With this lab session we will learn how to detect whether a collision has occurred or not and how to avoid it. In particular we will learn how to:

- Create a path that is not collision free
- Detect a colliding target position and modify the path accordingly
- Avoid a collision between multiple volumes

How to proceed

CREATE A PATH THAT IS NOT COLLISION FREE

1. Follow the instructions 1-7 of the previous lab session (“Programming an industrial robot in a virtual environment”) to import the robot ABB IRB 140, associate it with the Virtual Controller 6.09 and equip the robot end effector with the tool “My tool” which is available in the ABB Library under “Equipment”.
2. Follow the instructions 14-15 of the previous lab session (“Programming an industrial robot in a virtual environment”) to import in the station the “Propeller table” and position it opportunely.
3. You will visualize the following robot station.

   ![Robot Station Image]

4. Create a new work-object: go to the tab “Home”, then from the menu “Other” click “Create Workobject”. Change its name to “wobj_collision”. Click “Create”.
5. Below “User Frame” select “Frame by points” and then “Three points” method.
6. Click on the icon “Surface selection” and on the icon “Snap mode edge”.

7. Activate the first input window of the Three-points window. Select three points on the surface of the object fixed to the table, then “Accept”, “Create”. We now have the new workobject (wobj_collision) in the corner of the table.
8. Go to “Home/Target/Create Target”. Then change to “Snap object” mode.

9. In the graphics window select one after the other the desired target positions (notice that when they are selected in a correct way they are marked as red stars). This mode helps you find the corners. **Create a path that is not collision free.** Click “Create”, then “Close”.

Now we have obtained a path that collides with existing components.

10. Right-click “Target_10” on the Path&Targets browser. Click “View tool at target” then select “MyTool”. A representation of the tool will show up in the graphics window. The tool aligns with the target position and orientation.
11. Right-click “Target_10” and choose “Modify Target->Rotate”. Rotate the target 180° around the y axis. Press “Apply” and the tool shows the resulting position of the tool with respect to that target. Then click “Close”.

12. Right-click “Target_10” and click “Copy orientation”.
13. Multi-select all the remaining targets, right-click and select “Apply orientation”.

14. Stop function “View tool at target” by deselecting “MyTool”.
15. Modify the instruction template: change velocity to “v50” and zone to “fine”.
16. Delete the path (menu “Modify”) and the unused targets (select T_ROB1, menu “Modify”, “Remove unused targets”)
17. Multi-select all targets and then right-click and choose “Add to new path”. A new path “Path_10” is created.
18. Expand “Path_10” to see the motion instructions set from the instruction template.

19. Right-click the first motion instruction, press “Modify instruction” and change “Motion type” to joint. In fact it is easier for the robot to reach a MOVEJ instruction as the first position. Click “Apply” and “Close”.

20. Under the “Home” tab, select the button “Path”, then click on “Create new path” to create a new empty path.

21. Right-click “Path_10” and select “Configuration->Autoconfiguration”. Choose a suitable starting configuration. (Choose the second option). This allows you to select among different joint configurations that allow the same position/orientation of the tool.

22. Rename the new empty path (Path_20) “main”. Main is the entry point when running a robot program.

23. Right-click “main” and select “Insert Procedure Call”. Choose “Path_10”.

24. Synchronize the created object from the station to the RAPID code. From “Home” tab, click the button “Synchronize” and “Synchronize to RAPID”. Select also “Path_10”, then click OK.

25. Click “RAPID” tab. Expand “RAPID”, then T_ROB1 and double click on Module 1. Here we see the robot program:

26. Run the simulation of the program by selecting the tab “Simulation” and then click play.

27. Save the station as “Station_with_collisions”.

---

\(^1\) T_ROB1 is the main task from the ABB controller (the first letter T stands for ‘task’). A task is a program that the robot controller will execute. Each task consists of a set of modules and all the modules are local in each task. Variables (VAR) and constants (CONST) are local with respect to each task, but when they are defined as “persistent” (PERS), they become reachable in all the tasks. The functions that do not return a value are denoted as PROCEDURES and bounded by PROC and ENDPROC.
COLLISION DETECTION

(You will find it in the folder lab3rs_CollisionDetection/Stations/Station_with_collisions.rsstn)

N.B this functionality is available only under a licensed version of RobotStudio. It basically allows to create two groups of objects that will be tested for collision against each other. Hence, the collision detection functionality detects whether a collision between two selected objects has occurred or not, but the robot does not stop in the case it collided with the specified object. The collision is only reported to the user through the graphics interface, then the user has to manually change the path if he/she wants to avoid collision.

1. From the “Simulation” tab, click on “Create collision set”. This command allows you to set up collision detection between moving parts. When you click, you will visualize “CollisionSet_1” in the Layout browser.

2. Expand “CollisionSet_1”. You can visualize two objects: object A and object B, respectively. These are the objects between which you want to detect a potential collision.

3. Drag the first element between you want to check collision, “MyTool”, and drop it on “ObjectsA”. Then perform the same operation for the propeller table: select “table_and_fixture_140” and drop it on “ObjectsB”.

If you now expand “object A” and “object B” you can visualize the object they have been associated with.
4. Right click on “CollisionSet_1” and select “Modify collision set”. Set “Near miss” equal to 5 mm and click “apply”, then “close”. Near miss is defined by default as the two objects being within 0.01 meter from each other. According to these settings, during the simulation when a collision between the objects will be detected, the colliding objects will be marked red in the graphics.

5. Press Play to run the simulation and check collisions.

6. During the execution of the path RobotStudio reports the collisions. To avoid collisions and create a collision-free path we need to modify the colliding target positions.

7. Go to the “Path&Targets” browser and select the first colliding target, by expanding TROB_1->workobject->wobj_collision.

8. Right-click on the first colliding target, in this case “Target_10” and select “View Tool at Target”.

9. Change the position of the Targets such that they do not collide anymore. (In this case select the target you want to modify, then right-click it, select “Modify target” -> “Offset position”.

   In this case, we should set:
   - Target_10: x=-5, y=0, z=-15;
   - Target_20: x=-5, y=-5, z=-15;
   - Target_30: x=0, y=-5, z=-15;
   - Target_40: x=10, y=0, z=-15;
   - Target_50: x=10, y=10, z=-15;

10. Synchronize the created object from the station to the RAPID code. From “Home” menu, click “Synchronize” and “Synchronize to RAPID”. Click ok.

11. Press Play to run the simulation (From the Simulation tab) and check whether you have obtained a free collision path or not. If not, iterate steps 9-11. If yes, save the station.
COLLISION AVOIDANCE

N.B this functionality is available only from RobotWare version 6.08 and does not necessarily requires a license. RobotWare is the app you need to download and install to use the virtual robot controller. It is available for download from the RobotApps page in the RobotStudio Add-Ins tab. It basically allows to prevent collisions between robots and equipments or between robots in a MultiMove system. In this case, differently from the collision detection mode, when a collision is predicted between two components, the simulation stops immediately before the predicted collision occurs.

1. Open RobotStudio and create a new solution with empty station
2. From the Home tab select the menu “ABB library” and import IRB140
3. From the Home tab select “Import Library/Equipment” and import “IRC5Dual-Cabinet”, then position it in x=500,y=-600,z=100 using the button “Move” in the “Freehand” menu (alternatively right-click on the robot name in the Layout browser and click “Position/Set Position”, choose the desired position then press “Apply” and “Close”).

![RobotStudio interface](image)

4. From the Home tab, press “Virtual Controller”, select “From Layout” and import the robot controller **6.09**. *(N.B the functionality we are going to use works only with RobotWare version 6.08 or newer)*
5. Select a tool for the IRB140. From the Home tab, select the “Import Library” button, then click “Equipment/training objects” and choose “MyTool”.
6. Attach it to the IRB140 end effector by dragging and dropping it to the robots in the Layout browser.
7. Create an empty path: from the Home tab, select “Path/Empty path”. You will then visualize Path_10 in the Paths&Targets browser. Change the velocity to v500 and the zone to z5, by exploiting the menus in the bottom toolbar.
8. From the Home tab, jog the robot end effector to a desired position (using the button Jog linear in the Freehand menu). Then press the button “Teach instruction” to save the pose and the motion instruction (MOVE in this case).

![RobotStudio interface with Teach instruction](image)

9. Iterate the procedure 9. to create an arbitrary path composed by at least 10 targets and associated motion instructions.

![RobotStudio interface with an arbitrary path](image)
10. Right-click on Path_10 and select “Move along the path” to let the robot move along the path just created.

11. On the tab “Controller” click on “Change options”, select “Motion Supervision” and put a mark on “Collision detection”, then click OK. Collision avoidance requires the RobotWare option “Collision Detection”.

12. A restart of the controller is required.

13. The state of the collision model in the controller can be shown in the 3D graphics, by clicking “Collision Avoidance/Show current state in graphics”.

14. RobotWare contains built-in collision model that can be shown in the graphic window. In this case the only colliding component of the station that RobotWare takes into account is IRB140.

15. When “Show predicted collision in graphics” is active the collision model will be displayed when a collision is predicted. At the current time no collision can be predicted since only the robot is included in the RobotWare collision model. We need to include also the tool and the cabinet, which are considered External Equipment. External Equipment must be configured manually.

16. To configure the external equipment from the “Controller” tab, select the “Collision Avoidance” menu and click “Configure”. This window will show up:
The window shows the 3D view of the actual collision avoidance model. All parts of the robot are automatically added.

17. To add the tool click “Add/Station Object” and select “MyTool”. The name of the object must be a valid RAPID identifier. The collision model exploits a simplified geometry that consists of convex hulls.

By default a single hull is created. Since in this case the geometry is too coarse, it is possible to subdivide the model by planes into multiple hulls.

18. To do that, select the “Convex hull” window. The best plane can be selected automatically, by clicking “Add/Automatic”.

19. Click “Generate hull” to update the geometry.

20. The “Position” window specifies the placement and attachment of the object. In this case it is filled up automatically since the tool is attached to the robot end effector in the RobotStudio station. Click “OK” to create the object. The collision model window is updated automatically:

21. Now, let us add the controller cabinet: click “Add/Station object” and select “IRC5_Dual-Cabinet”. This name is not a valid RAPID identifier: change it to “IRC5_Dual_Cabinet”. Click “OK”. The collision model is updated accordingly.

22. Click “Upload to controller” to commit the changes to the controller.

23. If we make the robot move along the path (Select “Path_10” and right-click “Move along path”), a collision is predicted between MyTool and the cabinet and a warning message is shown in the Output window. Notice that the robot stops before the predicted collision occurs.
The robot stops here

Colliding position