

Bild: ArtMinds Robotics GmbH

Roboteranwendungen umsetzen In drei Schritten programmiert

Die drei meist genutzten Verfahren, um Industrieroboter zu programmieren, sind die Teach-In-Programmierung, die grafische CAD-gestützte Offline-Programmierung und die klassische textuelle Programmierung. Welche Vor- und Nachteile diese Methoden haben und wie es gelingen kann, mit neuen Programmierverfahren den Anlagenanlauf zu beschleunigen, zeigt der folgende Beitrag.

Beim Teach-In-Verfahren erfolgt die Programmierung mit dem Teach-Pendant direkt am Roboter (Online-Programmierung). Sie zählt daher zu den einfachsten Methoden. Mit Hilfe von Knöpfen und Joysticks kann ein Bediener den Roboter in verschiedene Gelenkstellungen oder Stellungen seines Endeffektors bewegen. Die Positionen lassen sich auf diese Weise nacheinander aufzeichnen und die Punkte zu einer kontinuierlichen Bewegungsbahn zusammenfügen – ein einfacher Ansatz, der es jedoch keine spezifischen Möglichkeiten zulässt. Des Weiteren ist die Integration komplexer Programmlogik zur flexiblen Reaktion auf Sensorsignale von Kraft-Momenten-Sensoren oder Kameras nicht möglich. Daher ist die reine Teach-Pendant basierte Programmie-

rung auf einfache Industrieroboteranwendungen beschränkt. Grafische Offlineprogrammierung geht einen Schritt weiter, indem für die Berechnung optimaler Bewegungspfade die Geometrie der Roboteranlage sowie der Werkstücke genutzt wird. Die Handhabung der dafür benötigten Software ist komplizierter als die Nutzung eines Teach-Pendants. Anwender benötigen CAD/CAE-Erfahrung, um die Benutzerschnittstellen und den Programmierablauf optimal nutzen zu können. Die CAD-Modelle der Anlage und Werkstücke werden in diese Softwaresysteme importiert, so dass Roboter fernab der Anlage an einem Büroarbeitsplatz programmiert werden können. In der dreidimensionalen Simulationsumgebung lassen sich Wegpunkte schneller und fle-

xibler als beim Teach-In-Verfahren am realen Roboter definieren bzw. ändern. Der Entwurf möglichst optimaler Bewegungsprofile wird durch die Software unterstützt. Daher ist die automatische Berechnung kollisionsfreier Teilbewegungen in solchen Umgebungen recht einfach möglich. Die Herausforderung liegt jedoch meist in der Übertragung des Programms auf die reale Roboterumgebung, da aufgrund von Kalibrierungsungenauigkeiten oder Abweichungen zwischen Modell und Realität weitere Anpassungen nötig werden.

Einfacher programmieren

Komplexe Logik, die auf Sensorsignale reagiert und zu flexiblerem, robusterem Roboterverhalten führt, war lange Zeit

nicht mit rein grafischer Offlineprogrammierung umsetzbar. Sensor-basierte Roboteranwendungen mussten daher bis jetzt explizit textuell in der Programmiersprache des jeweiligen Roboterherstellers programmiert werden. Dieser Ansatz erfordert Erfahrung sowohl in textueller Programmierung als auch in sensor-basierter Robotik, das Personal muss also entsprechend qualifiziert sein. ArtiMinds Robotics adressiert die Schwächen der drei Verfahren. Mit der Robot Programming Suite (RPS) ist es möglich, eine direkte Programmierung am physischen Roboter mit der grafischer Offline-Programmierung zu mischen und so den Anlagen-

anlauf zu beschleunigen. Dabei können flexible, sensor-basierte Anwendungen mit Kraft-Momenten-Sensorik oder Vision Systemen realisiert werden, ohne textuell programmieren zu müssen. Der Programmierablauf besteht dabei aus drei Schritten:

- Mittels einer grafischen Benutzeroberfläche werden einzelne Abschnitte des gesamten Bewegungsablaufes per Drag&Drop von Bewegungsabläufen und umfangreich getesteten Teilprozessen, so genannten Templates, erstellt und zusammengefügt.
- Teach-in einiger weniger Schlüsselstellungen des Roboters für den Ab-

schnitt. Dies kann mit dem Teach-Pendant, durch direktes Führen von kollaborativen Robotern an ihrem Endeffektor oder am virtuellen Roboter in der Simulationsumgebung der Software erfolgen. Für eine Roboter-aufgabe kann beliebig und nahtlos zwischen den unterschiedlichen Teach-Verfahren gewechselt werden.

- Nach Berechnung der nominalen Robotertrajektorie, überlagerten Kraftreglern und Simulation der programmierten Roboterbewegung wird schließlich ein Programm in der nativen Programmiersprache des Zielrobotersystems erzeugt. Der generierte Code läuft eigenständig auf dem unmodifizierten Robotercontroller, sodass die via Netzwerk verbundene Roboterprogrammiersoftware am Ende problemlos abgetrennt werden kann.

Die Programmiersoftware integriert verschiedene Industrieroboterfamilien, Kraft-Momenten-Sensoren, Greifer und Bildverarbeitungssysteme. Tiefgreifende Kenntnisse der Hardwaresysteme sowie Erfahrung in textueller Programmierung werden nicht benötigt. Der Fokus der Roboterprogrammierung liegt somit ausschließlich auf der Entwicklung des geeigneten, flexiblen Bewegungsablaufs für die Automatisierungsaufgabe. ■

Der Autor Dr.-Ing. Sven R. Schmidt-Rohr ist CEO bei der ArtiMinds Robotics GmbH.

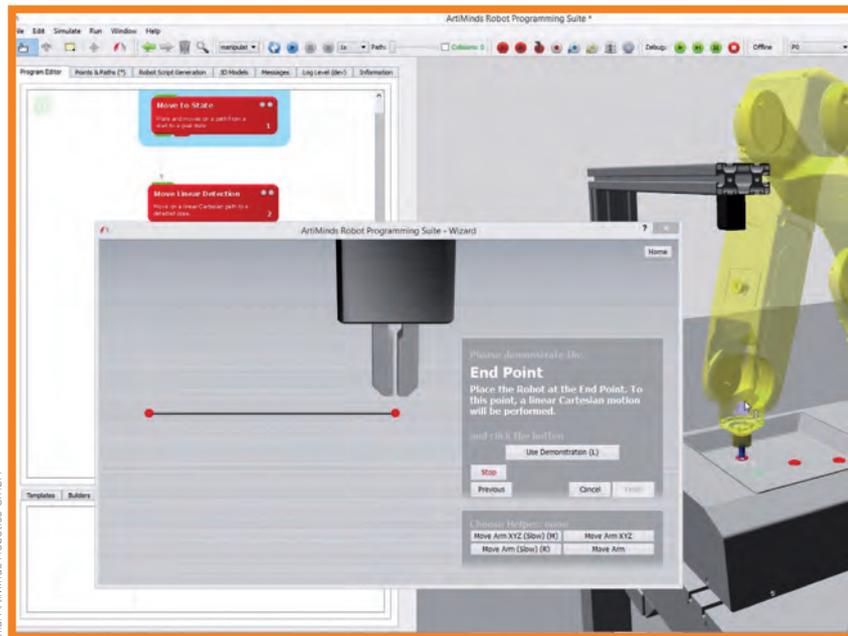


Bild: ArtiMinds Robotics GmbH

Intuitive Programmierung via Drag and Drop von Templates und Wizard-gestützte Parametrisierung

www.artiminds.com

INSERENTENVERZEICHNIS

abilis GmbH	Seite 54
ACATEC Software GmbH	Seite 43
AICOMP Consulting GmbH	Seite 55
Bachmann electronic GmbH	Seite 9
camos Software und Beratung GmbH	Seite 55
Datafox GmbH	Seite 61
EAS Engineering Automation Systems GmbH	Seite 2
ECS GmbH	Seite 35
FIS Informationssysteme und Consulting GmbH	Seite 75
GFOS mbH	Seite 59
Interflex Datensysteme GmbH & Co. KG	Seite 63
ISGUS GmbH	Seite 57

Kontron S&T AG	Seite 29
MAPAL Fabrik für Präzisionswerkzeuge Dr. Kress KG	Seite 13
Mensch und Maschine Deutschland GmbH	Seite 39
MESAGO Messe Frankfurt GmbH	Seiten 5, 23
Modus Consult GmbH	Seite 27
onoff AG	Seite 31
ORCON GmbH	Seite 37
ORISA Software GmbH	Seite 53
SEAL Systems AG	Seite 11
Shuttle Computer Handels GmbH	Seite 17
simus systems GmbH	Seite 3