

# Methoden des maschinellen Lernens als Notfallstrategie nach Seilrissen bei parallelen Seilrobotern

## Machine Learning Methods as an Emergency Strategy after Cable Breaks of Cable-Driven Parallel Robots

Patrick Gust\*, Christian Hürten, Roland Boumann, Tobias Bruckmann  
Universität Duisburg-Essen, Lehrstuhl für Mechatronik, 47057 Duisburg, Deutschland

\*Kontakt: patrick.gust@uni-due.de

### Kurzfassung

In einem aktuellen Projekt der Universität Duisburg-Essen, gefördert durch die Deutsche Forschungsgemeinschaft (DFG), werden *Strategien zur Schadensvermeidung nach Seilrissen in parallelen Seilrobotern* entwickelt. Die zunehmend anwendungsorientierten Forschungsarbeiten auf dem Gebiet der Seilrobotik, beispielsweise der automatische Mauerroboter CARLO oder die weltweit in Stadien eingesetzte SkyCam, werfen die Frage nach der Sicherheit von Seilrobotern auf. Die sehr kritische Fehlerquelle eines Seilrisses ist bereits mehrmals bei bekannten Seilrobotern aufgetreten. Um das Risiko für Mensch und Maschine zu reduzieren, sind aus der Literatur einige Notfallstrategien nach Seilrissen bekannt. Diese Ansätze sind in der Simulation analysiert und experimentell validiert worden, müssen aber Einschränkungen in Kauf nehmen, um die Echtzeitfähigkeit zu erreichen. In diesem Beitrag wird einen Ansatz aus dem Bereich des maschinellen Lernens vorgestellt.

### Abstract

In a current project at the University of Duisburg-Essen, funded by the German Research Foundation (DFG), *Strategies for Damage Prevention after Cable Breaks in Parallel Cable Robots* are being developed. The increasingly application-oriented research work in the field of cable robots, for example the automatic wall building robot CARLO or the SkyCam used in stadiums worldwide, raises the question of safety for cable robots. The very critical source of error caused by a cable break has already occurred several times with well-known cable robots. In order to reduce the risk for man and machine, several emergency strategies after cable breaks are known from the literature. These approaches have been analysed in simulation and experimentally validated, but must accept limitations in order to achieve real-time capability. This article presents an approach from the field of machine learning.

## 1 Einleitung

Ein Seilroboter ist ein parallelkinematischer Roboter, der als Endeffektor eine mobile Plattform besitzt, die über Seile mit computergesteuerten Winden aktuiert wird [1]. Die Forschung zu Seilrobotern veränderte zuletzt den Fokus von der Grundlagenforschung [2] hin zu anwendungsorientierteren Projekten [3,4]. Der Seilroboter CARLO nutzt den Vorteil der großen Arbeitsräume seilbasierter Robotersysteme aus, um automatisch Steine für den Bau eines Hauses zu setzen [3]. Ein anderes Beispiel ist die weltweit in Stadien eingesetzte Skycam, welche an Seilen befestigt über Sportevents schwebt und die Athleten filmt [4]. Der Wechsel zu anwendungsorientierten Forschungsarbeiten wirft die Frage nach der Sicherheit seilbasierter Robotersystem auf. Im Jahr 2011 stürzte beispielsweise eine Skycam wegen eines gerissenen Seils auf das Spielfeld und verfehlte einen Spieler nur knapp [5]. Ein weiteres Beispiel ist der Einsturz des baufälligen Arecibo-Teleskops im Jahr 2020 [6].

Aus der Literatur sind einige Ansätze für Notfallstrategien nach Seilrissen bekannt [7–10].

In [7] wird eine Notfallstrategie für einen Seilroboter mit drei Freiheitsgraden präsentiert, in der eine sichere Halte-

position mit Hilfe einer geradlinigen Trajektorie angefahren wird. Die Autoren wählen diesen relativ simplen Ansatz aufgrund der Echtzeitfähigkeit dieser Methode [8].

Boumann et al. stellen in [9] zwei modellbasierte Methoden zur Schadensvermeidung nach Seilrissen bei einem Seilroboter mit sechs Freiheitsgraden vor und validieren diese experimentell in [10]. Eine der Methoden ist eine einschrittige nichtlineare modellprädiktive Regelung (Nonlinear Model Predictive Control, NMPC) [9]. Diese NMPC minimiert die kinetische Energie des Endeffektors bei Detektion eines Seilrisses und verfolgt so das Ziel, den Endeffektor schnellstmöglich anzuhalten.

Modellprädiktive Regelungen vereinen das vorausschauende Modellwissen mit dem Vorteil der Rückführung einer klassischen Regelung und reagieren somit auf Prozessstörungen [11]. Ein Nachteil ist die notwendige Rechenzeit des Optimierungsalgorithmus [11].

In der Literatur finden sich alternative Ansätze, um NMPCs durch Verfahren des maschinellen Lernens zu ersetzen.

Sieberg et al. zeigen in [13], dass sich mithilfe eines Co-Active Neuro-Fuzzy Inference System (CANFIS) die Rechenzeit im Vergleich zu einer NMPC für die aktive Wankstabilisierung [12] um den Faktor 8.85 reduzieren lässt.



CANFIS ist eine Weiterentwicklung des Adaptive Neuro-Fuzzy Inference System (ANFIS, [14]) und besonders für MIMO Systeme geeignet [15]. Erste Untersuchungen im Vorfeld dieses Beitrages ergaben, dass ein CANFIS als Lösung für den hier beschriebenen Anwendungsfall nicht praktikabel ist.

Ein weiterer Ansatz aus dem Bereich des maschinellen Lernens wird von Wang et. al in [16] vorgestellt. Die Autoren bilden das Verhalten einer modellprädiktiven Regelung in einem Stromrichter durch ein künstliches neuronales Netz ab. Damit wird der Rechenaufwand deutlich reduziert und es werden genauere Systemmodelle höherer Ordnung verwendet.

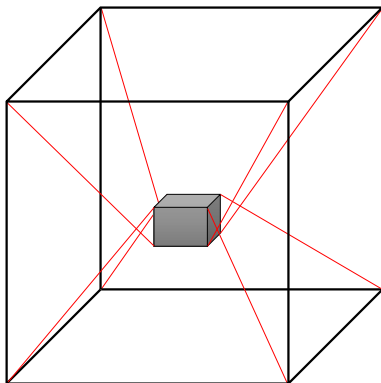
In diesem Beitrag wird die NMPC zur Schadensvermeidung nach Seilrissen [9] durch ein künstliches neuronales Netz ersetzt. Die durch Boumann et. al präsentierte NMPC dient als Grundlage zur simulativen Generierung von Trainingsdaten.

Der Beitrag stellt zunächst die Modellierung und Simulation paralleler Seilroboter vor. Dabei wird die NMPC als Notfallstrategie nach Seilrissen vorgestellt. Anschließend wird die angewandte Methodik erläutert, welche auf den Grundlagen von künstlichen neuronalen Netzen basiert. Die Ergebnisse des Trainings und der Simulation werden präsentiert und diskutiert. Abschließend folgt eine Zusammenfassung und ein Ausblick auf die nächsten Forschungsarbeiten.

## 2 Modellierung und Simulation paralleler Seilroboter

Ein paralleler Seilroboter (*engl.* Cable Driven Parallel Robot, CDRP) verbindet eine mobile Plattform, auch Endeffektor, über Seile und Umlenkrollen mit aktuierten Winden und ist beispielhaft in **Bild 1** abgebildet. Ein redundanter Seilroboter hat mehr Seile  $m$  als Freiheitsgrade  $n$  des Endeffektors, was zu einer Redundanz  $r$  führt. [1]

$$r = m - n \quad (1)$$



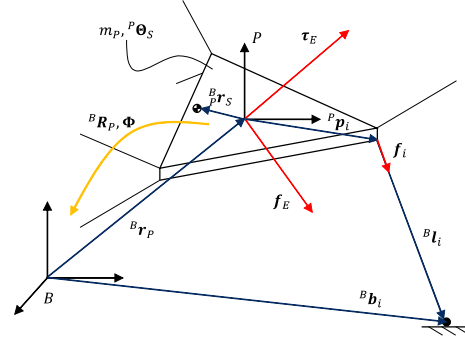
**Bild 1** Paralleler Seilroboter mit  $m = 8$  Seilen und  $n = 6$  Freiheitsgraden

Der in diesem Beitrag betrachtete Seilroboter gehört mit einer Redundanz von  $r = 2$  nach Verhoeven zu den sogenannten Redundantly Restrained Position Mechanisms

(RRPMs) [17]. Es handelt sich um den Seilroboter SEGESTA der Universität Duisburg-Essen [2].

### 2.1 Modellierung

Im folgenden werden die grundlegenden Ansätze und Gleichungen zur Simulation eines CDRPs nach Boumann et. al beschrieben [18]. **Bild 2** skizziert die Modellparameter eines CDRPs.



**Bild 2** Skizze der geometrischen Zusammenhänge eines Seilroboters, angelehnt an [18]

Es wird das körperfeste Koordinatensystem  $P$  des Endeffektors und das raumfeste Koordinatensystem  $B$  des Rahmens eingeführt. Die Pose  ${}^B\mathbf{x}_P$  des Endeffektors besteht aus der Position des Endeffektors  ${}^B\mathbf{r}_P$  und der Orientierung  $\Phi$  mit Bezug auf das Koordinatensystem  $B$ .

$${}^B\mathbf{x}_P = [{}^B\mathbf{r}_P \quad \Phi]^\top \quad (2)$$

Die Rotationsmatrix  ${}^B\mathbf{R}_P$  beschreibt die Orientierung des Endeffektors bzw. des Koordinatensystems  $P$  mit Bezug auf das Koordinatensystem  $B$ .

Die Geometrie des Seilroboters wird durch die Anknüpfungspunkte  ${}^P\mathbf{p}_i$  der Seile am Endeffektor sowie der Ausgangspunkte der Seile an den Umlenkrollen  ${}^B\mathbf{b}_i$  am Rahmen beschrieben. Hier wird zur Vereinfachung von einem punktförmigen Seilaustritt ausgegangen. Die Seilvektoren  ${}^B\mathbf{l}_i$  lassen sich, unter der Annahme gespannter, masseloser und linienförmig verlaufender Seile, aus der inversen Kinematik des Roboters ableiten:

$${}^B\mathbf{l}_i = {}^B\mathbf{b}_i - ({}^B\mathbf{r}_P + {}^B\mathbf{R}_P\mathbf{p}_i), \quad 1 \leq i \leq m \quad (3)$$

Die Seilkräfte  $f_i$ , die von den Motoren erzeugt werden, ziehen in Richtung der Seilvektoren am Endeffektor. Es wird der Richtungsvektor der Seilkraft  $\mathbf{v}_i$  eingeführt.

$${}^B\mathbf{f}_i = f_i \frac{{}^B\mathbf{l}_i}{\|{}^B\mathbf{l}_i\|_2} = f_i {}^B\mathbf{v}_i, \quad 1 \leq i \leq m \quad (4)$$

Der Index des Koordinatensystems wird im Folgenden der Einfachheit halber weggelassen.

Mit den eingepägten Kräften  $\mathbf{f}_E$  und Momenten  $\boldsymbol{\tau}_E$ , die auf den Endeffektor einwirken, wird das Kräftegleichgewicht gebildet.

$$\underbrace{\begin{bmatrix} -\mathbf{f}_E \\ -\boldsymbol{\tau}_E \end{bmatrix}}_{-\mathbf{w}_E} = \underbrace{\begin{bmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_m \\ \mathbf{p}_1 \times \mathbf{v}_1 & \dots & \mathbf{p}_m \times \mathbf{v}_m \end{bmatrix}}_{\mathbf{A}^\top} \underbrace{\begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix}}_{\mathbf{f}} \quad (5)$$

Vereinfacht wird der Zusammenhang durch die Strukturmatrix des Roboters  $\mathbf{A}^\top \in \mathbb{R}^{n \times m}$  sowie dem Seilkraftvektor  $\mathbf{f} \in \mathbb{R}^{m \times 1}$  und dem Kraftwinder  $\mathbf{w}_E \in \mathbb{R}^{n \times 1}$ .

Es ist zu beachten, dass die Seile am Endeffektor nur ziehen und niemals drücken. Um ein Erschlaffen der Seile zu verhindern, wird neben einer maximal erzeugbaren Seilkraft  $f_{max}$  eine minimale Seilkraft  $f_{min}$  definiert. Das überbestimmte Gleichungssystem (5) wird somit bei gegebenem Kraftwinder  $\mathbf{w}_E$ , einer Redundanz  $r = 2$  und gesuchter Seilkraftverteilung  $\mathbf{f}$  mit  $f_i \in [f_{min}, f_{max}]$  zu einem Optimierungsproblem [19].

Die dynamischen Bewegungsgleichungen des Seilroboters finden sich durch Aufstellen der Newton-Euler-Gleichungen [20] für den Endeffektor. Dazu wird die Masse des Endeffektors  $m_P$  mit dem Ortsvektor  ${}^P\mathbf{r}_S$ , welcher vom Koordinatensystems  $P$  zum Schwerpunkt  $S$  zeigt, eingeführt.  ${}^P\mathbf{r}_S$  beschreibt dessen schiefsymmetrische Matrix. Die Anwendung des Satzes von Huygens-Steiner liefert den Trägheitstensor  ${}^B\mathbf{I}_P$  des Endeffektors bezüglich des raumfesten Koordinatensystems  $B$  unter Verwendung des Trägheitstensors  ${}^P\mathbf{I}_S$  bezüglich des Schwerpunktes des Endeffektors.

$$\begin{aligned} & \begin{bmatrix} m_P \mathbf{E} & -m_P {}^P\mathbf{r}_S \mathbf{H} \\ -m_P {}^P\mathbf{r}_S & {}^B\mathbf{I}_P \mathbf{H} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}_P \\ \dot{\Phi} \end{bmatrix} + \dots \\ & \begin{bmatrix} m_P [(\dot{\mathbf{H}}\dot{\Phi}) \times_P \mathbf{r}_S + (\mathbf{H}\dot{\Phi}) \times ((\mathbf{H}\dot{\Phi}) \times_P \mathbf{r}_S)] \\ {}^B\mathbf{I}_P \dot{\mathbf{H}}\dot{\Phi} + (\mathbf{H}\dot{\Phi}) \times ({}^B\mathbf{I}_P \mathbf{H}\dot{\Phi}) \end{bmatrix} - \begin{bmatrix} \mathbf{f}_E \\ \boldsymbol{\tau}_E \end{bmatrix} \\ & = \mathbf{M}_P(\mathbf{x}_P) \ddot{\mathbf{x}}_P + \mathbf{K}(\mathbf{x}_P, \dot{\mathbf{x}}_P) + \mathbf{Q}(\mathbf{x}_P, \dot{\mathbf{x}}_P) = \mathbf{A}^\top \mathbf{f} \end{aligned} \quad (6)$$

Die Vektoren  $\dot{\mathbf{x}}_P$  und  $\ddot{\mathbf{x}}_P$  beschreiben die zeitlichen Ableitungen der Endeffektorpose.  $\mathbf{H}$  und  $\dot{\mathbf{H}}$  ergeben sich aus den kinematischen Kardan-Gleichungen [21].  $\mathbf{E} \in \mathbb{R}^{3 \times 3}$  ist eine Einheitsmatrix. Zur Vereinfachung werden die Endeffektormasse in der Massenmatrix  $\mathbf{M}_P(\mathbf{x}_P)$ , die Zentrifugal- und Korioliskräfte in dem Vektor  $\mathbf{K}(\mathbf{x}_P, \dot{\mathbf{x}}_P)$  und weitere Kräfte und Momente in dem Vektor  $\mathbf{Q}(\mathbf{x}_P, \dot{\mathbf{x}}_P)$  zusammengefasst.

## 2.2 Nichtlineare Modellprädiktive Regelung als Notfallstrategie nach Seilrissen bei Seilrobotern

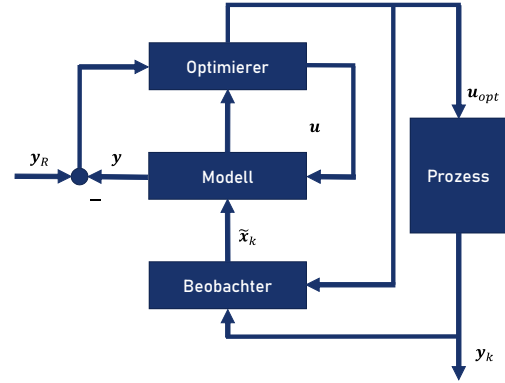
Eine durch Boumann et. al vorgestellte Notfallstrategie nach Seilrissen bei Seilrobotern ist die nichtlineare modellprädiktive Regelung [9]. Der Ansatz verfolgt das Ziel die kinetische Energie des Endeffektors  $E_{kin}$  zu minimieren.

$$E_{kin} = \frac{1}{2} m_P \dot{\mathbf{r}}_P^\top \dot{\mathbf{r}}_P + \frac{1}{2} \Omega^\top \mathbf{I} \Omega, \quad \Omega = \mathbf{H}\dot{\Phi} \quad (7)$$

Die kinetische Energie ist direkt an die Geschwindigkeit des Endeffektors gekoppelt. Eine erfolgreiche Minimierung der kinetischen Energie führt zu einem vollständigen Stop des Endeffektors. Wird der Endeffektor erfolgreich gestoppt, so befindet er sich automatisch im nach Seilriss verbleibenden Arbeitsraum.

Modellprädiktive Regelungen optimieren den Stellgrößenverlauf mit Hilfe eines diskretisierten Modells [11]. Adamy [11] erklärt diese in Analogie zu einem Schachspie-

ler, der versucht, die nächsten Züge seines Gegenspielers vorherzusagen, und entsprechend seine eigenen Züge optimiert. Die NMPC berechnet zu einem Zeitschritt  $k$  aus dem Zustand  $\tilde{\mathbf{x}}_k$  die Ausgangsgrößen  $\mathbf{y}$  über einen Prädiktionshorizont  $n_p$  und optimiert die Stellgrößen  $\mathbf{u}$  über einen Stellhorizont  $n_c$ , um einer Referenzgröße  $\mathbf{y}_R$  zu folgen (siehe Bild 3).



**Bild 3** Nichtlineare modellprädiktive Regelung (NMPC), angelehnt an [9, 11]

Im Folgenden wird der Aufbau der NMPC nach [9] vorgestellt. Das System des Seilroboters wird durch folgendes nichtlineares, diskretes Modell abgebildet:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) \quad (8)$$

$$\mathbf{y}_{k+1} = \mathbf{C}\mathbf{x}_k \quad (9)$$

Der Zustandsvektor  $\mathbf{x}_k$  setzt sich aus den Geschwindigkeit und der Pose des Endeffektors zusammen, welche sich aus der Vorwärtskinematik des Seilroboters ergibt:

$$\mathbf{x}_k = [\dot{\mathbf{r}}_{P,k} \quad \dot{\Phi}_k \quad \mathbf{r}_{P,k} \quad \Phi_k]^\top \quad (10)$$

Die Geschwindigkeit des Endeffektors in Translation und Rotation ist die Ausgangsgröße, da das Ziel der Regelung die Minimierung der kinetischen Energie und der vollständige Stop des Endeffektors ist.

$$\mathbf{y}_{k+1} = \begin{bmatrix} \dot{\mathbf{r}}_{P,k+1} \\ \dot{\Phi}_{k+1} \end{bmatrix} \quad (11)$$

Bei einem Seilriss werden für die mathematische Modellierung die dem Seil zugehörigen Zeilen der Strukturmatrix entfernt. Die Eingangsgröße  $\mathbf{u}_k$  besteht aus der Seilkraftverteilung  $\mathbf{f}_k^*$  der nicht gerissenen Seile.

$$\mathbf{u}_k = \mathbf{f}_k^* \quad (12)$$

Die nichtlineare Funktion  $f(\mathbf{x}_k, \mathbf{u}_k)$  ergibt sich aus der Diskretisierung und numerischen Integration von Gleichung (6). [9] definiert zur Optimierung der Eingangsgrößen die Kostenfunktion  $J$  mit der Gewichtungsmatrix  $\mathbf{G}$  und dem Gewichtungsfaktor  $g$ . Sie summiert über den Prädiktionshorizont die Differenzen der Geschwindigkeiten als Ausgangsgröße  $\mathbf{y}_{k+i}$  und der Referenzgröße  $\mathbf{y}_R = \mathbf{0}$  sowie über den Stellhorizont die Änderung der Seilkräf-

te, um Schwingungsanregungen durch große Kraftstöße zu vermeiden.

$$J = \sum_{i=1}^{n_p} \mathbf{y}_{k+i}^\top \mathbf{G} \mathbf{y}_{k+i} + \dots \quad (13)$$

$$\sum_{i=1}^{n_c} (\mathbf{f}_{k+i-1}^* - \mathbf{f}_{k+i-2}^*)^\top g(\mathbf{f}_{k+i-1}^* - \mathbf{f}_{k+i-2}^*)$$

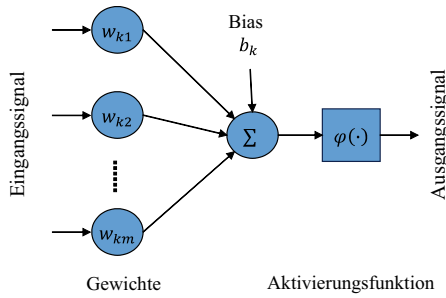
Der Optimierungsalgorithmus der NMPC minimiert für jeden Zeitschritt  $k$  die Kostenfunktion  $J$  durch Anpassung der Seilkräfte  $\mathbf{f}^*$ .

### 3 Künstliches Neuronales Netz

Der von Wang et. al vorgestellte Ansatz zur modellprädiktiven Regelung eines Stromrichters basiert auf einem künstlichen neuronalen Netz (KNN) [16]. Haykin zufolge ist ein großer Vorteil neuronaler Netze die Generalisierung [22]. Dadurch ist das KNN in der Lage, auch bei vorher unbekanntem Eingaben sinnvolle Ergebnisse zu erzeugen. Im folgenden werden die Grundlagen neuronaler Netze erklärt.

#### 3.1 Aufbau eines KNN

Ein KNN besteht aus einer Vielzahl sogenannter Neuronen. Die Skizze eines Neurons ist in **Bild 4** dargestellt.



**Bild 4** Aufbau eines Neurons in einem künstlichen neuronalen Netz, angelehnt an [22]

Die Eingänge  $\mathbf{q}_k$  werden mit Gewichtungparametern  $\mathbf{w}_k$  multipliziert und zusammen mit einem Bias  $b_k$  aufsummiert [22]. Eine Aktivierungsfunktion  $\varphi(\cdot)$  limitiert in der Regel die Ausgänge [22].

$$y_k = \varphi(\mathbf{w}_k^\top \mathbf{q}_k + b_k) \quad (14)$$

Es gibt unterschiedliche Typen der Aktivierungsfunktion, wobei in diesem Beitrag die Sigmoid-Funktion

$$\varphi(x) = \frac{1}{1 + e^{-x}}, \quad (15)$$

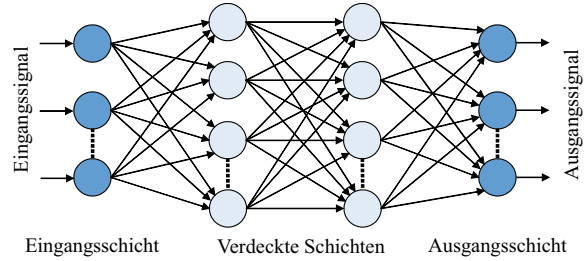
der Tangens-Hyperbolicus

$$\varphi(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (16)$$

und die ReLu-Funktion eingesetzt werden [22]

$$\varphi(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}. \quad (17)$$

Nach [23] besteht ein KNN aus vielen Neuronen, die zu einem Netz mehrerer Schichten verknüpft werden. Eine einfache Struktur eines KNN ist das sogenannte Feed-Forward-Netz, wobei jedes Neuron als Eingang alle Ausgänge der Neuronen der vorherigen Schicht bekommt [23]. Ein beispielhaftes Feed-Forward-Netz ist in **Bild 5** dargestellt.



**Bild 5** Feed-Forward-Netz, angelehnt an [22]

#### 3.2 Training

Bishop [23] beschreibt ein KNN als parametrierbare nicht-lineare Funktionen. Nach dem Autor muss, um die Parameter dieser nichtlinearen Funktion zu ermitteln, der sogenannte mittlere quadratische Fehler (*engl.* Mean-Square-Error, MSE) auf einem Trainingsdatensatz minimiert werden [23]. Ein Trainingsdatensatz besteht aus den Eingangsvektoren  $\mathbf{q}_n$  mit  $n = 1, \dots, N$ , wobei  $N$  die Anzahl an Datensätzen ist, und den zugehörigen Zielvektoren  $\mathbf{t}_n$ . Der MSE ergibt sich nach [23] beim Ausgang eines KNN  $\mathbf{y}(\mathbf{q}_n, \mathbf{W})$  und dem Gewichtungssatz  $\mathbf{W}$  zu:

$$MSE(\mathbf{W}) = \frac{1}{2} \sum_{n=1}^N \|\mathbf{y}(\mathbf{q}_n, \mathbf{W}) - \mathbf{t}_n\|^2. \quad (18)$$

Die Parameter des KNN werden mittels Fehlerrückführung (*engl.* Error-Back-Propagation) mit einer Korrektur  $\nabla \mathbf{W}$  angepasst, welche sich aus den partiellen Ableitungen der Fehlerfunktion über den Gewichtungssatz ergibt [22].

Zur Generierung der Trainingsdaten werden zunächst die Ein- und Ausgangsdaten der NMPC definiert. Neben der Geschwindigkeit des Endeffektors  $\dot{\mathbf{x}}_P$  ist die Strukturmatrix des Seilroboters elementar für die Seilkraftverteilung. Nach Gleichung (3), (4) und (5) ist diese abhängig von der Pose des Endeffektors  $\mathbf{x}_P$ . Zusätzlich zur Geschwindigkeit des Endeffektors minimiert die NMPC in der Kostenfunktion (Gleichung (13)) die Seilkraftänderungen, wozu die zuvor gestellten Seilkräfte  $\mathbf{f}_{\text{alt}}^*$  notwendig sind. Zusammenfassend ergibt sich der Eingangsvektor  $\mathbf{q}_n \in \mathbb{R}^{1 \times 19}$ :

$$\mathbf{q}_n = [\mathbf{x}_P \quad \dot{\mathbf{x}}_P \quad \mathbf{f}_{\text{alt}}^*]^\top \quad (19)$$

Der Ausgang der NMPC ist die zu stellende Seilkraftverteilung  $\mathbf{f}_{\text{neu}}^*$ . Diese sind die Zielvektoren für das Training des KNN.

$$\mathbf{t}_n = \mathbf{f}_{\text{neu}}^* \quad (20)$$

Die Trainingsdaten der NMPC werden mit Hilfe des nach [18] in Abschnitt 2.1 beschriebenen Simulationsmodells erzeugt. Die NMPC ist in diesem Simulationsmodell als Notfallstrategie nach Seilrissen hinterlegt. Durch Entfernen eines Seiles lassen sich Seilrisse in verschiedenen Posen simulieren. Als Robotermodell dient der Seilroboter SE-GESTA der Universität Duisburg-Essen [2].

Der Endeffektor wird an zufällig und gleichmäßig verteilten Positionen im Arbeitsraum des Seilroboters platziert. Durch Entfernen eines Seiles wird ein Seilriss simuliert und die Notfallstrategie ausgelöst. Bei einem erfolgreichen Abfangen des Endeffektors durch die NMPC werden die in Gleichung (19) und (20) definierten Daten gespeichert. Durch mehrfache Wiederholungen der Simulation wurden ca. 9,4 Mio. Datensätze erzeugt.

Die Trainingsdaten werden zu 70 % für das Training des Netzes, zu 15 % zur Validierung und zu 15 % zum Testen verwendet. Vorher werden sie mit einem Mittelwert  $\bar{q} = 0$  und einer Standardabweichung  $\sigma = 1$  standardisiert sowie alle Datenpunkte zufällig angeordnet.

## 4 Ergebnisse und Diskussion

Für den hier behandelten Anwendungsfall wird ein KNN mit mehreren verdeckten Schichten aufgebaut. Den genauen Aufbau liefert eine Hyperparameteroptimierung, deren Einstellparameter in **Tabelle 1** zusammengefasst sind. Die Ergebnisse der Hyperparameteroptimierung sind in **Tabelle 2** dargestellt. Beim Training wird eine stufenförmige Lernratenabnahme angewendet.

Das KNN wurde mit den Parametern aus **Tabelle 2** für 30 Epochen trainiert. Das Training brach automatisch ab, da sich der MSE auf dem Validierungsdatensatz nicht weiter reduzierte. Dadurch wird einer Überanpassung (*engl.* Overfitting) des Netzes auf den Trainingsdaten vorgebeugt. Zusätzlich sind in **Tabelle 2** die Ergebnisse des Trainings anhand des MSEs auf den Trainings-, Validierungs- und Testdaten dargestellt. Außerdem wird hier der MSE der Testdaten in den Kräfte Raum, also in die Seilkräfte am Endeffektor, umgerechnet.

Die Ergebnisse des Trainings zeigen, dass das Netz auf den Trainingsdaten etwas bessere Ergebnisse erzielt. Die vergleichbaren Werte zwischen Validierungs- und Test-MSE zeigen, dass das Netz in der Lage ist, generalisierte Ergebnisse zu erzielen.

Die Umrechnung der Daten zurück in den Kräfte Raum zeigt weiter, dass die durchschnittliche Abweichungen bei einer minimalen und maximalen zu stellenden Seilkraft von 15 N respektive 150 N relativ klein ist.

Im folgenden wird eine Simulation die Leistungsfähigkeit des KNN bewerten.

**Tabelle 1** Parameter der Hyperparameteroptimierung

Parameter	Minimalwert	Maximalwert	Schrittweite
Verdeckte Schichten	1	3	1
Neuronen	32	1024	32
Batchgröße	1024	524288	$2^{x+1}$
Lernrate	$10^{-5}$	$10^{-1}$	$10^{x+1}$
Epochen für Lernratenabnahme	1	16	1
Lernratenabnahme	0,88	1	0,01
Aktivierungsfunktion	ReLU, Tanh, Sigmoid		

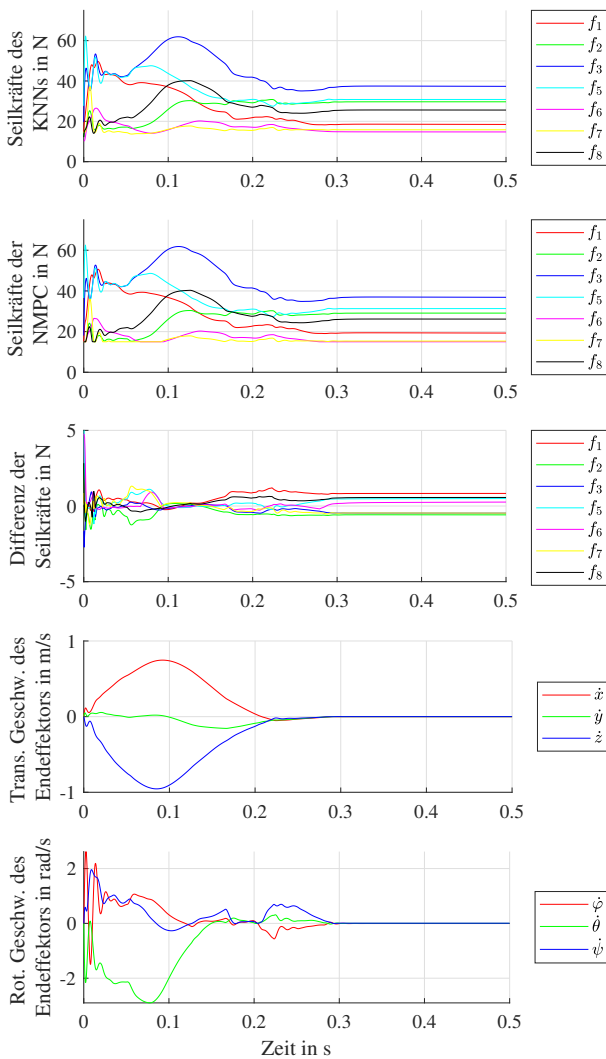
**Tabelle 2** Ergebnis der Hyperparameteroptimierung

Parameter	Wert
Verdeckte Schichten	3
Neuronen 1. Schicht	256
Neuronen 2. Schicht	256
Neuronen 3. Schicht	224
Batchgröße	1024
Lernrate	$10^{-3}$
Epochen für Lernratenabnahme	1
Lernratenabnahme	0,99
Aktivierungsfunktion	Sigmoid
<i>MSE auf den jeweiligen Datensätzen</i>	
Training	4,2096E-4
Validierung	5,8289E-4
Test	5,7992E-4
Test im Kräfte Raum	0,2585 N

### 4.1 Analyse und Simulation in MATLAB/Simulink

Die Simulation des Seilroboters in ©MATLAB/Simulink ermöglicht es, die NMPC durch das KNN als hinterlegte Notfallstrategie zu ersetzen. Somit lassen sich zu dem Testdatensatz weitere neue Daten erzeugen und das KNN in einer simulativen Testumgebung im geschlossenen Regelkreis erproben. In der Simulation werden die theoretischen Seilkräfte durch die NMPC weiter erzeugt und ermöglichen einen Vergleich zum KNN. Die Seilkraftverläufe der NMPC und des KNNs eines erfolgreichen Simulationsdurchlaufs sind in **Bild 6** abgebildet. Zusätzlich ist die Differenz zwischen den Berechnungsmethoden pro Seilkraft und die translatorische bzw. rotatorische Endeffektorgeschwindigkeit über die Zeit abgebildet.





**Bild 6** Ergebnisse der Simulation und Vergleich zwischen NMPC und KNN

Die Differenz der Kraftverläufe zwischen NMPC und KNN zeigt, dass das Netz die NMPC mit relativ kleinem Fehler abbildet. Es fällt auf, dass bei  $0,1 - 0,15$  s der Fehler kleiner ist, während der Endeffektor eine erhöhte Geschwindigkeit hat.

Die Verläufe der translatorischen Geschwindigkeiten zeigen, dass das Netz das Ziel der Regelung in kurzer Zeit erreicht und der Endeffektor in eine stabile Halteposition gelangt.

Zusammenfassend zeigt das KNN in ausgewählten Simulationsläufen das Potential eines realistischen Ansatzes für eine Notfallstrategie nach Seilrissen von parallelen Seilrobotern. Das KNN ist in der Lage, die nichtlinearen Abhängigkeit der NMPC auf Basis von simulierten Trainingsdaten zu erlernen und auf unbekanntem Daten weiterhin sehr gute Ergebnisse zu erzielen. Die Erfolgchancen für das Abfangen des Endeffektors in beliebigen Seilrissituationen müssen durch weitere Forschungsarbeiten erhöht werden.

## 5 Zusammenfassung und Ausblick

Im Rahmen dieses Beitrags wurde eine nichtlineare modellprädiktive Regelung als Notfallstrategie nach Seilrissen bei Seilrobotern durch ein künstliches neuronales Netz ersetzt. Die Generierung von Trainingsdaten erfolgte durch die Simulation von Seilrissen mit der hinterlegten NMPC als Notfallstrategie in ©MATLAB/Simulink. Bei dem vorgestellten KNN handelt es sich um ein Feed-Forward-Netz mit mehreren verdeckten Schichten. Die Hyperparameter, die das Netz und das Training beschreiben, wurden durch eine Hyperparameteroptimierung ermittelt.

Das KNN produziert auf den Trainings- und Validierungsdaten einen niedrigen MSE und ist in der Lage, auf dem Netz unbekanntem Testdaten, die Leistungsfähigkeit zu bestätigen. Durch weitere Seilrissimulationen mit dem KNN als Notfallstrategie im geschlossenen Regelkreises wurde die gute Generalisierbarkeit des Netzes nachgewiesen. In der Simulation ist das KNN in einigen Fällen in der Lage, den Endeffektor nach einem Seilriss abzufangen und in kurzer Zeit in eine sichere Halteposition zu bringen.

Der präsentierte Ansatz zeigt vielversprechende Ergebnisse, die Forschungsmöglichkeiten für zukünftige Untersuchungen eröffnen. Ein tiefergehender Vergleich des KNNs mit der NMPC wird zeigen, ob das Netz in der Lage ist, echtzeitfähige Ergebnisse zu erzielen. Eine Erweiterung des Prädiktions- und Stellhorizontes der NMPC bei der offline Generierung weiterer Trainingsdaten könnte die Zuverlässigkeit des Netzes erhöhen. Weitere Ansätze, wie die Verwendung von rekurrenten Netzen, haben das Potential die Nichtlinearität der NMPC besser abzubilden [24].

Der Beitrag bildet die Grundlage für weitere Arbeiten zur Erforschung von Methoden des maschinellen Lernens als Notfallstrategie nach Seilrissen bei parallelen Seilrobotern.

## 6 Danksagung

Die Arbeit an dieser Veröffentlichung ist gefördert durch die Deutsche Forschungsgemeinschaft (DFG) - Projektnummer 506498608.

## 7 Literatur

- [1] A. Pott, *Cable-Driven Parallel Robots: Theory and Application*, Bd. 120. in Springer Tracts in Advanced Robotics, vol. 120. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-76138-1
- [2] SEGESTA - *Seilgetriebene Stewart-Plattformen in Theorie und Anwendung*. [Online]. Available: [https://www.uni-due.de/mechatronik/forschung/SEGESTA\\_ARTIST.php](https://www.uni-due.de/mechatronik/forschung/SEGESTA_ARTIST.php)
- [3] Roske, T., Lipowsky, J., Palzer, U., Eden, W., Schäfers, M., Lemmen, P., Heidel, R., Boumann, R., Spengler, A. und Bruckmann, T.: (2021), *Automatisierter Bau von Kalksandstein-Mauerwerk mit Seilrobotern*. Mauerwerk, 25: 82-89. <https://doi.org/10.1002/dama.202110011>

- [4] G. Brown: *SkyCam*, garrettcam.de. Zugegriffen: 16. November 2023. [Online]. Verfügbar unter: <https://www.garrettcam.com/skycam>
- [5] J. Marshall, *No one hurt after ESPN SkyCam falls during Insight Bowl*, The San Diego Union-Tribune, San Diego, 30. Dezember 2011. Zugegriffen: 2. November 2023. [Online]. Verfügbar unter: <https://www.sandiegouniontribune.com/sdut-no-one-hurt-after-camera-falls-at-insight-bowl-2011dec30-story.html>
- [6] A. Witze, *Gut-wrenching footage documents Aricibo telescope's collapse*, Nature, Dez. 2020, Zugegriffen: 2. November 2023. [Online] doi: <https://doi.org/10.1038/d41586-020-03421-y>
- [7] G. Boschetti, R. Minto, und A. Trevisani: *Experimental Investigation of a Cable Robot Recovery Strategy*, Robotics, Bd. 10, Nr. 1, S. 35, Feb. 2021, doi: 10.3390/robotics10010035 .
- [8] G. Boschetti, C. Passarini, und A. Trevisani: *A Recovery Approach for Spatial Cable Robots After Cable Failure*, in Advances in Italian Mechanism Science: Proceedings of the Second International Conference of IFToMM Italy, Bd. 68, G. Carbone und A. Gasparetto, Hrsg., in Mechanisms and Machine Science, vol. 68. , Cham: Springer International Publishing, 2019, S. 457–464. doi: 10.1007/978-3-030-03320-0.
- [9] R. Boumann und T. Bruckmann: *Development of Emergency Strategies for Cable-Driven Parallel Robots after a Cable Break*, in Cable-Driven Parallel Robots: Proceedings of the 4th International Conference on Cable-Driven Parallel Robots, Bd. 74, A. Pott und T. Bruckmann, Hrsg., in Mechanisms and Machine Science, vol. 74. , Cham: Springer International Publishing, 2019, S. 269-280. doi: 10.1007/978-3-030-20751-9
- [10] R. Boumann, C. Jeziorek, und T. Bruckmann: *Validation of Emergency Strategies for Cable-Driven Parallel Robots After a Cable Failure*, in Cable-Driven Parallel Robots: Proceedings of the 6th International Conference on Cable-Driven Parallel Robots, Bd. 132, S. Caro, A. Pott, und T. Bruckmann, Hrsg., in Mechanisms and Machine Science, vol. 132. , Cham: Springer Nature Switzerland, 2023, S. 209-220. doi: 10.1007/978-3-031-32322-5
- [11] J. Adamy: *Nichtlineare Systeme und Regelungen*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018. doi: 10.1007/978-3-662-55685-6 .
- [12] P. M. Sieberg, S. Reicherts, und D. Schramm: *Nichtlineare modellbasierte prädiktive Regelung zur aktiven Wankstabilisierung von Personenkraftwagen*, 2018, doi: 10.17185/DUEPUBLICO/45323 .
- [13] P. M. Sieberg, C. Hürten, und D. Schramm: *Representation of an Integrated Non-Linear Model-Based Predictive Vehicle Dynamics Control System by a Co-Active Neuro-Fuzzy Inference System*, in 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA: IEEE, Okt. 2020, S. 572–577. doi: 10.1109/IV47402.2020.9304585 .
- [14] J.-S. R. Jang: *ANFIS: adaptive-network-based fuzzy inference system*, IEEE Trans. Syst., Man, Cybern., Bd. 23, Nr. 3, S. 665–685, Juni 1993, doi: 10.1109/21.256541 .
- [15] E. Mizutani und J.-S. R. Jang: *Coactive neural fuzzy modeling*, in Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia: IEEE, 1995, S. 760–765. doi: 10.1109/ICNN.1995.487513 .
- [16] D. Wang et al.: *Model Predictive Control Using Artificial Neural Network for Power Converters*, IEEE Trans. Ind. Electron., vol. 69, no. 4, pp. 3689–3699, 2022, doi: 10.1109/TIE.2021.3076721.
- [17] R. Verhoeven: *Analysis of the Workspace of Tendon-based Stewart Platforms*, Dissertation, Fakultät für Ingenieurwissenschaften, Universität Duisburg-Essen, Duisburg, 2004.
- [18] R. Boumann and T. Bruckmann, *Simulation and Model-Based Verification of an Emergency Strategy for Cable Failure in Cable Robots*, Actuators, vol. 11, no. 2, p. 56, 2022, doi: 10.3390/act11020056.
- [19] T. Bruckmann, *Auslegung und Betrieb redundanter paralleler Seilroboter*, Dissertation, Lehrstuhl für Mechatronik, Universität Duisburg-Essen, Duisburg, 2010. [Online]. Available: <https://nbn-resolving.org/urn:nbn:de:hbz:464-20101020-073248-8>
- [20] D. Schramm, M. Hiller, and R. Bardini, *Modellbildung und Simulation der Dynamik von Kraftfahrzeugen*, 3rd ed. Berlin, Heidelberg: Springer Vieweg, 2018. [Online]. Available: <http://www.springer.com/de/book/9783662544808>
- [21] M. Hiller, *Mechanische Systeme: Eine Einführung in die analytische Mechanik und Systemdynamik*. Berlin, Heidelberg: Springer, 1983.
- [22] S. S. Haykin, *Neural networks and learning machines*, 3rd ed. New York: Prentice Hall, 2009.
- [23] C. M. Bishop, *Pattern recognition and machine learning*. New York: Springer, 2006.
- [24] M. V. d. Hoop, M. Lassas, and C. A. Wong, *Deep learning architectures for nonlinear operator functions and nonlinear inverse problems*, Dec. 2019. [Online]. Available: <http://arxiv.org/pdf/1912.11090v3>

# DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT  
DUISBURG  
ESSEN

*Offen im Denken*

ub

universitäts  
bibliothek

In: Zehnte IFToMM D-A-CH Konferenz 2024

Dieser Text wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt. Die hier veröffentlichte Version der E-Publikation kann von einer eventuell ebenfalls veröffentlichten Verlagsversion abweichen.

**DOI:** 10.17185/duepublico/81705

**URN:** urn:nbn:de:hbz:465-20240304-164608-7



Dieses Werk kann unter einer Creative Commons Namensnennung - Nicht kommerziell - Keine Bearbeitungen 4.0 Lizenz (CC BY-NC-ND 4.0) genutzt werden.