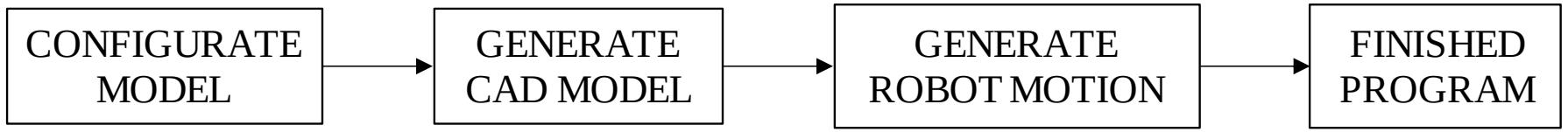
Automatic Production Planning

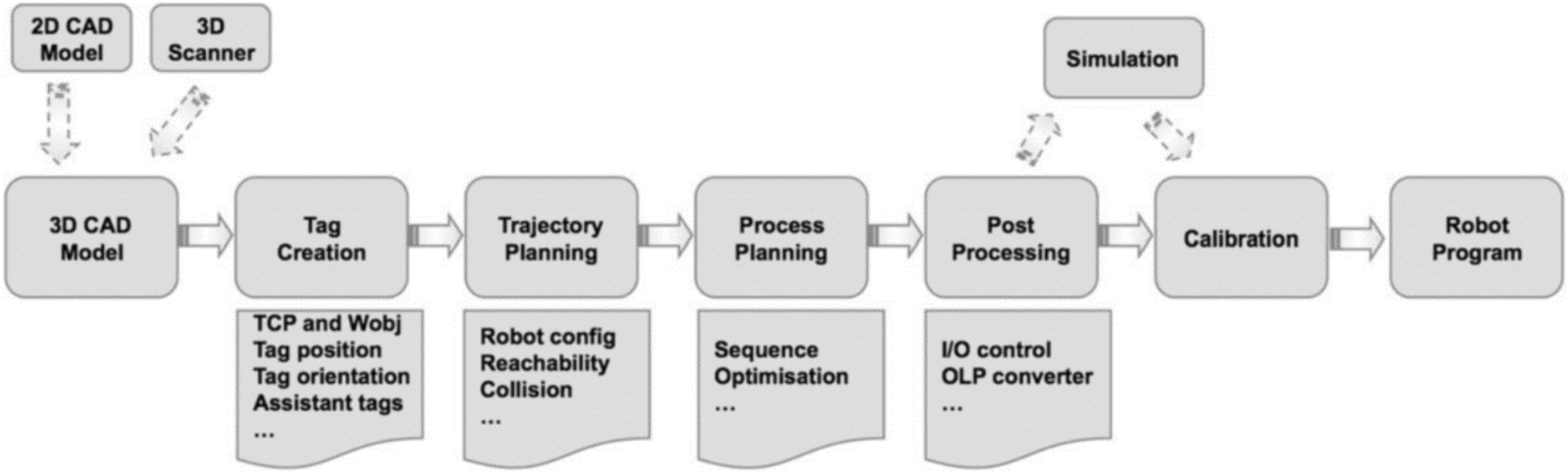
LINKÖPING UNIVERSITY

Design Automation Lab
mehdi.tarkian@liu.se

Automation process



Generate Robot Motion



Design Automation Lab
mehdi.tarkian@liu.se

Varions robot classes in Delmia

- AMP generates finally a RobotTask with Operations/RobotMotions(having attributes) and a set of tags(seam).
- RobotTask with Operations/RobotMotions(having attributes) and the set of tags have information. For example each tag has the information regarding XYZYPR values.
- RobotMotions have information regarding weld speed etc. The current implementation of AMP is such that these information are first populated in AMPTags.
- The list of AMPTags is put in an AMPPath. When AMP is nearing its' end of execution, tags and RobotTask with appropriate Operations/RobotMotions are created and populated with these information by retrieving it from the AMPTags(For each Operation/RobotMotion and tag there is a corresponding AMPTag from which these information are retrieved).
- AMPTags are non-persistent and they are just intermediate buffers for holding these information. Through the ability of executing VB script file by embedding the call to it through the "Execute" keyword, these AMPTags are made available to the user for further customization of these information.
- "Execute" keyword facilitates calling the VB script file's CATMain method by passing it the AMPPath object as the single argument.

Robotmotion object

- Tag group object
  - http://catiadoc.free.fr/online/interfaces/interface_TagGroupFactory.htm#CreateTagGroup
- RobotTaskFactory object
  - http://catiadoc.free.fr/online/interfaces/interface_RobotTaskFactory.htm#CreateRobotTask
- Createoperation object
  - http://catiadoc.free.fr/online/interfaces/interface_RobotTask.htm#CreateOperation
- Createrobotmotion object
  - http://catiadoc.free.fr/online/interfaces/interface_Operation.htm#CreateRobotMotion
- Robotmotion object
  - http://catiadoc.free.fr/online/interfaces/interface_RobotMotion

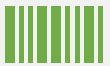
Create a new excel file

Save

Select all CATIA refs in preferences

Create a new module in the VB editor

Start assembling the codes in the upcoming slides



In order to test the code, open STATION_KOLLISION in CATIA

LINKÖPING UNIVERSITY

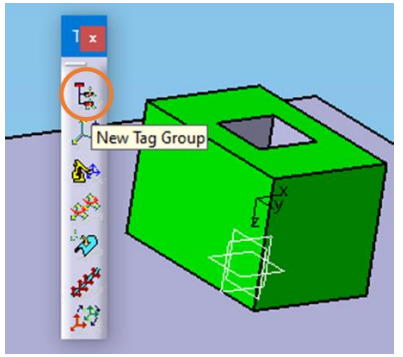
Design Automation Lab
mehdi.tarkian@liu.se

Setup code

```
Set CATIA = GetObject(, "CATIA.Application")

Dim processDocument1 As Document
Set processDocument1 = CATIA.ActiveDocument
Dim product1 As CATBaseDispatch
Set product1 = processDocument1.PPRDocument.Resources.Item("TagList.1")
Dim products1 As Products
Set products1 = product1.Products

Set product3 = product1
Dim products3 As Products
Set products3 = product3.Products
Dim product4 As Product
Set product4 = products3.Item("I_Station_kollision.1")
Dim products4 As Products
Set products4 = product4.Products
Dim product5 As Product
Set product5 = products4.Item("Product1.1")
```


Design Automation Lab
mehdi.tarkian@liu.se

Create tag group




```
Set ParentObject = product1
Set objTagGroupFact = product1.GetTechnologicalObject("TagGroupFactory")
Dim objTagGroup As TagGroup
Dim ModifyRef As Boolean

ModifyRef = False
objTagGroupFact.CreateTagGroup "My_Group", ModifyRef, ParentObject, objTagGroup
Set product7 = products3.Item("My_Group" & ".1")
```

Create tag

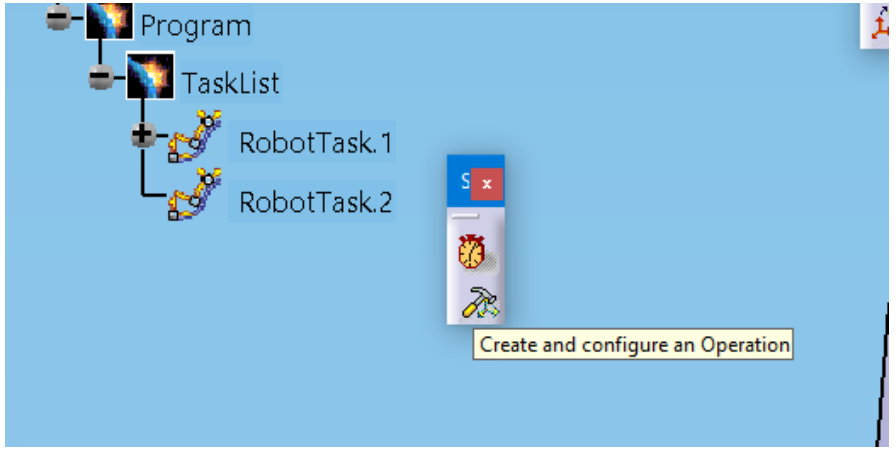

```
Set objTagFactory = product7.GetTechnologicalObject("TagFactory")

Dim objTag As Tag
Dim objTag2 As Tag

Dim TagName As String

TagName = "Tag1"

objTagFactory.CreateTag TagName, objTag
TagName = "Tag2"
objTagFactory.CreateTag TagName, objTag2
```

```
objTag.SetXYZ 900, -100, 100
objTag2.SetXYZ 900, -100, 800

Dim iPath As AMPPath
Dim params(6) 'As CATSafeArrayVariant

Dim AppTag As AMPTag
Dim DepTag As AMPTag
Dim count As Integer
```

Create Robot Task

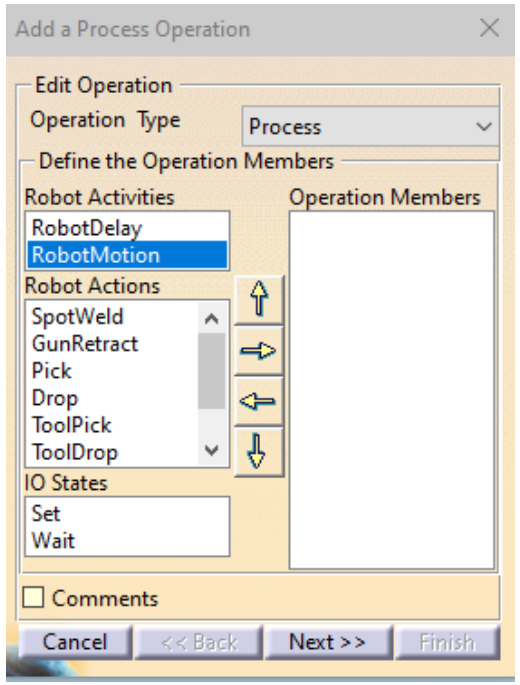

```
Set processDocument1 = CATIA.ActiveDocument
Set objRobot = processDocument1.PPRDocument.Resources.Item(7)

Set objRobotTaskFactory =
objRobot.GetTechnologicalObject("RobotTaskFactory")
objRobotTaskFactory.CreateRobotTask "my_robot_task", objRobotTask

Dim objRobotTask1 As Variant
Set objRobotTask1 = objRobotTask.GetTechnologicalObject("RobotTask")
```

Create Operation

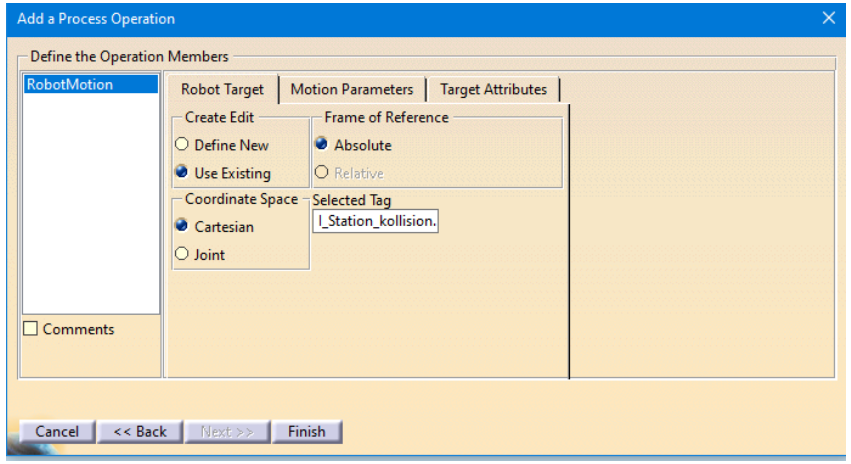



```
Dim objRefOperation As Operation
Dim objAfterOperation As Operation
Set objAfterOperation = Nothing
Dim objOperation As Operation
objRobotTask1.CreateOperation objRefOperation, objAfterOperation,
objOperation
```

Create Robot Motion




```
Dim objRobotmotion101 As RobotMotion
Dim objRefAct As AnyObject
objOperation.CreateRobotMotion objRefAct, 1, objRobotmotion101
objRobotmotion101.SetTagTarget objTag
‘objTag is basically the coord which the end-effector is supposed to reach and rotate to
```



Design Automation Lab
mehdi.tarkian@liu.se

Create after operation and motion

objRobotTask1.CreateAfterOperation objOperation, objOperation20

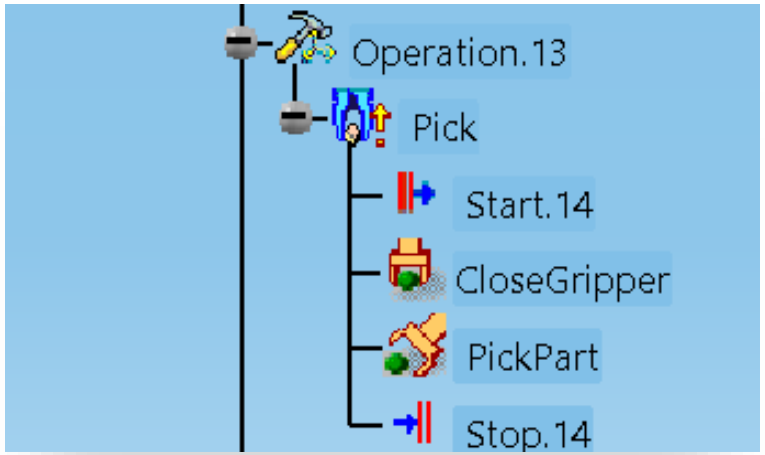
Dim objRefAct2 As AnyObject

objOperation20.CreateRobotMotion objRefAct2, 1, objRobotmotion102

objRobotmotion102.SetTagTarget objTag2

Pick and Place operation

- We have not yet been able to find the API objects reqiured to do pick and place operations, but we do have a solution.
- This solution is extremely tweaky
- There must be a manually created pick and place operation which we copy and paste
- However the values of the object needs to be modified one time before copy operation on the original object and then again on the copy.
- Basically the object has two parameters (A and B) and corresponding values SA and SB.
    - Because of an apparent bug, what happens during the copy paste operation is that the SA and SB swicth place from original A=SA, B = SB to A=SB, B=SA.
    - Furthermore only values of A can be modified from the API library.
    - Hence if we first modify the values SA before the copy operation then we have successfully placed an updated value for parameter B for the copied object.
    - After the copy operation we can modify the value of parameter A.



Design Automation Lab
mehdi.tarkian@liu.se

Copy and Paste of Pick & Place operations (1/2)

```
Dim objRobotTask10(3)
objRobotTaskFactory.GetAllRobotTasks objRobotTask10
Set selection1 = CATIA.ActiveDocument.Selection
Set opt2copy = objRobotTask10(0).ChildrenActivities.Item(4)

‘modifying the parameter A before copy operation
Set product100 = processDocument1.PPRDocument.Resources.Item(2)
opt2copy.ChildrenActivities.Item(2).ChildrenActivities.Item(3).GrabbingObject = product100
selection1.Clear
Set opt2copy = objRobotTask10(0).ChildrenActivities.Item(4)
selection1.Add (opt2copy)
selection1.Copy
```


Design Automation Lab
mehdi.tarkian@liu.se

Copy and Paste of Pick & Place operations (2/2)

```
Set opt2paste = objRobotTask10(1).ChildrenActivities.Item(objRobotTask10(1).ChildrenActivities.count - 1)
selection1.Clear
selection1.Add (opt2paste)
selection1.Paste

Set product100 = processDocument1.PPRDocument.Resources.Item(5).Products.Item(1)
Set opt2mod = objRobotTask10(1).ChildrenActivities.Item(objRobotTask10(1).ChildrenActivities.count - 1)
Set opt2mod2 = opt2mod.ChildrenActivities
Set opt2mod2a = opt2mod2.Item(2)
Set opt2mod2aa = opt2mod2a.ChildrenActivities.Item(3)

‘modifying the parameter A after copy operation
opt2mod2aa.GrabbingObject = product100
```

Almost complete

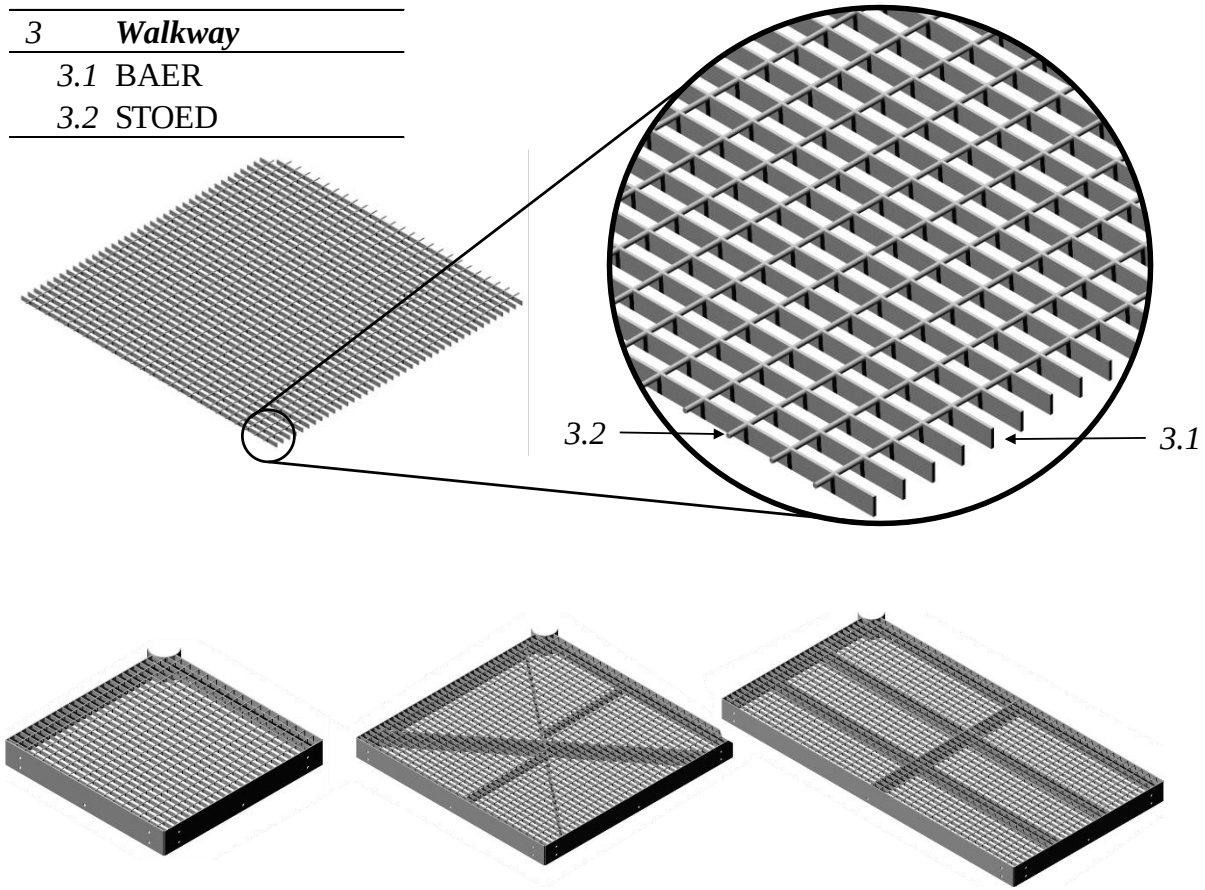
You have now been able to create automatic operations as first shown in the tutorial Introduction to production planning in Delmia

Try to make sense of what you've actually done and create a coherent sequence of operation for your robot

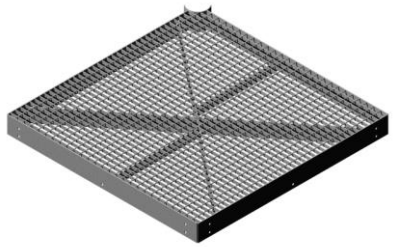
Design Automation Lab
mehdi.tarkian@liu.se

Master thesis: Automation of Offline Programming for Assembly and Welding Processes

3	***Walkway***
3.1	BAER
3.2	STOED



2	***Inner Structure***
2.1	SUPPORT NO CUT
2.2	SUPPORT ONE CUT
2.3	SUPPORT TWO CUT



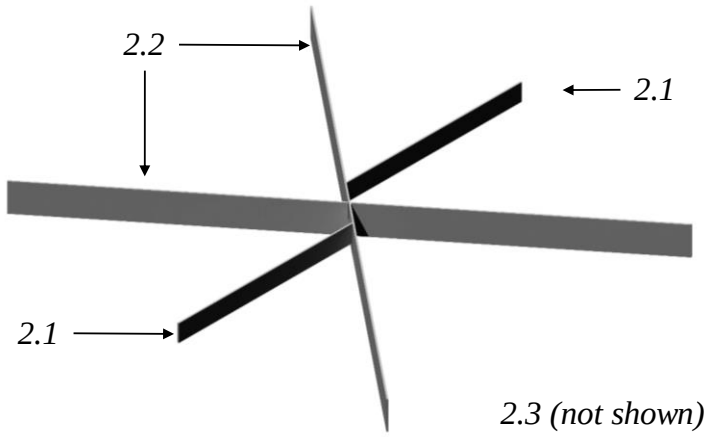
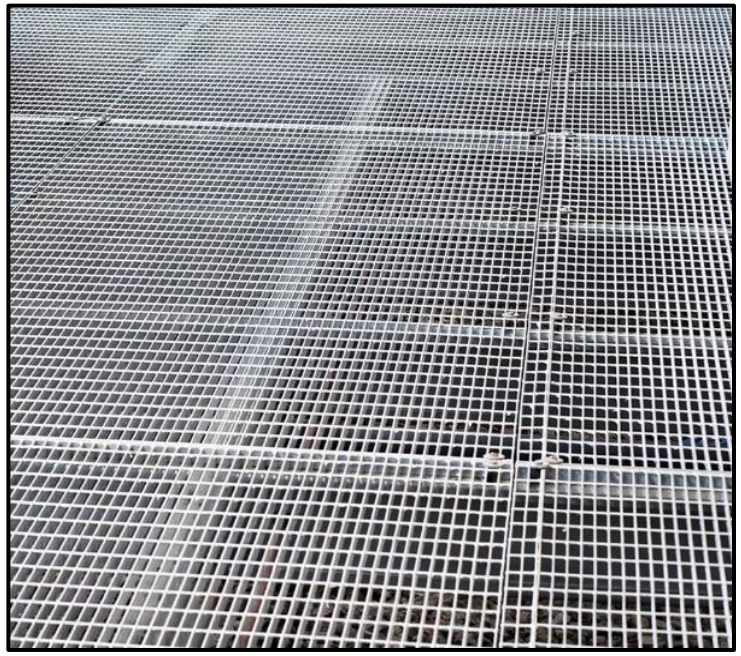


Figure 19: Three different configurations from the product line

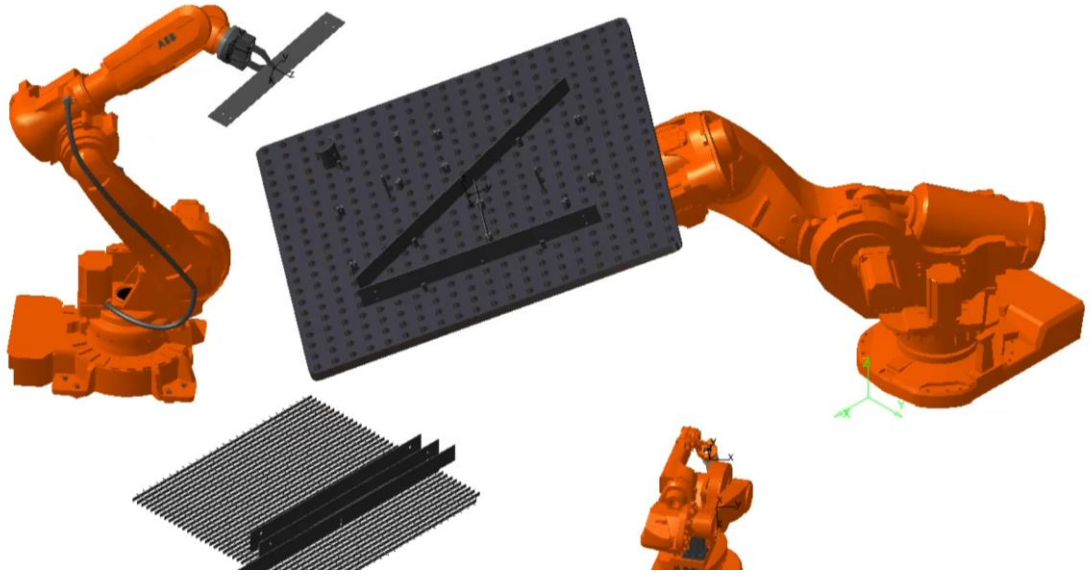
Background

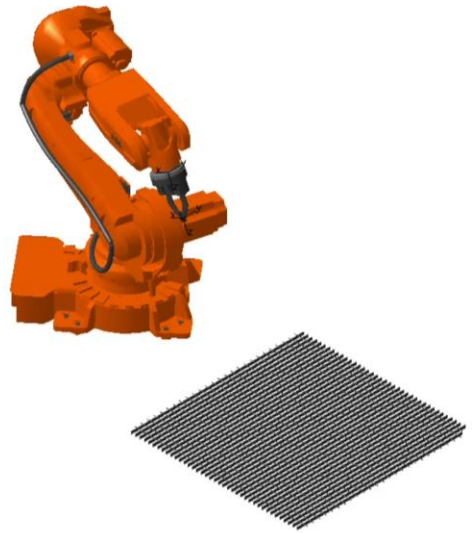
- Industrial gratings
  - *Assembled & welded by hand*
  - *Automate assembly process*
  - *Automate welding process*
- Programming robots is:
  - *Time-consuming*
  - *Expensive*
  - *Unrealistic for small batch sizes*

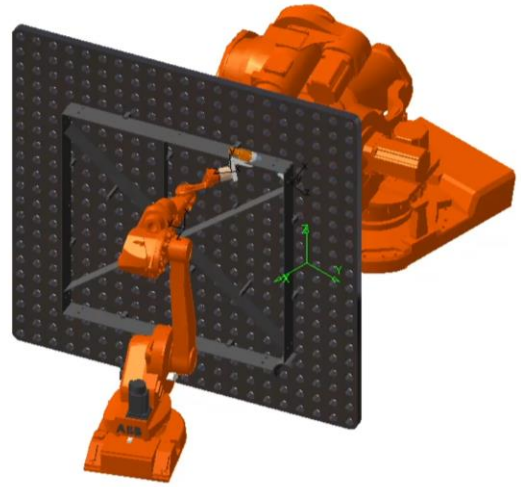


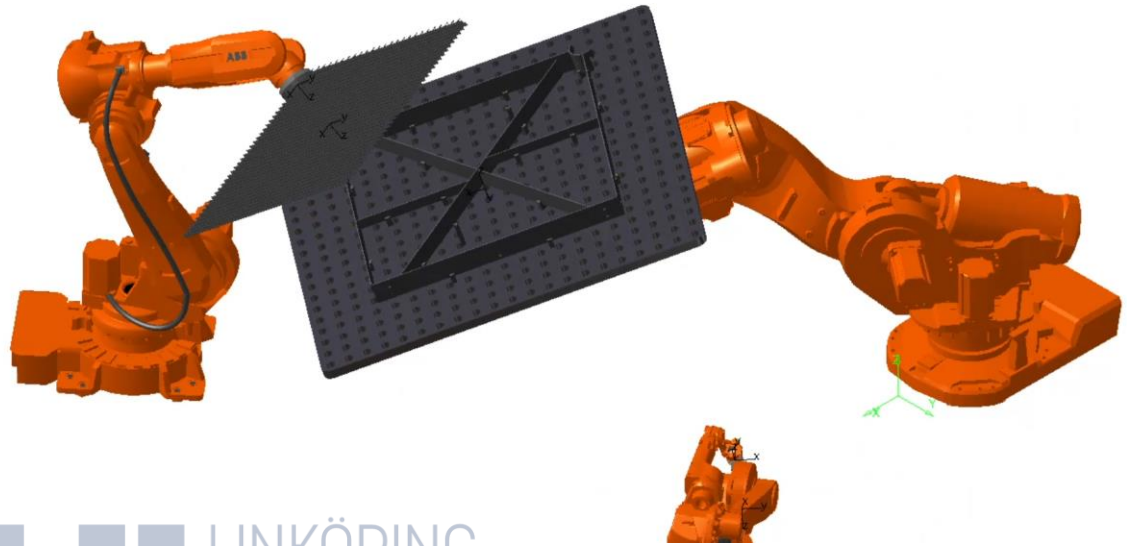
[1]

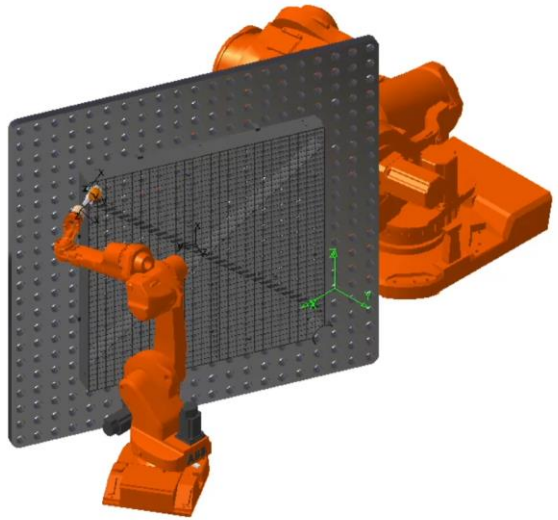
Design Automation Lab
mehdi.tarkian@liu.se











LINKÖPING UNIVERSITY

Design Automation Lab
mehdi.tarkian@liu.se