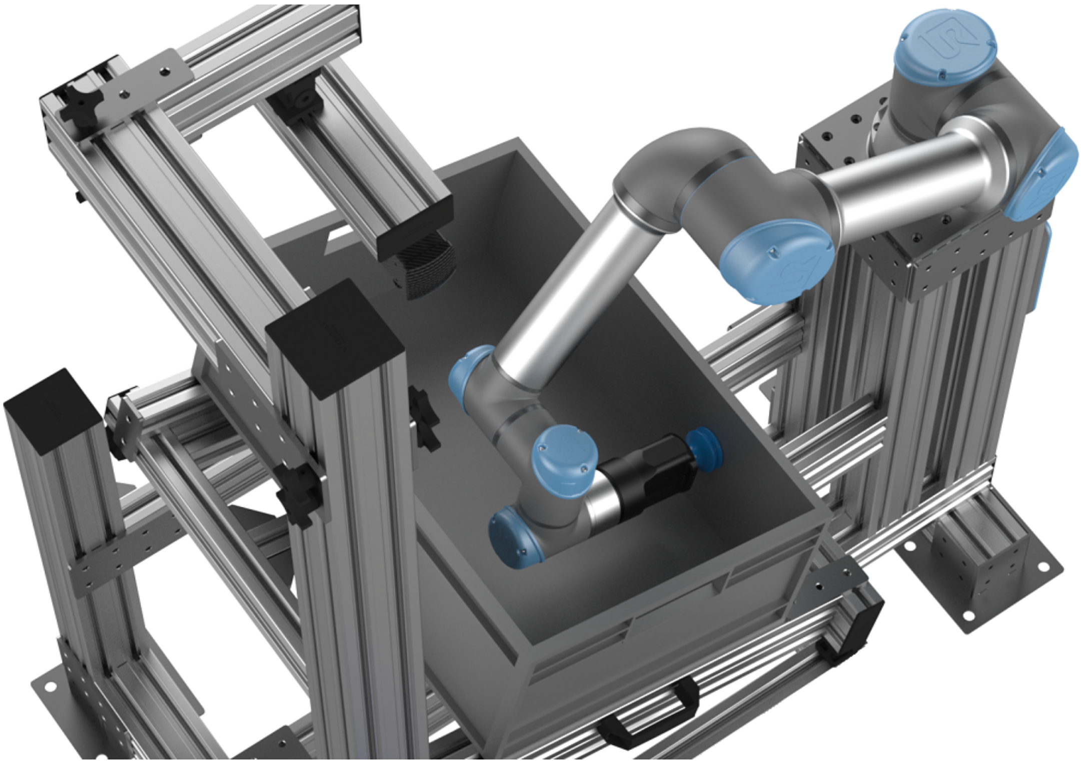
UNIVERSAL ROBOTS



ActiNav

NEXT-GENERATION MACHINE LOADING

ActiNav Next-Generation Machine Loading

Integrator Guide

Original instructions (en)

Contents

1. Introduction

1.1. About this document

The purpose of this Integrator Guide is to provide a comprehensive technical reference to use during the ActiNav integration process.

This guide is divided into the stages of a Bin Picking application integration process. It covers topics related to application qualification (e.g. how to assess parts for Bin Picking), application design (e.g. workcell and tool design), application installation and configuration (e.g. configuring the network and perception settings) and application programming (e.g. how path planning works).

1.2. Related materials

- ActiNav Viewer:
  www.universal-robots.com/articles/ur/actinav-viewer/
- Universal Robots e-Series robot User Manual:
  www.universal-robots.com/download/
- Mesh Lab Document:
  www.universal-robots.com/articles/ur/actinav/actinav-how-to-use-meshlab-to-resize-your-part-models/

1.3. Business Contact Details

Universal Robots A/S

Energivej 25, 5260 Odense S, Denmark

+45 89 93 89 89

https://www.universal-robots.com

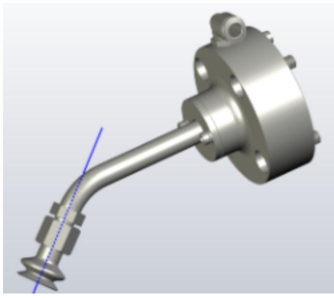
1.4. Terminology

This section describes common terms in the ActiNav Bin Picking process.

- **Tool**: In this document, "tool" is used to refer to the End-effector. End-effector is the technical term for any workpiece attached to the flange of the robot.
- **Path Planning**: The algorithm that calculates a series of robot poses and movements to pick and place a part without collisions among the robot, the part, the tool, the bin, and/or the environment.
- **Clearance Shape**: A simplified shape surrounding the robot, tool bin, part, and environment. Collisions between clearance shapes are not permitted, with limited exceptions (such as the tool contacting the part for a pick). Clearance shapes are generated automatically for the robot

and the bin (as part of teaching the bin), and are created by the user for the tool, part, and environment.

- **Terrain**: A height map or "shrink wrap" of everything in the bin. It is a dynamically generated clearance shape for avoiding collisions with unpicked parts (or anything else in the bin) during picking.
- **AMM Robot**: A semitransparent virtual robot in the ActNav Viewer. It shows what paths the AMM is considering during path planning.
- **Axis of Symmetry**: A characteristic of a part or a tool. Any rotation around this axis results in the same pick. In the exaples illustrated below, for a cylindrical part, the axis of symmetry runs the length along the middle of the part.



- **Free Spin**: The ability of a symmetrical tool to assume any rotation around its axis of symmetry dring picking. This allows many different picks after teaching a single one. Free spin automatically adjusts the placement so the part is always placed the same way regardless of the way it was picked.
- **Reference Frames**: Reference frames define the coordinate system for certain actions and definitions in ActiNav.
    - **Base or Robot**: The robot coordinate system, with its origin in the center of the robot base. Used for the coordinates of the environment and the place location, and for normal Waypoint programming.
    - **Tool Flange**: The coordinate system referenced to the middle of the tool flange on the robot arm. Used for the coordinates of the tool clearance shapes.
    - **Tool**: The coordinate system with its origin at the Tool Center Point (TCP). Used for specifying the post-pick retraction, and for teaching.
    - **Part**: The coordinate system with its origin at the part origin. Used for the coordinates for the part clearance shapes and for pick points.
- **Relative**: Used to refer to the defining coordinate system. A "part-relative" Custom Pose adjusts the robot pose to always put the part in the same part orientation. A "robot-relative" Custom Pose always assumes the same robot pose regardless of how the part is held.

1.4.1. ActiNav abbreviations

This section lists the acronyms and abbreviations used in this document.

Abbreviation	Meaning	Abbreviation	Meaning
AMM	Autonomous Motion Module	FNP	Find Next Part
CA	Conditional Action	ID	Inside Diameter (pick)
CoG	Center of Gravity	OD	Outside Diameter (pick)

2. Safety

2.1. Safety message types

Safety messages in this document contain information to help you avoid injury or equipment damage. This document contains the following safety message types.

WARNING
This safety message indicates a hazardous situation that, if not avoided, could result in death or serious injury.

CAUTION
This safety message indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.

NOTICE
This safety message indicates a situation that, if not avoided, could result in damage to equipment or property.

2.2. General safety precautions

This section contains general safety precautions, read it before installing and/or operating ActiNav.

WARNING
Failure to perform a risk assessment before installing and operating the ActiNav application can result in equipment damage or personnel injury.

- Perform a risk assessment before installing and operating the ActiNav application.
- Read the Universal Robots e-Series Robot User Manual and Service Manual.

WARNING
Performing installation or maintenance of equipment connected to a power source can lead to electric shock.

- Ensure that the equipment is disconnected from the power source before performing installation or maintenance.

CAUTION

Failure to perform installation or maintenance correctly can result in equipment damage or personnel injury.

- Only qualified personnel shall perform installation, start-up, and maintenance.
- Read the UR User Manual and the UR Service Manual.

NOTICE

This product is used exclusively with the Universal Robots e-Series robot. General safety considerations valid for the Universal Robots e-Series robot are also valid for this product.

2.3. ActiNav-specific safety precautions

This section contains safety precautions specific to the ActiNav set-up and applications.

The integrator is responsible for determining the need for specific ActiNav safety precautions and providing them.

WARNING

Failure to provide adequate safeguarding, as required by the risk assessment, can result in death or serious injury.

- The integrator shall ensure adequate safeguarding.

WARNING

Failure to turn off power to the robot when servicing, or repairing, any part of the ActiNav set-up can result in death or serious injury.

- Turn the power off during service and/or repair.

The integrator shall be responsible for installing and programming ActiNav, to lessen unexpected movement situations.

CAUTION

Robot inactivity can be perceived as a stop, resulting in equipment damage or personnel injury from unexpected movement.

- Check the program state to determine if the robot is completely stopped or temporarily inactive (waiting). If the program is running but the robot is not moving, the robot can move again unexpectedly.
- Configure warning behavior in the Motion Start Actions.

For more information on Motion Start Actions, see the ActiNav Operating Guide.

2.4. Tool-specific safety precautions

This section contains safety precautions specific to ActiNav tool usage.

The integrator shall design, or select, the tool (see requirements in EN-ISO 10218-2).

WARNING

The potential energy of the payload can lead to a throwing hazard, in the following cases:

  - Poor grasp (workpiece retention).
  - The workpiece is too heavy.
  - The tool is poorly designed.
  - The robot moves too quickly.

- The integrator shall design, or select, the proper gripper, taking into account: the robot movements, the part features and the inertia of the workpiece when gripped by the robot.

In the case of vacuum grippers and electromagnets, loss of power can cause the workpiece to fall from the tool as a result of gravity.

CAUTION

Loss of power can result in the tool dropping the workpiece.

- Turn the power off before removing the tool.
- The integrator shall ensure there are backup systems to keep the tool holding the workpiece, if power is lost unexpectedly.

2.5. Integration-specific safety precautions

This section contains safety precautions specific to ActiNav's integration processes.

WARNING

The robot's pick and place movement can be irregular, causing the robot to swing over singularities.

- Always bolt down the robot securely.

WARNING

Failure to draw clearance shapes on top of safety planes may cause ActiNav to violate the safety place, causing the robot to stop, and/or drop the part.

- Always draw the Clearance Shapes on top of safety planes.

NOTICE

Failure to meet the minimum requirements in part geometry can render your ActiNav application unworkable.

- Make sure to meet all the minimum requirements when definining the geometry of the part.

NOTICE

Using a part model file size larger than 8 MB can cause import error/s, as bigger files are harder to resize.

- Keep all file sizes under 8 MB.
- Use models in the 2-5 MB range to avoid issues.

NOTICE

Placing a part in a machine backwards, or not at all, can jam or damage the machine.

- Incorporate a sensor in the fixture to ensure you place the part correctly in the regrip fixture.

NOTICE

When configuring bin placement, angling the bin can cause parts to roll toward the bin floor, leading to increased cycle time.

- Do not use angled bins.
- Do not use angled 3D sensors.

2.6. 3D sensor-specific safety precautions

This section lists labels and warnings specific to the 3D sensor. All labels are located on the back panel of the 3D sensor, except for the Laser aperture label.

CAUTION

The 3D sensor is a class 3R laser product. This class of laser has a low risk of injury.

- Avoid looking directly into the beam.

NOTICE

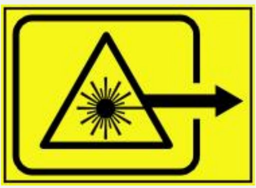
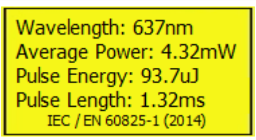
Allowing the robot to rest, or to settle onto another object, during sample taking, can cause inaccuracy in the robot position and incorrect location data being read.

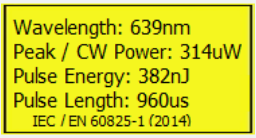
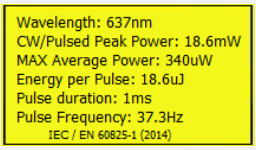
- Do not allow the robot to settle onto other objects while taking samples.

NOTICE

Bumping or nudging an aligned system can cause the alignment to be lost.

- Do not bump the sensor after you align the system.

Label	Description	Laser class
	Laser radiation hazard warning symbol.	
	Laser aperture label. Designates the place from which laser radiation is emitted.	
	Label with manufacturer address, product name, and model, CE and FCC marks, disposal directions and country of origin.	
	Laser radiation warning with laser class label. The serial number of the device can be found above the warning labels.	3R
	Label specifying wavelength, average power, pulse energy and pulse length of the laser. Used on scanner models M, L, XL.	3R

Label	Description	Laser class
	Label specifying wavelength, average power, pulse energy and pulse length of the laser. Used on scanner models M, L, XL.	3R
	Laser radiation warning with laser class label. The serial number of the device can be found above the warning labels.	2
	Label specifying wavelength, average power, pulse energy and pulse length of the laser. Used on scanner models M, L, XL.	2

2.7. Intended use

ActiNav is designed for performing the following operations:

- Locating objects in a bin.
- Picking the objects using a tool attached to a robot arm.
- Moving the objects to a destination location.
- Placing objects in a specific way at the destination location.

Use the product only within the range of its technical specifications. Any other use of the product is considered improper and unintended.

For full warranty details, refer to the Warranty chapter in the Universal Robots e-Series Robot User Manual.

> **NOTICE**
> Damage to the robot, ActiNav and other equipment, resulting from any improper or unintended use, is not covered by warranty.

> **NOTICE**
> Universal Robots is not liable for any damage resulting from any improper or unintended use.

> **NOTICE**
> ActiNav is not intended to handle Electro Static Discharge (ESD) parts or items.

NOTICE

ActiNav is not intended to operate under intense, harsh or direct light.

- Intense light can result in reduced performance.
- See the technical specifications for the 3D Sensor.

2.8. Foreseeable misuse

Use of ActiNav for a purpose other than the intended use is considered to be a misuse.

This includes, but is not limited to:

- Failure to follow the safety precautions written in the product documentation.
- Failure to perform the risk assessment.
- Failure to install visual indicators around the product workspace informing the personnel that the robot arm moves automatically.
- Overloading the robot arm.
- Use of the product outdoors.
- Operating the product in conditions where the environment parameters exceed the values specified in the technical specifications of the product.
- Use in potentially explosive environments.
- Use in medical and life critical applications.

2.9. Risk assessment

For information on residual risks, refer to section Risk Assessment in the Universal Robots e-Series robot User Manual.

3. Application qualification

3.1. Parts

A viable bin picking candidate is a part with multiple pick points. These are locations on the part that allow it to be picked. The part shall be able to lie inside a bin and display any of these pick points, regardless of how the part is presented in the bin.

3.1.1. Part size

ActiNav has no maximum part size requirement, other than the robot payload. The recommended maximum part size is a 6 inch (150 mm) cube. The minimum part size ActiNav can pick is about 0.4 inch (10 mm) in length and at least 0.3 inch (8-10 mm) in width and height.

The minimum part size is a function of the resolution of the medium sized 3D sensor, and ActiNav's ability to find parts near the bin walls or bin floor. ActiNav's perception algorithms remove data closer than 2 mm to the bin wall or bin floor. It requires at least 5 mm more to discern the presence of a part.

3.1.2. Part picking: Feature and Tools

Part features are the sides or surfaces and angles (if present) that define the characteristics of a part. A tool is usually mounted on the robot arm to interact directly with the part.

Part features

The dimensions of the contact areas, on the pick points, shall be large enough to support the weight of the part when picked. The safety margin shall be 5 mm to allow for:

- Poor contact
- Robot accelerations
- Leaks (vacuum tool)

In the example below the part is a cylindrical object that can be picked by expanding internal fingers on either end, or by external fingers around the smaller diameter, using the same tool.

Tools

ActiNav's robot control allows a tool to hold, or grasp, a part in any position. See 2.3. ActiNav-specific safety precautions on page 12.

ActiNav supports the following tool types:

- **Vacuum and magnetic**: The part shall have sufficient contact area and multiple pick points.
- **Internal gripper fingers**: The part shall have multiple pick points on the inside.
- **External gripper fingers**: The part shall have multiple pick points on the outside.

You can combine internal and external gripper fingers.

Forced Monitored Pick

A Force Monitored Pick allows you to pick a part by grasping the outer edges. If a certain force is exceeded, ActiNav abandons the outside pick. There shall be enough clearance around the part for the tool's fingers to fit without colliding with adjacent parts.

A good example of this is a shaft with multiple diameters. If you add the thickness of the tool's fingers to the smaller diameter of the shaft, and the combined diameter is still small enough, you can use an external grasp.

Vacuum example:

- A part weighs 100 g (0.98 Newtons at sea level, or 3.5 oz)
- The vacuum generates a maximum of 90% vacuum
- Air pressure at sea level is 1 bar (100 kPa or 14.5 psi)
- So the minimum contact area is 5 x 0.98 N / 100 kPa / 0.9 = 0.54 cm^2 = a vacuum cup with a 10 mm (0.4 inch) diameter contact area.

3.1.3. Part geometry

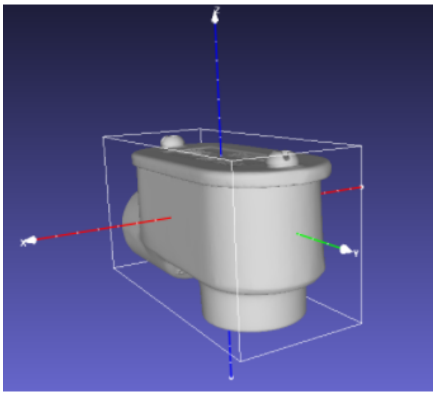
The shape or geometry of a determines Bin Picking performance. The part shall be pickable (see: 3.1.2. Part picking: Feature and Tools on the previous page), so it can be placed. A part can be placed into an intermediate fixture, or placed directly at the final destination.

A part with more pick points creates many opportunities to find a path to be extracted from an unstructured bin. So more pick points mean better system performance. Insufficient pick points on a part can cause the system to run slower, or can prevent every part from being cleared from the bin. The part's pick points shall be similar enough to be picked by the same tool.

When considering the viability of a part for ActiNav, you can imagine a box around the part, simlar to the example below, which assumes all part axes refer to the part's sides.

- +Z is Top
- +Y is Front
- +X is Left

- -Z is Bottom
- -Y is Back
- -X is Right



Part geometry rules

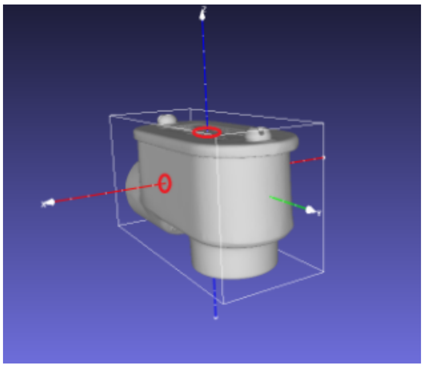
The rules below are the minimum requirements for ActiNav to have commercially acceptable performance.

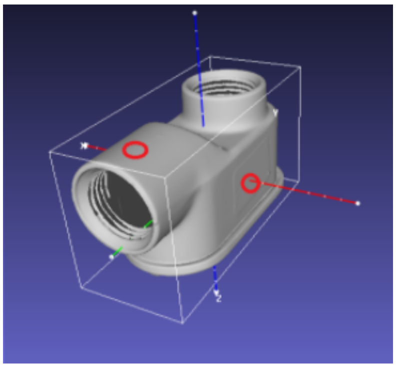
Rule	
1	The part shall be pickable from at least 1 ± axis and/or sides. Plus one other point.
2	A part that is not flat shall be pickable from all of the sides that may present in a bin.
3	The delta ▵ part thickness shall be at least 3 mm to 4 mm from the bin floor.

The only exception to these rules is if the part lies flat inside a bin (see: Example Case 3: Flat parts on page 22).

Example Case 1: No axial symmetry

In this example the part is a conduit. There is no rotational symmetry around any of the axes. Dissecting the conduit proves the part geometry rules.

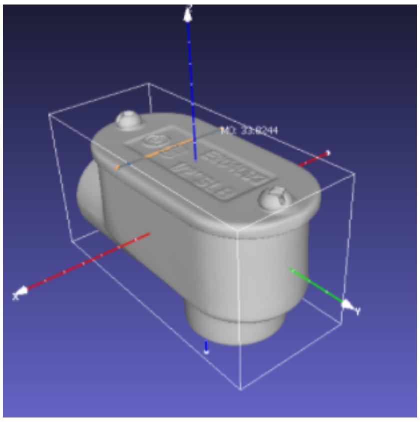




Rule	
Following rule 1	You can pick from both sides along at least one axis. In this case, the pick can be from 2 ± axes, namely ± X and ± Z. There is a surface on the part that can be picked with the same vacuum tool for both axes. The pick points do not need to be on the axis; they just need to be pickable on that side of the cube.

Rule	
Following rule 2	You can pick from four sides: +Z Top, -Z Bottom, +X Left, and -X Right.
Following rule 3	The part thickness, when measured, is above 3-4 mm from the bin floor. This rule is difficult to conceptualize, however other example cases in this chapter will better illustrate it.

In the example below, all 3 rules are fulfilled making the conduit, with 4 pickable sides, a great part to use for ActiNav.

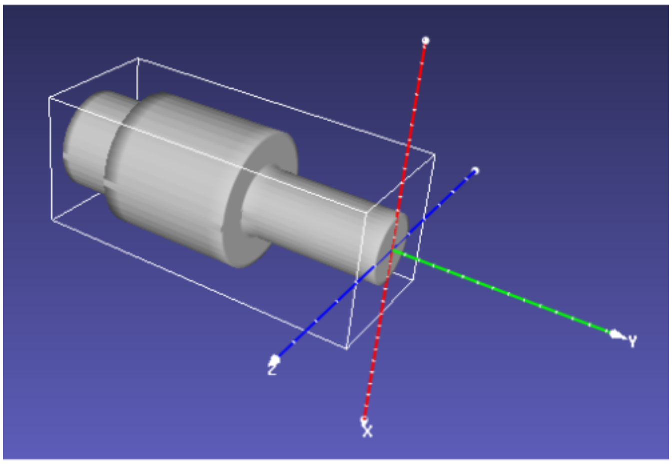


Example Case 2: Axial symmetry

In this example the part is cylindrical.

Cylindrical parts have straight parallel sides and a circular cross-section. A cylindrical part shall naturally lie on its side in the bin. Assuming the tool can pick the part from the side, cylindrical parts are the most ideal parts for ActiNav, as there are infinite pick points around a symmetrical axis. There is always a pickable side exposed, giving cylindrical parts the fastest cycle times.

The cylindrical part, illustrated below, always has a pickable side in the +Z, -Z, +X, and -X axes.

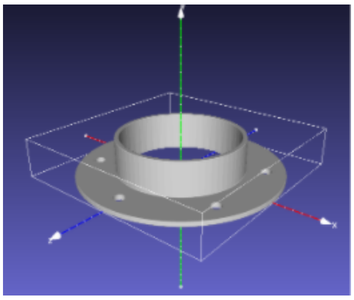


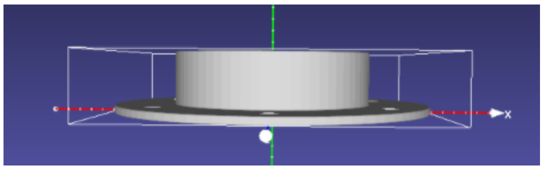
You can improve the cycle time even further if it does not matter how the part is placed as long as it is placed on its axis of symmetry.

Cylindrical flat part

In this example the part is cylindrical and flat. It does not lie on its side in the bin.

It is possible to pick the part from the side, but there are not enough parts with a pickable side exposed in the bin. The bin walls and adjacent parts can cause obstructions.





A good practice is to take into account the worst-case scenario when doing tests. In this case, the worst-case scenario would be to leave just a handful of parts lying flat on the bin floor. If the part lies flat, the side is not visible. The system shall detect it based on the top view.

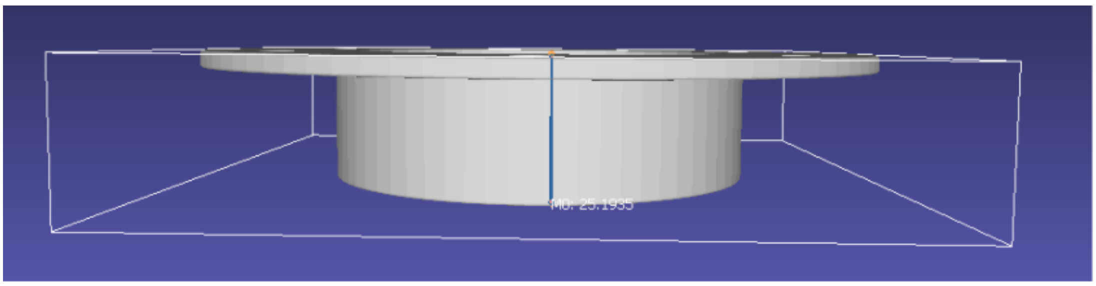
Example Case 3: Flat parts

In this example, the part is a circular flange that can be picked using internal fingers. Flat parts are only pickable from the top and bottom of the part. Follow the rules below:

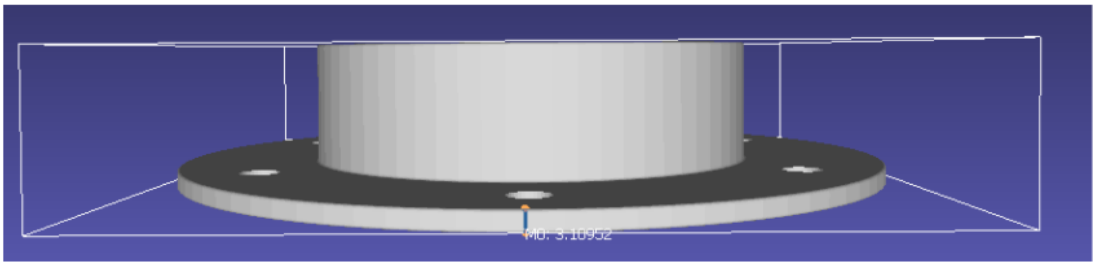
Rule	
1	You can pick from at least 1 ± axis ±Y
2	The exception is flat parts so only 2 pick points are required.
3*	The delta △ part thickness is at least 3-4 mm thick from the bin floor. Rule 3 has less to do with picking the part and is more about the 3D sensor seeing the part.

*This rule can be confusing, so this flat circular part provides a great example of a fringe case.

The image below shows the delta △ part thickness is 26.1935 mm from the bin floor, making the part easy to detect in this orientation.



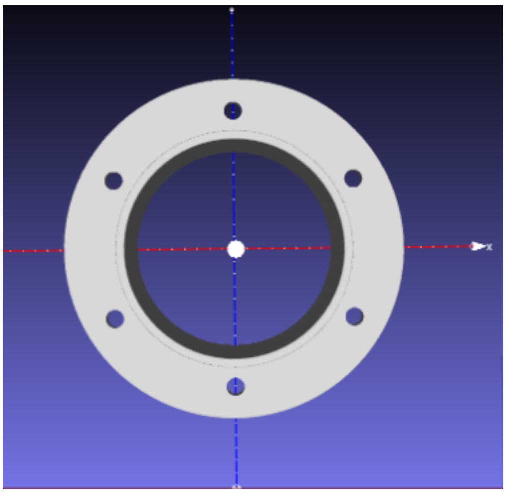
The image below is on the edge of what the 3d sensor can detect, because the part thickness to the bin floor is 3.109 mm.

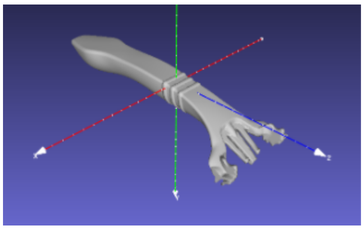
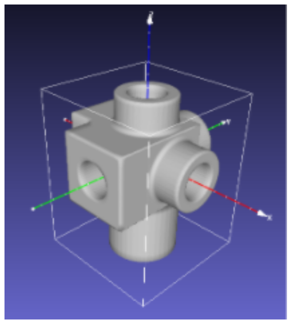


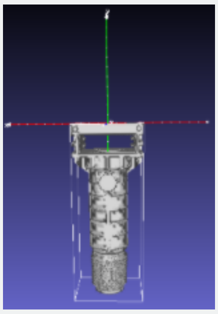
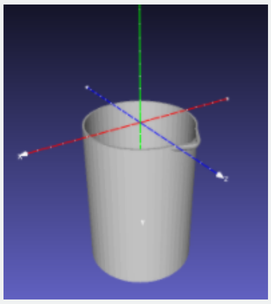
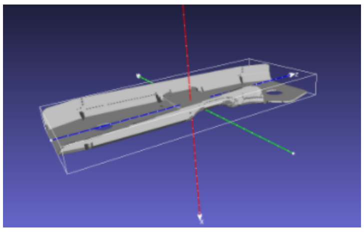
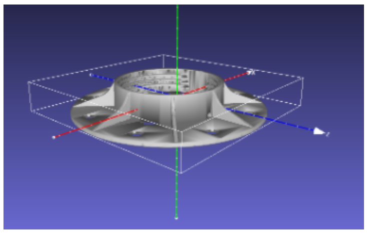
To the user the part above does not appear flat, but the 3D camera detects the part from the top, without depth. So the part appears as 2 circles:

1. The outer circle of the flange
2. The inner circle of the vertical cylinder

The camera finds the delta △ thickness of the part by referencing the bin floor to the first height it can measure. In this case it detects 3.109 mm as the first edge.



Examples of other parts and their classifications		
No distinct geometry		

Examples of other parts and their classifications		
Cylindrical parts		
Flat parts		

Once you determine the part geometry is acceptable, and you have enough pickable points, you then need to examine material and surface finish.

Part material and surface finish

Part material

In bin picking, the type of material making up the part impacts percption and quality of grasp by the tool. The nature of the part material (solid, opaque/translucent, transparent, or reflective) is also important.

ActiNav supports many materials including:

- metal
- plastic
- ceramic
- fiberglass
- wood

ActiNav does not support transparent materials like glass, and reflective materials because they can cause the 3D sensor's laser to reflect multiple times around the bin, causing noise and/or "holes" in the image.

Matte materials are acceptable to ActiNav, but opaque/translucent parts are more challenging to match.

If you encounter issues with opaque/translucent materials, contact your distributor or UR SSE to try to modify some of the perception parameters for the particular part.

Surface finish

Surface finish can determine whether or not a part is pickable, depending on the type of tool. For a vacuum, parts with a cast, forged, or embossed finish can be challenging to pick. For grippers, there should be a safety factor of 5 times the amount of force required for a vacuum tool, to grasp the part vertically.

Example: If it takes 20 N to lift the part vertically with a vacuum cup you should allow 80 N of force for a gripper to grasp the part.

This is because ActiNav generated paths are not repeatable, so the moment force of swinging the part around can be greatly increased (see: 2.3. ActiNav-specific safety precautions on page 12). Apply the safety factor to each of the pick points because part CoG (Center of gravity) can make a difference in the amount of force applied to your grasp.

Part models

ActiNav accepts the following part model formats:

- STL
- STEP - can be either AP203 or AP214.
- OBJ - object mesh files produced when using a 3D scanner to create a part model.

ActiNav fully supports OBJ files but using an OBJ file format is unlikely, unless you or a third party scans the parts.

Resizing the model optimizes perception by either adding or subtracting points to the model. Depending on the size of the model loaded, ActiNav resizes your file to automatically be simplified to 1000 K vertices.

Models below 200 KB may have too few points to reliably match the 3D sensor data. However, a small STEP file may still be able describe the part precisely.

Axis of symmetry

Part origin determines how well the tool picks the part, except if you use the Axis of Symmetry.

Using the Axis of Symmetry feature (see: 6.2.3. Part and tool symmetry on page 49) on the part pick, when rotating about the axis of symmetry of the part, requires a “perfect” origin. This is due to calculating an offset to the origin, based on the Pick Rule.

If the origin is not “perfect”, then the resulting picks cause the tool to push hard into the part, while other picks do not even touch the part. You can keep the origin as close to the pick location as possible to cause the least amount of rotation error.

Part model size is a factor in how quickly perception can match to your part. Use a surface model instead of a solid body model, as using a solid body model decreases the quality of the match to your part.

The external surfaces of the part are visible during matching. So having more internal data bogs down matching. Follow the link below to learn how to create a surface model:

www.universal-robots.com/articles/ur-articles/actinav-creating-a-simplified-part-model-using-solidworks/

Flat parts use a different perception algorithm that require a slightly different part model. You can use either a STEP file, or an OBJ file, without simplifying the model. The only restriction on flat parts is the model requires less than 20,000 vertices.

To resolve file size issues, or resize a part model, you can review the following Mesh Lab document:

www.universal-robots.com/articles/ur/actinav/actinav-how-to-use-meshlab-to-resize-your-part-models/

3.2. Cycle time

The current average bin pick cycle time of 10 to 90 seconds, can become faster as new software is released.

The 10-second minimum cycle time refers to the time it takes to get 1 part out of the bin and into a known orientation. The minimum cycle time does not account for additional robot motion.

During a Bin Picking program, the cycle time can get progressively faster, or slower, depending on how many pickable parts are in the bin. Depending on the part geometry, it may be possible to pick a part and place it in a known orientation, but the part's orientation can make it impossible to place the part into the machine.

This is common with Bin Picking. In such a case, you can regrip the part (see: 4.2. Regrip Fixtures Design on page 28).

Regripping the part requires additional time (about 8 seconds) for the additional motions after a regrip. Regripping has the advantage of increasing placement accuracy.

If it is critical to keep up with a determined cycle time for an application, you can purchase an additional robot to grab the part from the regrip station. The cycle times remain within a lower range.

4. Application design

4.1. Tool design

ActiNav is the only Bin Picking solution on the market using very small tools.

The best type of tool is usually offset from the final wrist of the robot and bent at a 45 degree angle. So the desired TCP is around 130 in the Z, and about a 40 mm offset in Y (half the diameter of the tool flange, to clear the tool flange), and 0 at the X. This configures a similar TCP to a standard vacuum end effector. Grippers can be larger when the bin is larger.

The tool shall be able to access the part, inside the bin, from a vertical angle.

You can test the tool in different positions around the bin to determining if it fits the application.

To test a tool

1. Manually move the tool around the inside of the bin.
2. Try to pick parts lying in different positions and orientations.
3. You can try linear moves from Pre-Pick to Pick, for example. Make sure the tool can reach any part and have enough force to lift it.

   The idea is to test here is the Pre-Pick approach angle.

The tool cannot reach a part if the intended pick point is angled toward a bin wall, which prevents the robot arm from picking other parts.

You can try to fit the tool about 30-50 mm above the part to give you an idea of the Pre-Pick offset and make sure the tool is perpendicular to the desired pick point.

4.1.1. Three-jawed gripper

Avoid using three-jawed grippers because it is unlikely your part is pickable from all sides. Although the three-jawed gripper does self center the part the best, you need a tool that can grab a part from multiple surfaces.

A three-jawed gripper also requires an additional 45 degree mounting bracket, because you cannot actuate a three-jawed gripper with "fingers" bent at 45 degrees. This makes it much more challenging to get to your pick approach angle within your bin because your Pre-Pick offset is now 50 mm plus the size of your EE.

If you plan to use a pneumatic or electric gripper, the recommendation is to use a two-jawed gripper.

4.2. Regrip Fixtures Design

Designing a regrip fixture allows the tool to access the part from every pick point taught. Regrip fixtures ensures the part uses gravity to self center, so the part is always in a known orientation.

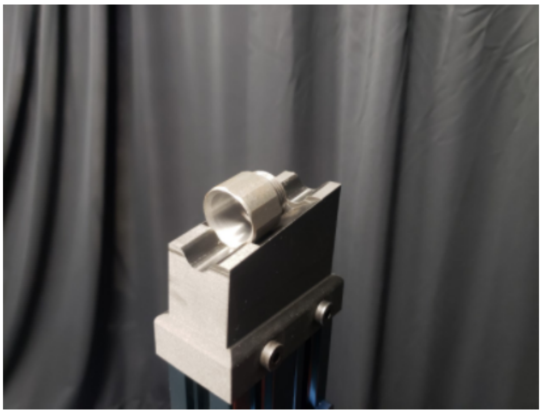
Bin picking applications usually require regrip fixtures to account for the part's pickable sides, CoG and tool access.

4.2.1. Regrip fixture design requirements

- Make sure your regrip fixture can access all of the pick points on the part.
- You can drop the part above the regrip fixture and allow gravity to self center the part.
- Use funnels and chamfers to improve leading in the part to be centered.
- You can use an industrial vibrator to shake parts into better orientation for picking.
- You can also use "V" shaped ramps to gain access to place the part top or bottom.

The examples below illustrate a ramp and a regrip fixture:

In the image on the left, the part can be placed from the top, bottom, and the sides. It can be also placed in any orientation without using a regrip. The ramp leads to a conveyor that leads into a machine for processing.



The image on the right shows a regrip fixture in use.There is a "V" groove to vibrate the part into the proper position to be regripped and placed into the machine. The part can be picked internally top and bottom and can be picked externally on the smaller diameter with the threads. If the part is placed, top or bottom, the lip of the regrip fixture holds the part in case the fingers have too much friction on the tool retraction.

You can access the external grasp by placing vertically down. This is also a great example of the artificial clearance shapes (see:6.3.2. Artificial clearance shapes on page 55) needed to force the tool to place the part vertically.

4.2.2. Why does the application need a regrip fixture?

Many machine tending applications have a single acceptable robot pose for how to enter into a machine. Regripping ensures the part can achieve the acceptable robot pose.

It is common for the type of Pick to cause a part to be incorrectly picked, making it impossible to place into a machine. A part can be upside down, making it impossible to place it correctly into the machine. Regripping corrects the Pick.

Although you can pick one side of a non-symmetrical part to place it into the machine, all of the other sides shall allow regripping.

Example:

- Try to pick your cell phone up with only two fingers.
- Now try flipping the phone over using only those two fingers.
- Notice this is an impossible task.
- You can balance the phone on the edge of the table to be able to regrip it.
- Let go of the phone, flip your arm to the bottom side of the phone and then flip it over.

Similarly, the only way to flip a part is by using a regrip.

It is sometimes possible for a symetrical part to be placed directly into a machine. Symmetrical parts do not have preference in how the part is being placed along that axis. Even in this minor use case, a symmetrical part can require a regrip, for accuracy.

ActiNav has about ±3 mm of accuracy (although there have been much tighter tolerances (±1 mm) for some part types). For this reason, if your final place target requires high accuracy, .005 inches for example, a regrip fixture can allow gravity self center the part to the required tolerance. If you switch to regular UR waypoints, robot repeatability accuracy is increased to 0.03 mm.

4.2.3. Limitations of 3D vision in determining part orientation

3D vision is a bit more challenging to decipher than just a regular 2D image.

The way 3D vision is improved is by telling the 3D sensor what type of part it should be looking for (This is why we use a CAD model).

But even if it knows what the part looks like, parts can lie at different angles and may not have enough point cloud data to guarantee that we have matched to a part perfectly.

This does not always mean that we can't pick the part but it may mean we can't get the "true" part orientation.

With a sensor and some basic logic you can then infer that the orientation you placed it in the Regrip Fixture is either true or false and you can then adjust your regrip motion accordingly.

4.2.4. Incorporating sensors in the fixture design

A fixture design requires sensors when:

1. There needs to be high repeatability of placing the part into the machine.
2. There is not enough data to determine “true” part orientation, due to not having enough features on the part to determine orientation.

4.3. Workcell design

4.3.1. Placing the bin

Angled bins are common for human-centric workstations, but horizontal bins are better suited to ActiNav.

Parts rolling toward the floor of an angled bin resets the perception, and/or path planning queue, increasing cycle time.

It can be difficult to dislodge parts from the bin due to the pressure of other parts leaning against them.

Angled 3D sensors can work, but they have the risk of creating “shadows” (parts of the bin that are not visible to the sensor) and/or low-resolution areas (the native resolution is “smeared” over an angled plane).

The workcell configuration sometimes requires some offset of the sensor relative to the bin, however it should be limited.

4.3.2. Relationship between robot, bin, sensor and standard workcell designs

Vention published three standard ActiNav stand designs that optimize the relationship between the bin, sensor and the robot.

These designs make the entire bin accessible to a robot with most tools. The designs optimize sensor resolution while keeping it away from the robot.

Using the designs are not a requirement, but you can look at them and try to copy the critical dimensions.

Vention also allows you to make a local copy of the design and alter it.

- ActiNav Bin Picking Frame UR5e Medium:
  vention.io/designs/actinav-bin-picking-frame-ur5e-medium
- ActiNav Bin Picking Frame UR10e Medium:
  vention.io/designs/actinav-bin-picking-frame-ur10e-medium
- ActiNav Bin Picking Frame UR10e Large:
  vention.io/designs/actinav-bin-picking-frame-ur10e-large

4.3.3. Ideal bin dimension

Medium Bin (US-container)		Large Bin (US-container)	
Max size	Min size	Max size	Min size
22 x 23.5 x 8.5 inches (559 x 597 x 216 mm)	15 x 21 x 5.5 inches (381 x 533 x 140 mm)	42.5 x 30 x 35 inches (1080 x 762 x 889 mm)	22 x 23.5 x 8.5 inches (559 x 597 x 216 mm)
Medium Bin (Euro-container)		**Large Bin (Euro-container)**	
Max recommended	Min recommended	Max recommended	Min recommended
600 x 400 x 280 mm	600 x 400 x 120 mm	800 x 600 x 600 mm	600 x 400 x 200 mm

You can place the bin so either the width, or the length, is closest to the robot.

You can also leave the bin wide to have enough room for the last 3 joints, plus the tool, to enter the bin.

There shall be enough room for the last 3 joints of the robot to enter the bin.

5. Application installation and configuration

5.1. Networking

AMM

www.universal-robots.com/articles/ur/actinav-manuals-and-cad-files/

Using Flexible IP addressing

Before using the Flexible IP addressing, start to connect using the standard IP addresses below:

- 192.168.0.2 (robot)
- 192.168.0.1. (AMM)

To use Flexible IP Addressing

1. Verify your URcap is correctly installed and your System Status is Green and Ready.
2. Access the next section to change your IP address.

Communication to PLC

www.universal-robots.com/articles/ur/ethernet-ip-guide/

To configure the PLC

1. Download the .eds file to control the program and call other machine signals.
2. Use the UR dashboard server to call the program/s.
3. Use the Ethernet IP to control everything else.

5.2. Aligning the 3D sensor

The 3D sensor requires alignment from the sensor to the robot. You can detach the 3D sensor from the frame to mount it separately and use hand-eye alignment to line up the 3D sensor with the robot.

For the best coordination of 3D sensor and robot arm, use sample points within the volume of the bin to align the 3D sensor. Spread out the sample points to avoid accidentally making a well aligned section.

For the best alignment, when adding the samples, place the alignment marker (hemisphere) at the four lower corners and the upper four corners inside the bin. You can also add a few more samples towards the center area of the bin at varying heights, to improve accuracy.

The image below shows the alignment marker in one corner of a bin.

You can remove the bin from its position before alignment, to articulate the alignment marker to emulate the bin interior, by getting the alignment marker to the bin floor. If the robot arm size, or joint positions prevents this, then you can get the aligment marker as low possible inside the bin and point it toward the sensor.

The image on the left shows the aligment marker in the interior of the bin, pointing toward the sensor. The image on the right shows the alignment marker with the bin removed, pointing toward the sensor.

Using either method, the software is able to match parts on the bin floor.

5.2.1. 3D sensor alignment troubleshooting

Lost alignment can appear as successful validated Picks, starting to drift. Lost alignment can cause the Pick to be too high or to press in too much.

You can use the method below to do a quick, unofficial troubleshoot of the 3D alignment.

To check 3D sensor alignment

1. Freedrive the robot arm above the bin, so the last joint is horizontal.
2. Then, in PolyScope, access the **Installation** tab.
3. Tap **URCap ActiNav > Part Page > Validate Tab**.
4. In the viewer, select **Tool View** and turn on the red point cloud.
5. Tap **Validate** to trigger a scan.

If the alignment is good, the red point cloud matches the robot CAD precisely. The alignment is bad, if the points float slightly above the robot or slightly in it.

A more quantifiable method of checking sensor alignment is to add another sample to the sensor alignment process. If you find the new sample makes the average offset error much greater than the previous one, you can assume the alignment is lost.

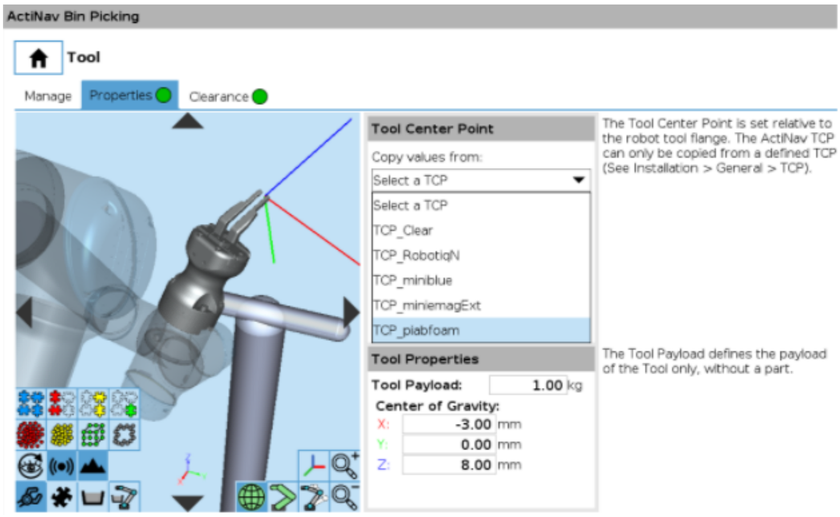
For example: If a UR5e with a medium sized sensor previously had 11 samples, with an offset error of 0.72 mm, and you fix the hemisphere on another sample with a higher offset error (1 mm or even 1.72 mm), then it would be best to redo sensor alignment.

5.3. Teaching the TCP

Everything in ActiNav is based on teach-by-demonstration, so teaching the TCP, accurately, is important. A minor mistake made during teaching the TCP can lead to redoing every step in your Installation, including the Pick Rules and the Place Rules. This section describes how to accurately teach the TCP.

To correctly place the TCP

1. In PolyScope, tap the **Installation** tab.
2. Under **General**, select **TCP** and use Freedrive to teach the TCP.
3. After teaching the TCP, under **Tool TCP**, select the TCP you taught in the drop-down.
4. Verify the TCP orientation is correct.
5. Ensure the positive Z axis points away from the tool.



You can use either method described below to accurately teach a TCP.

To teach the TCP using external CAD software

This metod requires the tool to be drawn and the surface model to be created, to get an X,Y,Z output from CAD software.

1. Type in the numbers of your ideal TCP output from an external CAD software. This is always the most preferred method to guarantee a “perfect” TCP.
2. Enter the correct orientation (positive or negative) for the axes, and verify them.

To teach the TCP using the UR Wizard

This is a viable method, but results can be 1-2 mm off.

1. Start to teach your 4 points by picking the corner of something and teaching your first point.
2. Then rotate with respect to the tool about Rx, Ry and Rz.

   Many TCPs are taught incorrectly because the operator only varied the X, Y, and Z position and not the tool rotation.

If you are using a vacuum cup, make sure the TCP is at the exact point where the cup is collapsed against the part. This helps to adjust a Pick.

To validate the TCP

Having a correct CAD model does not mean the tool is assembled properly. When using CAD, verify the TCP is correct.

1. Check the point cloud to verify the TCP is correct.

   Verification is similar to the 3D sensor alignment (see: 5.2.1. 3D sensor alignment troubleshooting on the previous page).
2. Use the aligned 3D sensor to check the TCP, scan it from multiple angles and adjust accordingly.

For example: If you are using a vacuum tool, you can point the vacuum cup toward the sensor, scan, and check the taught TCP is in the center of the cup with the red point cloud. Turning on the red point cloud shows the overall scan and the points hitting the cup's edge as a ring. The CAD in the viewer is a close representation, unless you change vacuum cups without updating the model.

5.4. Teaching the bin

ActiNav uses 24 points to accurately model the bin, to account for wall thickness and varying bin wall heights. ActiNav cannot use a model with eight points, one point for each corner of a bin's cube shape.

Teaching a bin wall assumes the TCP is touching that bin wall. This is best accomplished by positioning the tool perpendicular to the bin wall.

Once the bin is taught, you can adjust the bin wall thickness. Bin walls are clearance shapes, even though they are not programmed the same way as regular clearance shapes. If you see the robot trying to pick a part from the outside of the bin, as if the wall were not there, the walls are too thin. Make the bin wall a realistic thickness.

5.4.1. To teach the bin

1. Freedrive the robot arm with the tool attached, to teach the position and geometry of the bin.
2. Make sure the TCP touches the inside of each bin wall.
3. Follow the wizard format and move clockwise around the bin, starting with the bin wall closest to your robot.
4. Model four points on the inside of each wall, and add one more point to define the height. Then model four points on the bin floor.

At the default thickness of 10 mm, 2.5 mm of this is inside of the planes you taught, and 7.5 mm is on the outside of the plane.

Any changes to the bin wall thickness are applied outwards from the planes you taught. Adding 2.5 mm accounts for minor inconsistencies between human error and the robot repeatability.

Example: if you taught the TCP with a collapsed bellows cup, you shall collapse it the same way. It is often difficult to do this. A ribbed bin also makes it difficult to know if you are truly perpendicular. This is one of the main reasons why parts below 3-4 mm thickness are unacceptable.

See 5.4.2. Bin clipping below to avoid match extraneous parts.

5.4.2. Bin clipping

Clipping removes extraneous points from the point cloud generated by the 3D sensor. So you clip out anything that is not inside of (or slightly above) the bin, including the bin walls and bin floor.

To clip the bin correctly, perform clipping when zero parts are in the bin.

There is only one way to clip a bin properly. However you can use the method illustrated below for very flat parts (see: 5.4.3. Bin clipping for flat parts on the next page).

To clip the bin

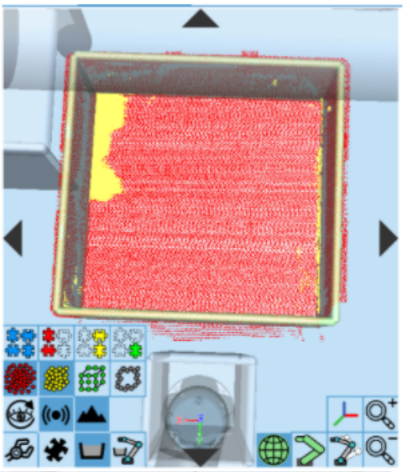
1. Turn on the red point cloud data.
2. Now look at the yellow data. These are the points left after clipping.

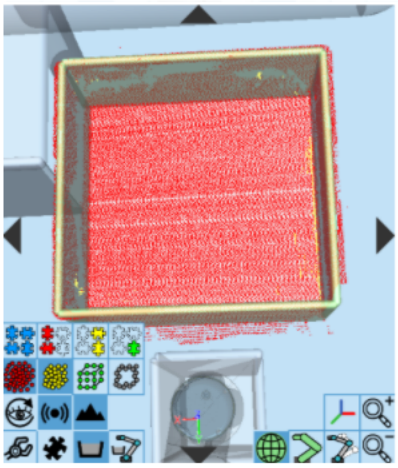
   In an empty bin, there should be no yellow point cloud data.

Failure to clip out extraneous data can cause mismatches during picking. If you see the robot trying to pick the bin itself, you probably need to adjust the clipping.

Similarly, an overclipped point cloud could cause parts on the bin floor, or near the sides, to be missed. Some parts, especially in the bottom corners, may not be picked because the tool cannot reach those spaces. That is not a clipping problem.

The images below illustrate a bin before clipping (left) and a clipped bin (right)





In most cases, where parts do not exceed the top of the bin:

- Set the Top clipping to 0.
- Clip in from your bin walls so that you cannot see any yellow point cloud data.
- Then do the same for the bin floor.
- If you see any small spots of yellow data, continue clipping until you can't see them.

If you are seeing small spots of yellow data floating around your bin then this is likely due to noise (especially with shiny, new bins) and you can ignore this.

5.4.3. Bin clipping for flat parts

With very flat parts, regular bin clipping can remove the point cloud for the parts on the bin floor. In such a case, you can adjust how you clip the point cloud and reteach the bin floor. To do this, you must set a “New” TCP to your tool.

To set a new TCP

Before you tap Play, you must return the "New"tool TCP to the original TCP you taught.

1. Teach the “New” TCP 20 mm straight out from the tool in the Z direction.
2. Then reteach the bin floor, so it now appears 20 mm lower than it actually is.
3. You can now revert to the original Tool TCP and begin clipping up from the bin.

Although ActiNav adjusts your Picks automatically, you must use the original TCP to avoid collision with the part and failed Picks.

Setting a new TCP is recommended due to how the clearance shapes on the bin are created. Clearance shapes on the bin are created by drawing planes when teaching. Since planes have thickness, it is impossible to get a perfect plane with four points, with the plane appearing wavy. By teaching the bin floor much lower and then clipping up much higher, you can reduce some of the waviness. So the four points are estimated to then make the planes perpendicular.

Minor changes in clipping the bin floor make little difference with normal volumetric parts. However, clipping the bin floor does have a big impact on very flat parts.

5.5. Clearance shapes

Clearance shapes are simplified shapes that surround the robot arm, bin, part, environment, and the tool. Clearance shapes are generated automatically for the robot and the bin (as part of teaching the bin). You create clearance shapes for the part, environment, and the tool. Collisions among clearance shapes are usually not permitted. There are limited exceptions, such as when the tool comes into contact with the Part during a Pick.

In ActiNav, you can check clearance shapes by moving the tool around and checking the ActiNav Viewer. Turn on the TCP triad to see whether the clearance shapes on the Viewer match the physical parts.

5.5.1. Part clearance shapes

As shown in the Clearance shape collision rules table (see: 5.5.5. Clearance shape collision rules on page 41), clearance shapes are used to avoid collisions with the environment, and with the tool after placement. So these shapes can be crude. Part clearance shapes are not used to avoid collisions with adjacent parts, that is done by the Terrain (see: 5.5.4. Terrain on the next page).

5.5.2. Environment clearance shapes

Environment clearance shapes can generally be quite crude. They should be kept very simple to increase the efficiency of path planning. Environment clearance shapes may also be used to create exclusion zones, to limit robot movement in undesirable areas. Creating exclusion zones can constrain robot motion too much, which can make path planning difficult.

You can also use environment clearance shapes to constrain robot poses during part placement.

For example, a cylindrical part may be pickable anywhere around its circumference, but if you place it on a flat table, the robot can take any valid pose, to place it. If this interferes with placing the part, some clearance shapes around the place location (for example, a lozenge on either side to form a wall) can prevent the robot from assuming certain poses.

If you draw a Clearance Shape on top of the safety plane, the robot does not cross that plane during ActiNav movements.

Environment clearance shapes are not safety certified. If you need certified safety, use the UR safety planes in PolyScope.

5.5.3. Tool clearance shapes

Tool clearance shapes are static, but tools can move.

Tool clearance shapes can usually be based off a previous version of the tool, such as tools with different vacuum cups or fingers. Vacuum cups can compress, tool fingers can move. Failure to correctly base the tool clearance shape can cause collisions with the bin (too small), or make the part unplaceable (clearance shape extends beyond the part after picking)

Drawing tool clearance shapes

Tool clearance shapes shall be drawn very tightly, since the tool manoeuvers close to the bin walls, near the terrain, and inside of a placement fixture.

- For vacuum cups with bellows, draw the clearance shape more in line with the collapsed cup than the expanded one. If this leads to collisions, make the environment and the bin clearance shapes larger.
- For grippers with fingers, draw the clearance shape for the finger position when the part is being held.
- You may need to make subtle adjustments to the final place environment shapes or to your tool clearance shapes. You can have your TCP just outside your last clearance shape on your tool, around 1-2 mm. This is so that when you teach the final place you are much less likely to have a collision between the Tool and the final Place location.

Bin clearance shapes

Teaching the bin and setting the wall thickness generates bin clearance shapes. You can observe bin clearance shapes in the ActiNav Viewer. Bin clearance shapes are usually larger sized than other clearance shapes.

To alter a bin clearance shape beyond the wall thickness, you can use the following methods to teach the bin inaccurately.

- Teach a smaller bin volume, or teach it higher, than the actual bin.
- Or teach the bin using a temporary TCP that is further out from the tool, to push out the floor.

See 5.4.3. Bin clipping for flat parts on page 38.

5.5.4. Terrain

The Terrain in not a clearance shape. The Terrain is a dynamically created shape that controls some collision avoidance (see:6.2. Understanding the Bin Picking loop on page 47).

5.5.5. Clearance shape collision rules

This table shows when collisions are allowed.

	Environment	Bin	Robot	Tool	Part
Robot	Never	Never			
Tool	Never	Never			
Part	• Pre-Pick to Post-Pick • Pre-Place to Place	Pre-Pick to Post-Pick	Never	Pre-Pick to Place	
Terrain			Never	Only when very near the part during pre-pick to pick	Pre-Pick to Post-Pick

5.6. Perception settings

In this section, any reference to time refers to the time it takes to match a part.

Processing starts with triggering a scan and acquiring the point cloud data, and always takes 1.5 seconds to 3 seconds to acquire the image. Processing normally occurs in parallel with other robot activity. It is not noticeable.

Matching occurs when the green point cloud overlay of your CAD model fits the yellow point cloud. This overlay represents the Confidence Score.

The yellow point cloud represents the remaining data after you clip away the excess data from the red (raw) point cloud data.

5.6.1. Volume detection

Normally, when people think of how to solve the vision problem it tends to be very challenging and you need to be a vision expert.

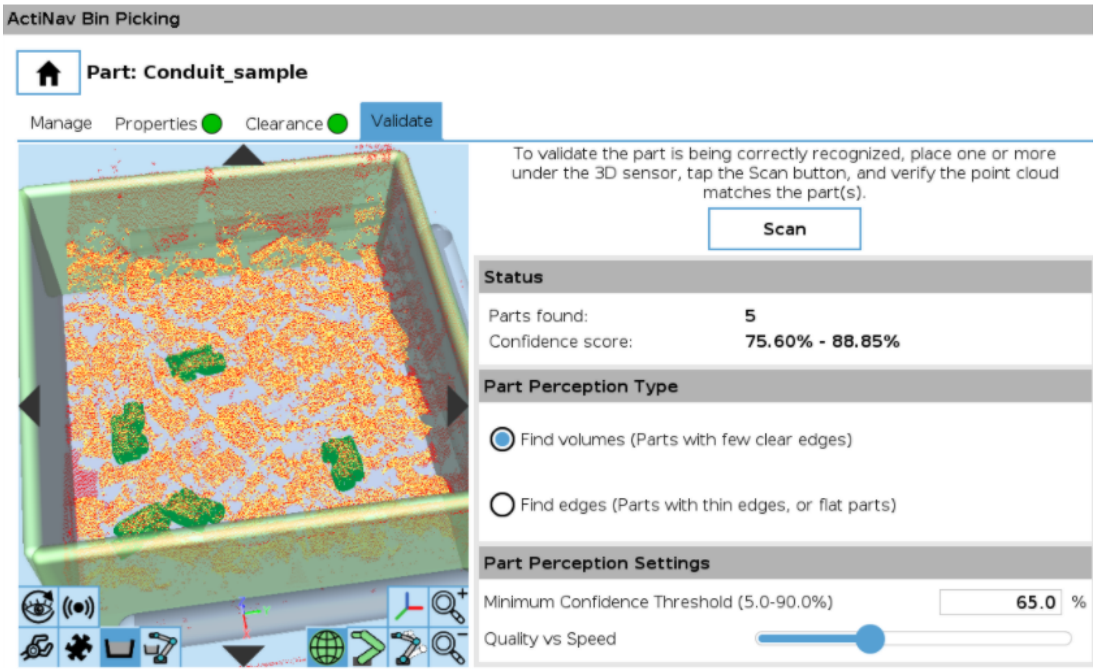
ActiNav takes a lot of the advanced work out of the equation for you. As you follow the online tutorials you should be able to follow the processes of matching to 5 parts.

To match parts:

1. Place the parts in your bin and hit scan.
2. Then rotate the parts 90 degrees and repeat the processes.
3. Select a confidence score about 5-10 % less than the lowest reported confidence score.

Defining features on the part is a huge advantage to being able to match to a part in its correct orientation. Parts with very few defining features can make matching difficult.

ActiNav matching works by matching to a 3D CAD model. The system is intuitively recognizes parts, unless insufficient data reaches the 3D sensor for a particular part. This may be due to issues with pose quality relative to speed.



Where things can get a little tricky is the quality vs speed slider. The way the quality vs speed slider works is based on the resolution of the match you would like to take. A great way to think about this is thinking of sending a laser grid across your computer keyboard.

Say you send light down to the keyboard and back and you are getting 5 lines per key.

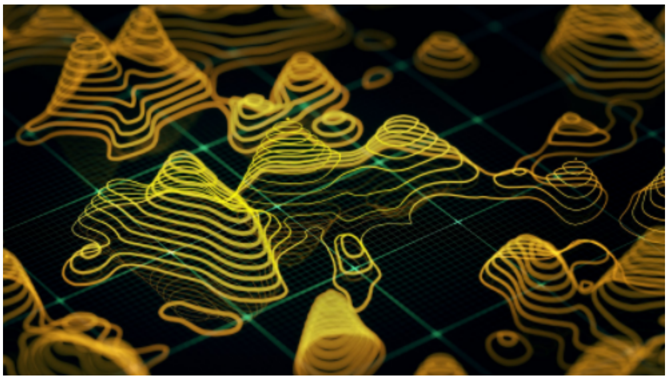
- By increasing the quality, instead of only getting 5 lines per key you actually get 15 lines per key.
- By leaning towards speed you only get 1 line per key.
- By having more lines per key you slow down how long it takes to match but you get a higher resolution.
  Resolution is not related to the size of the part, but rather to the smallest salient feature of a part.

For example: If a 100 x 100 mm part has a 5 mm protrusion that is key to determining the part pose, you will want high quality. If the protrusion is irrelevant to determining the part pose, high speed may be OK. (Note that if small features are important, they must not only be resolvable, but must also be visible in the bin. If a feature is hidden by another part half the time, or is on the other side of a flat part, it can't be resolved no matter what you set the slider to).

5.6.2. Edge detection

Edge Detection uses additional time to process a part into ActiNav (5-15 minutes, the first time). Edge Detection allows a part to be matched using only the edges of the part. However a part may have few defining features, so volumetric matching will likely fail to return correct matches. Edge Detection allows the matching algorithm to rely not only on defining features but also on 3D edges.

A great way to think about this is a topographical map, illustrated below, stitching the lines together to create a 3D edge profile of the part.



Ratio of Edge Thickness to Diameter

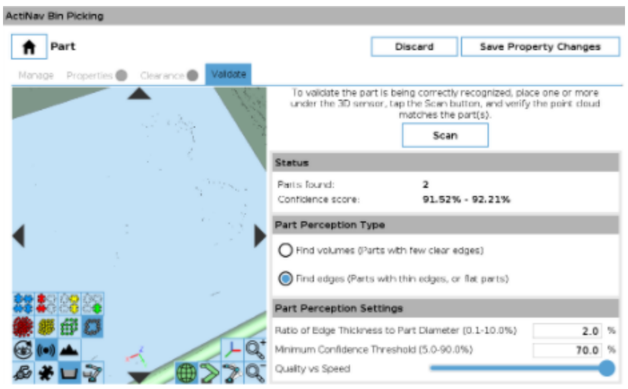
The Ratio of Edge Thickness to Part Diameter helps define what is an edge and what is not an edge. It is an approximation.

You can increase the speed slider all the way to the right as the resolution of the match is handled by the Ratio of Edge Thickness to Part Diameter parameter.

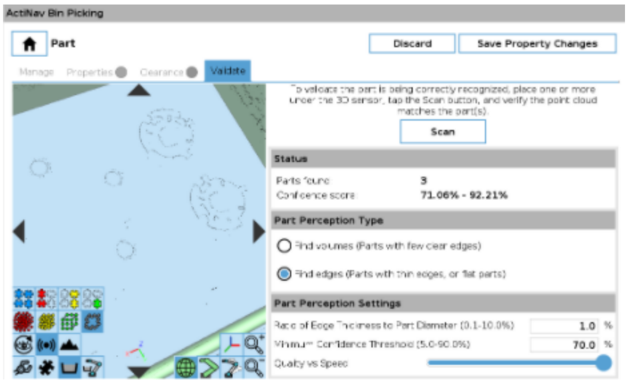
To see the edges of your part, turn off all of the other vision parameters except the gray point cloud data. The gray point cloud data appears once you adjust the Ratio of Edge Thickness to Part Diameter.

The images below track the progression of how many 3D edges are detected on the part as the Ratio of Edge Thickness to Part Diameter parameter decreases.

2%



1%



0.1%

This progression shows more of the contours of the part you are picking, as the percentage decreases.

- If the ratio is set too high, some edges are not be detected and will likely reduce matching accuracy.
- If the ratio is set too low, false edges are detected, likely due to noise in the scan that can also reduce matching accuracy.

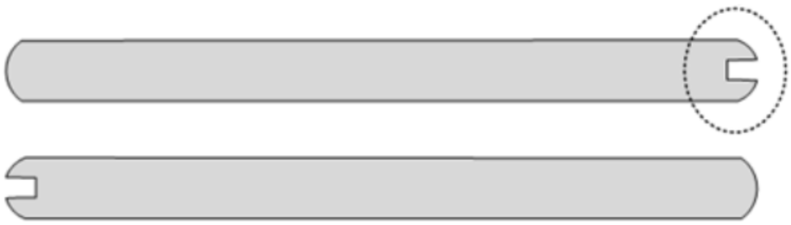
The ratio is set to allow enough "good" edges to be detected while limiting the number of false edges.

At this point you can turn on the green point cloud data to see how well you have matched to the part. The matchis good if green point cloud data overlaps the edges of the part. If there is no overlap, you can increase the confidence threshold.

5.6.3. Complex part detection

Complex parts are relatively easy to match because they many defining features (potential pick points). Simple parts can have so few defining features that having one of them obscured can cause the part to be matched incorrectly.

The example below shows a simple part. Covering the notch with another part, or having insufficient sensor resolution to find the notch, means the part orientation cannot be determined. If both ends are obscured there is no way to determine what kind of pick point the sensor detects. Both off-center picks and incorrectly flipped picks (180 degrees rotated about the axis of non-symmetry) could become common.

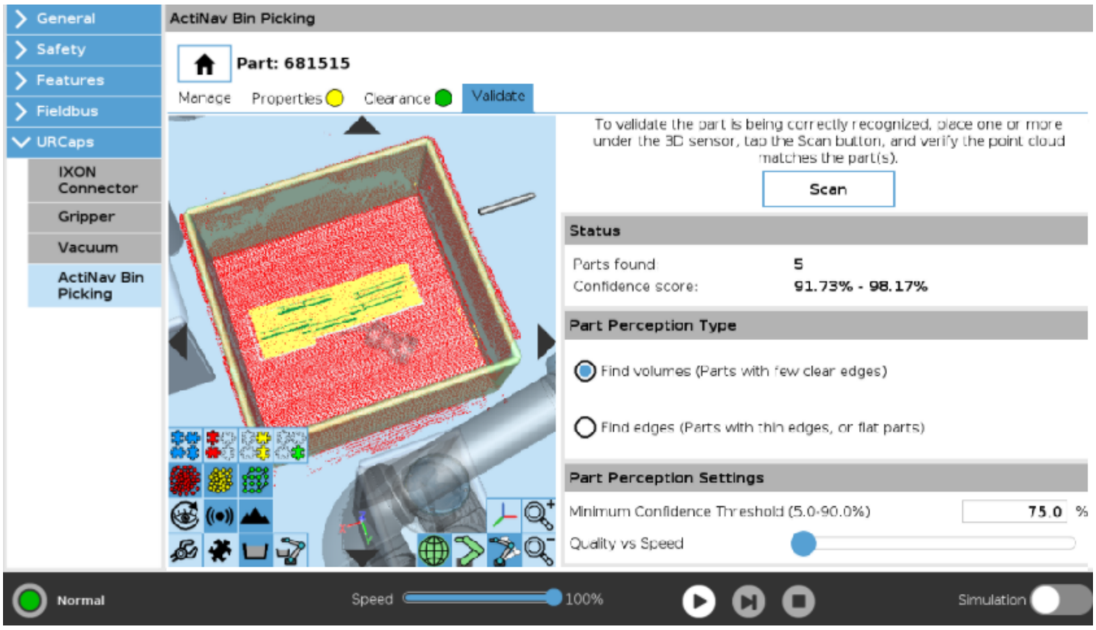


Other difficult Parts to discern orientation are simple part models such as blocks, cubes, and rods with no defining features. This problem increases if the simple Parts are stacked. The issue here is the only defining features on the parts are their ends. So it can be a challenge to distinguish one Part from another. The following image illustrates a “false confidence” when matching to a simple rod.

ActiNav gets a very high confidence score and matches a lot of points, but the match is incorrect.

A high confidence score only means that a high percentage of the part template's exposed surface area is matched to corresponding points in the scanned point cloud.

If a part has very few identifying features, this can be a mismatch or an inaccurate pose. Pay attention to the overlapping of the yellow point cloud to the green point cloud. The two point clouds should match up perfectly, if they do not then you may be getting a false part match.



The degree of shine on a Part can also impact the accuracy of the point cloud data, because too much light is reflected away from the 3D sensor. A shiny Part can reflect the light against bin walls causing noise, which makes matching difficult.

You can still try to use current settings for a shiny part. If the part is not matched, contact your local UR Sales Rep or Distributor.

6. Application programming

6.1. Understanding ActiNav path planning

Path planning is the computational problem to find a sequence of valid configurations that moves:

1. The tool to the object
2. The object from the source to destination
3. The tool back to the next object

You can solve the path planning problem by constraining the robot, and defining:

- A Bin Picking loop
- Pick Rules and Place Rules
- Clearance shapes for the robot to avoid

6.1.1. Clearance shapes in path planning

Clearance shapes allow the AMM to prevent contact among the ActiNav elements, by allowing you to draw protective "bubbles" around the bin, the tool, the environment, the robot and the parts.

There are some exceptions:

- You cannot place a part with a clearance shape on a fixture.
- You cannot pick a part with a clearance shape, because the clearance shape prevents collision.

To learn about clearance shapes, see: 5.5. Clearance shapes on page 39 and Tech Note: www.universal-robots.com/articles/ur/actinav-clearance-shapes-creation-and-use/

To learn how to teach the robot, pick and place rules, access the online training videos: academy.universal-robots.com/video-tutorials/ and select ActiNav.

6.2. Understanding the Bin Picking loop

This section explains what occurs on the backend of the software.

From the Home Position, ActiNav automatically triggers the first scan to establish a point cloud of different types of data, separated as follows:

- Raw (red): The detected points, including non-useful points that can be part of, or outside of, the bin.
- Clipped (yellow): The clipped version of the raw point cloud. The perception algorithm looks for parts within the clipped cloud.

- Matched (green): The parts matched by the perception algorithm. For computational efficiency, this can only be a subset of the possible matches.

Establishing a point cloud lets you match to a small rectangle. After one scan the rectangle continuously moves around the bin, allowing 10 potential parts, at a time, to be matched. Matching is based on the highest piles of parts, which have more exposed pickable parts.

Once the parts are matched, you can create the terrain by “shrink wrapping” the rest of the scan.

6.2.1. Terrain

The terrain is generated by the parts left in the bin. The terrain appears as an obstacle to the AMM as it tries to find a way to pick a part while avoiding collisions.

During a Pick, the tool can collide with a part. However, the tool cannot collide with the terrain beyond a small area around the part.

For example, reflective parts can cause a “noisy” terrain that does not match the content of the bin. Noisy terrain can prevent the AMM from finding a valid path to pick a part.

You cannot use part clearance shapes to create the terrain.

6.2.2. Recognising Pick and Place options

Once the condition to make a part pickable is met, you can enable the robot to recognize a viable Pick and Place option. Viable Pick and Place options break down into the Pick Rules and the Place Rules described below.

The Pick Rule defines how to pick the part by configuring the Do Pick node.

The Place Rule defines how to place the part by configuring the Do Place node.

The table below illustrates the rules and node types.

Pick Rules		Place Rules	
Do Pick node		Do Place node	
	Pre-Pick		Pre-Place
	Pick		Place
	Post-Pick		Post-Place

The Do Pick node and the Do Place node describe tool actions inside the Bin Picking loop.

The Do Pick node and the Do Place node are separate from each other, so you can create multiple Pick Rules with only one Place Rule.

6.2.3. Part and tool symmetry

This section describes the main ActiNav features needed to understand Pick Rules and Place rules.

- **Free Spin in tool Z**: This is an unrestricted rotation about the tool's Z axis. Free Spin in tool Z relaxes the constraints in the last joint of the robot, so all pick options surrounding the Pick Point become valid.

  A valid Pick Point can still have failures at another point in the Pick Rules. For example, in the Pre-Pick and/or in the Post-Pick.

  The Free Spin in tool Z only works if the tool is symmetrical about that Pick Point. Common tools using this feature are circular vacuum cups and circular magnets.

- **Frame Pick**

  Frame Pick is the opposite of the Free Spin in tool Z. The Frame Pick lines up the TCP to the exact rotation about the part you have specified in the Pick Rule. Teach a Frame Pick means teaching the same Pick Point rotated about the Z axis by 180 degrees to make the same grasp viable.

- **Axis of Symmetry (about the part)**:

  The Axis of Symmetry describes the same Pick Point on a part, rotated about a specified axis to be considered valid. The Axis of Symmetry requires “Perfect Part Origin”, which is particularly useful if the orientation of the part is not important. For example, cylindrical parts are axially symmentrical. This means only the left and the right of the part matter, but the cylinder is still valid if it is rotated about the long axis.

Except for Post-Place, the part is always described in terms of how to Pick and how to Place. This is quite different from describing the robot's pose. When you specify picks and places, focus on the part, not the robot.

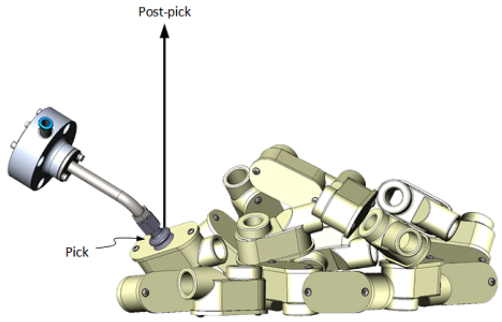
For more details abou the ActiNav Pick and Place Rules see: 6.2. Understanding the Bin Picking loop on page 47.

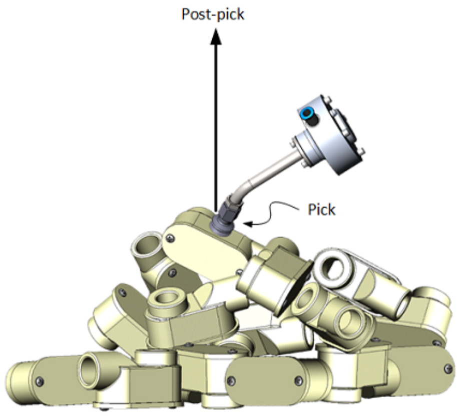
6.2.4. Pick rules

This section describes how to configure Pick Rules. Set up the Pre-Pick before configuring the Pick Rule, and the Post-Pick after.

Setting up a Pick Rule, with Pre-Pick and Post-Pick

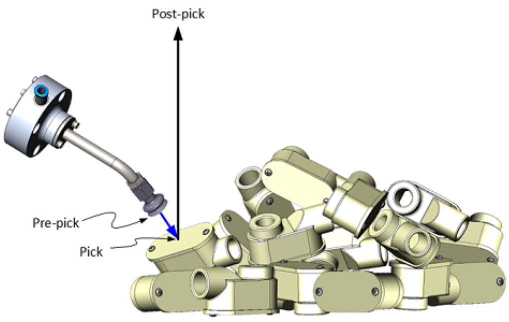
- The Pre-Pick offset shall be between 30 mm and 50 mm to allow enough room for the tool to hover just above the part, without getting too close to the terrain.
- The Pre-Pick position shall have sufficient clearance from tool to part clearance shape, and from tool to terrain.
- Collision checking among the part, terrain, and tool is disabled during the move between Pre-Pick and Pick.
- Consider Free Spin and/or Axis of Symmetry about the Part. The Part becomes a part of the tool, at the exact moment of “collision”. ActiNav automatically accounts for part size within path planning.
- For the Post-Pick, the lift shall be in the positive Z direction (robot base coordinates) so it does not retract to the Pre-Pick location.
- Your Post-Pick offset should generally be the same height as the bin wall height. Although this is not a requirement, it simplifies path planning and takes out an extra set of calculations to manoeuvre out of the bin.

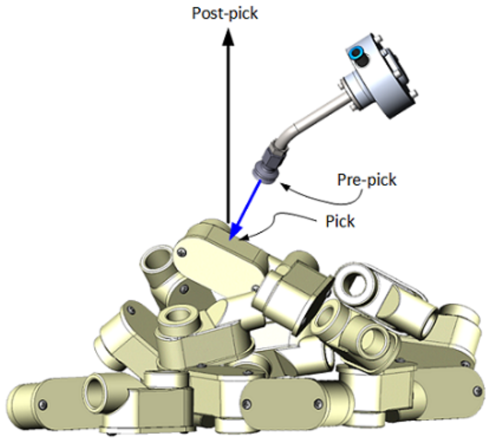




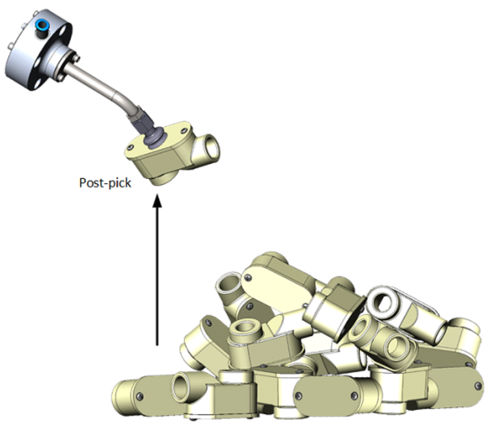
At every section of the Pick Rule the robot avoids anything inside the defined environment, bin, terrain etc. other than the part you are picking.

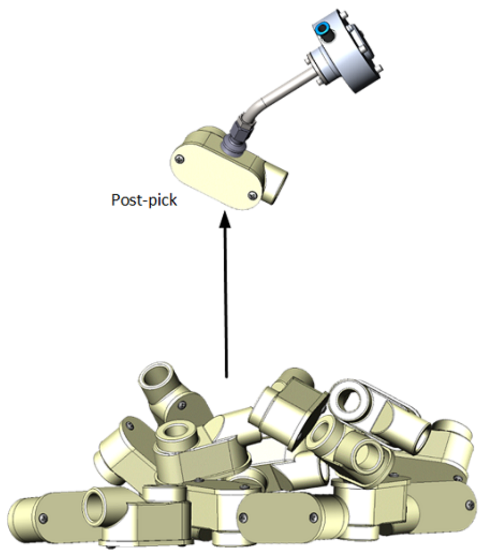
To set up a Pick Rule





1. Set the Pre-Pick offset to specify a linear move (relative to the part) from Pre-Pick to Pick.
2. Verify there are no collisions between the Tool and the part to be picked.
3. Describe exactly where and how to pick the part.
4. Then describe where you would like the Tool TCP to interact with the part.
5. Now set-up the Post Pick offset to tell the robot how far to lift the part linearly out of the bin.



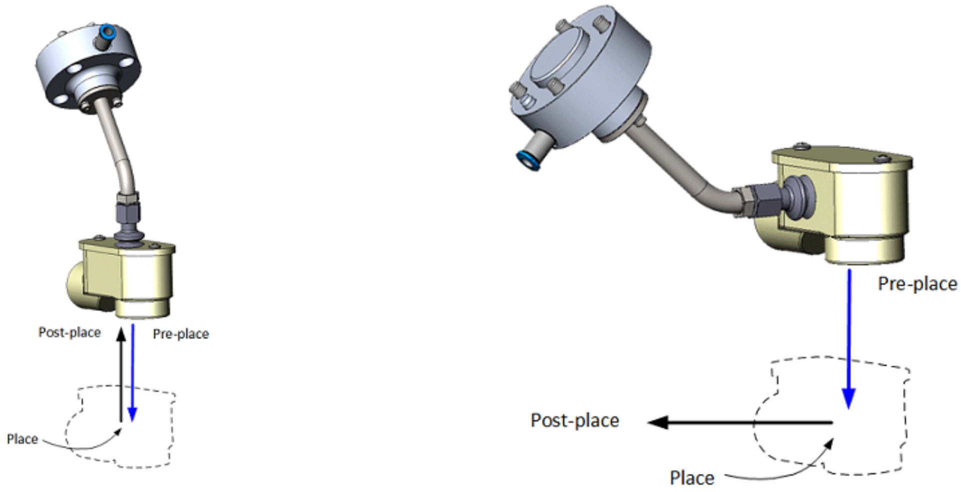


The part is permitted to collide with the bin any time from Pick to Post-Pick. This allows parts very close to the bin wall to be picked.

If the Post-Pick offset is large and your bin has an overhang the top, the part can collide with it. In such a case, you can try padding the overhang to make it even.

6.2.5. Place rules

This section describes how to configure Place Rules. Pre-Place is a location near to the part's final Place location. After you describe how to pick the part and how the part exits the bin (see: 6.2.4. Pick rules on page 50), you can describe the Place Rule. Set up the Pre-Place before configuring the Place Rule, and the Post-Place after.



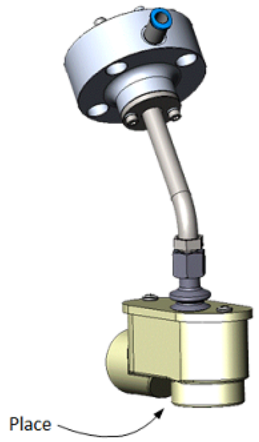
Setting up a Place Rule, with Pre-Place and Post-Place

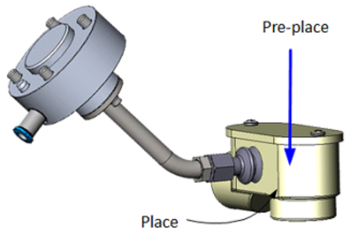
- At first, the robot avoids everything inside the environment to get to the Pre-Place location. No collisions are allowed.
- Applying ActiNav Bin Picking rule two allows collisions between the Part Clearance Shapes and the Environment Clearance Shapes from Pre-Place to Place.
- You cannot allow collisions between Tool Clearance Shapes and Environment Clearance Shapes.
- Make sure the Environment Clearance Shapes are thouroughly modeled for the place fixture, to avoid path planning failure.
- The robot places the part according to the current Place Rule. This is why the Do Pick node is separate from the Do Place node.

  So if the robot picks the part from the top, the part is placed from the top. If the robot picks the part from the right side, then the part is placed sideways, so the orientation matches the part place pose.
- If ActiNav does not validate any paths to the place location, check both the Tool Clearance Shapes and Environment Clearance Shapes. Move the tool to the Place location and look for collisions on the ActiNav Viewer.
- If you have Free Spin enabled, or multiple Pick Rules with the same Place Rule, there could be several tool poses for the same part placement. Check all of the poses. Not all poses have

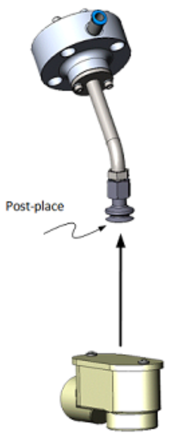
to be possible for ActiNav to work (for example, only 180° of Free Spin is possible on placement, whereas 360° of Free Spins is possible on picking), but this may slow down path planning.

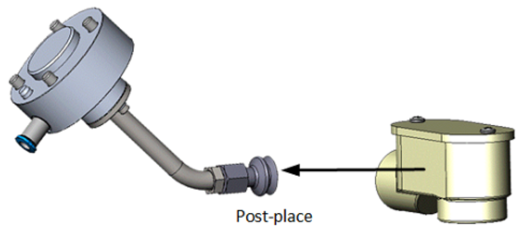
To set up Place rules





1. Verify there are no collisions between the Tool and the desired Part.
2. Set-up Pre-Place. For the final Place location, you can allow interaction between the part and the environment.
3. Describe the exact orientation in which to place the Part.
4. For the Post-Place Rule, describe how the tool moves relative to the part. This does not include describing the desired part pose.





In most cases the Post Place Rule is a negative Z retraction in tool coordinates. When configuring the Place Rule, you can use the option to retract in just tool Z. After the retraction, the robot finishes the loop by taking a collision-free path back to the Home Position.

6.2.6. Path Planning

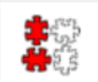
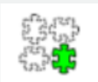
Path Planning allows collision-free movements, unless rule one or rule two are in effect. See 5.5. Clearance shapes on page 39.

ActiNav looks for a valid Pick and Place path for the robot. Pick and Place are considered valid when the robot executes all the actions, listed below, without colliding with anything of the elements making up the environment.

- Enter the bin
- Pick the part
- Exit the bin with the part
- Move to a pre-place near the final place
- Place the part
- Retract from tool Z

How Path Planning works

ActiNav continously works in the background to validate matched parts, but the robot arm only moves when a part with validated path is matched.

Path planning submenu		
	Part matched, looking for path	Before starting a path plan, when ActiNav finds a part, the part is represented as a blue puzzle piece. You can plan as many as paths as required to keep matching parts, and to be added to the queue for path planning.
	Part matched, no path found	In some cases, a part is matched but cannot find a valid path to execute, due to an error in validation. The part is represented as a red puzzle piece. The part is ignored.
	Part matched, path planning in progress	You can see the paths being calculated by turning on the yellow puzzle pieces.
	Part matched, path is in the queue for picking and placing.	You can see the paths being executed by turning on the green puzzle ieces.

6.3. Advanced program tips

6.3.1. Home Position, obstacles, and unexpected motion

The ideal Home Position resembles the pose of lamp in the image on the right.
In the Home Position:

- The robot arm shall face the direction intended for placing the part.
- The robot's joint angles shall be relatively centered on the rotation scale.

Setting this "ideal" Home Position prevents the robot arm from swinging over the bin and disrupting subsequent scans.

If the robot arm swings almost 270 degrees around its base, to return to the Home Position, you can make the following corrections:

1. Correct the joint angles.
2. Flip the robot's shoulder.
3. Then flip the robot arm over the elbow.

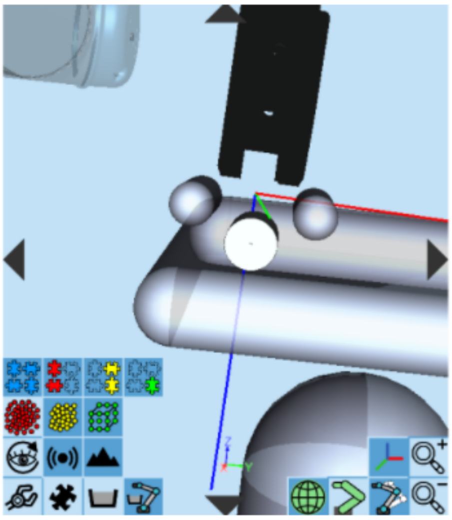
When the correction is made, the robot moves freely through the environment.

ActiNav uses all the space, not covered by clearance shapes, to design different paths for the robot to move through. This use of unoccupied space, results in the robot passing near environment elements. It is part of the advanced motion control that ActiNav provides. See 2.5. Integration-specific safety precautions on page 13.

ActiNav programming can make the robot arm appear to move unpredictably. The environment is user-defined, so ActiNav is always limited by the instructions in the program and a properly defined environment. See 2.3. ActiNav-specific safety precautions on page 12 .

6.3.2. Artificial clearance shapes

You can use artificial clearnce shapes to force path planning to pick or place a part in a desired position. Path planning often requires adjustment to execute the desired motion. Making the adjustments constrains the motion of the robot and technically slows down the path planning process.



In the example illustrated on the left, you can have the part placed on the table, with the tool moving through the easiest path. So instead on being placed vertically, the part can be placed less than 180 degrees when using axis of symmetry.
If you want path planning to drop the part vertically instead of at different angles, you can add artificial clearance shapes around the final place target.

6.3.3. Do Custom Pose

The Do Custom Pose command is a simple way to teach the robot to exit the Bin Picking Loop and switch to using Polyscope waypoints. The Do Custom Pose starts the transfer to exit the Bin Picking Loop, allowing you to use the waypoints and conditions in PolyScope. The last waypoint you teach in

PolyScope, before returning to the Bin Picking Loop, shall be at the same robot pose as the original Do Custom Pose. This is to accommodate smooth switching between PolyScope waypoints and the Bin Picking Loop.

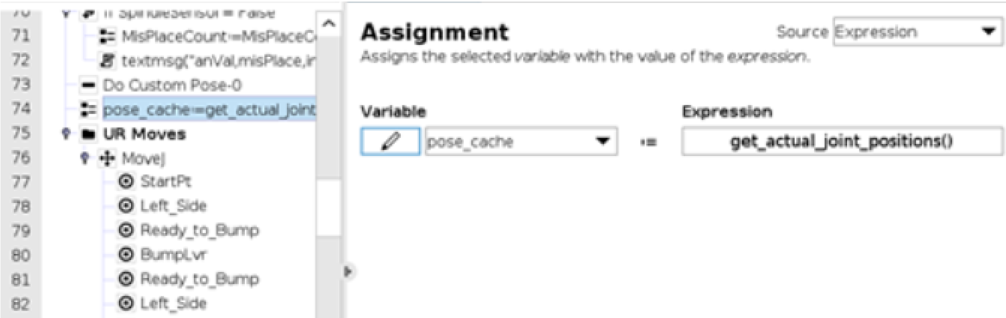
PolyScope waypoint moves are still susceptible to collision, so avoid obstacles when pre-defining the robot path.

To use Do Custom Pose

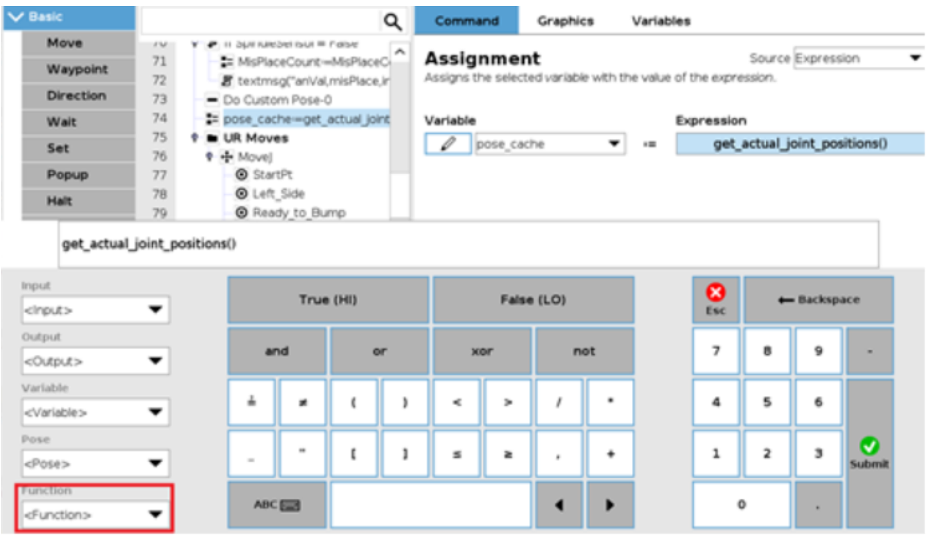
1. Under the **Do Custom Pose** node, create an **Assignment** and name the new variable **pose_cache"**.

   The new variable allows the program to support any additional changes and prevents any unwanted adjustments.

2. In the Expression dropdown, set the variable as equal to **get_actual_joint_positions()**.



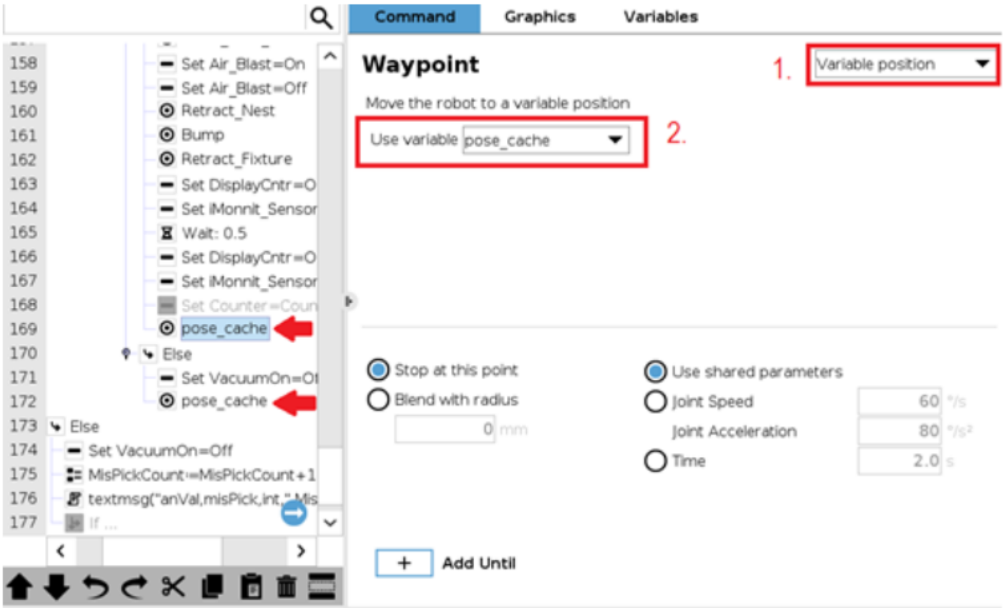
3. Navigate to the end of the regular waypoint moves and add a new waypoint to transfer motion back to ActiNav.



   If you have multiple Polyscope waypoint sections, that can hand motion back to ActiNav, the final waypoint appears as shown in the image above.

4. For the new waypoint:

- Change the Position type to **Variable position**.
- In the Use variable drop-down, select **pose_cache**.



Adding a collision-free move

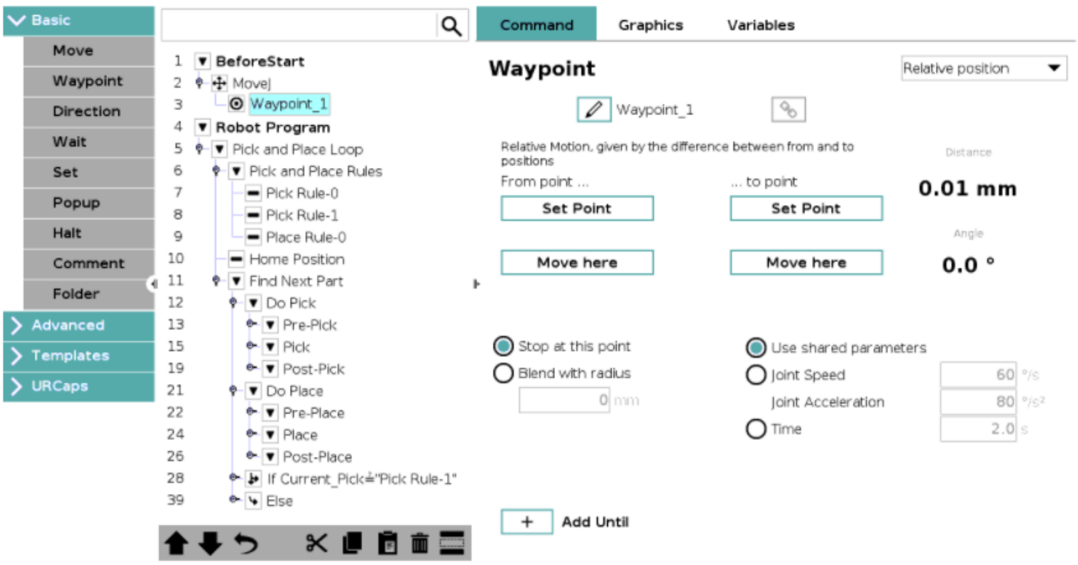
A program that includes PolyScope waypoints must start at the first waypoint before the program plays. The collision-free move allows you to skip the first waypoint and maintain a clear path to the Home Position.

To add a collision-free move

1. Create a **BeforeStart** program tree and add in a MoveJ waypoint.
2. In the drop down, select **Relative position**.
3. Set the waypoint anywhere in space and set the **From point… to point** in the same position.

   The distance between them should be >.02 mm.

This now always give a collision-free move from any point to the Home Position.



6.3.4. Technical notes and resources

Native palletizing vs PolyScope palletizing

To find out more about when to use ActiNav palletizing and Polyscope palletizing, see:
www.universal-robots.com/articles/ur/palletizing-with-actinav/

Threading, Sub Programs, Conditional Logic

To find out more about ActiNav functional rules and limitations in the PolyScope environment, see:
7. Annex I: Functional limitations and workarounds on page 66

Current Pick Variable for changing place location based on the current Pick

To find out more about using currently selected Pick Rules as Conditional Expressions, see:
www.universal-robots.com/articles/ur/actinav/actinav-using-the-currently-selected-pick-rule-as-a-conditional-expression/

Variable tool actions based on current Pick Variables

To find out more about the different tool actions based on Pick type, see:
www.universal-robots.com/articles/ur/actinav/actinav-how-to-select-different-tool-actions-based-on-pick-type/

6.3.5. Force Monitored Picking

External grasps

External grasps monitor the force applied by the tool from Pre-Pick to Pick. This monitoring minimizes the number of protective stops that come from picking the part too deeply.

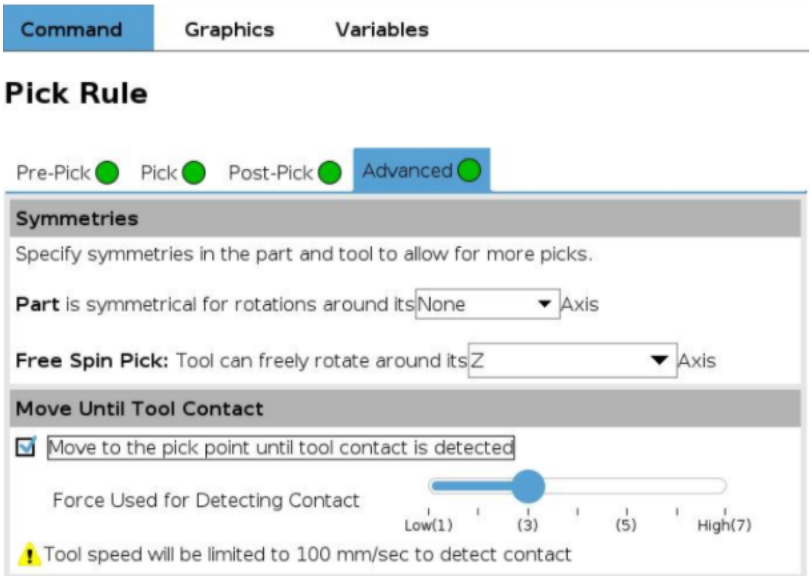
External grasping in picking

ActiNav offers two features to facilitate external grasping of parts: Force Monitored Picking and Tool Space for External Grasps. Force Monitored Picking applies to tools using External Grasps, vacuum tools and magnetic tools.

To enable Force Monitored Picking:

1. Under **Advanced**, access the section in the Pick Rule node.
2. Check the **Move until Tool Contact** box to begin testing how much force is required to apply from low to high. There is some variation from part to part and from tool to tool so the desired range is from 1-7.

When external grasping is enabled, you cannot have a Wait command in your program at Pick.



Tool space for external grasps

This feature is not available for vacuum tools or magnetic tools. Once the monitoring force at pick is enabled, Tool space for external grasps allows the tool to collide with the terrain by opening up the space around the part in the point cloud data so your tool can fit. This causes collisions with adjacent part/s. Enable Until Tool Contact to avoid getting protective stops.

To enable Until Tool Contact:

1. In the Installation, select the new **Advanced** tab to set up your tool.
2. In the **Advanced** tab, check the box to allow the tool to be used for External Grasps.
3. Once you enable External Grasps, set the distance slider.

You can measure the thickness of one 1 of your fingers, then set the distance slider to a measurement larger your finger. This allows enough space for your fingers to fit around your part.

6.3.6. Grip check and validate using Robotiq Hand-E

The conditional logic in Robotiq's Grip Check does not comply with the rules of ActiNav's conditional logic, as Grip Check is continuously checking. The work around for this, is to apply a conditional so the checking stops being continuous.

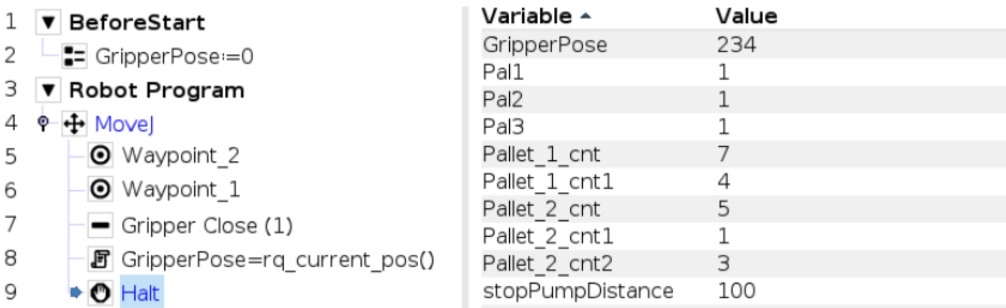
Using a Grip Check to validate the finger position ensures you:

- Got the part.
- Got the correct grasp.
- Did not accidentally pick up 2 parts.

The first thing you need to do is test the stroke of all of your grasps you would like to apply. You are looking for the range of values.To test:

1. Open a blank program.
2. Add in an assignment variable and label it **GripperPose**.
3. Set the value to 0.
4. Program a simple Home Position to the pick with your part on the table.
5. Once you have the value ranges you can open your Bin Picking program.

   In the image below, waypoint 1 is the Home Position and waypoint 2 is the Pick.



6. Grip the part using either a grip close or open, depending on the grasp you would like to test.
7. Write in the script command: `GripperPose=rq_current_pos()` and Halt your program.

8. Now place your part under the robot and play the program to the desired grasp.

   You may need to alter the gripper position depending on which grasp you are trying to get values for.

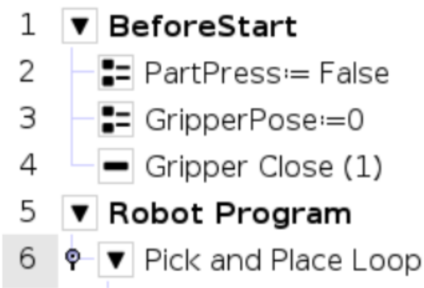
9. Once the robot grips the part variables, check the **GripperPose** value.

   Robotiq should output a value between 0 and 255.

Repeat this process for each of the grasps you would like to check and write these ranges down.

You shall add in the **GripperPose** variable as before and set the value to 0. You shall also create another variable, call it **PartPress** and set that value to false.

Now you can access the program and write down the 3 subset conditional.



The idea is to check if the part is in the correct gripper position. You can test this grasp at Post-Pick to ensure there is no interference from other parts inside the bin.

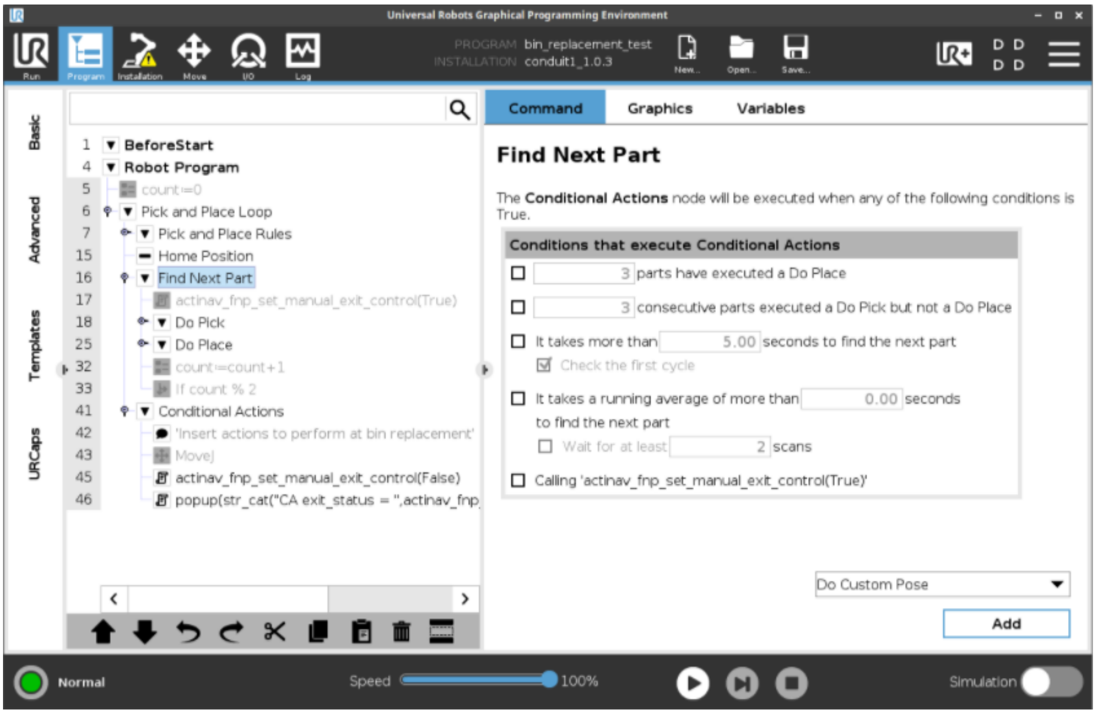
The conditional is as follows. If the current pick is either ID or OD, check the gripper position. State if the gripper position is within a certain range (You gathered these values in the above test), then check did we actually detect a part. State a Grip Check within the Gripper Pose conditional if a part is, or isn't, detected. If a part is detected, then Part Press is True, else Part Press is False.

```
Post-Pick
  'Insert actions to perform at post-pick'
  GripperPose=rq_current_pos()
  If Current_Pick≟"Top" or Current_Pick≟"Bottom"
    If GripperPose<200 and GripperPose>180
      Grip Check
        If object detected
          PartPress≔ True
        Else
          PartPress≔ False
  If Current_Pick≟"Side" or Current_Pick≟"Side2"
    If GripperPose>150 and GripperPose<180
      Grip Check
        If object detected
          PartPress≔ True
        Else
          PartPress≔ False
  Wait: 0.25
If PartPress≟ True
  Do Place
```

After writing down the 3 sub conditionals, wait .25 seconds to make sure you did not miss the variable output. Variable outputs can be instantaneous and you want to make sure that they settle.

Finally you can jump into the mis pick conditional. If Part Press is True, then you can do your Do Place, Else Open/Close your gripper (At this point you will still be at Post Pick) and you will drop the part back into the bin. Then the robot will jump back to the Home Position and try again.

6.3.7. Find Next Part Conditions and Status



The Find Next Part (FNP) node checks enabled conditions to determine if it should exit early while waiting for the next validated motion sequence. The Find Next Part node also exits, becoming available as a status variable for other nodes to act on.

Find Next Part node

The Find Next Part node offers several configurable condition checks. You can enable/disable any combination of exit checks. The exit check configuration parameters and enable/disable parameters are respectively displayed as URscript functions that return a boolean, indicating whether or not the value is accepted. If the value is not accepted, a reason displays in the UR log.

For example:

```
actinav_fnp_set_<name_of_condition>_<name_of_parameter>(value)
```

and

```
actinav_fnp_enable_<name_of_condition>(value)
```

While waiting for a validated motion sequence, the Find Next Part node checks each enabled condition. If a condition check triggers, the Find Next Part node immediately sets the FNP exit status variable and exits (no pick attempted).

Action	Description	Configuration
Do Place Executions	Exit after N parts have executed a Do Place.	N number of parts that have executed a Do Place
Do Pick Do Place *Execution Delta	Exit after N parts have executed Do Pick but not Do Place.	N number of parts that have executed Do Pick but not Do Place
FNP Time	Exit when the time waiting for the next validated motion sequence exceeds the time limit. Motivation: Provided as a catch-all when the other condition checks are not triggered but the robot has been motionless fir an extended period of time, waiting for a pick	Time limit (seconds) Check during the first FNP? (boolean default:true)
Average FNP Time	Exit when the running average of the last N FNP wait times exceeds the time limit.	Time limit (seconds) Wait for at least N samples? (boolean default:true) N number samples to wait for
Manual Control	Exit when actinav_fnp_set_manual_exit_control() is called with True.	None

*Delta here means the difference between the number of picks and the number of places, where the number of places can be smaller due to missed picks.

Conditional Actions node

The Conditional Actions node is the designated location for conditional action(s). It is optional and serves a descriptive purpose.

```
Program
Robot Program
Pick and Place Loop
Pick and Place Rules
Home Position
Find Next Part
Conditional Actions
```

On execution, the Conditional Actions node exhibits the following behavior:

```
If (ec_should_perform_bin_replacement()):
pose_cache = get_actual_joint_positions()
<execute child nodes (user actions)>
```

```
movej(pose_cache)

ec_reset_FNP_exit_checks()
```

7. Annex I: Functional limitations and workarounds

7.1. Using SubPrograms

You can implement the ActiNav program node as a subprogram from the main PolyScope program. A given PolyScope program allows a single ActiNav program loop.

An ActiNav program loop is not re-entrant, so ActiNav does not support multiple loops. ActiNav's Program Validator searches for multiple program nodes in the ActiNav program loop, and issues an error to prevent them from running.

You can use

7.2. Using a second ActiNav loop in a program

You can insert a second program loop into a Polyscope program, relative to the ActiNav program node, in the following structures:

- Nested: a second ActiNav program node is inserted as a child of an existing ActiNav program node. This use case is detected by ActiNav's Program Validator, an error is issued and the program is invalidated.
- Peered: a second ActiNav program node is inserted as an additional child of a program node that is already parent to the ActiNav program node. ActiNav uses a URScript to force a scripting error to be generated when a second ActiNav loop is present.

7.3. Using threads

The ActiNav program flow is sequential, so ActiNav does not support calling the ActiNav program node, or any subnodes, in a PolyScope thread. You can use threads not running ActiNav-related functionality in a program containing an ActiNav program node.

For example: Consider a thread that monitors an I/O signal. Processing that thread can occur in parallel with execution of ActiNav calls, so if the thread employs heavy use of URController resources (CPU, memory, etc) it can impact ActiNav performance and functionality.

7.4. Using conditions as threads

You can select the Check Expression Continuously box to run a condition in a thread in PolyScope. However Check Expression Continuously does not work if that condition is an ActiNav functionality, like an ActiNav move. See: Using threads.

PolyScope has no API for ActiNav to check if the Check Expression Continuously box is selected. The ActiNav Program Validator scans for any conditions in the ActiNav program node. If any conditions are found, a warning is issued to check for this configuration and disable if present.

7.5. Running at safety-certified collaborative speeds

ActiNav operates at maximum speeds as managed by UR. For any source driving the robot other than UR, PolyScope continuously checks the speed of the robot against its internal, non-published model to verify operation within safety parameters.

If a source is found to be operating too close to the limit of the safety parameters, PolyScope issues speed scaling change requests to reduce the robot's speed. If the robot does not slow down, then PolyScope triggers a protective stop.

This means Actin continuously tries to balance maintaining top speed with PolyScope generated requests to reduce speed.

Enabling ActiNav to run at higher speeds requires significant engineering integration from UR to address how speed scaling is applied.

7.6. Using PolyScope switch conditions

ActiNav does not support the use of PolyScope switch conditions.

You can use `IF/ELSE/ELSE IF` conditions which are supported by ActiNav as switch statements.

7.7. Using conditional logic

In ActiNav converging logic restricts conditional logic.

Converging logic implies any condition block in the ActiNav program tree, containing an ActiNav Move Node (either directly or as a descendant), cannot have a subsequent sibling that is an ActiNav Move Node or a conditional block containing an ActiNav Move Node (either directly or as a descendant).

In the following example the `IF/ELSE` statement is valid, but the ActiNav_Move node is not valid since both conditional moves converge:

```
IF (A)
ACTINAV_MOVE(position 1)
ELSE
ACTINAV_MOVE(position 2)
ACTINAV_MOVE(position 3)
```

The above example can be implemented correctly in ActiNav as follows and is functionally equivalent:

```
IF (A)
ACTINAV_MOVE(position 1)
```

```
ACTINAV_MOVE(position 3)

ELSE

ACTINAV_MOVE(position 2)

ACTINAV_MOVE(position 3)
```

Software Version: 5.10
Document Version: 1.0

99520