzahl oder Algorithmus niemals Fehler unter $10\ g\ l^{-1}$ erreicht. Dafür wurde stattdessen ein regressives neuronales Netzwerk genauer betrachtet.

Für den pO_2 in beiden Reaktoren wurden die Vorbereitungsschritte zu der Ermittlung der Epochenanzahl $epoch$ und Anzahl an versteckten Neuronen nH durchgeführt. Auch wurden wieder verschiedene Inputkonfigurationen überprüft und die mit dem geringsten RMSE ausgewählt.

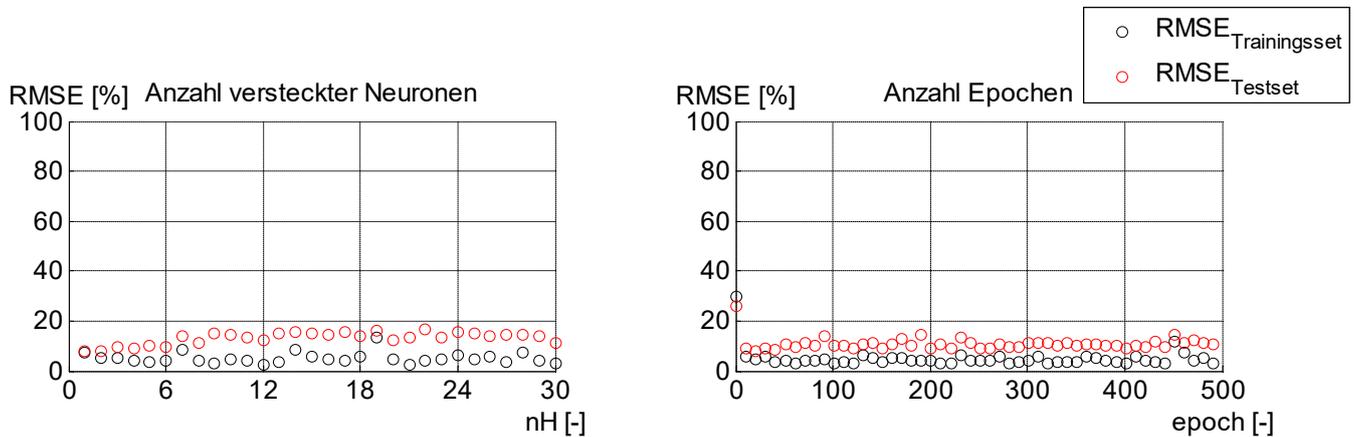


Abbildung 21: Die RMSE des durch das Feedforward Netzwerk bestimmten Sauerstoffpartialdrucks im ED10 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen nH und Epochenanzahlen $epoch$.

Für den pO_2 im ED10 wurden für die Variation von nH und $epoch$ weniger starke Streuungen um beobachtet. Es ist in Abbildung 21 zu erkennen, dass bei steigendem nH der $RMSE_{Testset}$ im geringen Maße zu- und der $RMSE_{Trainingsset}$ abnahm, um dann um einen konstanten Wert zu streuen. Die Epochenanzahl $epoch$, bis auf bei sehr kleinen Werten, zeigte kaum Einfluss auf den Fehler des Netzes. Für weitere Tests wurden dementsprechend eine kleine Neuronenanzahl von 5 und eine Epochenanzahl von 50 gewählt, da höhere Werte keine wesentliche Verbesserung zeigten und um die Lerndauer gering zu halten. Als Inputarray wurde N_{St} , x_{OG} , x_{CG} , Δm_{R1} , Δm_{R2} und c_{XL_T2} als optimal ermittelt und für das endgültige Erstellen der pO_2 -„Black Box“ für den ED10 verwendet.

Das endgültige Netzwerk für die Bestimmung des pO_2 im ED10 mit $nH = 5$, epoch = 50 und den Inputs = N_{St} , x_{OG} , x_{CG} , Δm_{R1} , Δm_{R2} und c_{XL_T2} wurde trainiert und die Fehler berechnet.

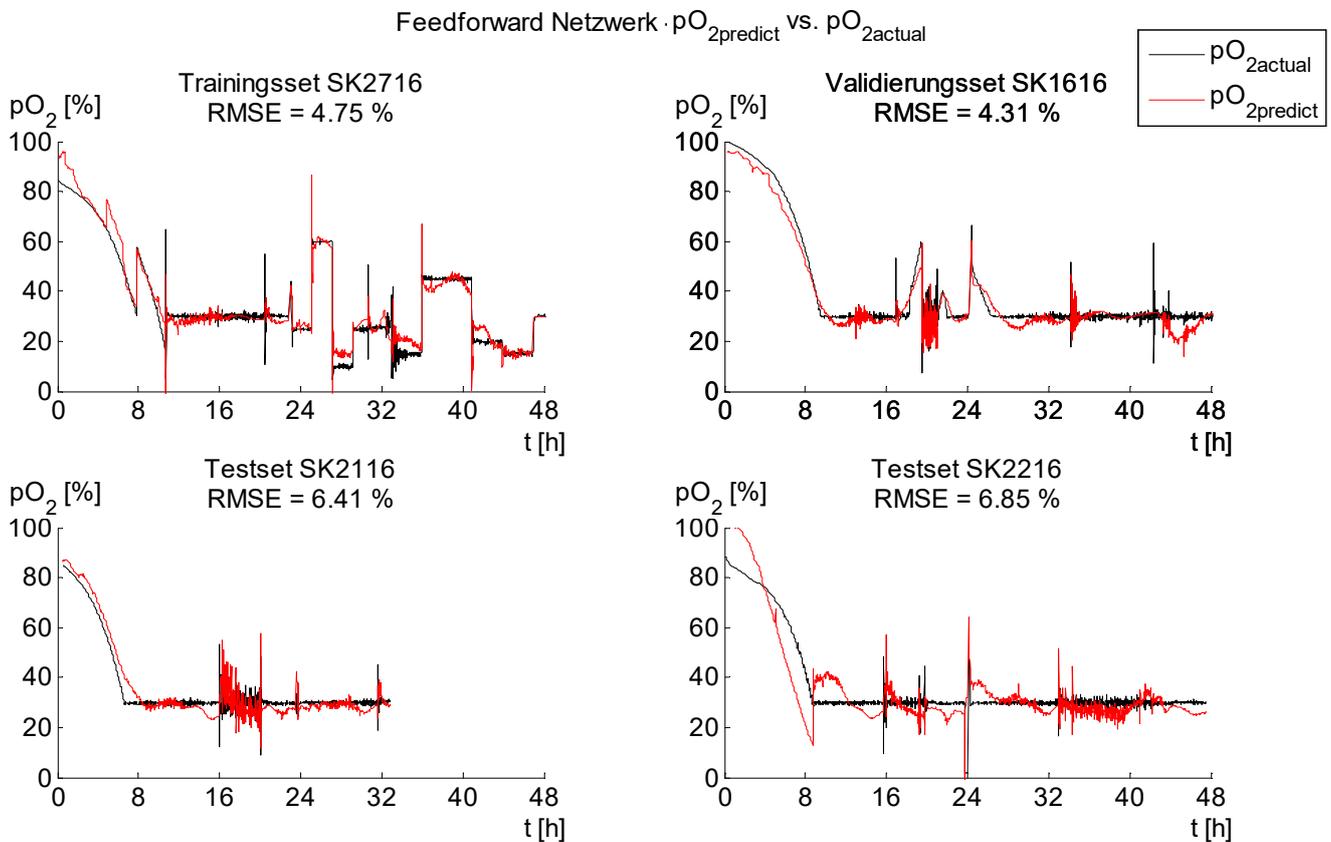


Abbildung 22: Gemessener und durch Feedforward-"Black Box" bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im ED10 und die berechneten RMSE.

In Abbildung 22 ist der gemessene und mit Hilfe der „Black Box“ ermittelte pO_2 dargestellt. Die deutliche Verbesserung gegenüber der „White Box“ und des Kalman-Filters beziehungsweise der PCA ist deutlich zu erkennen. Der maximale RMSE liegt bei 6,9 %, was sehr viel geringer ist als zum Beispiel der „White Box“, bei dem der RMSE immer über 30 % lag. Dennoch zeigt die Abbildung noch deutliches Verbesserungspotential. Insbesondere SK2216 zeigte teilweise starke Abweichungen zum tatsächlichen Verlauf des pO_2 . Dies könnte aber auch wiederum an der Batch- zu Batch-Variabilität liegen. Zwischen SK2116 und SK2216 wurde ein großer Unterschied beobachtet. Zwar wurden beide mit der gleichen Startzellkonzentration angeimpft, aber dennoch fiel der pO_2 bei SK2216 sehr viel langsamer auf die geregelten pO_{2w} von 30 % ab. Dieser Set-Point wurde bei SK2216 erst bei der doppelten Zellkonzentration erreicht als bei SK2116. Und auch im Vergleich mit SK1616 und SK2716 fiel

dies auf – bei diesen beiden Kultivierungen wurde pO_{2w} bei fast der gleichen Zellkonzentration erreicht wie bei SK2116. Aber trotz dieser Batch- zu Batch-Variabilität wurde der Verlauf einigermaßen gut beschrieben, anders als bei den vorhergehenden Berechnungen, und dieses Modell wäre geeignet, um zumindest große Ausfälle der pO_2 -Sonde zu detektieren.

Da das Feedforward-Netzwerk kein zufriedenstellendes Ergebnis für die Methanolkonzentration im C30 ergab, wurde dafür zudem noch die Möglichkeit betrachtet, ein regressives closed loop Netzwerk zu benutzen. Bei diesem ging der Ausgang y_i wieder als Input mit ein. Da die ethanolsonde, mit dessen Daten gearbeitet wurde, eine relativ große Trägheit aufwies, wurde daher vermutet, dass das regressive Netzwerk die Methanolkonzentration besser beschreiben könnte.

Bei den regressiven Netzwerken kommt ein weiterer Hyperparameter hinzu, die Verzögerungsschrittzahl tp .

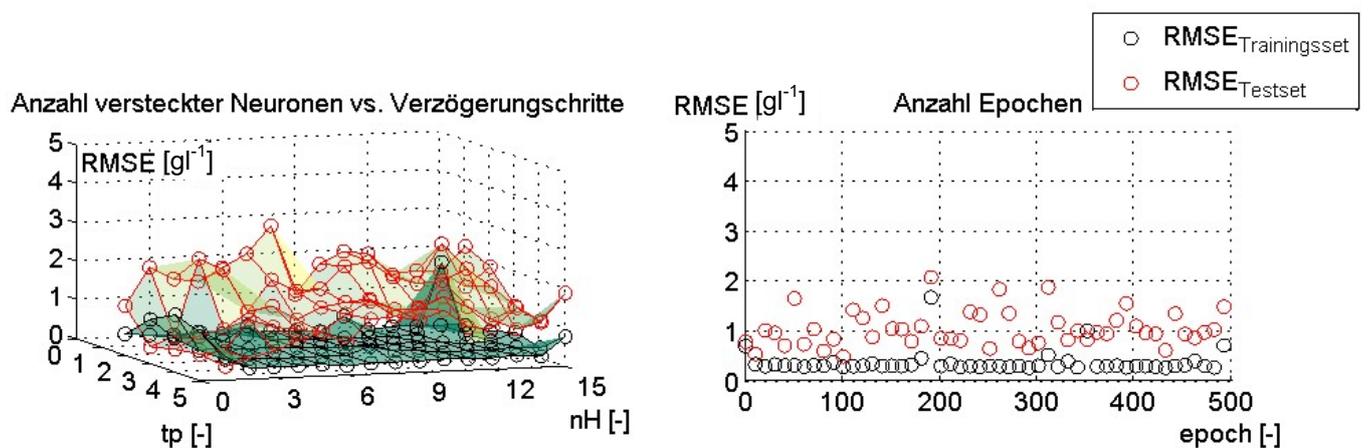


Abbildung 23: Die RMSE der durch das regressive closed loop Netzwerk bestimmten Methanolkonzentration im C30 von Trainingsset (SK2716 und SK1616) und Testset (SK2116 und SK2216) für verschiedene Anzahl versteckter Neuronen nH , Verzögerungsschritte tp und Epochenanzahlen epoch.

Im ersten Teil von Abbildung 23 ist zu erkennen, dass eine höhere Anzahl an Verzögerungsschritten sich insbesondere auf den RMSE der Testsets positiv auswirkte. Bei weniger Verzögerungsschritten sind die ermittelten Fehler höher. Ebenfalls ist zu beobachten, dass eine geringere Anzahl versteckter Neuronen sich ebenfalls auch positiv auf den RMSE der Testsets auswirkte. Aber dabei ist zu beachten, dass bei zu wenig versteckten Neuronen nH wiederum der RMSE des Trainingssets anstieg. Daher wurde als Kompromiss dafür eine Neuronenanzahl nH von 3 gewählt, da darüber der $RMSE_{\text{Testset}}$ signifikant anstieg. Als Verzögerungsschrittzahl wurde $tp = 5$ gewählt. Werte darüber waren nicht möglich, da die Trainingsdauer mit zunehmender Verzögerungsschrittzahl sehr stark zunahm und die Menge nach oben hin durch die Speicherzuordnung von MATLAB[®] begrenzt war. Die Epochenanzahl

epoch zeigte kaum einen Effekt auf die Vorhersagegüte, daher wurde epoch = 131 gewählt. Mit diesen Hyperparametern wurde x_{CG} , m_L , Δm_{R2} und c_{XL_T2} als passendes Inputarray ermittelt.

Mit den ermittelten Hyperparametern und Inputarray wurde ein regressives neuronales Netzwerk trainiert und getestet.

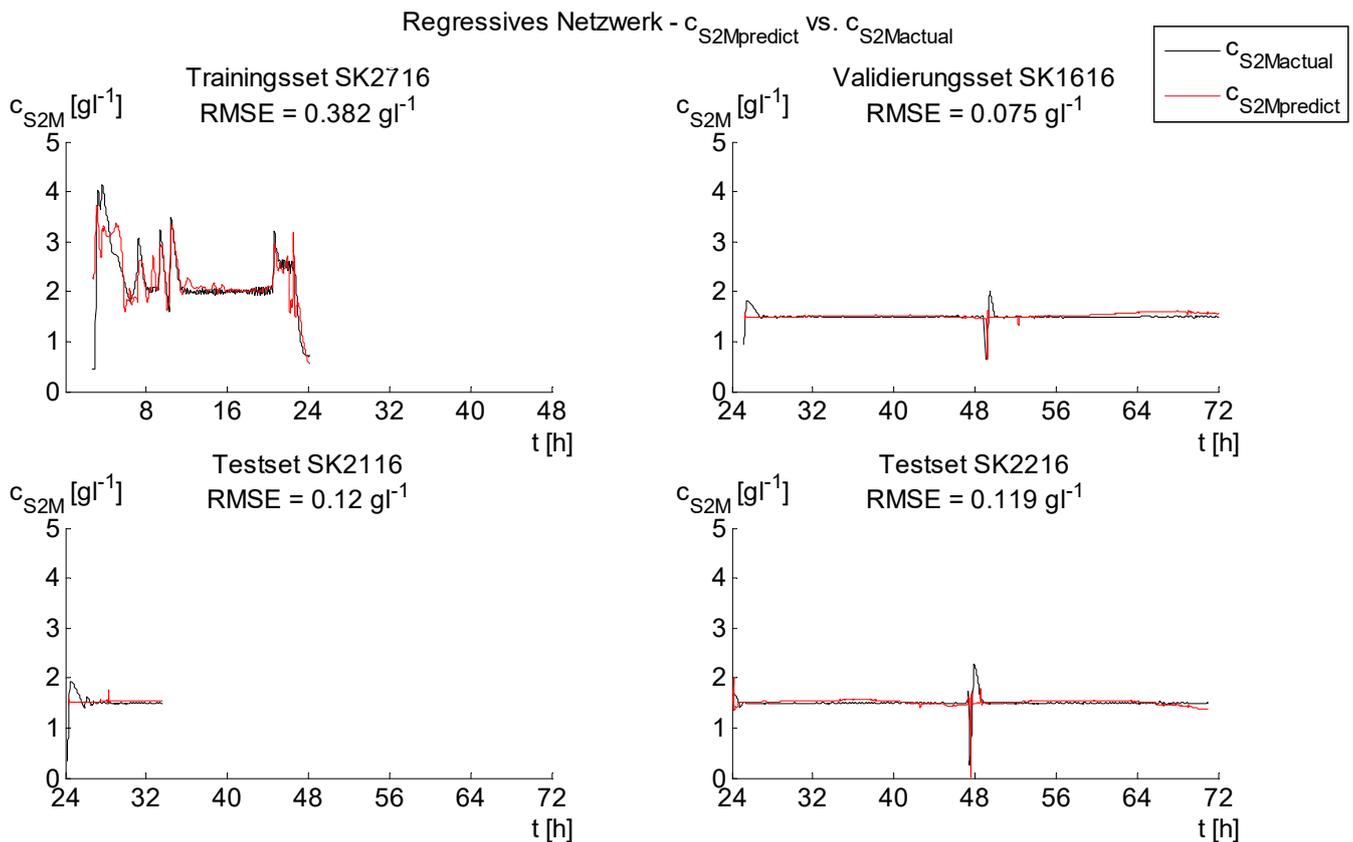


Abbildung 24: Gemessene und durch regressive "Black Box" bestimmte Methanolkonzentration c_{S2M} für alle vier Kultivierungen im C30 und die berechneten RMSE.

In Abbildung 24 ist das Ergebnis aus dem endgültigen trainierten regressiven Netzwerk für die Methanolkonzentration im C30 zu erkennen. Zwar ist der RMSE der geringste aller bisher betrachteten Modelle für c_{S2M} , aber dennoch ist die Beschreibung nicht optimal. Dies war besonders am Anfang jedes Zyklus bei Validierungsset und Testsets zu erkennen, bei denen die Anfangs-peaks nicht beschrieben wurden. Dennoch zeigte diese Art der Beschreibung der Methanolkonzentration sich am aussichtsreichsten.

Für die Methanolkonzentration im ED10 und die Sauerstoffpartialdrücke in beiden Reaktoren zeigte sich das regressive Netzwerk als nicht geeignet. Die $RMSE_{\text{Testset}}$ für die Bestimmung des pO_2 lagen unabhängig von t_p , nH und Inputkonfiguration immer bei über 20 % bis fast 40 %, was im Vergleich mit dem normalen Feedforward-netzwerk unzulänglich war. Für c_{S2M} des ED10 ergaben sich sogar noch schlechtere Ergebnisse, $RMSE_{\text{Trainingsset}}$ und $RMSE_{\text{Testset}}$ waren dort immer über 1 gl^{-1} . Daher wurden für die anderen drei Zielgrößen, c_{S2M} im ED10 und pO_2 in ED10 und C30 die regressiven neuronalen Netzwerke nicht weiter betrachtet.

Tabelle 6: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der „Black Box“ Neuronalen Netzwerke.

Zielvariable	Reaktor	Durchschnittlicher RMSE aller Kultivierungen	
		Feedforward-Netzwerk	Regressives Netzwerk
c_{S2M}	ED10	$0,44 \text{ gl}^{-1}$	$3,00 \text{ gl}^{-1}$
$c_{S2Mw} = 1,5 \text{ gl}^{-1}$	C30	$0,32 \text{ gl}^{-1}$	$0,17 \text{ gl}^{-1}$
pO_2	ED10	5,6 %	36 %
$pO_{2w} = 30 \%$	C30	2,6 %	75 %

In Tabelle 6 sind die RMSE der verschiedenen Netzwerke aufgelistet. Es ist zu erkennen, dass das regressive Netzwerk für die Berechnung von pO_2 und c_{S2M} im ED10 ungeeignet war, aber einen kleinen RMSE für c_{S2M} im C30 aufwies. Das normale Feedforward Netzwerk zeigte auch relativ geringe Fehler. Zwar waren die Ausgaben noch nicht optimal, aber diese Art der Modellbildung zeigte sich vielversprechend.

4.3.4 Neuronale Netzwerke – „Grey Box“

Für die Betrachtungen der „Grey Box“ wurden dieselben Hyperparameter wie für die „Black Boxes“ verwendet, da die Annahme besteht, dass diese die jeweilige Zielgröße bereits am besten beschreiben. Für die „Grey Boxes“ wurden pro Zielgröße mehrere Wege untersucht - für die Methanolkonzentration c_{S2M} wurden die beiden „Black Box“-Ausgaben $q_{S2X}(t)$ und $y_{gC_2}(t) = V_L(t) \cdot V_M(t)^{-1} \cdot q_{S2X}(t) \cdot C_{XL}(t)$ betrachtet.

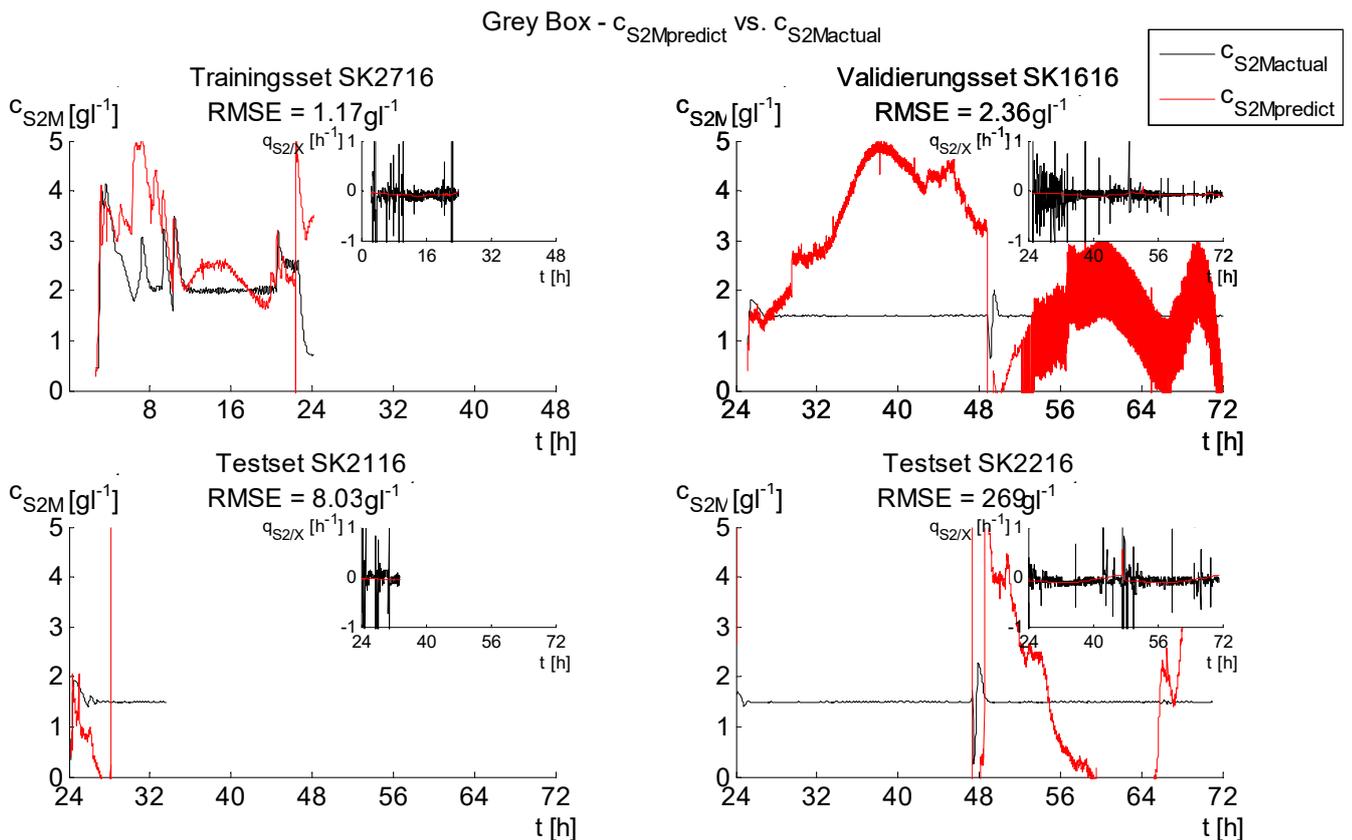


Abbildung 25: Gemessene und durch „Grey Box“ bestimmte Methanolkonzentration c_{S2M} für alle vier Kultivierungen im C30. Die Substrataufnahmerate $q_{S2X}(t)$ wurde über den „Black Box“-Anteil bestimmt.

In Abbildung 25 ist der Verlauf der gemessenen und der durch die „Grey Box“ berechneten Methanolkonzentration c_{S2M} zu erkennen. Es wurden Stichproben zu verschiedenen Neuronenanzahlen und Inputkonfigurationen durchgeführt, da aber keine der Konfigurationen eine Verbesserung des durchschnittlichen RMSE für die Testsets unter 100 gl⁻¹ aufwies, wurden keine weiteren Optimierungsversuche durchgeführt. Diese großen Fehler rühren wahrscheinlich daher, dass die berechnete Substrataufnahmerate sehr sprunghaft verlief,

wie in den kleinen Plots in Abbildung 25 zu erkennen ist, und diese Sprünge nicht adäquat bei den maximal eingesetzten zweckmäßigen Neuronenzahlen dargestellt werden konnten. Es wurden nie Neuronenzahlen über 100 betrachtet, da eine hohe Trainingsdauer unerwünscht war und die Speicherzuweisung von MATLAB® nach oben hin begrenzt war. Aufgrund der starken Sprunghaftigkeit des $q_{S2/X}$ konnte dieser aber bei kleineren Neuronenzahlen nicht durch ein neuronales Netzwerk ausreichend beschrieben werden.

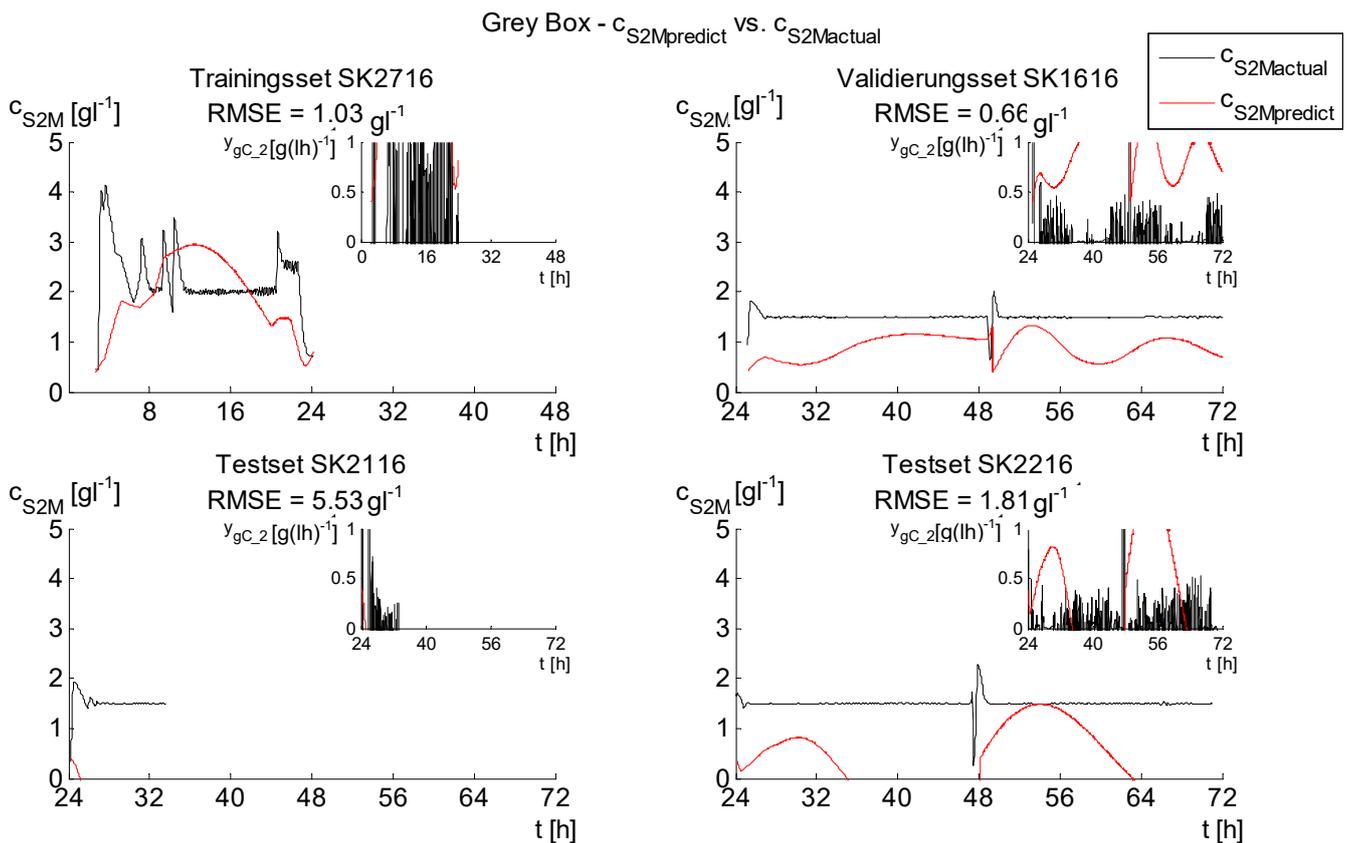


Abbildung 26: Gemessene und durch „Grey Box“ bestimmte Methanolkonzentration c_{S2M} für alle vier Kultivierungen im C30. $y_{gC_2}(t) = V_L(t) \cdot V_M(t)^{-1} \cdot q_{S2X}(t) \cdot c_{XL}(t)$ wurde über den „Black Box“-Anteil bestimmt.

Auch in der zweiten Variation der „Grey Box“ für c_{S2M} wurde ähnliches beobachtet, wie in Abbildung 26 erkennbar. Die starke Sprunghaftigkeit von y_{gC_2} konnte nicht ausreichend beschrieben werden, und eine gute Beschreibung von c_{S2M} fand nicht statt.

Dies wurde auch für c_{S2M} im ED10 beobachtet, daher sind diese Ergebnisse hier nicht weiter dargestellt.

Für die Modellierung des pO_2 wurden jeweils zwei Wege betrachtet, die zudem auch wieder in zwei mögliche „Black Box“-Ausgaben gesplittet wurden. Es wurden die quasi stationären Gleichgewichte $OTR = Q_{O_2}$ und $OTR = OUR$ betrachtet. Für jeweils beide Wege wurden zudem zwei verschiedene Parameter als Ausgabe des „Black Box“-Anteiles betrachtet. Für $OTR = Q_{O_2}$ waren dies der $k_L a(t)$ und $y_{g_OQ_2}(t) = Q_{O_2}(t) \cdot (k_L a(t) \cdot C_{OLmax}(t))^{-1}$.

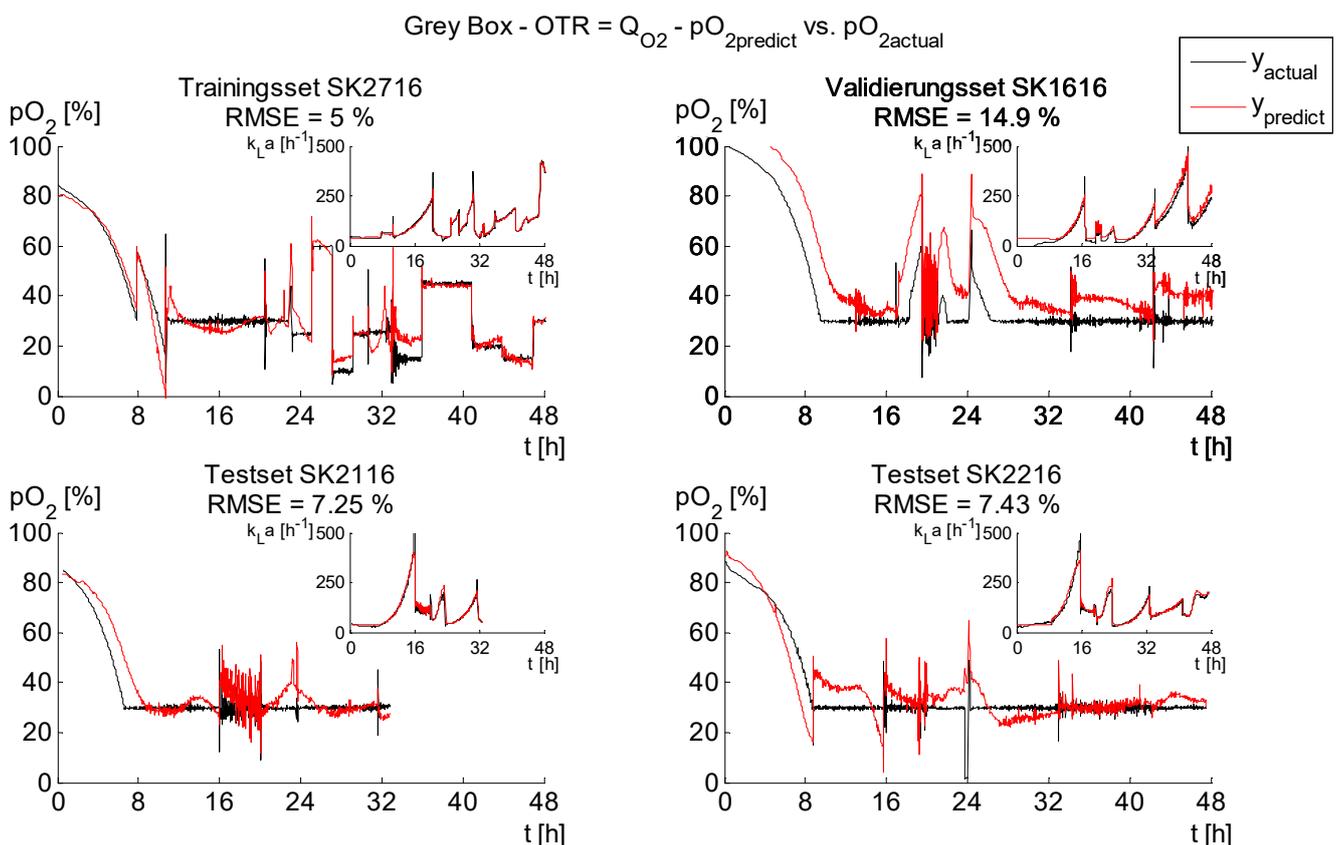


Abbildung 27: Gemessener und durch „Grey Box“ bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im ED10. Der volumenbezogene Sauerstoffübergangskoeffizient $k_L a(t)$ wurde über den „Black Box“-Anteil bestimmt.

In Abbildung 27 ist zu erkennen, dass die Beschreibung des pO_2 mit diesem Modell zwar besser war als mit der reinen „White Box“, aber dennoch schlechter als mit der „Black Box“. Die „Grey Box“ ist ein Hybrid aus diesen beiden unter der Vermutung, dass diese die Schwächen des jeweiligen reinen Verfahrens ausgleichen. Dies war hier nicht der Fall - die RMSE, insbesondere im Validierungsset, waren zu groß, um dieses Modell weiter in Betracht zu ziehen. Dies galt auch für die pO_2 -Bestimmung im C30.

Der zweite „Black Box“-Ausgabe-Parameter der betrachtet wurde war

$$y_{g_oQ_2} = Q_{O_2}(t) \cdot (k_{La}(t) \cdot c_{OLmax}(t))^{-1}.$$

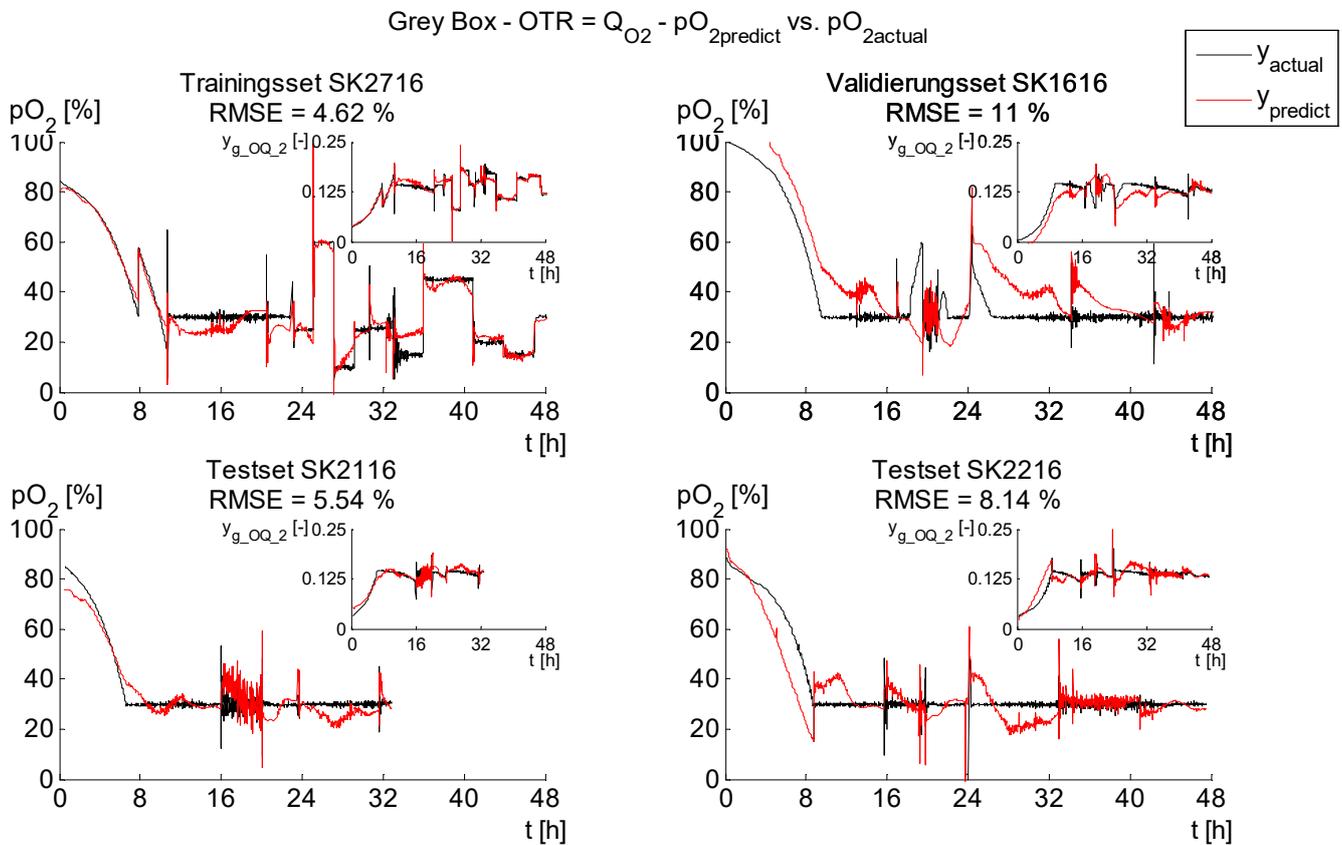


Abbildung 28: Gemessener und durch „Grey Box“ bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im ED10. $y_{g_oQ_2}(t) = Q_{O_2}(t) \cdot (k_{La}(t) \cdot c_{OLmax}(t))^{-1}$ wurde über den „Black Box“-Anteil bestimmt.

In Abbildung 28 ist zu erkennen, dass dieser Ausgabeparameter zwar zu etwas kleineren RMSE führt als nur der k_{La} , die Verbesserung aber im Hinblick auf die reine „Black Box“ nicht signifikant war. Daher eignete sich auch diese „Grey Box“ nicht für die Bestimmung des pO_2 im ED10. Auch dies galt im selben Maße für den Sauerstoffpartialdruck im C30. Lediglich das Trainingsset SK2716 wurde einigermaßen genau beschrieben mit einem RMSE von unter 2 %, alle anderen Kultivierungen wiesen sehr große RMSE auf mit diesem Modell.

Die zweite „Grey Box“ wurde über das quasi-stationäre Gleichgewicht $OTR = OUR$ versucht zu ermittelt. Als erster „Black Box“-Anteil wurde $y_{g_oo_1}(t) = q_{O/X}(t) \cdot OTR_{max}(t)^{-1}$ gewählt.

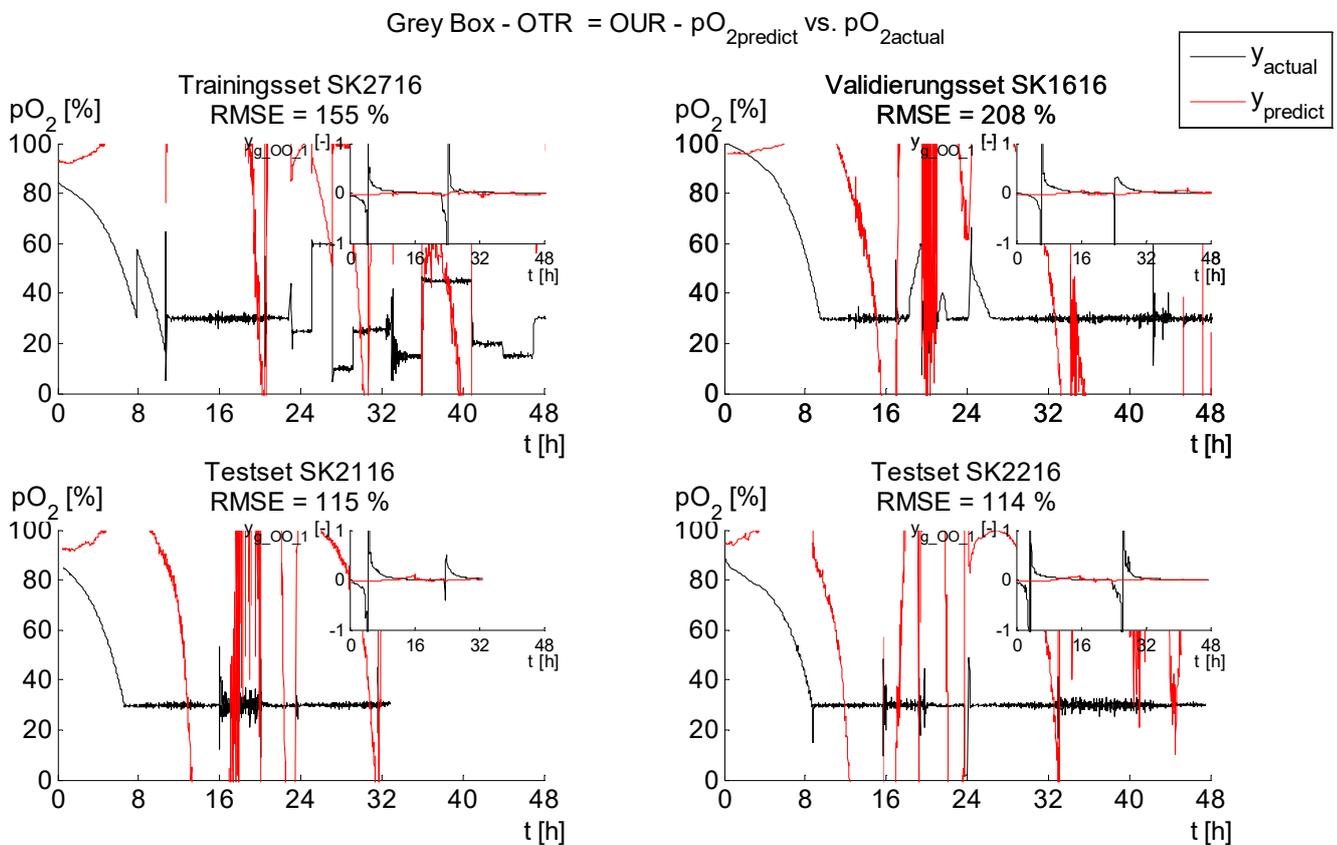


Abbildung 29: Gemessener und durch „Grey Box“ bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im ED10. $y_{g_oo_1}(t) = q_{O/X}(t) \cdot OTR_{max}(t)^{-1}$ wurde über den „Black Box“-Anteil bestimmt.

In Abbildung 29 ist deutlich zu erkennen, dass diese Art des Modells absolut ungeeignet war für die Ermittlung des pO_2 . Die sprunghaften Änderungen in $y_{g_oo_1}$ konnten nicht beschrieben werden. Auch ist erkennbar, dass sehr kleine Abweichungen von berechnetem $y_{g_oo_1}$ und dem durch die „Black Box“ ausgegebene, zu sehr großen Abweichungen in der Berechnung des pO_2 führten. Wegen dieser beiden Gründe führte die Ausgabe der „Black Box“ zu sehr großen Schwankungen im Modellergebnis. Dies galt auch für den C30, wenn auch in etwas geringerem Maße.

Als zweiter „Black Box“-Anteil für die „Grey Box“ $OTR = OUR$ wurde $y_{g_oo_2} = q_{O/X}(t) \cdot C_{XL}(t) \cdot OTR_{max}(t)^{-1}$ gewählt.

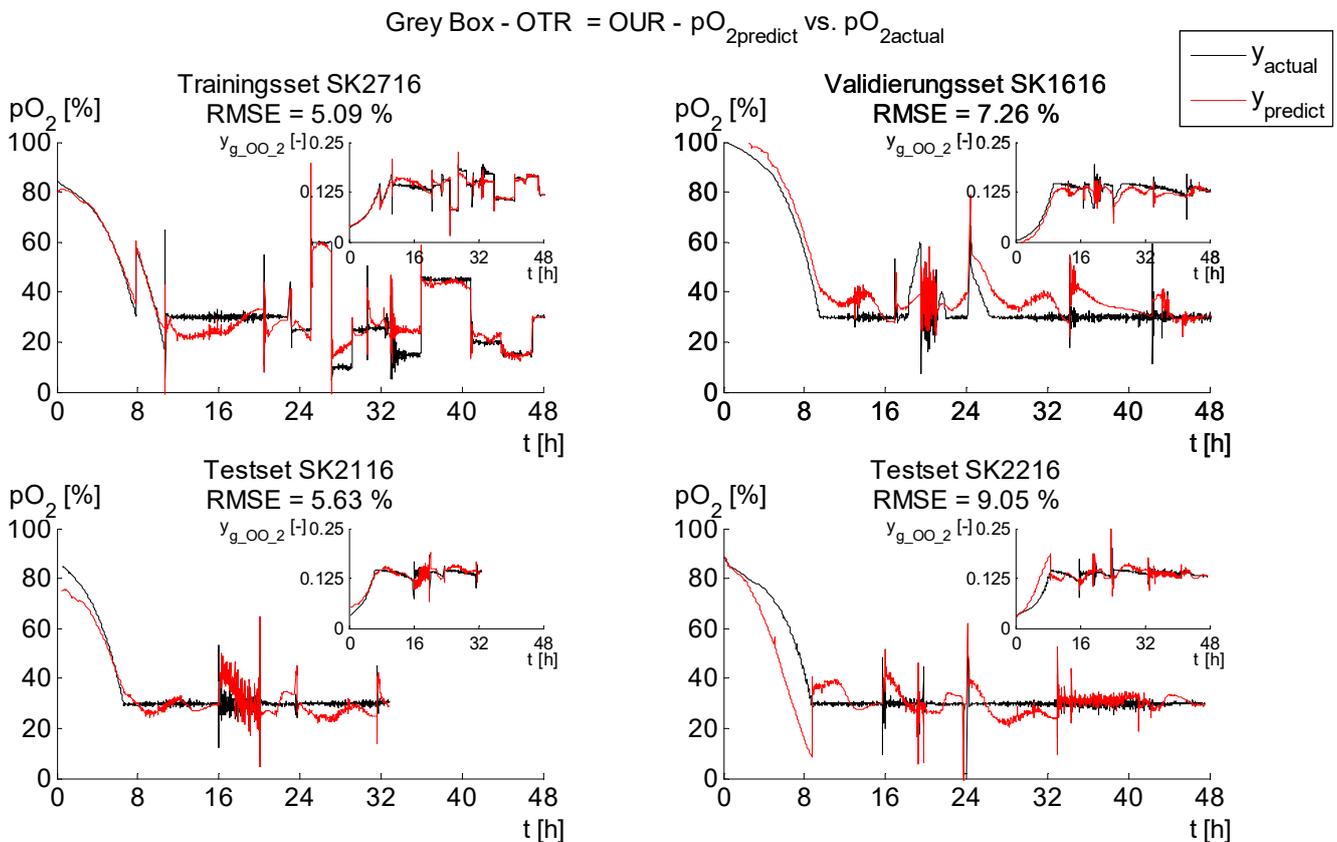


Abbildung 30: Gemessener und durch „Grey Box“ bestimmter Sauerstoffpartialdruck pO_2 für alle vier Kultivierungen im ED10. $y_{g_oo_1}(t) = q_{O/X}(t) \cdot OTR_{max}(t)^{-1}$ wurde über den „Black Box“-Anteil bestimmt.

Beim Vergleich von Abbildung 30 mit Abbildung 28 ist zu erkennen, dass zwischen $y_{g_oo_2}$ und $y_{g_oo_1}$ kein Unterschied besteht. Dies war auch zu erwarten, da beide Parameter in Gleichung (24) und (30) den gleichen Teil der pO_2 -Gleichung beschreiben. Daher traf auf diese beiden „Grey Boxes“ das gleiche zu. Auch hier war eine Verbesserung zu dem ersten „Black Box“-Anteil, $y_{g_oo_1}$, zu erkennen, aber wies dennoch schlechtere Ergebnisse auf als die reine „Black Box“. Das gleiche traf auch auf den C30 zu.

Tabelle 7: Durchschnittlicher RMSE der Zielvariablen berechnet mit Hilfe der „Grey Box“.

Zielvariable	Reaktor	Durchschnittlicher RMSE aller Kultivierungen			
„Black Box“-Anteil:		$q_{S2/X}$		$y_{gC_2} = V_L \cdot V_M^{-1} \cdot q_{S2X} \cdot C_{XL}$	
c_{S2M}	ED10	121,3 gl^{-1}		16,8 gl^{-1}	
$c_{S2Mw} = 1,5 \text{ } gl^{-1}$	C30	70,1 gl^{-1}		2,3 gl^{-1}	
„Black Box“-Anteil:		$k_L a$	$y_{g_{OQ_2}} = Q_{O_2} \cdot (k_L a \cdot c_{OLmax})^{-1}$	$y_{g_{OQ_1}} = q_{O/X} \cdot OTR_{max}^{-1}$	$y_{g_{OQ_2}} = q_{O/X} \cdot c_{XL} \cdot OTR_{max}^{-1}$
pO_2	ED10	8,6 %	7,3 %	148 %	6,8 %
$pO_{2w} = 30 \%$	C30	28,7 %	9,6 %	52,8 %	8,1 %

Alle durchschnittlichen RMSE der „Grey Box“ wurden in Tabelle 7 aufgelistet. Es war deutlich, dass die „Grey Box“ für die Bestimmung der Methanolkonzentration absolut nicht geeignet war, was an den großen Fehler dieser Modelle erkennbar war.

Auch für den pO_2 zeigte die „Grey Box“ keine signifikanten Vorteile gegenüber der „Black Box“. Bei den „Black Box“-Anteilen $k_L a$ für den C30 und $y_{g_{OQ_1}}$ für beide zeigten sich sehr große Abweichungen, die dieses Modell für die Überwachung der pO_2 -Sonde unbrauchbar machen. Und da auch die anderen „Grey Box“-Modelle für den pO_2 keine Verbesserung zeigten, ist deren Einsatz nicht vonnöten.

4.3.5 Vergleich der Modelle

In Tabelle 8 sind alle durchschnittlichen RMSE aller Modelle für beide Zielvariablen aufgelistet. Es wurde jeweils der kleinste Fehler für die Zielvariable und den Reaktor rot markiert. Es ist erkennbar, dass die besten Ergebnisse durch die „Black Box“ neuronalen Netzwerke erreicht wurden. Alle anderen Modellierungswege zeigten größere Fehler.

Tabelle 8: Durchschnittliche RMSE der Zielvariablen aller Kultivierungen und aller Modelle

Reaktor	Modell		durchschnittlicher RMSE aller Kultivierungen			
			$c_{S2M} [g\ l^{-1}]$ $c_{S2Mw} = 1,5$		$pO_2 [\%]$ $pO_{2w} = 30$	
ED10	„White Box“		1,6		36,3	
	Kalman-Filter		2,2		551,0	
	PCR		5,4		17,0	
	multivariate lineare Regression		2,4		10,6	
	„Black Box“	Feedforward	0,4		5,6	
		Regressiv	3,0		36,0	
	„Grey Box“ Variante 1		$q_{S2/X}$	121,3	$k_L a$	8,6
					$Y_{g_OQ_2}$	7,3
„Grey Box“ Variante 2		Y_{g_C2}	16,8	$Y_{g_OQ_1}$	148,0	
				$Y_{g_OQ_2}$	6,8	
C30	„White Box“		4,5		35,8	
	Kalman-Filter		6,5		1982	
	PCR		4,2		9,5	
	multivariate lineare Regression		0,4		6,7	
	„Black Box“	Feedforward	0,3		2,6	
		Regressiv	0,17		75,0	
	„Grey Box“ Variante 1		$q_{S2/X}$	70,1	$k_L a$	28,7
					$Y_{g_OQ_2}$	9,6
„Grey Box“ Variante 2		Y_{g_C2}	2,3	$Y_{g_OQ_1}$	52,8	
				$Y_{g_OQ_2}$	8,1	

5 Zusammenfassung

Ziel dieser Arbeit war die Entwicklung eines „Soft Sensors“ zur Überwachung der beiden Hardwaresonden für die Sauerstoffpartialdruckmessung und die Methanolkonzentrationsmessung in den Reaktoren ED10 und C30. Zudem wurde auch noch die Zellkonzentration mit der Basentitration korreliert. Für die Entwicklung des „Soft Sensors“ wurden verschiedene Modellierungswege betrachtet. Zuerst wurde die „White Box“ untersucht, bei der bekannte Grundlagengleichungen verwendet wurden. Dafür wurden die Parameter $q_{S2/X}$ und k_{La} nach van't Riet bestimmt. Es wurde versucht, die Modelle über den Extended Kalman-Filter und die Hauptkomponentenregression zu erzeugen. Es wurden die Regressionsparameter für eine multivariate lineare Regression bestimmt. Zuletzt wurden Feedforward und regressive neuronale Netzwerke betrachtet in der Form als „Black Box“ und als „Grey Box“ mit verschiedenen „Black Box“-Ausgabeparametern.

Um dieses Ziel zu erreichen wurden vier Kultivierungen durchgeführt. Bei SK2716 wurden, um eine gute Datenvariabilität zu erreichen, die Set-Points pO_{2w} , N_{Stw} , F_{R2relw} und c_{S2Mw} variiert. Anhand der Daten dieser Kultivierung wurden dann die verschiedenen Modelle entwickelt. Die Kultivierung SK1616 diente bei den neuronalen Netzwerken als Validierungsdatensatz um eine gute Generalisierung der Netze zu gewährleisten. Für alle anderen Modelle diente sie, zusammen mit SK2116 und SK2216 als Testdatensatz.

Die „White Box“ eignete sich nicht zur Beschreibung von c_{S2M} und pO_2 , da die RMSE sehr groß waren. Für pO_2 war der durchschnittliche Fehler immer über 30 % und der RMSE von c_{S2M} immer über 1 g l^{-1} . Wahrscheinlich lag das an der Rigidität der Modelle, sodass die Batch- zu Batch-Variabilität nicht beschrieben werden konnte und dass die Bestimmung der Parameter nicht optimal war.

Der Kalman-Filter eignete sich absolut nicht zur Beschreibung der Zielvariablen, da zum einen die Grundlagengleichungen, die schon in der „White Box“ zu großen Fehlern führten, hier wiederverwendet wurden und zum anderen die Mathematik hinter dem Kalman-Filter nicht zu dem untersuchten Problem passend war.

Die PCR und die multivariate lineare Regression zeigten beide ebenfalls zu hohe RMSE, die PCR wies dabei höhere auf. Gründe dafür waren wahrscheinlich, dass beide Verfahren auf Linearitäten in der Beziehung der Variablen beruhen, was bei den Daten hier nicht gegeben war.

Die „Grey Box“ neuronalen Netzwerke zeigten keine Verbesserung gegenüber den „Black Box“ Netzwerken, und führte teilweise sogar zu viel größeren Fehlern. Diese großen Fehler rührten vor allem daher, dass die berechneten Parameter zum Teil eine starke Sprunghaftigkeit aufwiesen, was durch die Netzwerke nicht beschrieben werden konnte.

Die „Black Box“ neuronalen Netzwerke wiesen die kleinsten RMSE auf und beschrieben die Zielvariablen am besten. Dabei beschrieb das Feedforward-Netzwerk den pO_2 in beiden Reaktoren und $c_{S_{2M}}$ im ED10 besser, das regressive Netzwerk dagegen den $c_{S_{2M}}$ im C30. Aber trotz der kleineren Fehler war die Beschreibung durch die Netzwerke noch nicht optimal.

Zusammenfassend lässt sich sagen, dass sich die „Black Box“ neuronalen Netzwerke am vielversprechendsten zeigten. Aber dennoch waren die Ausgabefehler noch zu groß um dieses tatsächlich als „Soft Sensor“ einzusetzen. Als „Soft Sensor“ zur Überwachung der Hardwaresonden wären ein RMSE für pO_2 von unter 1,5 % und für $c_{S_{2M}}$ $0,075 \text{ g l}^{-1}$, was 5 % vom Set-Point entsprach, in Ordnung gewesen. Dies wurde aber von keinem der Modelle erreicht. Für weitere Betrachtungen könnten noch andere Wege betrachtet werden, wie die Support Vector Regression, die Gaussian Mixture Regression, andere Grundlagengleichungen oder sogar die Neuentwicklung von Grundlagengleichungen. Zudem könnten noch andere Hyperparameter der neuronalen Netzwerke variiert werden, wie Lernraten oder die Fehlerfunktion. Des Weiteren könnten auch andere Netzwerktypen untersucht werden. Zuletzt könnten noch weitere Datenaufnahmewege in den Prozess eingebaut werden, wie zum Beispiel NIR-Spektroskopie, die etwas mehr Unabhängigkeit zu vorhandenen Daten aufweisen. Es existiert also noch Optimierungspotential, das noch verfolgt werden könnte.

6 Literaturverzeichnis

Browne AJ, Buckee GK, Nye JWS. Patent: Measurement of alcohol WO1990013026 A1. Crisp Instrumentation Ltd. 1990.

Chui CK, Chen G. Kalman Filtering with Real-Time Applications. Springer. 2009.

Clark LC, Wolf R, Granger D, Taylor Z. Continuous recording of blood oxygen tensions by polarography. *Journal of Applied Physiology* 10 (6). 1953. Seiten 189-193.

Foresee FD, Hagan MT. Gauss-Newton Approximation to Bayesian Learning. *International Conference on Neural Networks* 3. 1997. Seiten 1930-1935.

Hagan MT, Menhaj MB. Training Feedforward Networks with the Marquardt Algorithm. *IEEE Transactions on Neural Networks* 5 (6). 1994. Seiten 989-993.

Hecht-Nielsen R. Theory of the Backpropagation Neural Network. *International Joint Conference on Neural Networks*. 1989. Seiten 593-605.

Hornik K. Approximation Capabilities of Multilayer Feedforward Networks. *Neural Networks* 4(2). 1991. Seiten 251-257.

Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural networks* 2 (5). 1989. Seiten 359-366.

Kadlec P, Gabrys B, Strandt S. Data-driven Soft Sensors in the process industry. *Computers and Chemical Engineering* 33. 2009. Seiten 795-814.

Kessler W. Multivariate Datenanalyse für die Pharma-, Bio- und Prozessanalytik. Ein Lehrbuch. Wiley-VCH. 2007.

Lin B, Recke B, Knudsen JKH, Jørgensen SB. A systematic approach for soft sensor development. *Computers & Chemical Engineering* 31 (5-6). 2007. Seiten 419-425.

Mandenius CF, Gustavsson R. Mini-review: soft sensors as means for PAT in the manufacture of bio-therapeutics. *Journal of Chemical Technology and Biotechnology* 90. 2015. Seiten 215-227.

Nelles O. Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models. Springer. 2001.

Rumelhart DE, Hinton GE, Williams RJ. Learning representation by back-propagating errors. Nature 323. 1986. Seiten 533-536.

Anhang

Anhang 1: "Learning the Extended Kalman Filter" von Yi Cao, Cranfield University

<https://de.mathworks.com/matlabcentral/fileexchange/18189-learning-the-extended-kalman-filter> (13.07.2016, 10:21)

```
function [x,P]=ekf(fstate,x,P,hmeas,z,Q,R)
% EKF Extended Kalman Filter for nonlinear dynamic systems
% [x, P] = ekf(f,x,P,h,z,Q,R) returns state estimate, x and state covariance, P
% for nonlinear dynamic system:
%     x_k+1 = f(x_k) + w_k
%     z_k    = h(x_k) + v_k
% where w ~ N(0,Q) meaning w is gaussian noise with covariance Q
%     v ~ N(0,R) meaning v is gaussian noise with covariance R
% Inputs:  f: function handle for f(x)
%          x: "a priori" state estimate
%          P: "a priori" estimated state covariance
%          h: function handle for h(x)
%          z: current measurement
%          Q: process noise covariance
%          R: measurement noise covariance
% Output:  x: "a posteriori" state estimate
%          P: "a posteriori" state covariance
%
% Example:
%{
n=3;          %number of state
q=0.1;       %std of process
r=0.1;       %std of measurement
Q=q^2*eye(n); % covariance of process
R=r^2;       % covariance of measurement
f=@(x) [x(2);x(3);0.05*x(1)*(x(2)+x(3))]; % nonlinear state equations
h=@(x) x(1); % measurement equation
s=[0;0;1];   % initial state
x=s+q*randn(3,1); %initial state with noise
P = eye(n);  % initial state covariance
N=20;       % total dynamic steps
xV = zeros(n,N); %estimate % allocate memory
sV = zeros(n,N); %actual
zV = zeros(1,N);
for k=1:N
    z = h(s) + r*randn; % measurments
    sV(:,k)= s; % save actual state
    zV(k) = z; % save measurment
    [x, P] = ekf(f,x,P,h,z,Q,R); % ekf
    xV(:,k) = x; % save estimate
    s = f(s) + q*randn(3,1); % update process
end
for k=1:3 % plot results
    subplot(3,1,k)
    plot(1:N, sV(k,:), '-', 1:N, xV(k,:), '--')
end
%}
% By Yi Cao at Cranfield University, 02/01/2008
%
```

```

[x1,A]=jaccsd(fstate,x); %nonlinear update and linearization at current
state
P=A*P*A'+Q; %partial update
[z1,H]=jaccsd(hmeas,x1); %nonlinear measurement and linearization
P12=P*H'; %cross covariance
% K=P12*inv(H*P12+R); %Kalman filter gain
% x=x1+K*(z-z1); %state estimate
% P=P-K*P12'; %state covariance matrix
R=chol(H*P12+R); %Cholesky factorization
U=P12/R; %K=U/R'; Faster because of back substitution
x=x1+U*(R'\(z-z1)); %Back substitution to get state update
P=P-U*U'; %Covariance update, U*U'=P12/R/R'*P12'=K*P12.

```

```

function [z,A]=jaccsd(fun,x)
% JACCSD Jacobian through complex step differentiation
% [z J] = jaccsd(f,x)
% z = f(x)
% J = f'(x)
%
z=fun(x);
n=numel(x);
m=numel(z);
A=zeros(m,n);
h=n*eps;
for k=1:n
    x1=x;
    x1(k)=x1(k)+h*i;
    A(:,k)=imag(fun(x1))/h;
end

```