

Optimierung eines PID-gesteuerten DC-DC-Konverters mit maschinellem Lernen

Patryk Krzyzanski

Erstprüfer: Prof. Dr.-Ing. Bernhard Wicht

Zweitprüfer: Dr.-Ing. Markus Olbrich

Leibniz Universität Hannover

Abteilung: Institut für Mikroelektronische Systeme

October 22, 2023

Abstract

In dieser Arbeit wird die Verwendung neuronaler Netze zur Optimierung und Steuerung eines PID-regulierten DC-Konverters untersucht. Das Ziel besteht darin, ein System zu entwickeln, das in der Lage ist, die altersbedingte Degradation von Schaltungskomponenten wie Kapazität und Induktivität zu überwachen und anzupassen, um die Leistung des Konverters aufrechtzuerhalten. Ein besonderer Fokus liegt auf dem Trainingsprozess und der Architektur des neuronalen Netzes. Der Trainingsprozess wird mithilfe von Methoden wie Deep Deterministic Policy Gradient (DDPG) und Bayesscher Optimierung umgesetzt. Das Training und die Schaltungssimulation werden unter Einsatz von Transientenanalyse mit SystemC durchgeführt, um eine präzise Bewertung und Auswertung der Simulationsergebnisse zu ermöglichen. Es werden Techniken zur Optimierung der Hyperparameter des neuronalen Netzes vorgestellt. Herausforderungen und Lösungsansätze im Kontext der neuronalen Netzarchitektur und des Trainings werden diskutiert. Abschließend werden die erzielten Ergebnisse und ihre Implikationen für zukünftige Forschungen präsentiert.

Contents

1	Einleitung	2
1.1	Hintergrund und Motivation	2
1.2	Problemstellung	2
1.3	Relevanz und Forschungslage	2
1.4	Forschungsmethode und Ansatz	2
1.5	Ziele und Aufbau der Arbeit	2
2	Grundlagen	3
2.1	Elektrotechnik	3
2.1.1	Buck-Konverters	3
2.1.2	Degradation von Kondensatoren und MOSFETs in DC-DC-Konvertern	4
2.1.3	PID-Regler	5
2.1.4	Pulsweitenmodulation und ihre Darstellung	6
2.2	Informationstechnologie	6
2.2.1	Einführung in Neuronale Netzwerke	6
2.2.2	Vorwärtspropagation in Neuronalen Netzwerken	8
2.2.3	Gradientenberechnung	9
2.2.4	Rückpropagation in Neuronalen Netzwerken	9
2.2.5	Backpropagation im Training neuronaler Netze	10
A	Anhang	11

Chapter 1

Einleitung

1.1 Hintergrund und Motivation

Die effiziente Übertragung elektrischer Energie von einer Quelle zu einem Verbraucher stellt eine entscheidende Herausforderung in der Elektrotechnik dar. Im Mittelpunkt stehen dabei Gleichspannungswandler (DC-DC Konverter), die in vielfältigen Anwendungen von Energieübertragungssystemen bis hin zu mobilen Geräten eine Rolle spielen [17, p. 70].

1.2 Problemstellung

Jedoch werden diese Systeme durch Degradationseffekte, besonders von Schlüsselkomponenten wie Kapazitäten, zunehmend beeinträchtigt. Solche Degradationen können die Lebensdauer und Effizienz von elektronischen Systemen nachhaltig schädigen und erfordern dringende wissenschaftliche Untersuchung [9, p. 1].

1.3 Relevanz und Forschungslage

Erste Studien in diesem Bereich, wie die von Jeong et al. und Kulkarni et al., haben bereits die drastischen systemischen Auswirkungen solcher Degradationen demonstriert [11, p. 3]. Dadurch wird die Notwendigkeit unterstrichen, innovative Lösungsansätze für dieses Problem zu entwickeln.

1.4 Forschungsmethode und Ansatz

Ein vielversprechender Lösungsansatz ist die Nutzung von künstlichen neuronalen Netzen (KNN). Diese bieten, wie die Arbeiten von Brunton und Kutz sowie Almajlawi et al. nahelegen, eine hervorragende Plattform für die Steuerung komplexer Systeme und könnten somit eine Alternative zu traditionellen Reglern bieten [3, p. 270] [2, p. 8].

1.5 Ziele und Aufbau der Arbeit

Diese Arbeit zielt darauf ab, die Anwendungsmöglichkeiten von KNN zur Überwachung und Kompensation der Degradation in DC-Konvertern systematisch zu untersuchen. Im Fokus stehen dabei die Architektur des neuronalen Netzes, verschiedene Trainingsmethoden und -umgebungen, sowie die spezifischen Herausforderungen und Lösungsansätze im Kontext des Trainingsprozesses.

Chapter 2

Grundlagen

Einleitung zum Kapitel

Dieses Kapitel dient als umfassende Grundlage für die Erforschung der Rolle neuronaler Netze in der Optimierung und Steuerung von PID-regulierten DC-Konvertern. Im Fokus stehen sowohl die Grundlagen der DC-DC-Konvertertechnologie als auch spezielle Herausforderungen, die in diesem Kontext auftreten können, wie beispielsweise die altersbedingte Degradation von Schaltungskomponenten. Darüber hinaus bietet das Kapitel einen Überblick über moderne Optimierungsmethoden wie DDPG (Deep Deterministic Policy Gradients) und Bayessche Optimierung, die in der aktuellen Forschung Bedeutung erlangt haben.

Der Inhalt dieses Kapitels zielt darauf ab, den Leser umfassend auf die Herausforderungen, technischen Lösungen und innovativen Ansätze in diesem sich schnell entwickelnden Forschungsfeld vorzubereiten.

2.1 Elektrotechnik

2.1.1 Buck-Konverters

Hauptkomponenten und Funktionen eines DC-DC-Konverters Die Wandlung von Gleichspannung (DC) in eine andere Gleichspannung ist ein kritischer Aspekt in der Elektronik und Energieversorgung. Ein weit verbreitetes Schaltungsdesign, das diese Funktion ausführt, ist der Buck-Konverter. In der Literatur wird dieser als eine Standardmethode für DC-DC-Wandlung beschrieben [17, p. 66].

MOSFET-Transistor Der MOSFET-Transistor agiert als elektronischer Schalter, der den Stromfluss in der Schaltung reguliert. Im Vergleich zu alternativen Schaltelementen bietet der MOSFET eine signifikante Effizienzsteigerung durch minimale Leistungsverluste. Dies wird durch Phänomene wie Trägermobilität und die damit verbundene Widerstandsfähigkeit gegenüber thermischen Ausfällen ermöglicht [4, p. 29].

Induktivität (Spule) Die Induktivität dient der temporären Energiespeicherung in Form eines magnetischen Feldes, das beim Stromfluss durch die Spule generiert wird. Dies ist insbesondere relevant in Anwendungen wie Solenoid-Antriebsschaltungen, wo die Induktivität als Energiespeicher und -überträger fungiert [4, p. 54].

Diode Die Diode ist so ausgerichtet, dass sie den Strom nur in einer Richtung passieren lässt. Dies ist insbesondere wichtig, wenn der MOSFET-Transistor deaktiviert ist. Als passive Schalter werden oftmals schnelle Erholungsdiolen oder Schottky-Dioden aufgrund ihrer exzellenten Schalteigenschaften verwendet [4, p. 29].

Kondensator Der Kondensator dient der Glättung der Ausgangsspannung und speichert Energie für die Last. Er spielt eine wichtige Rolle in der Dynamik der Schaltung und ermöglicht eine stabilere Energieversorgung [10, p. 54].

Regelung und Anwendungen In der Praxis werden Buck-Konverter oft von einer nicht-idealen Spannungsquelle gespeist und müssen daher unter variablen Eingangsspannungen und Lastströmen arbeiten [4, p. 124, 120, 113]. Daher ist eine geschlossene Regelungsschleife erforderlich, um eine konstante Ausgangsspannung sicherzustellen.

Buck-Konverter finden eine breite Anwendung in verschiedenen elektronischen Geräten und Systemen. Ihr hoher Wirkungsgrad, der in der Regel zwischen 75% und 98% liegt, macht sie besonders attraktiv.

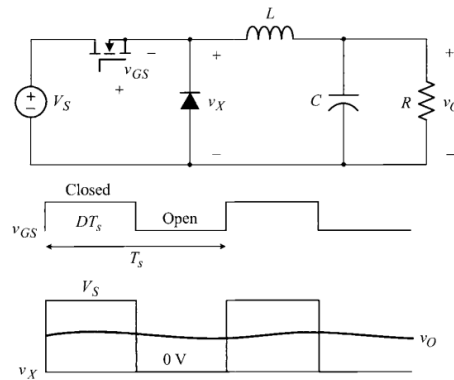


Figure 3.2 Buck converter and major waveforms.

Figure 2.1: Schematische Darstellung eines DC-DC Konverters. Quelle: [4, Seite 88]

2.1.2 Degradation von Kondensatoren und MOSFETs in DC-DC-Konvertern

Die Zuverlässigkeit und Effizienz von DC-DC-Konvertern sind zunehmend von der Degradation ihrer Schlüsselkomponenten, insbesondere von Kondensatoren und MOSFETs, beeinträchtigt.

Kondensatoren Kondensatoren sind anfällig für verschiedene Arten von Ausfallmodi, darunter Änderungen des Verlustfaktors ($\tan \delta$), der Impedanz und des Dissipationsfaktors. Diese Parameter sind entscheidend für die Beurteilung der Zuverlässigkeit eines Systems. Ebenso ist die erhöhte Äquivalente Serienresistenz (ESR) von Elektrolytkondensatoren, die elektrischen und thermischen Belastungen ausgesetzt sind, ein weiterer entscheidender Faktor für die Degradation.[9, pp. 1]

MOSFETs Bei MOSFETs kann die Degradation aufgrund von thermischen Spannungen zu einem Gate-Source-Kurzschluss oder einem Drain-Source-Kurzschluss führen. Die Degradation der Transistoren erhöht deren Leistungsverluste und beschleunigt damit den Degradationsprozess weiter.[17, pp. 190]

Kontrolle und Überwachung Aktuelle Forschungsbemühungen konzentrieren sich auf die Entwicklung von Kontrollalgorithmen, um die Degradation zu verzögern und die Zuverlässigkeit der Konverter zu erhöhen. Dazu gehören auch Verfahren zur Schätzung des Zustands der Degradation in Echtzeit. [4, p. 24, p. 310-311]

Integration mit neuronalen Netzen für Kondensatoren und MOSFETs Neuronale Netze können verwendet werden, um aktiv entgegensteuernde Maßnahmen zur Verlangsamung der Degradation von Schlüsselkomponenten wie Kondensatoren und MOSFETs in DC-DC-Konvertern einzuleiten. Durch die kontinuierliche Analyse von Betriebsparametern wie Temperatur und Spannung sind diese Netze in

der Lage, den Zustand der Degradation in Echtzeit zu erfassen. Sobald kritische Zustände erkannt werden, können die neuronalen Netze automatisch die PID-Koeffizienten des Konverters anpassen, um die Degradation zu minimieren und die Zuverlässigkeit des Systems zu erhöhen. [12, p. 22]

2.1.3 PID-Regler

Der PID-Regler (Proportional-Integral-Derivativ) ist eine weit verbreitete Regelungsstrategie in industriellen Steuerungssystemen und verschiedenen Arten von Anwendungen. Er ist unerlässlich für die Regelung von Prozessen wie Geschwindigkeit, Temperatur und Spannung [8, p. 2].

Proportionalanteil (P) Diese Komponente erzeugt einen Ausgangswert, der proportional zum aktuellen Fehlerwert ist. Die proportionale Reaktion kann durch Multiplikation des Fehlers mit einer Konstanten namens K_p eingestellt werden, die als Proportionalverstärkung bezeichnet wird.

$$P_{\text{out}} = K_p \times e(t) \quad (2.1)$$

Integralanteil (I) Diese Komponente befasst sich mit der Akkumulation vergangener Fehler. Wenn der Fehler über einen längeren Zeitraum vorhanden war, wird er akkumuliert (Integral des Fehlers), und der Regler wird den Steuerausgang in Beziehung zu einer Konstanten K_i ändern, die als Integralverstärkung bekannt ist.

$$I_{\text{out}} = K_i \times \int e(t) dt \quad (2.2)$$

Differentiantanteil (D) Diese Komponente liefert einen Steuerausgang, um die Änderungsrate des Fehlers zu kompensieren. Der Beitrag des Differenzierungsanteils zur gesamten Steueraktion wird als Differenzierungsverstärkung K_d bezeichnet.

$$D_{\text{out}} = K_d \times \frac{d}{dt} e(t) \quad (2.3)$$

[14, p. 1744]

Die PID-Regelungsgleichung Die PID-Regelungsgleichung kombiniert diese drei Komponenten, um den Steuerausgang zu erzeugen:

$$\text{Steuerausgang} = P_{\text{out}} + I_{\text{out}} + D_{\text{out}} \quad (2.4)$$

$$\text{Steuerausgang} = (K_p \times e(t)) + (K_i \times \int e(t) dt) + (K_d \times \frac{d}{dt} e(t)) \quad (2.5)$$

Einstellung der Verstärkungsfaktoren Die Konstanten K_p , K_i , und K_d werden eingestellt, um die optimale Systemleistung zu erreichen; ein schlecht eingestellter PID-Regler kann instabil, langsam oder schwingend sein.

Anwendungen bei Gleichstrom-Gleichstrom-Wandlern Im Kontext von Gleichstrom-Gleichstrom-Wandlern können PID-Regler helfen, die Ausgangsspannung zu stabilisieren, indem sie die Ausgangsspannung kontinuierlich mit der gewünschten Spannung vergleichen und handeln, um den Fehler durch Anpassung des Tastverhältnisses des Schaltelements zu minimieren [2, p. 4].

Fazit Der PID-Regler ist eine vielseitige und weit verbreitete Regelungsstrategie. Seine Anpassungsfähigkeit und Effizienz machen ihn ideal für eine breite Palette von Anwendungen, von industriellen Prozessen bis zu modernen Technologiesystemen. Für eine erweiterte Diskussion über verschiedene Varianten von PID-Reglern, wie zum Beispiel den Fuzzy PID-Controller, könnten Sie das Paper "Shi2020AdaptiveController" verwenden [15, p. 9].

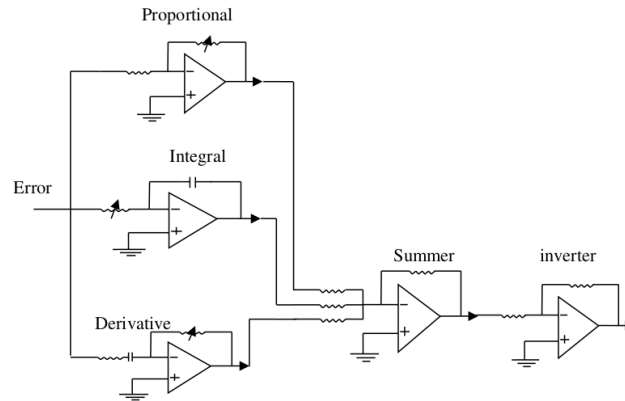


Figure 2.2: Schematische Darstellung eines PID-Regler. Quelle: [16, p. 18]

2.1.4 Pulsweitenmodulation und ihre Darstellung

Pulsweitenmodulation (PWM) ist eine Schlüsseltechnik in DC-DC-Wandlern, die zur Steuerung der Schaltkomponenten eingesetzt wird, um die Ausgangsspannung oder den Ausgangsstrom zu regulieren. Sie ermöglicht eine präzise Kontrolle, indem sie die 'Einschaltzeit' des Schalters im Vergleich zur gesamten Zykluszeit (Einschaltzeit + Ausschaltzeit) variiert.[13, p. 2]

Tastverhältnis D Das Tastverhältnis D wird mathematisch als das Verhältnis der Einschaltzeit zur gesamten Zykluszeit beschrieben:

$$D = \frac{\text{Einschaltzeit}}{\text{Einschaltzeit} + \text{Ausschaltzeit}} \quad (2.6)$$

Proportionalanteil (P) Das Tastverhältnis spielt eine wichtige Rolle, da es den Mittelwert der Ausgangsspannung oder des Ausgangsstroms bestimmt. Bei der PWM wird ein Steuersignal mit einem hochfrequenten Trägersignal verglichen, um die 'Ein'- und 'Aus'-Zustände des Schalters festzulegen. Das Steuersignal stammt oft von höheren Regelkreisen wie PID-Reglern, die den Fehler zwischen Soll- und Istwert minimieren sollen.

Die Hauptmotivation für die Verwendung von PWM in Steuerungssystemen ist die Anpassung des Mittelwerts der Ausgabe an ein Referenzsignal. Zusätzlich wird versucht, harmonische Verzerrungen und Schaltverluste zu minimieren [13, p. 2].

In der Abbildung oben ist eine typische PWM-Schaltung dargestellt. Die PWM-Block und die Spannungsrückführungsschaltung im DC-DC-Wandler arbeiten zusammen, um sicherzustellen, dass die Ausgangsspannung der Referenzspannung im stationären Zustand folgt. Hierbei wird ein Steuersignal v_{con} und ein Rampensignal V_{ramp} verwendet, um die Impulsbreite des aktiven Schalters zu modulieren. Das rechte Diagramm (b) zeigt die Steuersignale und ihre Relation zueinander, wodurch das Schaltverhalten des Wandlers beeinflusst wird [4, p. 114].

2.2 Informationstechnologie

2.2.1 Einführung in Neuronale Netzwerke

Neuronale Netzwerke bilden das rechnerische Grundgerüst für eine Vielzahl von Aufgaben in den Bereichen maschinelles Lernen und künstliche Intelligenz. Sie sind von dem komplexen Netzwerk an Neuronen im menschlichen Gehirn inspiriert und versuchen, biologische Lernprozesse nachzuahmen [1, p. 21]. In diesem Zusammenhang bieten sie ein robustes und flexibles Rahmenwerk zur Lösung komplexer Herausforderungen [6, p. 29].

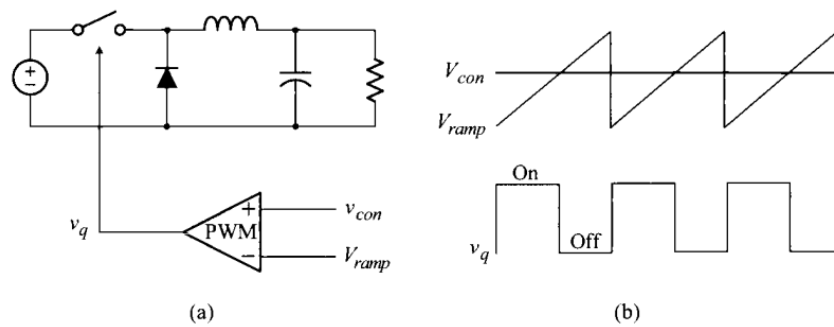


Figure 2.3: Schematische Darstellung eines PWM-Modulator. Quelle: [4, p. 114]

Vorteile von Neuronalen Netzwerken

Neuronale Netzwerke bieten mehrere entscheidende Vorteile, die ihren Einsatz in unterschiedlichen Anwendungsbereichen attraktiv machen:

- **Parallelität:** Sie sind für die parallele Verarbeitung konzipiert und ermöglichen daher schnelle Berechnungen sowie Echtzeitverarbeitung.
- **Nichtlineare Funktionsapproximation:** Die Netzwerke sind besonders gut darin, nichtlineare Funktionen zu approximieren [6, p. 214], was sie vielseitig einsetzbar macht.
- **Modellgeneralisierung:** Neuronale Netzwerke können aus einer begrenzten Datenmenge generalisieren und somit präzise Vorhersagen für unbekannte Eingaben treffen.

Diese Vorteile bilden die Grundlage für ihre breite Anwendbarkeit, die im nächsten Abschnitt erläutert wird.

Unterschied zwischen Biologischen und Künstlichen Neuronalen Netzwerken

Künstliche neuronale Netzwerke bestehen aus rechnerischen Einheiten, den sogenannten Neuronen. Diese sind durch anpassbare Gewichtungen verbunden, die der Stärke synaptischer Verbindungen in biologischen Systemen ähneln. Lernen erfolgt durch die Anpassung dieser Gewichtungen, ähnlich wie sich die Stärken synaptischer Verbindungen in biologischen Systemen als Reaktion auf Reize ändern [1, p. 21].

Deep Learning als Spezialisierung

Deep Learning stellt einen spezialisierten Unterbereich des maschinellen Lernens dar, der neuronale Netzwerke mit drei oder mehr Schichten verwendet. Diese tiefen Netzwerke führen eine hierarchische Merkmalsextraktion durch, die es ihnen ermöglicht, immer komplexere Muster und Merkmale zu erkennen, während die Daten durch die Schichten fließen.

Zusammenfassung und Ausblick

Zusammenfassend bieten neuronale Netzwerke ein leistungsfähiges Rahmenwerk für eine Vielzahl von Aufgaben, von einfacher Mustererkennung bis hin zu komplexen Entscheidungsfindungsprozessen. Der Einsatz von mehrschichtigen Architekturen und nichtlinearen Aktivierungsfunktionen erweitert die Fähigkeiten traditioneller maschineller Lernalgorithmen [1, p. 40].

Mit dieser Grundlage werden die folgenden Abschnitte einen vertieften Einblick in die mathematischen Aspekte von neuronalen Netzwerken bieten. Insbesondere werden wir uns darauf konzentrieren, wie neuronale Netzwerke bei der Optimierung und Steuerung von PID-regulierten DC-Konvertern Anwendung finden. Dabei liegt der Fokus auf der Berücksichtigung von Alterungsprozessen und Abnutzung von Schaltungselementen wie Kapazitäten und Induktivitäten und wie diese Einflüsse mathematisch modelliert und optimiert werden können.

2.2.2 Vorwärtspropagation in Neuronalen Netzwerken

Schicht-für-Schicht-Propagation

Beginnend mit der Eingabeschicht $A^{[0]}$, die im Wesentlichen die Eingabedaten X sind, berechnet jede nachfolgende Schicht $Z^{[l]}$ und $A^{[l]}$ entsprechend den oben genannten Gleichungen. Dies bildet das Kernstück der Vorwärtspropagation.

Dimensionalität und Netzwerkarchitektur

Die Anzahl der Neuronen in jeder Schicht und die Art der verwendeten Aktivierungsfunktion können die Leistung des Netzwerks erheblich beeinflussen. Es ist wichtig, die Dimensionalität jeder Schicht während der Entwurfsphase zu berücksichtigen, um ein effektives Lernen sicherzustellen.

Die Vorwärtspropagation ist ein wesentlicher Prozess in neuronalen Netzwerken, der die Übertragung von Eingabedaten durch die Netzwerkarchitektur ermöglicht, um die Ausgabe zu erzeugen [14, p. 1421]. Sie ist eine Abfolge von mathematischen Operationen, die Gewichtungen, Biases und Aktivierungsfunktionen involvieren [5, p. 73].

Gewichtsmatrix $W^{[l]}$ und Bias-Vektor $b^{[l]}$

Die Gewichtsmatrix für die Schicht l wird als $W^{[l]}$ bezeichnet, und $b^{[l]}$ ist der Bias-Vektor für dieselbe Schicht [7, p. 46]. Diese Parameter werden während des Backpropagation-Prozesses trainiert, um den Fehler zwischen der vorhergesagten und der tatsächlichen Ausgabe zu minimieren [1, p. 41].

$$Z^{[l]} = W^{[l]}A^{[l-1]} + b^{[l]} \quad (2.7)$$

Aktivierungsfunktionen

Eine Aktivierungsfunktion, normalerweise durch σ bezeichnet, transformiert die gewichtete Summe $Z^{[l]}$ in die aktivierte Ausgabe $A^{[l]}$ [14, p. 1421].

$$A^{[l]} = \sigma(Z^{[l]}) \quad (2.8)$$

$$A^{[l]} = \sigma \left(\begin{pmatrix} w_{1,1}^{[l-1,l]} & w_{1,2}^{[l-1,l]} & \cdots & w_{1,m}^{[l-1,l]} \\ w_{2,1}^{[l-1,l]} & w_{2,2}^{[l-1,l]} & \cdots & w_{2,m}^{[l-1,l]} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,1}^{[l-1,l]} & w_{n,2}^{[l-1,l]} & \cdots & w_{n,m}^{[l-1,l]} \end{pmatrix} \begin{pmatrix} A_1^{[l-1]} \\ A_2^{[l-1]} \\ \vdots \\ A_m^{[l-1]} \end{pmatrix} + \begin{pmatrix} b_1^{[l]} \\ b_2^{[l]} \\ \vdots \\ b_n^{[l]} \end{pmatrix} \right) \quad (2.9)$$

Schicht-für-Schicht-Propagation

Beginnend mit der Eingabeschicht $A^{[0]}$, die im Wesentlichen die Eingabedaten X sind, berechnet jede nachfolgende Schicht $Z^{[l]}$ und $A^{[l]}$ entsprechend den oben genannten Gleichungen [14, p. 1421].

Dimensionalität und Netzwerkarchitektur

Die Anzahl der Neuronen in jeder Schicht und die Art der verwendeten Aktivierungsfunktion können die Leistung des Netzwerks erheblich beeinflussen [14, p. 1408]. Es ist wichtig, die Dimensionalität jeder Schicht während der Entwurfsphase zu berücksichtigen, um ein effektives Lernen sicherzustellen [5, p. 73].

2.2.3 Gradientenberechnung

Sie haben eine Kostenfunktion C_0 definiert als:

$$C_0 = \sum_{j=0}^{n_L-1} (a_j^{[L]} - y_j)^2 \quad (2.10)$$

Um die Kostenfunktion zu minimieren, müssen Sie den Gradienten in Bezug auf alle Gewichtungen und Bias berechnen. Mit Hilfe der Kettenregel können Sie die partielle Ableitung der Kostenfunktion in Bezug auf jedes Gewicht wie folgt ausdrücken:

$$\frac{\partial C_0}{\partial w_{jk}^{[L]}} = \frac{\partial w_{jk}^{[L]}}{\partial z_j^{[L]}} \frac{\partial z_j^{[L]}}{\partial a_j^{[L]}} \frac{\partial a_j^{[L]}}{\partial C_0} \quad (2.11)$$

Wenn Sie den Nabla-Operator verwenden, um den Gradienten der Kostenfunktion in Bezug auf alle Gewichtungen in der Matrixform darzustellen, erhalten Sie:

$$\nabla_{W^{[L]}} C_0 = \begin{pmatrix} \frac{\partial C_0}{\partial w_{11}^{[L]}} & \frac{\partial C_0}{\partial w_{12}^{[L]}} & \dots & \frac{\partial C_0}{\partial w_{1m}^{[L]}} \\ \frac{\partial C_0}{\partial w_{21}^{[L]}} & \frac{\partial C_0}{\partial w_{22}^{[L]}} & \dots & \frac{\partial C_0}{\partial w_{2m}^{[L]}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial C_0}{\partial w_{n1}^{[L]}} & \frac{\partial C_0}{\partial w_{n2}^{[L]}} & \dots & \frac{\partial C_0}{\partial w_{nm}^{[L]}} \end{pmatrix} \quad (2.12)$$

Dabei ist $\nabla_{W^{[L]}} C_0$ die Matrix der partiellen Ableitungen der Kostenfunktion C_0 in Bezug auf jede Gewichtung in $W^{[L]}$.

2.2.4 Rückpropagation in Neuronalen Netzwerken

Für die Rückpropagation definieren wir den Fehler in der Ausgabeschicht durch:

$$\frac{\partial C_0}{\partial a_j^{[L]}} = 2 \left(a_j^{[L]} - y_j \right)$$

wobei $a_j^{[L]}$ die Aktivierung der j -ten Einheit in der Ausgabeschicht und y_j der tatsächliche Wert für diese Einheit ist.

Fehler Rückpropagieren Der nächste Schritt besteht darin, den Fehler durch das Netzwerk zurückzupropagieren.

Um den Beitrag jeder Gewichtung und jedes Bias zur Gesamtkostenfunktion zu ermitteln, berechnen wir die Ableitung der Aktivierungsfunktion $a_j^{[L]}$ in Bezug auf die lineare Kombination $z_j^{[L]}$:

$$\frac{\partial z_j^{[L]}}{\partial a_j^{[L]}} = \sigma' \left(z_j^{[L]} \right)$$

wobei $\sigma' \left(z_j^{[L]} \right)$ die Ableitung der Aktivierungsfunktion ist.

Um die Ableitung der linearen Kombination $z_j^{[L]}$ in Bezug auf die Gewichtung $w_{jk}^{[L]}$ zu berechnen, verwenden wir:

$$\frac{\partial z_j^{[L]}}{\partial w_{jk}^{[L]}} = a_k^{[L-1]}$$

Kettenregel Anwenden Jetzt kombinieren wir alle diese Teile mit der Kettenregel:

$$\frac{\partial C_0}{\partial w_{jk}^{[L]}} = 2 \left(a_j^{[L]} - y_j \right) \cdot \sigma' \left(z_j^{[L]} \right) \cdot a_k^{[L-1]}$$

Gradienten der Kostenfunktion Schließlich, um den Gradienten der Kostenfunktion C_0 in Bezug auf alle Gewichtungen darzustellen, verwenden wir:

$$\nabla W^{[L]} C_0 = \left(2 \left(a^{[L]} - y \right) \odot \sigma' \left(Z^{[L]} \right) \right) A^{[L-1]T}$$

Mit diesen Gradienten können Sie die Gewichtungen und Biasse im neuronalen Netzwerk aktualisieren, indem Sie einen Optimierungsansatz wie den Gradientenabstieg verwenden.

2.2.5 Backpropagation im Training neuronaler Netze

Mathematische Grundlagen

Die Gradientenmatrix für die Gewichtungen in Bezug auf die Kostenfunktion C_0 unter Verwendung des Nabla-Operators kann in der Matrixform dargestellt werden:

Dabei setzt sich jede Komponente der Matrix wie folgt zusammen:

$$\frac{\partial C_0}{\partial w_{ij}^{[L]}} = 2(a_i^{[L]} - y_i) \cdot \sigma'(z_i^{[L]}) \cdot a_j^{[L-1]}$$

So erhalten wir die gesamte Matrix der partiellen Ableitungen:

$$\nabla_{W^{[L]}} C_0 = \begin{pmatrix} 2(a_1^{[L]} - y_1) \cdot \sigma'(z_1^{[L]}) \cdot a_1^{[L-1]} & \dots & 2(a_1^{[L]} - y_1) \cdot \sigma'(z_1^{[L]}) \cdot a_m^{[L-1]} \\ 2(a_2^{[L]} - y_2) \cdot \sigma'(z_2^{[L]}) \cdot a_1^{[L-1]} & \dots & 2(a_2^{[L]} - y_2) \cdot \sigma'(z_2^{[L]}) \cdot a_m^{[L-1]} \\ \vdots & \ddots & \vdots \\ 2(a_n^{[L]} - y_n) \cdot \sigma'(z_n^{[L]}) \cdot a_1^{[L-1]} & \dots & 2(a_n^{[L]} - y_n) \cdot \sigma'(z_n^{[L]}) \cdot a_m^{[L-1]} \end{pmatrix}$$

Die Größe dieser Matrix ist $n \times m$, wobei n die Anzahl der Neuronen in Schicht L und m die Anzahl der Neuronen in Schicht $L - 1$ ist.

Jeder Eintrag in dieser Matrix gibt die Änderungsrate der Kostenfunktion C_0 in Bezug auf das entsprechende Gewicht an. Mit dieser Matrix können Sie die Gewichtungen in Schicht L aktualisieren, um die Kostenfunktion zu minimieren.

Appendix A

Anhang

Bibliography

- [1] Charu C. Aggarwal. *Neural Networks and Deep Learning*. Library of Congress Control Number: 2018947636. Yorktown Heights, NY, USA: Springer International Publishing AG, part of Springer Nature, 2018. ISBN: 978-3-319-94462-3. DOI: 10.1007/978-3-319-94463-0.
- [2] Muhanad D. Hashim Almawlawe, Muhammad Al-badri, and Issam Hayder Alsakini. “Performance Improvement of a DC/DC Converter Using Neural Network Controller in comparison with Different Controllers”. In: *Unknown Journal* Unknown Volume.Unknown Number (2023), Unknown Pages.
- [3] Steven L. Brunton and J. Nathan Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. University Printing House, Cambridge CB2 8BS, United Kingdom: Cambridge University Press, 2019. ISBN: 978-1-108-42209-3. DOI: 10.1017/9781108380690. URL: <https://www.cambridge.org/9781108422093>.
- [4] Byungcho Choi. *Pulsewidth Modulated DC-to-DC Power Conversion: Circuits, Dynamics, and Control Designs*. IEEE Press series on power engineering. Hoboken, New Jersey: IEEE Press, Wiley, 2013. ISBN: 978-1-118-18063-1.
- [5] François Chollet. *Deep Learning with Python, Second Edition*. Covers topics from the fundamentals of machine learning and deep learning to advanced topics in computer vision and text processing. Shelter Island, NY: Manning Publications Co., 2021. ISBN: 9781617296864. URL: <https://www.manning.com/>.
- [6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [7] Jeff Heaton. *Introduction to the Math of Neural Networks*. Ed. by WordsRU.com. Cover Art: Carrie Spear. Heaton Research Inc., May 2012. ISBN: 978-1475190878.
- [8] Emad A. Hussein and Musa H. Waly. “Proportional-Integral (PID) Controller Design Using Genetic Algorithm (GA)”. In: *Al-Qadisiya Journal For Engineering Sciences* 4.2 (2011).
- [9] Jaeyoon Jeong, Sangshin Kwak, and Seungdeog Choi. “Degradation-Sensitive Control Algorithm Based on Phase Optimization for Interleaved DC–DC Converters”. In: *Machines* 11 (2023). Received: 1 May 2023, Revised: 29 May 2023, Accepted: 30 May 2023, Published: 5 June 2023, p. 624. DOI: 10.3390/machines11060624. URL: <https://www.mdpi.com/journal/machines>.
- [10] Nihal Kularatna. *DC Power Supplies: Power Management and Surge Protection for Power Electronic Systems*. Version Date: 2011916, No claim to original U.S. Government works. Boca Raton, London, New York: CRC Press, Taylor & Francis Group, 2012. ISBN: 978-0-415-80248-2.
- [11] Chetan Kulkarni et al. “Model-based Avionics Systems Fault Simulation and Detection”. In: *[Journal Name - Not Provided]* (2023).
- [12] Miguel Morales. *Grokking Deep Reinforcement Learning*. With a forew. by Jr. Charles Isbell. 20 Baldwin Road, PO Box 761, Shelter Island, NY 11964: Manning Publications Co., Dec. 2020. URL: <https://www.manning.com>.
- [13] Satish Kumar Peddapelli. *Pulse Width Modulation Analysis and Performance in Multilevel Inverters*. Library of Congress Cataloging-in-Publication Data Applied for. Hyderabad, Telangana State, India: Walter de Gruyter GmbH, 2017. ISBN: 978-3-11-046817-5. URL: <http://www.degruyter.com>.

- [14] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4th ed. Pearson Series in Artificial Intelligence. Cover images by various photographers and artists. Hoboken, NJ: Pearson Education, Inc., 2021. ISBN: 978-0-13-461099-3. URL: <https://lccn.loc.gov/2019047498>.
- [15] Qian Shi et al. “Adaptive Neuro-Fuzzy PID Controller based on Twin Delayed Deep Deterministic Policy Gradient Algorithm”. In: *Neurocomputing* (2020). Journal Pre-proof, Communicated by Prof. Zidong Wang. DOI: <https://doi.org/10.1016/j.neucom.2020.03.063>.
- [16] Tapan Kumar Swain and Vaibhav Baid. “Analog Fabrication of PID Controller”. This work was carried out under the supervision of Prof. Sandip Ghosh for the partial fulfillment of the requirements for the award of End semester thesis Submission in Electrical Engineering at National Institute of Technology, Rourkela (Deemed University). Bachelor’s Thesis. Rourkela, Orissa-769008: National Institute of Technology, Rourkela, May 2014.
- [17] Mike Wens and Michiel Steyaert. *Design and Implementation of Fully-Integrated Inductive DC-DC Converters in Standard CMOS*. Series Editors: Mohammed Ismail; Mohamad Sawan. ESAT-MICAS, Dept. Elektrotechniek, K.U. Leuven. Leuven, Belgium: [Publisher Name - Not Provided], 2022.