

BECKHOFF

TITELSTORY

Maschinelles Lernen
automatisierungsgerecht umgesetzt

OFFENER SOFTWARE-DIENST

Funktionsbausteine für IO-Link
automatisiert erstellen

MANAGED GBIT-SWITCHES

Von der Stange
und nach Maß

Sonderausgabe zur **SPS smart production solutions**

A8100

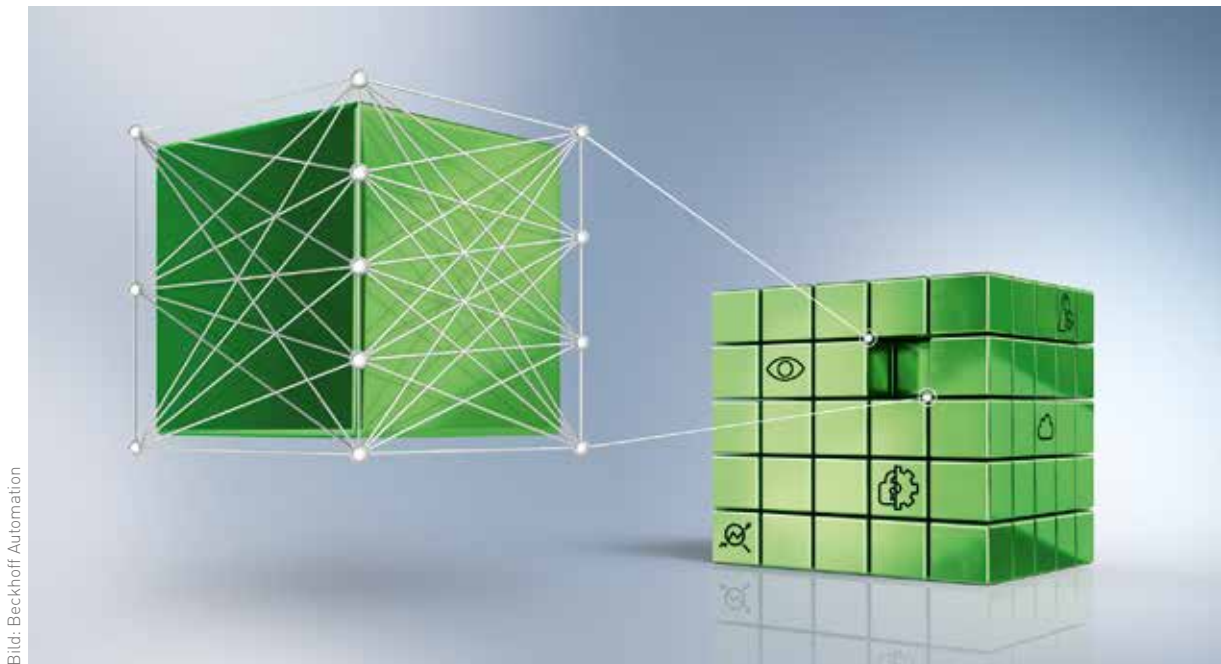


Bild: Beckhoff Automation

01 Mit dem direkt in die Steuerungstechnik integrierten TwinCAT 3 Machine Learning verschmelzen die Welten von Data Scientists und Maschinenbauern und es eröffnen sich durch Synergieeffekte völlig neue Möglichkeiten der Maschinenoptimierung

Maschinelles Lernen automatisierungsgerecht umgesetzt

Das maschinelle Lernen (ML) gilt als vielversprechendste Technologie, um mit moderner Automatisierungstechnik zukünftig noch weitere Optimierungspotenziale erschließen zu können. Voraussetzung dafür ist eine nahtlos in die Steuerungstechnik integrierte Funktionalität, wie sie nun mit TwinCAT 3 Machine Learning zur Verfügung steht.

Text: Fabian Bause, Benjamin Jurke, Klaus Neumann

Auch wenn die praktische Anwendung des maschinellen Lernens im industriellen Umfeld noch am Anfang steht, ein unmittelbarer Nutzen ist bereits heute klar: Bereiche, wie Robotik, mathematische Optimierung, Anomaliedetektion oder modellprädiktive Regelung, würden direkt durch eine entsprechende Integration profitieren. Das haben zahlreiche

Anwendungsstudien der letzten Jahre gezeigt. Mit den Produkten TwinCAT 3 Neural Networks Inference Engine und TwinCAT 3 Machine Learning Inference Engine bietet Beckhoff den Automatisierern und Maschinenbauern nun die Möglichkeit, die Inferenz – d. h. die Ausführung eines trainierten ML-Modells – industriegerecht, nahtlos und in Echtzeit zu integrieren. [1]

SPS smart production solutions

➔ Beckhoff Automation: Halle 7, Stand 406

Der Workflow beginnt mit der Datenaufnahme

Ein wesentlicher Bestandteil, um Trainingsprozesse und die Inferenz der gelernten Modelle mithilfe von geeigneten Da-

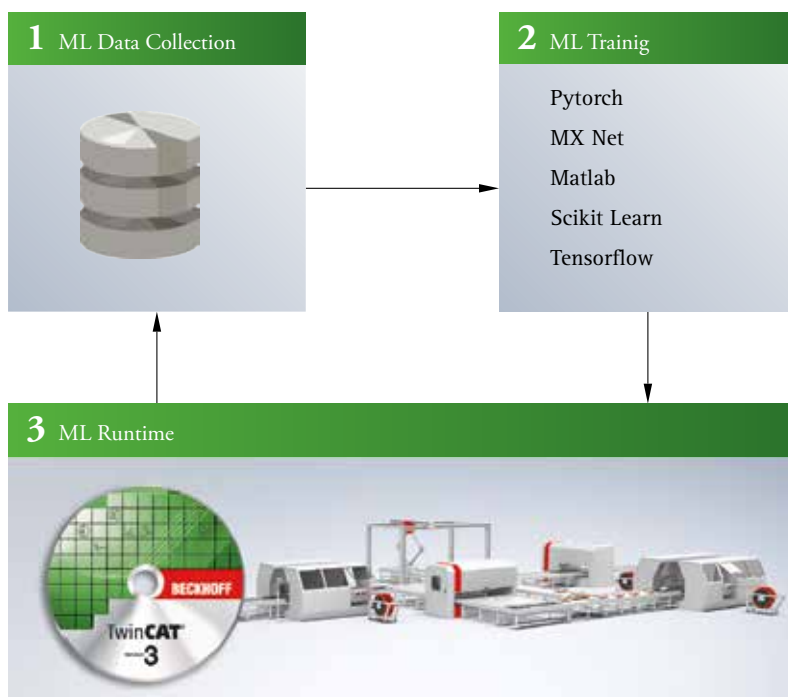
ten in Gang zu setzen, ist eine effektive Dateninfrastruktur. Viele Anlagen zeichnen schon heute eine große Datenmenge auf, die für das maschinelle Lernen nutzbar ist. Die nahtlose Integration von Machine Learning in TwinCAT 3 ermöglicht es, diverse Frameworks, wie Tensorflow, Pytorch und Matlab, direkt anzuwenden. Auf diese Weise verschmelzen die Welten von sogenannten Data Scientists und Maschinenbauern zunehmend und es ergeben sich immense Synergieeffekte (Bild 1).

Der Kreislauf des Workflows beginnt mit der Aufnahme von Daten aus einem (teil-)automatisierten Prozess (Bild 2). Welche Daten aufzunehmen und auszuwerten sind, hängt natürlich von der jeweiligen Applikation ab. Zielführend ist in der Regel, mit einem spezifischen und abgegrenzten Anwendungsfall zu beginnen und dann die dafür notwendigen Daten zu definieren. Hier ist das Domänenwissen, d. h. das Know-how des Maschinenbauers, von zentraler Bedeutung.

Diese Domänenexperten kennen empirische und physikalische Zusammenhänge zwischen definierten Zielgrößen und Prozessdaten, sie geben abstrakten Variablen eine Semantik. In Zusammenarbeit mit ihnen ist daher zu erarbeiten, ob die gesuchten oder davon ableitbare Zielgrößen zeitlich korrelierbar mit weiteren Informationen gewonnen werden können. Dabei kann es sich um Maschinendaten aus zusätzlicher Sensorik, aus anderen Systemen wie dem ERP-System oder um händisch durch Mitarbeiter in textueller Form oder per Maschinen-HMI eingegebene Daten handeln. Die Verfügbarkeit und Qualität der Zielgrößen definiert den Unterschied zwischen überwachtem und unüberwachtem Lernen (Supervised bzw. Unsupervised Learning).

Neben der Art der Daten und dem Auswertungsverfahren sind die einzusetzenden Werkzeuge von zentraler Bedeutung für den Workflow. Aus den Bereichen TwinCAT 3 Measurement und TwinCAT 3 Connectivity bieten sich folgende Produkte an:

- In der Engineering-Phase ist TwinCAT 3 Scope – mit seiner Speicherfunktion u. a. als Binärdatei, CSV und TDMS – ein einfaches und flexibles Tool zur Datenaufnahme und -speicherung.
- Ist eine SQL- oder no-SQL-Datenbank (lokal, im Netzwerk oder in der Cloud) verfügbar, ist der TwinCAT 3 Database Server ein ideales Werkzeug. Mit ihm lassen sich auch große Datenmengen während der Maschinenlaufzeit aggregieren.
- Ist die Maschine mit einer Public oder Private Cloud über einen Message Broker verbunden, bietet sich die Produktfamilie TwinCAT 3 IoT an, um die Daten über den Broker in einem Datastore oder Warehouse zu speichern.



02 Workflow von der Datenaufnahme über das Training bis hin zur Integration des trainierten ML-Modells in die TwinCAT 3 Runtime (XAR)

- Im laufenden Betrieb eignet sich der TwinCAT 3 Analytics Logger, um komfortabel und zyklussynchron große Datenmengen aus der Maschinensteuerung in einer lokalen Datei oder über einen Message Broker in einem Data Lake oder einer Datenbank abzulegen.
- Entscheidende Fragen zum Einsatz eines bestimmten Werkzeugs können beispielsweise sein:
- Sollen die Daten über die Maschinenlaufzeit hinweg durchgängig gespeichert werden?
 - Darf der Quellcode der Maschine angepasst werden?
 - Wie hoch sind die Abstrakte und die Anzahl kommunizierter Daten?
 - Welche IT-Restriktionen bzw. Security-Mechanismen müssen eingehalten werden?

Training und Nutzung eines ML-Modells

Sind Trainingsdaten verfügbar, können diese in ein externes ML-Framework eingelesen und verarbeitet werden. Aus Sicht des Anwenders bietet sich hier eine Fülle von passenden ML-Frameworks an. Viele davon basieren auf Python und sind Open-Source-Software, andere sind kommerziell verfügbar wie Matlab und SAS. Als Schnittstelle zu den Daten dienen zahlreiche Standardfunktionen und Toolboxes bzw. Bibliotheken, um auf die abgelegten Daten zuzugreifen.

Die Arbeit mit dem ML-Framework beinhaltet die Datenaufbereitung und -selektion, Auswahl und Training eines geeigneten ML-Modells sowie dessen Evaluation. Das Ergebnis dieses Engineering-Schritts ist ein trainiertes ML-Modell, welches den Zusammenhang zwischen Eingangs- und Zieldaten repräsentiert. Ebenso offen wie die Schnittstelle zu den Daten ist die Schnittstelle für das Deployment des trainierten ML-Modells in TwinCAT 3 ge-



Bild: Beckhoff Automation

03 Der auf der Hannover Messe 2019 gezeigte ML-XTS-Demonstrator zeigte, wie man eine Anlage mithilfe von TwinCAT 3 Machine Learning und TwinCAT 3 Motion Control energie- und verschleißoptimiert ansteuern kann

wählt. Hier wird das Open Neural Network Exchange (ONNX)-Dateiformat als Industriestandard unterstützt.

Das Deployment des ML-Modells in der TwinCAT 3 Runtime ist der finale, zweiteilige Schritt:

- Das exportierte ML-Modell wird in ein geschütztes Binärformat (.bml) konvertiert.
- Die .bml-Datei wird dann auf dem Zielsystem (Embedded-PC oder Schaltschrank-PC) abgelegt.

Die Konvertierung in das optional geschützte Dateiformat .bml dient dem Schutz geistigen Eigentums. Denn in dieser Form ist nicht nachvollziehbar, wie das trainierte Modell aufgebaut ist. Auch für externe Dienstleister ist dieser Austauschweg mit dem Kunden ein Weg, um Geschäftsgeheimnisse zu wahren.

Zum Laden und Ausführen des trainierten ML-Modells dient ein neuer Baustein für TwinCAT 3. Es handelt sich hierbei um ein Standard-„TcCOM“-Modul, das ein generisches Inferenzmodul darstellt. Darüber stehen folgende neue Funktionen zur Verfügung:

- TwinCAT 3 Machine Learning Inference Engine: für klassische ML-Algorithmen wie Support Vector Machine und Principal Component Analysis,
- TwinCAT 3 Neural Network Inference Engine: für neuronale Netze wie Multilayer Perceptrons und Convolutional Neural Networks.

Anlagenoptimierung durch maschinelles Lernen

Ein besonders interessanter Aspekt ist die Nutzung mathematischer Optimierungsverfahren aus dem Bereich der Optimalsteuerung (Optimal Control). Durch die geeignete Wahl eines Kriteriums und der Nebenbedingungen können viele unterschiedliche Problemstellungen auf eine allge-

meine Weise optimal gelöst werden. Diese Methoden haben sich u. a. in der Robotik bereits bewährt. Eine direkte Anwendung in einer zyklusbasierten Steuerung wie der SPS kommt allerdings nicht infrage. Diese Verfahren liefern zwar sehr gute Ergebnisse, leiden aber an einer nichtdeterministischen Laufzeitkomplexität und potenziell divergentem Verhalten. Zudem sind die verwendbaren Algorithmen sequenziell und lassen sich, aufgrund der prinzipbedingt eingeschränkten Parallelisierbarkeit, nicht beliebig durch bessere Hardware beschleunigen. Sie sind schlicht nicht performant genug für das Erreichen adäquater Zykluszeiten einer Echtzeitsteuerung.

Maschinelles Lernen – insbesondere mit neuronalen Netzen – kann hier Abhilfe schaffen. Denn neuronale Netze sind strukturell bedingt von deterministischer Laufzeitkomplexität und können prinzipiell jeden Algorithmus abbilden. Eine entsprechende Imitation entsteht durch den vorab stattfindenden Trainingsprozess. Ein langsamer Optimierungsprozess kann somit als „Lehrer“ für ein schnelles neuronales Netz, den „Schüler“, eingesetzt werden.

Dieses neuronale Netz ist dann bei einer geeigneten Einbettung in die Softwareinfrastruktur in einer Echtzeitanwendung einsetzbar. Der Softwareinfrastruktur kommt hierbei eine besondere Rolle zu, da neuronale Netze aufgrund ihrer mathematischen Struktur inhärente Ungenauigkeiten aufweisen. Diese müssen für den Einsatz in der Automatisierung vorab behoben werden. Zudem ist wichtig, dass die neuronalen Netze ausreichend und fehlerfrei trainiert wurden.

Live-Demonstrationen zu den Praxisvorteilen

Auf der Hannover Messe 2019 demonstrierte Beckhoff bereits konkret, wie neuronale Netze für die Steuerung in Echtzeit eingesetzt werden können – mit der TwinCAT 3 Neural Networks Inference Engine und deren Integration in TwinCAT 3 Motion Control. Den Kern des entsprechenden Demonstrators bildeten zwei XTS-Systeme und ein dazwischen platziertes Fließband. Die jeweils zehn XTS-Mover wurden nach und nach auf Markierungen des Fließbands synchronisiert und für eine Strecke von einem Meter parallel zur jeweiligen Markierung gefahren. Anschließend koppelten sie aus und bewegten sich für die nächste Synchronfahrt wieder an den Anfangspunkt des Fließbands (**Bild 3**).

Ein XTS wurde mit einem klassischen Verfahren gesteuert, in diesem Fall das im Motion-Control-Bereich häufig verwendete zeitoptimale Sieben-Phasen-Profil. Insbesondere während des Auskoppelns und kurz vor dem erneuten Auf-synchronisieren zeigte sich hierbei allerdings ein hoher Verschleiß- und Energieverbrauchswert, da in diesen Bereichen stark beschleunigt bzw. abgebremst wird. Das zweite, mithilfe eines speziell für diese Anwendung trainierten neuronalen Netzes gesteuerte XTS arbeitete hingegen deutlich effizienter. Umgesetzt war dies über spezifische Funktionsbausteine für die Steuerung, mit denen sich das trainierte Netz sehr einfach aus der PLC oder aus C++ heraus nutzen lässt. Inzwischen lassen sich auch einfach Inputs und Outputs im TwinCAT System Manager während der Engineering-Phase verbinden. Das trainierte Modell funktioniert also wie ein mathematischer Prozessor. Im gezeigten XTS-Anwen-



04 Die ML-Anwendungen als virtueller Sensor und zur Anomaliedetektion (Bild) wurden auf der Hannover Messe 2019 ebenfalls demonstriert

dungsfall wurde das neuronale Netz hierfür durch einen Optimierungsalgorithmus trainiert, der auf einem einfachen Energie- und Verschleißmodell basiert.

Die Twincat 3 Neural Network Inference Engine ermöglichte die Echtzeit-Verwendung in der Bewegungssteuerung (NC, Motion Control). Aufgrund der besonderen Anforderungen im Motion-Control-Bereich muss das erzeugte Bewegungsprofil bis zur Beschleunigung an das vorangegangene Profil stetig anschließen und darf etwaige Maximalwerte für Geschwindigkeit und Beschleunigung nicht überschreiten. Dies wird durch Twincat 3 berücksichtigt und ohne Aufwand für den Anwender sichergestellt. Es zeigte sich, dass das vom neuronalen Netz gesteuerte XTS einen deutlich geringeren Gesamtenergieverbrauch und weniger Verschleißerscheinungen aufweist als das klassisch gesteuerte System. Denn das neuronale Netz erzeugt „weichere“ Profile, verteilt also die für die Synchronisation notwendige Dynamik über einen größeren Bereich des XTS.

Dieses Beispiel zeigt, dass eine durch Software realisierte Effizienzsteigerung möglich ist, die im Wesentlichen durch mathematisch objektive Optimierungsverfahren zustande kommt. Solche Verfahren sind sehr allgemein und lassen sich daher zielgerichtet auch für andere Optimierungskriterien einsetzen. Auf diese Weise können Anlagen weitaus zielorientierter entwickelt und deren Effizienz zusätzlich gesteigert werden.

Datenbasierte Modellierung bei virtueller Sensorik und Anomaliedetektion

Eine Grundvoraussetzung für viele Optimierungsansätze in der Automatisierung ist ein geeignetes Modell des zu optimierenden Systems. Leider sind solche Modelle nicht immer vorhanden oder teilweise zu unpräzise. Zudem ist das manuelle Erstellen eines solchen Modells aufwendig und damit kostenintensiv. Eine Alternative ist das rein datenbasierte Erstellen eines Modells. Hierzu werden Systemdaten aufgenommen, die durch maschinelle Lernverfahren zum entsprechenden Modell führen.

Ein Beispiel ist die virtuelle Sensorik, bei der ein Modell mit geeigneten Sensoren als Eingang eine andere bestehende

Sensorausgabe vorhersagt. Hierzu werden Daten der Eingangssensoren aufgezeichnet und für das spätere Training vorbereitet. Beim Training definiert man den zu ersetzenden Sensor als sogenanntes Target und die anderen Sensoren als Input. Ist eine Vorhersage der Sensorausgabe nach dem Training und einer Validierung präzise möglich, lassen sich der entsprechende physikalische Sensor und somit auch die entsprechenden Kosten einsparen.

Auch zu diesem Beispiel zeigte Beckhoff auf der Hannover Messe 2019 eine konkrete Umsetzung: ein Lüfter mit Drucksensor sowie Vibrations- und Drehgeschwin-

digkeitserfassung. Damit konnte verdeutlicht werden, dass sich der erzeugte Luftdruck durch die mithilfe von Twincat 3 Scope ausgelesenen Werte für Vibration und Drehgeschwindigkeit präzise vorhersagen lässt. Die ebenfalls präsentierte Anomaliedetektion bestand aus zwei baugleichen Lüftern, die jeweils über einen Druck-, Vibrations- und Geschwindigkeitssensor verfügten. Einer der Lüfter war vollkommen funktionstüchtig; der andere wies eine absichtliche Beschädigung an einem Lüfterflügel auf. Über die mit dem Twincat 3 Database Server aufgezeichneten Daten und mithilfe der Twincat 3 Machine Learning Inference Engine konnte – in Echtzeit direkt in der PLC oder aus C++ heraus und ohne einen aufwendigen Modellierungsprozess – eine Vorhersage über den Zustand des jeweiligen Lüfters getroffen werden (Bild 4). (hz)

Literatur

[1] Machine Learning von Beckhoff Automation:
www.beckhoff.de/machine-learning

Autoren



Dr. Fabian Bause ist Produktmanager Twincat bei Beckhoff Automation in Verl.



Dr. Benjamin Jurke ist Mitarbeiter für R&D Machine Learning bei Beckhoff Automation in Verl.



Dr. Klaus Neumann ist Software-Entwickler Maschinelles Lernen bei Beckhoff Automation in Verl.

info@beckhoff.de