Faculty of Engineering of the University of Porto

Visual Components: Process Modelling Machining and Palletizing Cell

INDUSTRIAL ROBOTICS

Master in Electrical and Computer Engineering

Sandro Magalhães
António Paulo Moreira

2nd Semester of 2019/2020

1 Introduction

The Visual Components is a 3D simulation software for manufacturing. The simulations are usually created using different features from the software. However, the most common are Process Modelling and Works Library. Both can be involved in factory simulation, defining interactions between robots, mobile robots, humans, and the rest of the components.

This laboratory work introduces to the Process Modelling capabilities of the Visual Components. We intended to mount a manufacturing cell on Visual Components. This cell will receive wheels for machining, and, after that, the industrial manipulator will palletize them.

This practical work was performed using Visual Components 4.2 Premium but should work with other versions of Visual Components.

2 Configuring the workspace for machining the wheels

We are going to start by placing all the components in the Visual Components workspace. For that, start with an empty workspace **File** > **Clear All** (Ctrl+N). Now, it is needed a **Feeder** to provide the wheels, a **Conveyor** to transport them, and a reference **Car Tyre**. So, in the **Home** tab, go to the **eCatalog** (left panel), and add to the workspace:

- **eCatalog** > **Models by Type** > **Process Flow Components** > **Feeder**
- **eCatalog** > **Models by Type** > **Conveyors > Conveyor**
- **eCatalog** > **Models by Type** > **Products and Containers** > **Car Tyre**

If the **eCatalog** panel is not visible, do it visible in the **Home** tab > **Windows** group > **Show** > **eCatalog.**

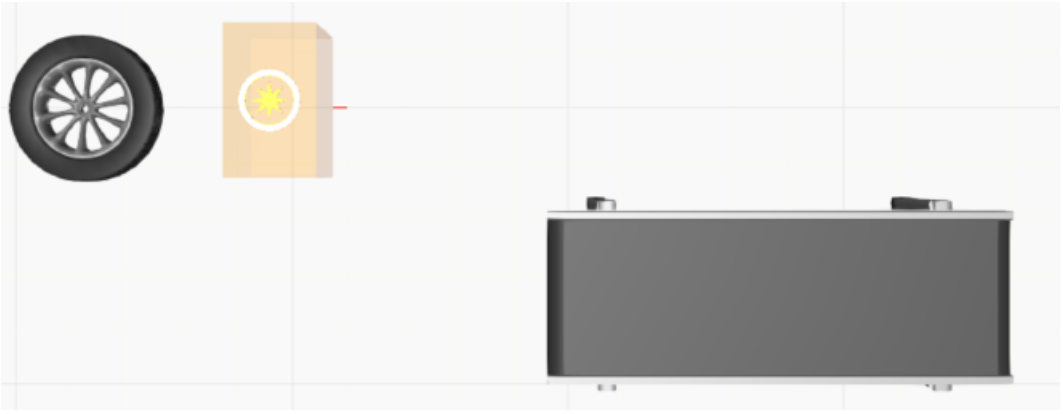


Figure 1 Added components to the 3D World

The **Car Tyre** may be placed in any place.

To allow the objects provided by the **Feeder** flow to the conveyor, we need to connect them. Select **Home** > **PnP.** The **PnP** is the Plug and Play tool, which allows an easy connection between components. Grasp the **Conveyor** and approximate it to the **Feeder**. A green arrow will appear

between the **conveyor** and the **feeder** (Figure 2a). Continue moving until the conveyor attaches to the feeder. Release the conveyor and check that the small yellow arrow is now green (Figure 2b).

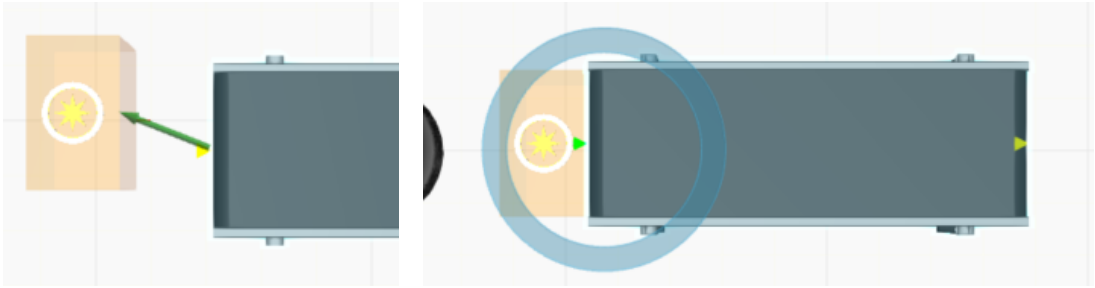


Figure 2 Attaching the conveyor to the feeder. (a) Green arrow to attach using PnP (b) Conveyor attached to the Feeder - green arrow

Run the simulation (Figure 3), and check that the **Feeder** is providing cylindrical objects, and they are flowing to the conveyor (Figure 4). Reset the simulation.

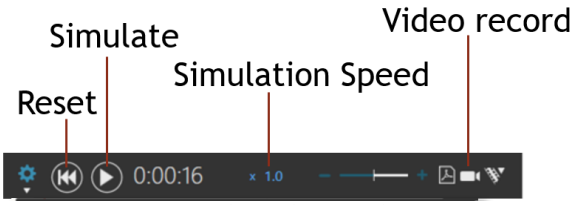


Figure 3 Simulation buttons description

Figure 4 shape feeder providing cylinders to the conveyor

Now, configure the **Feeder** to provide Car Tyres. For that, select the **Feeder** and select the tab **Process** > **Products**. In the **Product Type Editor** (left panel), select **VC_Cylinder.** In the **Properties** (right side panel), change the name to **CarTyre** and using the **Pick component from the 3D world** tool  , select the **Car Tyre** in the workspace.

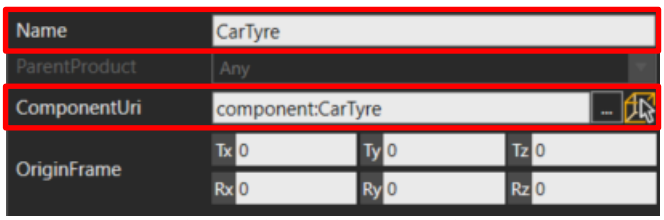


Figure 5 Configuring the product of the feeder to provide Car Tyres instead of cylinders

Now, it is needed to adjust the size of the **Feeder** and the **Conveyor**. Select the **Feeder** and, in the **Component Properties** panel, change the **ConveyorLength** and the **ConveyorWidth** to **700 mm.**

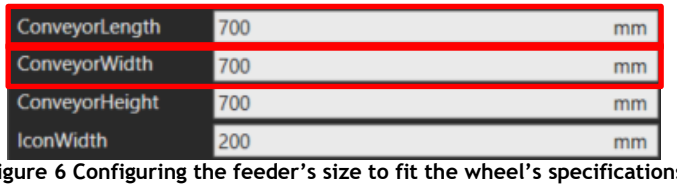


Figure 6 Configuring the feeder's size to fit the wheel's specifications

Proceed similarly to the conveyor, and change the **ConveyorLength to 3000 mm.**

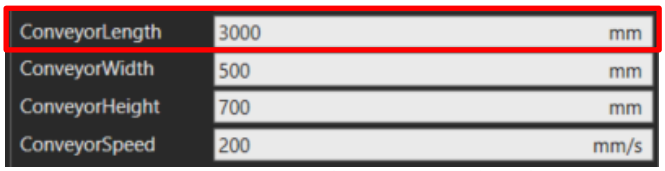


Figure 7 Increasing the conveyor's length

Using the **Home** > **PnP** tool detach and attach the conveyor again to the feeder, to fix the conveyor width and the flow.

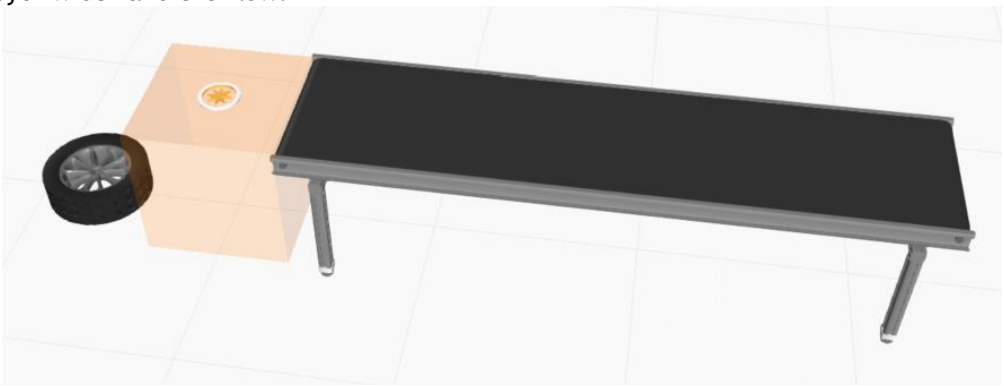


Figure 8 Final result for providing car tyres

Now, add the robots that will process and manipulate the Car Tyres. Add an **eCatalog** > **Models by Type** > **Process Transport Controllers** > **Robot Transport Controller** at the end of the **Conveyor** (Figure 9)**.**

Figure 9 Adding the robot controller

The robot will need to move planarly to reach the machines and the objects. For that, add an **eCatalog** > **Models by Type** > **Robot Positioners** > **Visual Components** > **Robot Floor Track** to the workspace. Using the **Home** > **PnP** tool, attach the **Robot Floor Track** to the **Robot**

Transport Controller. Keeping the **Robot Floor Track** selected, in the **Component Properties** panel (right side panel), rotate the **Robot Floor Track,** i.e., **Rz** = 90.

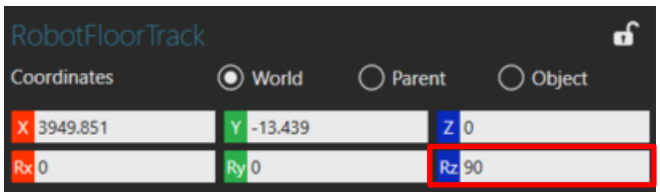


Figure 10 Rotating the Robot Floor Track to be orthogonal to the conveyor

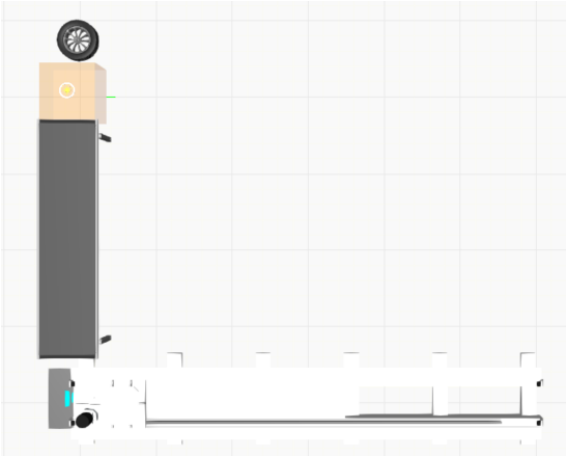


Figure 11 Final configuration of the Robot Floor Track

Finally, pick an **eCatalog** > **Models by Type** > **Robots** > **Visual Components** > **Generic Articulated Robot v4** and attach it to the **Robot Floor Track** (Figure 12).

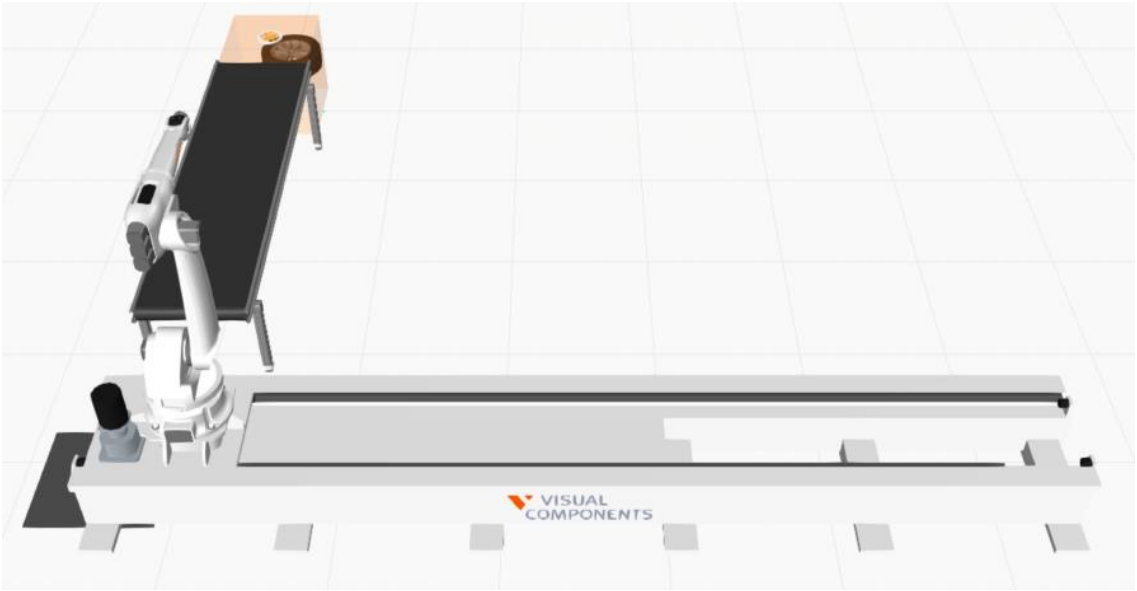


Figure 12 Mounting the robot manipulator in the robot floor track

The robot will also need a tool to grasp the objects. Select the **eCatalog** > **Models by Type** > **Robot Tools** > **Visual Components** > **Generic 3-Jaw Gripper** and attach it to the robot (Figure 13).

Figure 13 Attaching the gripper to the robot manipulator

For processing the wheels, it is needed two ProMill's and two ProLathe's machines. Pick two **eCatalog** > **Models by Type** > **Machines** > **Visual Components** > **Process Machine - ProMill** and place them near the **Robot Floor Track,** in the lateral side of the conveyor. Proceed similarly, adding two **eCatalog** > **Models by Type** > **Machines** > **Visual Components** > **Process Machine - ProLathe** on the other side of the **Robot Floor Track**.

It is possible to remove the panel in the **Components Properties** of each machine. Select each machine or all machines at once, and, in the **Component Properties** panel, unselect the **ShowPanel** option.

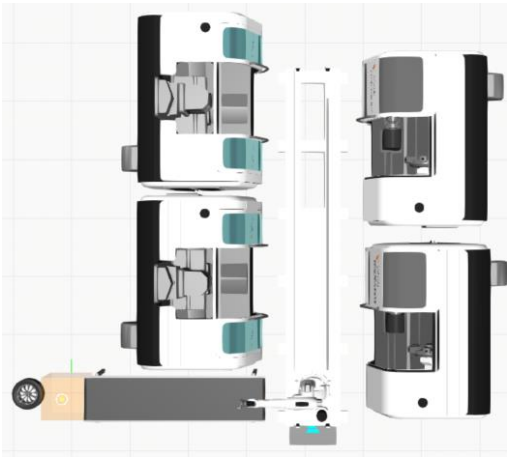


Figure 14 Final result of the machining cell

3 Programming the cell to machine the tyres

Run the simulation now, and check whether the wheels flow to the conveyor. However, they do not stop at the end, neither the robot picks them. So, first, add an **eCatalog** > **Models by Type** > **Process Flow Components** > **From Conveyor Processor** and, using the **Home** > **PnP** tool, attach it to the end of the conveyor (Figure 15).

Figure 15 Attaching the conveyor process to the end of the conveyor to allow the robot to pick the car tires

Now, run the simulation, and check whether the wheels stop when they reach the end of the conveyor, waiting for the robot to grasp them. Reset the simulation. It is time to program the robot to grasp the wheels and transport them to the machines.

Move to the **Process** tab and select **Process** > **Flow**. One blue circle is shown on the conveyor and each machine. These are the nodes of the components. Select the **fromConveyor** node (click on the name instead of the circle), Figure 16a, next select the robot transporter, Figure 16b, and then the **ProMill** node (click on the name instead of the circle), Figure 16c. Check if the robot controller symbol is visible in the flow lines, Figure 16d, else select the shown symbol and in the **Properties**, change the **Implementer** to **RobotController::TC**.

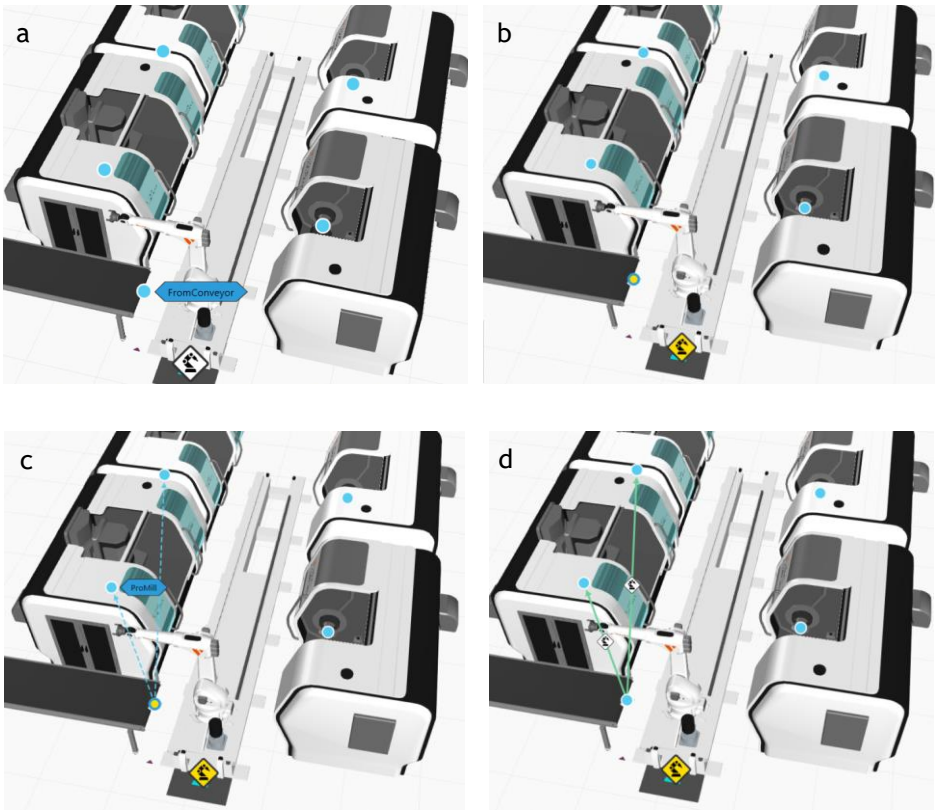


Figure 16 Defining the flow process modelling. (a) Select the FromConveyor node - click on the name (b) Select the robot controller - becomes yellow when selected (c) Select the ProMill machines (d) Final result of the flow

After the wheels being processed in the ProMill machine, they should be processed in the ProLathe machine. Therefore, select, first, the **ProMill** node and check if the **Robot Controller** is selected (Figure 17a), and then the **ProLathe** nodes (Figure 17b).

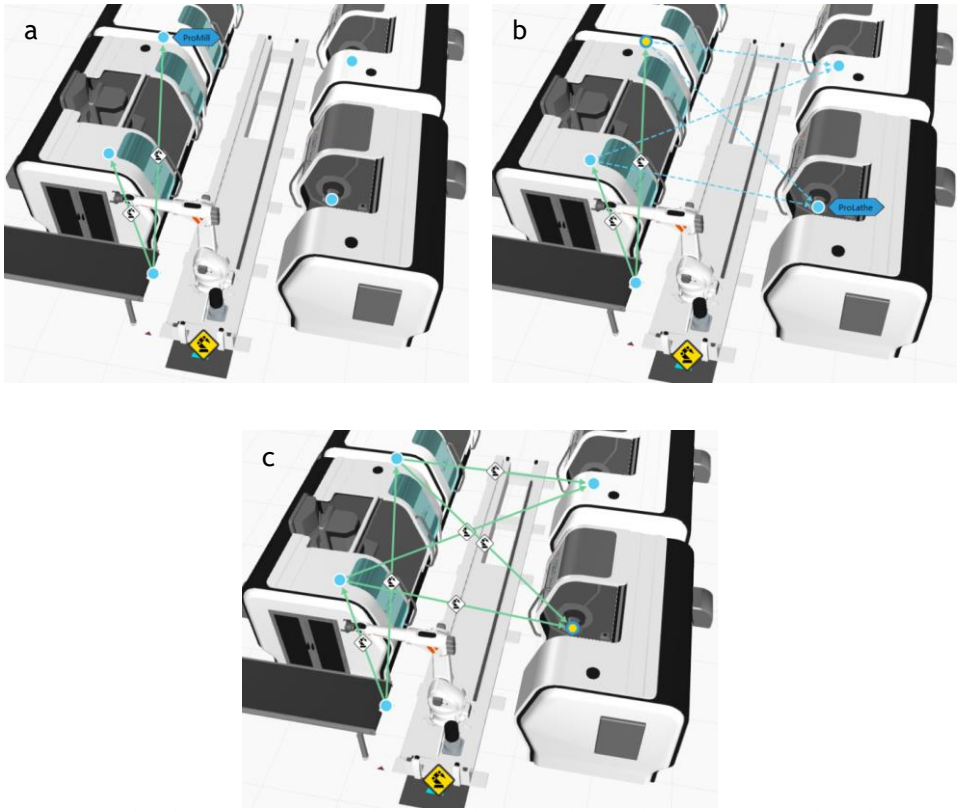


Figure 17 Defining the flow process for machining in the ProLathe machines (a) Assure the robot controller is selected - yellow - and select the ProMill node - click on the name (b) Select the ProLathe node (c) Final result.

The **Process Flow Editor** should look like:

Figure 18 Flow Process for machining the car tyres

If the flow is not visible, use the eye's symbol to make the **Flow Group #1** visible.

Now, run the simulation, and check that the robot picks the wheels and transport them between machines.

Video: https://educast.fccn.pt/vod/clips/2p6uthdq2x/html5.html

4 Configuring the palletizing cell

Now, it is intended that the robot palletizes the wheels after they were machined. Similarly to what was done for providing wheels, it is needed a new **Feeder**, **Conveyor,** and reference **Pallet.** So, add to the 3D World:

- **eCatalog** > **Models by Type** > **Process Flow Components** > **Feeder**
- **eCatalog** > **Models by Type** > **Conveyors** > **Conveyor**
- **eCatalog** > **Models by Type** > **Products and Containers** > **Pallet 1200x1000**

Figure 19 Added the new components for palletizing

Now, fix the **Feeder** dimensions. Select the last added **Feeder** and, in **Component Properties**, change the **ConveyorLength** to 1500, the **ConveyorWidth** to 1000 and the **ConveyorHeight** to 500.

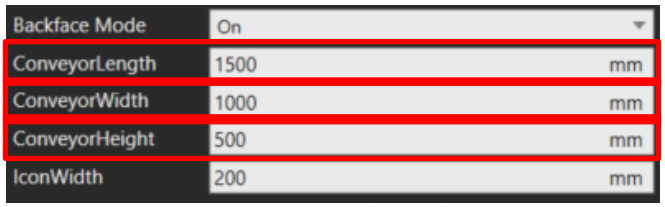


Figure 20 Configuring the Feeder's size

Select, now, the last added **Conveyor**, and change the **ConveyorLength** to 2500 and unselect the option **ShowEndIdler**.

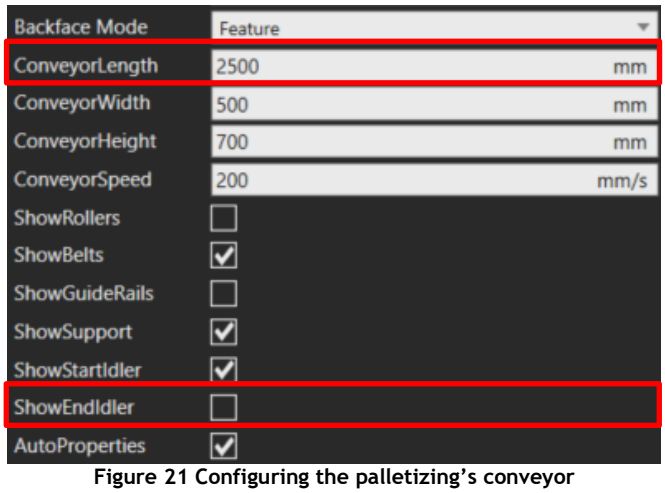


Figure 21 Configuring the palletizing's conveyor

Finally, using the **Home** > **PnP** tool, attach the **Conveyor** to **Feeder**. Select the **Feeder** and in **Component Properties** > **ProductCreator** change the **Part** option to **New Product Type**.

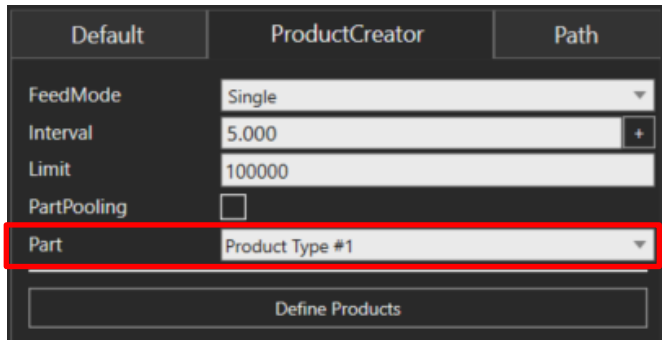
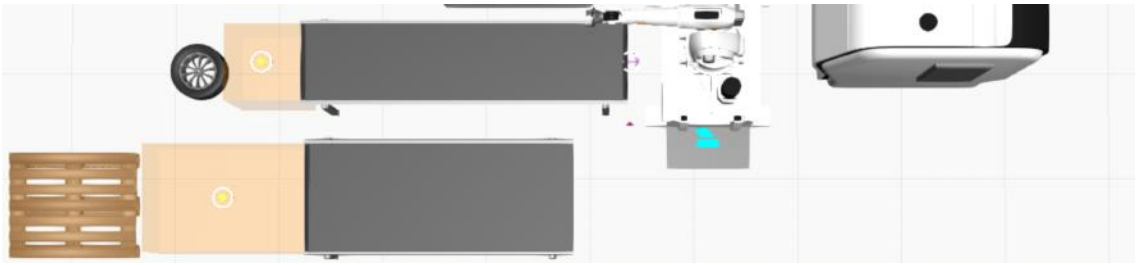


Figure 22 Create a new product for the new shape feeder to provide pallets

Go to the **Process** tab and select **Process** > **Products.** In the **Product Type Editor**, select the new product **Product Type #1**. Now in the **Properties** panel, rename the product to **Pallet** and using the **Pick component from 3D world** tool , select the **FinPallet** (Pallet 1200x1000) in the workspace.

Figure 23 Set the new product to a pallet

Run the simulation, and check whether the **Feeder** is providing pallets, and they flow to the **Conveyor**. Reset the simulation.



Now, add an **eCatalog** > **Models by Type** > **Process Template Components** > **Inline Process** to the workspace, and another **eCatalog** > **Models by Type** > **Conveyors** > **Conveyor**. Select the **Inline Process** and, in **Component Properties**, change the **ConveyorLength** to 2000 mm and unselect the **ShowStartIdler** and **ShowEndIdler**.

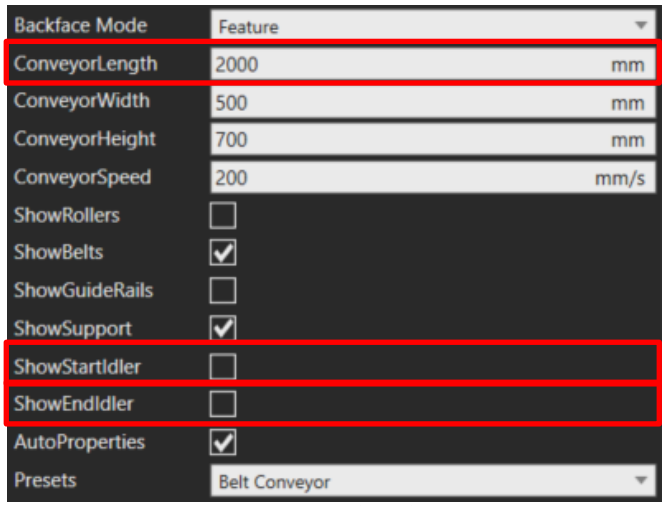


Figure 24 Configuring the inline conveyor

Select the last added **Conveyor** and unselect the **ShowStartIdler**. Using the **Home** > **PnP** tool, plug the **Inline Process** to the conveyor connected to the Pallet feeder, and then connected the last added conveyor to the **Inline Process.**

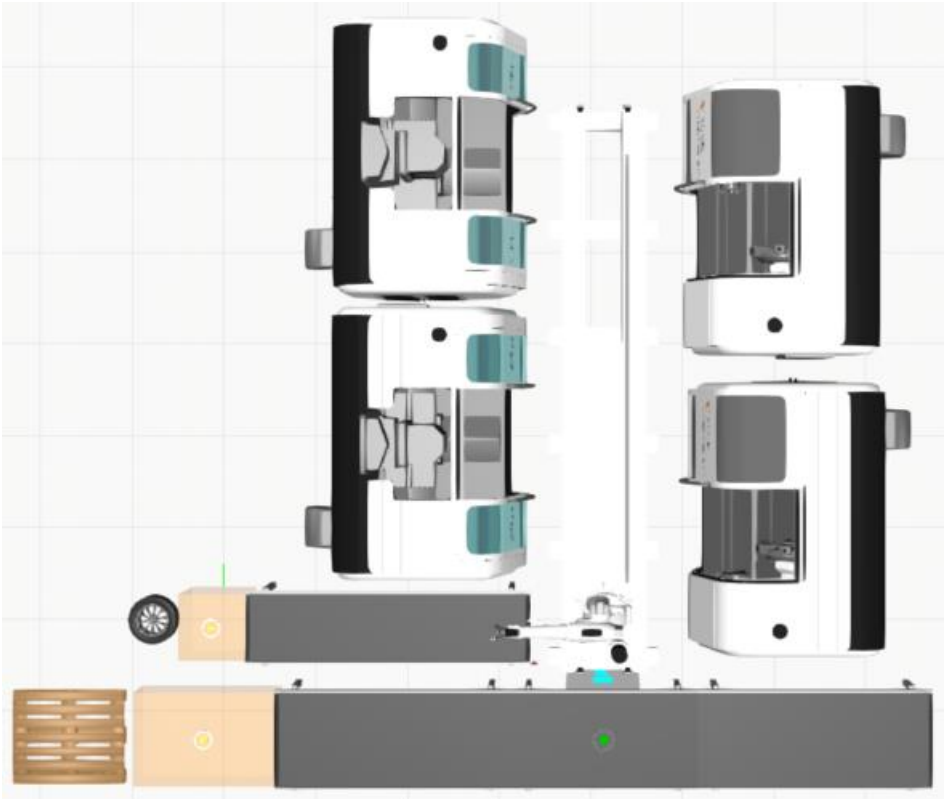


Figure 25 Final configuration of the machining and palletizing cell

Run the simulation, and check if the Pallet stops at the **Inline Process.**

5 Programming the robot to palletize

The programming strategy to palletize is similar to the strategy to program the robot to transport the wheels to the machines. So, In the **Process** tab, select **Process** > **Flow**. Now, select the **ProLathe** node, then select the **Robot Transporter** (whether not selected) and, finally, the **Inline Process** node.

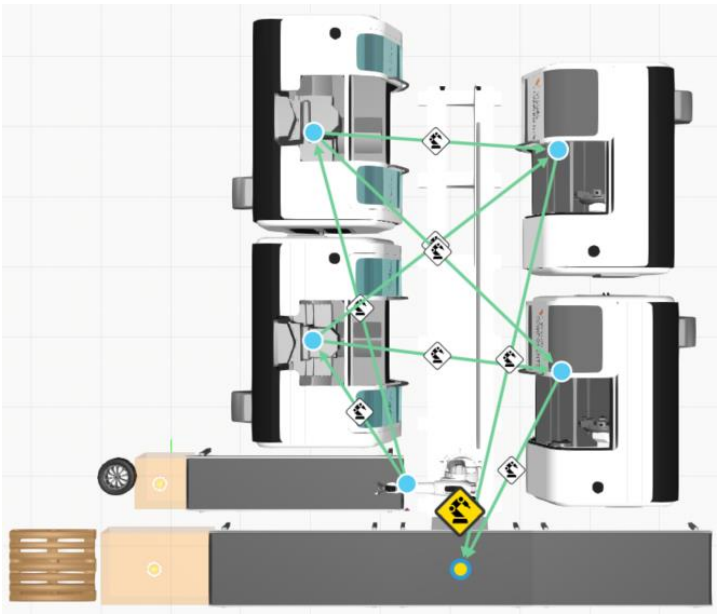


Figure 26 Connecting the machining flow to the palletizing flow

The **Process Flow Editor** looks like:

Figure 27 Resulting process flow

Run the simulation and check that the robot, now, moves the processed wheels to the pallet. However, it puts the wheel in the middle inner of the pallet and does not limit the number of wheels.

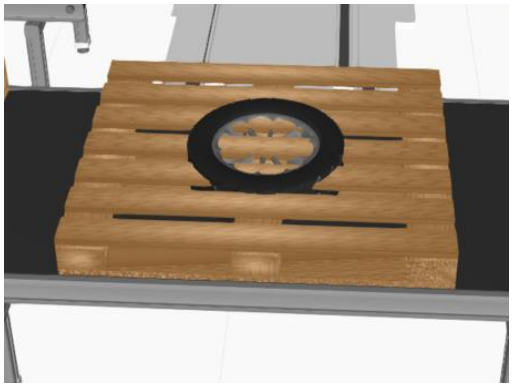


Figure 28 The robot places the car tires in the pallet. However, it is overlapped with the pallet.

For fixing this, it is needed to configure the **Inline Process** node. Select **Process > Processes**, and click on the **Inline Process** in the workspace. A new window **Inline Process::ProcessExecutor_HIDE_** is shown (Figure 29).

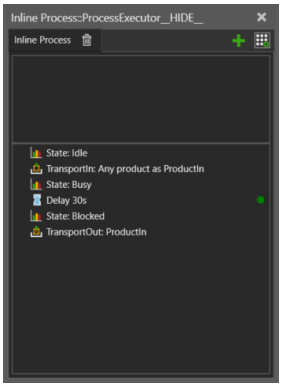


Figure 29 Standard configuration of the conveyor's inline process

Select the **TransportIn** statement and, in **Statement Properties**, change the **Source** to **From Component Container**, and the **ProductVariableName** to **Pallet.**

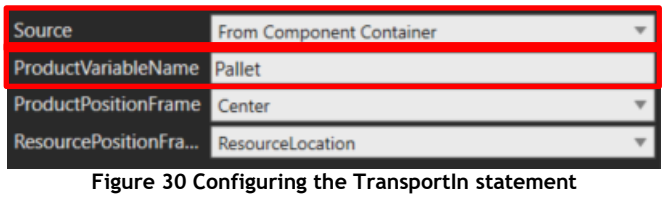


Figure 30 Configuring the TransportIn statement

Remove the **Delay 30s** statement and add a **Transport Pattern in** instead. For that, click on **Statements**  button and choose the **Transport Pattern In** option**.** In the **Statement Properties** for this new statement, change the **ProductVariableName** to **CarTyres**, the **ProductPositionFrame** to **Center,** and the **ResourcePositionFrame** to **ResourceLocation.** We also need to adjust the offsets to start palletizing in one of the corners, stacking three wheels, and avoiding the overlapping. So, change the **Statement Properties** as in the Figure 31. The offset defined the position of the first wheel. It was chosen considering the dimensions of the car tyre and the pallet. It was intended to place the first wheel in a quarter of the pallet, avoiding the overlap with the pallet. After it was defined the pattern structure to place the car tyres, i.e, in this case, a matrix of $2 \times 2 \times 3$, avoiding the overlapping with other wheels or the pallet. Unselect also the **AcceptAllProductTypes** option, to accept only products from **Flow Group #1.**

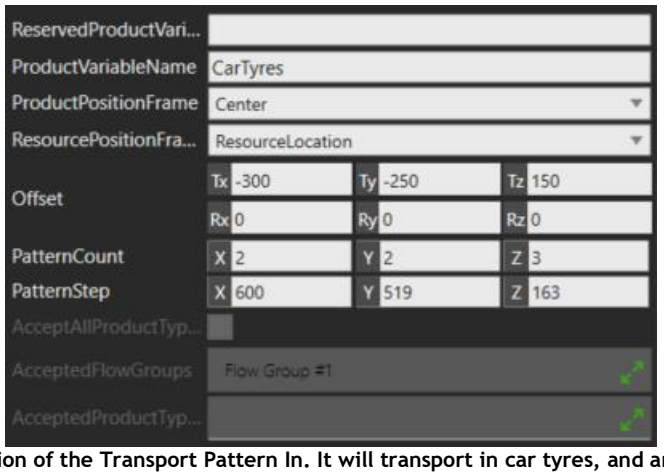


Figure 31 Configuration of the Transport Pattern In. It will transport in car tyres, and arrange them in a 2x2 matrix in three layers. It offsets the first wheel to start near one corner and then moves the reference frame to place the other car tires.

Now, add an **Attach** statement after the **TransportPatternIn** statement. This statement will join the wheels to the pallet, holding the wheels, when the pallet is moving. In **Statement Properties** of the **Attach** statement, change the **Parent** to **Pallet** and the **Child** to **CarTyres**.

Figure 32 Configuration of the attach to group the car tyres and the pallet

Finally, select the **TransportOut** statement and in **StatementProperties**, change the **ProductVariableName** to **Pallet,** and the destination to **To Component Container**.

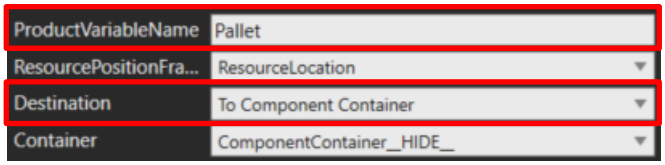


Figure 33 Configure the conveyor to move the pallet when it is full.

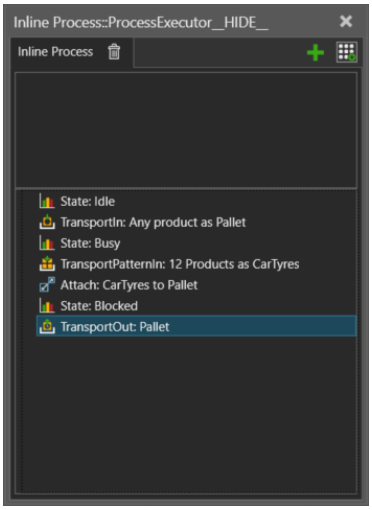


Figure 34 Final configuration of the conveyor's Inline Process to palletizing

Run the simulation and check that the robot palletizes the wheels. When the pallet has three wheels stacked, the conveyor moves and changes to an empty pallet.

If the robot stops during the palletizing process, it means that the conveyor is out of range of the robot (Figure 35). To solve this, adjust the conveyors positions, becoming near to the robot. Some times, the simulation also stops, some joints of the robot become red (Figure 36), and a similar error message is printed in the output panel (Figure 35). This means that the manipulator reached the joints limits (Figure 36). Also, approximate the conveyors to the robot.

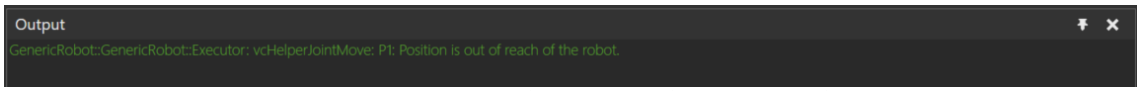


Figure 35 Out of reach message in the output panel

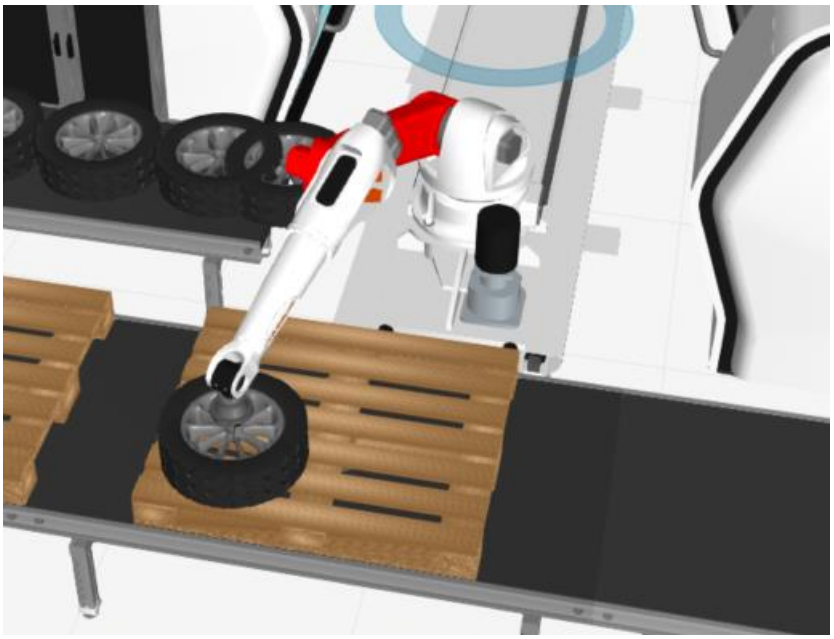


Figure 36 Robot reached a joint limit

Video: https://educast.fccn.pt/vod/clips/1u0e4kofx7/html5.html

6 Finalising the cell

In manufactories, robotics cells must be closed to protect human operators of incidents related to the robots' operations. So, similarly, we will build a fence around the cell to enclose the robot. For that, add to the centre of the cell an **eCatalog** > **Models by Type** > **Facilities - Factory** > **Visual Components** > **Fence Builder.** In **Component Properties** > **Fence Defaults,** change the **Max Length** to 1400 mm. Now, in **Component Properties** > **Default**, click on **Start Drawing** and draw the limits of the cell.

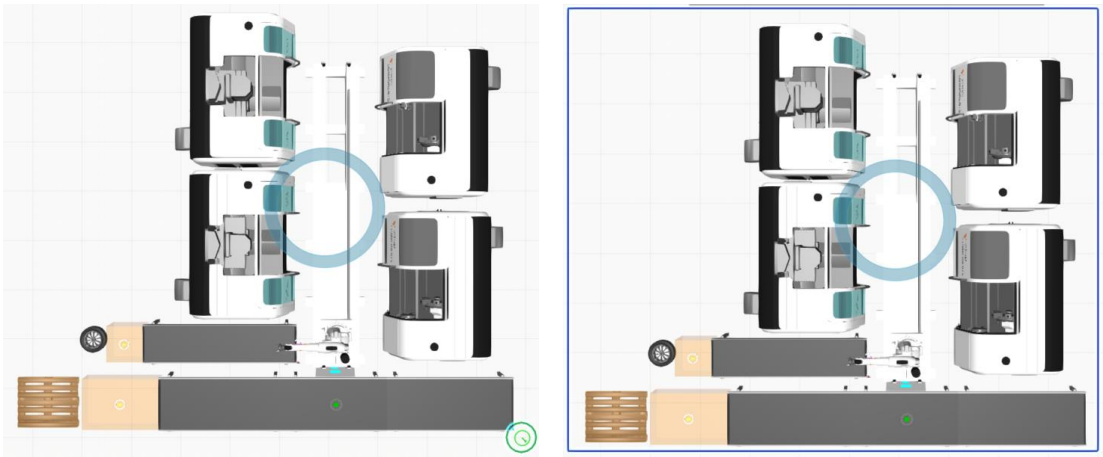


Figure 37 Building the fence. (a) Place the fence builder in the center of the cell and the green circle in once corner to start drawing the fence (b) the blue line is the drawn fence

Once the limits of the fence are specified, click in **Component Properties** > **Default** > **Populate Fences**.

Now, the tutorial about Process Modelling in Visual Components finished. Feel free to continue testing the Visual Components and beautify the cell.
Create a video with the final result.

Video: https://educast.fccn.pt/vod/clips/168igfyfsk/html5.html