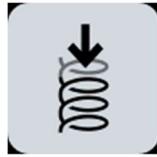


Prototyping App Package (ehem. Research App Package)

Cartesian Compliance



Die App *Cartesian Compliance* erlaubt das Einstellen der Nachgiebigkeit des Arms in allen kartesischen Achsen. Die einzustellenden Achsen sind durch die Einstellung der Transformationsmatrix vom Flansch zum Endeffektor definiert, da die Nachgiebigkeit im Endeffektor-Koordinatensystem definiert ist. Je niedriger ein Wert für eine Achse gewählt wird, desto nachgiebiger wird der Arm in dieser Achse. Die eingestellten Werte werden in allen nachfolgenden Apps beibehalten.

Joint Compliance



Die App *Joint Compliance* erlaubt das Einstellen der Nachgiebigkeit des Arms auf Gelenkebene. Für jedes Gelenk kann hier eine Steifigkeit definiert werden. Analog zur App *Cartesian Compliance* gilt hier auch, je niedriger der gewählte Wert, desto weicher und nachgiebiger wird der Arm in der entsprechenden Achse sein. Die eingestellten Werte werden ebenfalls beibehalten für nachfolgende Apps und können das Bahnfolgeverhalten des Roboters negativ beeinflussen. Sie können jedoch auch genutzt werden, um ein Führen des Arms durch die Umgebung zu implementieren.

Thresholds



Die App *Thresholds* erlaubt das Einstellen von Kollisionsschwellen. Da der Roboter über eine externe Kraft- und Momenteschätzung auf Basis interner Sensorik verfügt, kann der Arm beim Überschreiten von bestimmten Kraft- oder Momenteschwellen seine aktuell ausgeführte Tätigkeit abbrechen. Die Schwellen sollten immer in Hinblick auf die auszuführende Aufgabe gewählt werden. Die eingestellten Werte gelten nun für alle folgenden Apps.

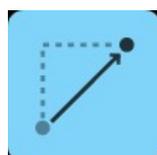
Die App erlaubt im ersten Schritt das Einstellen der Kraftschwellen. Hier kann für jede kartesische Richtung eine spezielle Schwelle eingestellt werden. Dies erlaubt das genaue Anpassen der Schwellen an den konkreten Arbeitsablauf. Im nächsten Schritt können noch die Drehmomentschwellen für jedes Gelenk eingestellt werden. Sollte eine beliebige Drehmoment- oder Kraftschwelle überschritten werden, wird die aktuelle Ausführung unterbrochen und kann nach Behebung des Fehlers fortgeführt werden.

Repeat



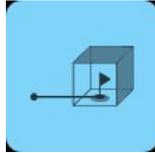
Die *Repeat* App erlaubt das wiederholte Ausführen einer Reihe von Apps. Hierzu muss die Anzahl der Wiederholungen eingestellt werden sowie die Aktion, die am Ende eines jeden Durchlaufs ausgeführt werden kann. Hier kann aus 3 Aktionen gewählt werden. *Wait* wartet für die eingestellte Zeit zwischen jeder Iteration, *User* wartet auf eine haptische Bestätigung durch den Operator und *Repeat* beginnt einfach von vorn.

Relative Motion



Die App *Relative Motion* bewegt den Arm relativ zu seiner aktuellen Position im kartesischen Raum entlang eines einstellbaren Vektors. Dabei kann konfiguriert werden, ob die Bewegung relativ zum Endeffektor oder im Koordinatensystem der Basis ausgeführt werden soll.

Move Contact



Die App *Move Contact* erlaubt das Abfahren einer eingelernten Abfolge von Punkten im kartesischen Raum. Der Unterschied zu *Cartesian Motion* ist die unterschiedliche Reaktion auf Kollision in einem streng definierten Bereich. Die App bietet die zusätzliche Möglichkeit, dass auch das Bemerkens eines Kontakts in einem streng definierten Bereich das Erreichen der Zielpose signalisiert. Somit kann zum Beispiel ein Stapel implementiert werden. Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Joint Compliance* App eingestellt wurden.

Die Konfiguration erfolgt zunächst analog zur App *Cartesian Motion*, indem die Trajektorie eingelernt wird und die Geschwindigkeit und Beschleunigung konfiguriert werden. Die maximalen Werte können abhängig von Konfiguration, Endeffektorgewicht und eingestellter Maximalgeschwindigkeit abweichen.

Zusätzlich kann nun noch der Radius um den letzten Punkt definiert werden, indem ein Kontakt als erfolgreiches Beenden der Bewegung gewertet wird.

Apply Force



Die App *Apply Force* lässt den Roboter für einen benutzerdefinierten Zeitraum die eingestellte Kraft applizieren. Die Kraft wird in der aktuellen Position des Arms aufgetragen. Nicht konfigurierte Endeffektoren oder andere Gewichte, die die externe Kraftschätzung des Arms beeinträchtigen, werden auch die Genauigkeit der applizierten Kraft negativ beeinflussen. Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Cartesian Compliance* App eingestellt wurden.

Die Konfiguration der App besteht aus dem Einstellen des gewünschten Zeitraumes, in welchem die Kraft angewendet wird. Zusätzlich ist es erforderlich die maximale Distanz anzugeben, in die sich der Arm während der Ausführung der App bewegen darf.

Hierzu werden verschiedene Schwellen für Abweichung in X, Y und Z eingestellt sowie zusätzlich eine Schwelle für die

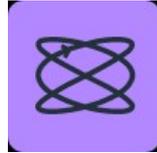
Line



Die App *Line Motion* bewegt den Arm mit einer vorgegebenen Geschwindigkeit entlang einer Linie. Zusätzlich ermöglicht die App das Auftragen einer Kraft während der Bewegung. Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Joint Compliance* App eingestellt wurden. Sollte die Option *Apply Force* gewählt sein, werden die Werte aus der *Cartesian Compliance* App genutzt.

Die Parametrierung der App erfolgt folgendermassen: Zuerst wird durch das Einlernen zweier Punkte die Richtung, welcher der Arm folgen soll, definiert. Danach wird die Zeitdauer eingestellt, die der Roboter der Linie folgen soll sowie die Geschwindigkeit, mit der die Linie abgefahren werden soll. Zusätzlich muss nun die maximale Distanz in allen kartesischen Richtungen angegeben werden, die sich der Arm von seinem Ausgangspunkt entfernen darf. Soll der Arm zusätzlich zur Bewegung noch eine Kraft aufbringen, muss die aufzubringende Kraft eingestellt werden. Dies erfolgt analog zur Konfiguration der *Apply Force* App.

Lissajous Motion



Die App *Lissajous Motion* ermöglicht das Abfahren verschiedener Lissajous Figuren. Zusätzlich kann während des Abfahrens eine Kraft aufgebracht werden. Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Joint Compliance* App eingestellt wurden. Sollte die Option *Apply Force* gewählt sein, werden die Werte aus der *Cartesian Compliance* App genutzt.

Zur Parametrierung wird zuerst die Dauer der Bewegung eingestellt. Danach kann die Bewegung genauer konfiguriert werden. Hierzu werden die Anzahl sowie die Amplitude der Bewegungen in beide Richtungen konfiguriert. Danach kann die Ebene, in der die Bewegung ausgeführt wird, ausgewählt werden. Soll zusätzlich noch eine Kraft angewendet werden, kann diese analog zur *Apply Force* App. Eingestellt werden.

Spiral Motion



Die App *Spiral Motion* führt zum Ausführen einer Spiralbewegung des Arms. Zusätzlich kann während der Bewegung eine Kraft aufgetragen werden. Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Joint Compliance* App eingestellt wurden. Sollte die Option *Apply Force* gewählt sein, werden die Werte aus der *Cartesian Compliance* App genutzt.

Im ersten Schritt der Parametrierung der App wird die Bewegungsdauer eingestellt. In den nächsten Schritten werden der Anstieg, die Breite und die Drehrichtung der Spirale definiert. Danach kann die Ebene, in der die Bewegung ausgeführt wird, ausgewählt werden. Soll zusätzlich noch eine Kraft angewendet werden, kann diese analog zur *Apply Force* App. Eingestellt werden.

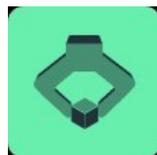
Gripper Move



Die App *Gripper Move* erlaubt dem Benutzer während der Ausführung einer Aufgabe die Greiferbreite zu verändern, zum Beispiel um die Hand vor dem Greifen eines Objektes zu öffnen.

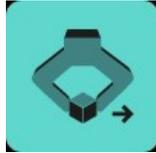
Hierzu kann zum einen die gewünschte Greiferbreite, am Arm oder mithilfe des Sliders, eingestellt werden sowie die Geschwindigkeit, mit der der Greifer verfahren werden soll.

Gripper Grasp



Die App *Gripper Grasp* erlaubt das Greifen von Objekten. Die Konfiguration erfolgt in gleicher Weise wie die Konfiguration der *Gripper Move* App. Zusätzlich können noch die Greifkraft und die Masse des gegriffenen Objektes konfiguriert werden. Diese sollte möglichst genau konfiguriert werden, um negative Auswirkungen auf die Performance der Kraftapplikation oder folgender Bewegungen zu vermeiden.

Transport Motion



Die App *Transport Motion* erlaubt das Abfahren einer eingelesenen Abfolge von Punkten im kartesischen Raum. Der Unterschied zur *Cartesian Motion* App ist, dass zusätzlich während der Fahrt überwacht wird, ob ein gegriffenes Objekt verloren wird und der Verlust des Objektes die Bewegung terminiert.

Zur Ausführung dieser App werden die Nachgiebigkeitswerte genutzt, die in der *Joint Compliance* App eingestellt wurden. Die Konfiguration erfolgt zunächst analog zur App *Cartesian Motion*, indem die Trajektorie eingelesen wird und die Geschwindigkeit und Beschleunigung konfiguriert werden. Die maximalen Werte können abhängig von Konfiguration, Endeffektorgewicht und eingestellter Maximalgeschwindigkeit abweichen.

Modbus Out



Die App *Modbus Out* setzt den konfigurierten Modbus-Ausgang auf den konfigurierten Wert.

Zur Konfiguration muss der entsprechende Ausgang ausgewählt werden. Das bedeutet die Konfiguration des Moduls, der Karte und des Ausgangs sowie des zu setzenden Wertes.

Modbus Pulse



Die App *Modbus Pulse* setzt den konfigurierten Modbus-Ausgang auf den konfigurierten Wert. Nach der eingestellten Wartezeit wird der Ausgang wieder zurückgesetzt.

Zur Konfiguration muss der entsprechende Ausgang ausgewählt werden. Das bedeutet die Konfiguration des Moduls, der Karte und des Ausgangs sowie des zu setzenden Wertes. Im nächsten Schritt kann die Wartezeit eingestellt werden.

Modbus Wait



Die App *Modbus Wait* wartet, dass der konfigurierte Modbus-Eingang den erwarteten Wert annimmt. Sobald dies geschieht, beendet sich die App und die darauffolgende App wird ausgeführt.

Zur Konfiguration muss der entsprechende Eingang ausgewählt werden. Das bedeutet die Konfiguration des Moduls, der Karte und des Eingangs sowie des zu erwartenden Wertes.

Wait



Die App *Wait* bietet die Möglichkeit, die Ausführung für einen bestimmten Zeitraum pausieren zu lassen. Der angestrebte Wartezeitraum kann in Sekunden konfiguriert werden.

User Interaction



Die App *User Interaction* lässt den Roboter auf eine haptische Bestätigung durch den Anwender warten. Nach dem erfolgreichen Triggern einer haptischen Geste in einer der konfigurierten Richtungen, wird die App beendet und die folgende App ausgeführt.

Als erster Schritt wird die Kraftschwelle definiert, die überwunden werden muss, um die Geste zu triggern. Im nächsten Schritt werden alle Richtungen ausgewählt, in die ein Kontakt erkannt werden soll.