

Online- und Offline-Programmierung kombinieren

# Roboterprogrammierung neu gedacht

*Komplexe Roboteranwendungen durchlaufen in ihrem Lebenszyklus meist drei Phasen: Planung, Programmierung und Instandhaltung.*

*Eine Programmierlösung, die sich in allen drei Phasen und über die verschiedenen Aufgabenfelder hinweg einsetzen lässt, bringt hier Vorteile – besonders dann, wenn Anwender flexibel zwischen Online- und Offline-Programmierung hin und her schalten können.*

**U**nter Offline-Programmierung versteht man das Programmieren eines Roboters in der Simulationsumgebung, während bei der Online-Programmierung direkt am realen Roboter programmiert bzw. getestet werden kann. Typischerweise verlaufen Roboter-Automatisierungsprojekte in drei Schritten, die spezielle Online- oder Offline-Arbeiten umfassen: Planung, Programmierung und Optimierungen während der Instandhaltung. Jede Phase erfordert spezielle Tools, individuelles Know-how und eine gute Abstimmung, weil oft verschiedene Personen dafür zuständig sind. Nicht selten kommt es an den Schnittstellen zwischen den einzelnen Phasen zu Problemen.

getestet werden kann. Der Planer ist daher auf Simulationswerkzeuge angewiesen, um die geometrische Auslegung der Roboterzelle, die Werkzeuge und die Programmabläufe offline festzulegen und zu überprüfen. Hierzu führt er Erreichbarkeitsanalysen und Kollisionsüberprüfungen durch, optimiert die Reihenfolge der Roboteraktionen, bestimmt möglichst effiziente Bewegungsbahnen und legt Sicherheitsbereiche fest.

In der nächsten Phase hat der Programmierer die schwierige Aufgabe, das Ergebnis der Planung, das oft nur abstrakt in Form von CAD-Daten, Bildern und Ablaufbeschreibungen vorliegt, in ein komplexes Roboterprogramm umzusetzen und in der Planung fehlende oder fehlerhafte Annahmen zu korrigieren. Um den geplanten Prozess in die Realität umzusetzen,

Bild 1: Mit der Robot Programming Suite (RPS) lassen sich Roboter aller namhaften Hersteller programmieren sowie kompatible Peripherie-Hardware wie elektrische oder pneumatische Greifer und taktile oder optische Sensoren nutzen.

Zu Beginn gibt es naturgemäß noch keinen „realen“ Roboter, auf dem programmiert und

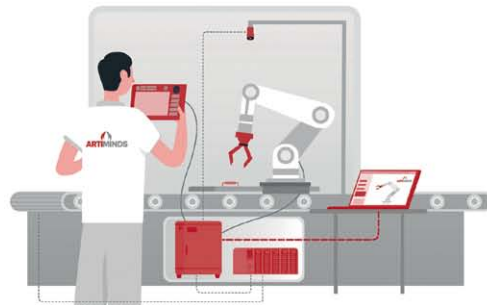


Bild: Artiminds Robotics und Gordenkoff/stock.adobe.com

## PLANUNG & PROGRAMMIERUNG



## INBETRIEBNAHME



## ANALYSE & OPTIMIERUNG IM LAUFENDEN BETRIEB

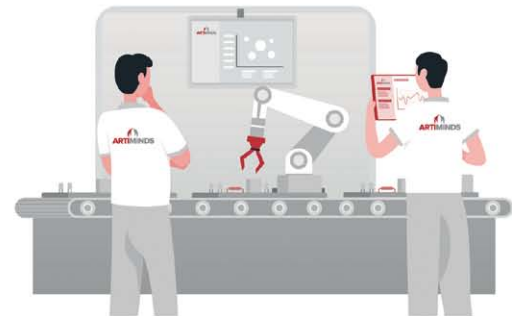


Bild: ArtiMinds Robotics

Bild 2: Die Robot Programming Suite (RPS) lässt sich in allen drei genannten Phasen einsetzen – Planung, Programmierung und Instandhaltung.

Roboterbewegungen flexibel zu berechnen, mit der SPS zu kommunizieren, Werkzeuge und Bildverarbeitungssysteme anzusteuern oder komplexe Sensoren wie Kraft-Momenten-Sensoren einzubinden, muss er herstellerspezifischen Roboter-Code schreiben. Sind Roboter unterschiedlicher Hersteller im Einsatz, führt dies unweigerlich zu größerem Aufwand.

Sobald die Anwendung in der Produktion läuft, wird vom Instandhaltungs-Team erwartet, dass es nicht nur eines, sondern mehrere komplexe Roboterprogramme für unterschiedliche Roboterhersteller schnell (online) korrigieren und verbessern kann sowie Änderungen aufgrund des Schichtbetriebs nachvollziehbar dokumentiert. Hierfür muss der Programmierer das komplexe Roboterprogramm in Form von Dokumentation, Kommentaren und Eingabemasken bestmöglich vorbereiten, damit das in Roboter-Code formulierte Prozesswissen möglichst nicht verloren geht.

### Ein neuer Ansatz

Angesichts der Probleme, die diese drei Entwicklungsphasen mit sich bringen, hat der 2013 als Spin-off des KIT (Karlsruher Institut für Technologie) gegründete Roboterprogrammier-Software-Hersteller ArtiMinds Robotics in seiner Software „Robot Programming Suite“ (RPS) einen durchgängigen Ansatz verwirklicht, der die Online- und Offline-Programmierung nahtlos miteinander kombiniert (Bild 1). So muss für alle Phasen der robotergestützten Automatisierung nur ein Produkt eingesetzt werden (Bild 2). Der neue Ansatz ist leicht in die bestehenden Abläufe zu integrieren, weil die Software grafisch mithilfe vordefinierter Anwendungsbausteine programmiert wird, mithin automatisch Roboter-Code erzeugt wird, der in der Standardsteuerung des Roboterherstellers läuft. Daher fallen zum einen die Schnittstellenprobleme zwischen verschiedenen Tools weg. Zum anderen können Rollen

auch stärker verschmelzen, was eine flexiblere Nutzung von Ressourcen erlaubt und die Effizienz der Zusammenarbeit zwischen den einzelnen Bereichen steigert. So entstehen beispielsweise in der Planungsphase bei gleichem Aufwand wie zuvor realitätsnähere Ergebnisse, und der Roboterprogrammierer kann diese dann mit deutlich weniger Aufwand umsetzen. Das Risiko, dass kritische Probleme erst in der Programmierphase oder bei der Inbetriebnahme auftreten, wird geringer. Auf Basis der Datendurchgängigkeit und eines eigenen Applikationskatalogs kann man sich zudem leicht roboterübergreifende Standards erarbeiten. »Außergewöhnlich bei unserem Konzept ist, dass man damit nicht nur spezielle Teillösungen, sondern komplexe Roboter-Automatisierungsprojekte in ihrer Gesamtheit realisieren kann«, erläutert Dr.-Ing. Andreas Hermann, Senior Team Leader Advanced Robotics bei ArtiMinds. »Und das mit den Robotern aller namhaften Hersteller.«

Durch die einfache Möglichkeit zum Wechsel zwischen online und offline bringt die Software weitere Vorteile, die sich unter anderem bei der Programmierung zeigen. Ein Beispiel macht das deutlich: »Beim Entgraten von Bauteilen oder bei der Inspektion von Kanten müssen Roboter oft komplexe Trajektorien abfahren«, führt Hermann aus. »Diese bewegungsorientiert zu programmieren und mit dem realen Werkstück abzugleichen ist alles andere als trivial.« Hier geht ArtiMinds mit seiner RPS einen anderen Weg. Mit dem Tool „CAD2Path“ lassen sich in der Offline-Welt CAD-Daten des Bauteils einlesen und daraus automatisiert Bewegungsbahnen aus dem Modell erzeugen. Für den Abgleich mit der realen Welt wechselt der Programmierer dann in den Online-Modus, bewegt den Roboterarm an einige charakteristische Punkte des Werkstücks und teacht diese ein. Positionen vom digitalen Zwilling können auf den realen Roboter übertragen werden oder umgekehrt (Bild 3). Innerhalb kurzer Zeit lassen sich so komplexe Pfade programmieren. Auch Trans-

ferbewegungen, bei denen es nicht auf Genauigkeit ankommt, können offline oft einfacher definiert und bei Bedarf dann online um ihre exakte Zielposition ergänzt werden.

### Schnell verschiedene Lösungen testen

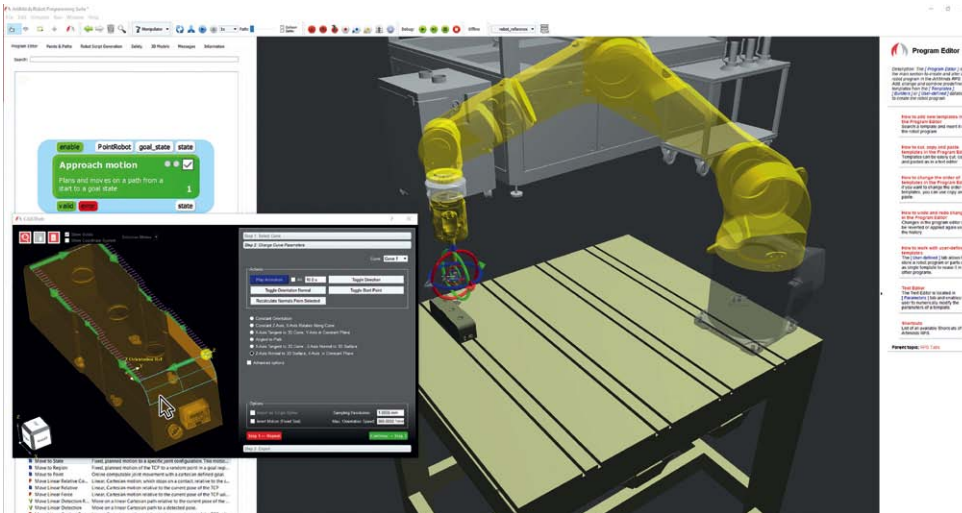
»Durch die Möglichkeit zum einfachen Umschalten zwischen beiden Welten lassen sich verschiedene Lösungsansätze schnell direkt auf dem Roboter evaluieren, weil Teaching, Programmieren und Testen in einer engen Verbindung stehen«, stellt Hermann fest. »Das ist gerade bei herausfordernden Anwendungen, die wegen der Komplexität bei der Programmierung häufiger getestet werden müssen, von Vorteil.« Ein Beispiel sind sensorbasierte Anwendungen mit Kraft-Momenten-Sensoren oder Kamerasystemen, die einen deutlich höheren Programmieraufwand und mehr Know-how erfordern als einfache Bewegungen. »Programmierer können die Technik quasi weg-abstrahieren und sich rein auf die Prozesse konzentrieren«, verdeutlicht Hermann. »Das macht die RPS auch zu einem Tool für Rapid Prototyping sensorbasierter Montageaufgaben. Die Entwicklungszyklen aus Programmieren, Nachteachen, Code Erzeugen und Testen werden kürzer, und unterschiedliche Lösungsstrategien lassen sich effizient evaluieren.«



Andreas Hermann, ArtiMinds Robotics

„Unsere Robot Programming Suite kombiniert Online- und Offline-Programmierung.“

Bild: ArtiMinds Robotics



**Bild 3:** Die harmonisierte Kombination aus online und offline erleichtert die Programmierung. So lassen sich beispielsweise importierte CAD-Pfade an Punkten des realen Objekts kalibrieren.

Instandhalter im erstellten Roboter-Code mit den Standardtechniken des Roboterherstellers gezielt Änderungen wie etwa das Nachziehen eines Wegpunkts vornehmen. Er kann beispielsweise einzelne getauchte Positionen mit wenig Aufwand nachträglich durch Touch-up verbessern und die Aktualisierungen in die Software zurückspielen. Das ist sehr wichtig, weil sonst Dokumentation und Realität nicht mehr übereinstimmen und bei der Realisierung weiterer Anlagen essenzielle Änderungen verloren gehen.« Mit der Zusatz-Software LAR (Learning & Analytics for Robots) ist es zudem möglich, Roboterdaten aus dem realen Betrieb zu analysieren. Damit lassen sich Produktionszyklen in Bezug auf Stabilität, Genauigkeit und Geschwindigkeit optimieren. Gleichzeitig ist es mit der LAR möglich, verschiedene Programm-

stände miteinander zu vergleichen. »So bleiben Änderungen transparent, und es lässt sich sicher sagen, ob beispielsweise eine nachträgliche ‚Optimierung‘ wirklich besser ist als die ursprüngliche Version«, erklärt Hermann. »Zudem profitiert der Instandhalter von einer PDF-Dokumentation inklusive hinterlegter Kommentare, die der Programmierer einfach per Knopfdruck aus der Programmier-Suite erstellen kann.«

Nicht nur für Neueinsteiger ist die Programmier-Suite interessant. Auch für Unternehmen, die bereits über eine Bibliothek aus Roboter-Code verfügen und diese weiterverwenden wollen, bietet sie verschiedene Lösungsansätze. »Einerseits lässt sich Code aus vorherigen Anwendungen integrieren und für zukünftige

Zudem vereinfacht sich der Einsatz von elektrischen Greifern, Bildverarbeitung oder Kraft-Momenten-Sensoren, und der Anwender bleibt beim Setup flexibel. Auch bei Komponenten mit nur einfachen Funktionen ist der Initialaufwand für die Einbindung meist groß, weil zahlreiche Protokolle implementiert und Treiber erstellt werden müssen. Hier unterstützt die Programmier-Suite durch ein umfangreiches Sortiment von Protokollen, wodurch dieser Aufwand entfällt. Schließlich ist die nahtlose Kombination aus Online- und Offline-Programmierung auch für die Instandhaltung von Vorteil, und die RPS erleichtert den Austausch mit den Programmierern. »Ziel unserer aktuellen Entwicklungen ist es, den Disconnected-Modus immer weiter auszubauen«, sagt Hermann. »Das heißt, selbst ohne RPS kann der

Projekte weiterverwenden«, legt Hermann dar. »Andererseits ist auch der umgekehrte Weg denkbar, dass beispielsweise für komplexe sensorbasierte Teilaktionen Code mit der Programmier-Suite erzeugt und in der Programmierumgebung des Roboters eingebunden wird.« Generell wolle ArtiMinds in diesem Zusammenhang den Anwendern keine Vorgaben machen, wie sie den Code nutzen sollen und inwieweit ein schrittweiser Umstieg möglich ist. »Die kombinierte Herangehensweise aus Online- und Offline-Programmierung im Zusammenspiel mit den vorhandenen Regelungsalgorithmen und der Treiberbibliothek erlauben es, prozessorientiert zu arbeiten und diesen Fokus nicht durch zeitaufwändige Detailimplementierungen zu verlieren«, resümiert Hermann. (ak) ■