

# Geometrisk algoritm och kombinatorisk optimering med tillämpning inom bilindustri

Edvin Åblad

Fraunhofer-Chalmers Centre  
Chalmers tekniska högskola och Göteborgs universitet  
1 November, 2018



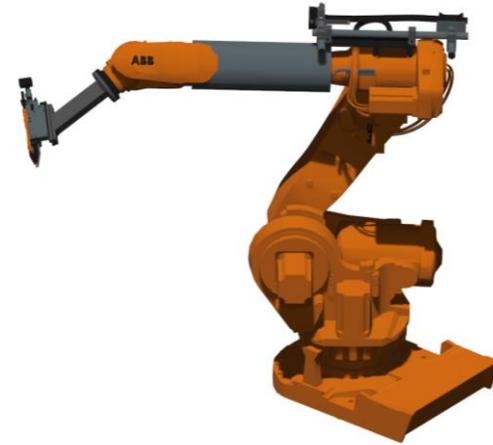
UNIVERSITY OF  
GOTHENBURG  
**CHALMERS**



FRAUNHOFER CHALMERS  
RESEARCH CENTRE FOR INDUSTRIAL MATHEMATICS

# De närmaste 25 minuterna

- ➔ Introduktion av FCC
  - Simulering & optimering
- ➔ Hur jag hamnade på FCC
- ➔ Simulering av kablar
- ➔ Optimering av robotstationer
- ➔ Kurser jag haft mest nytta av



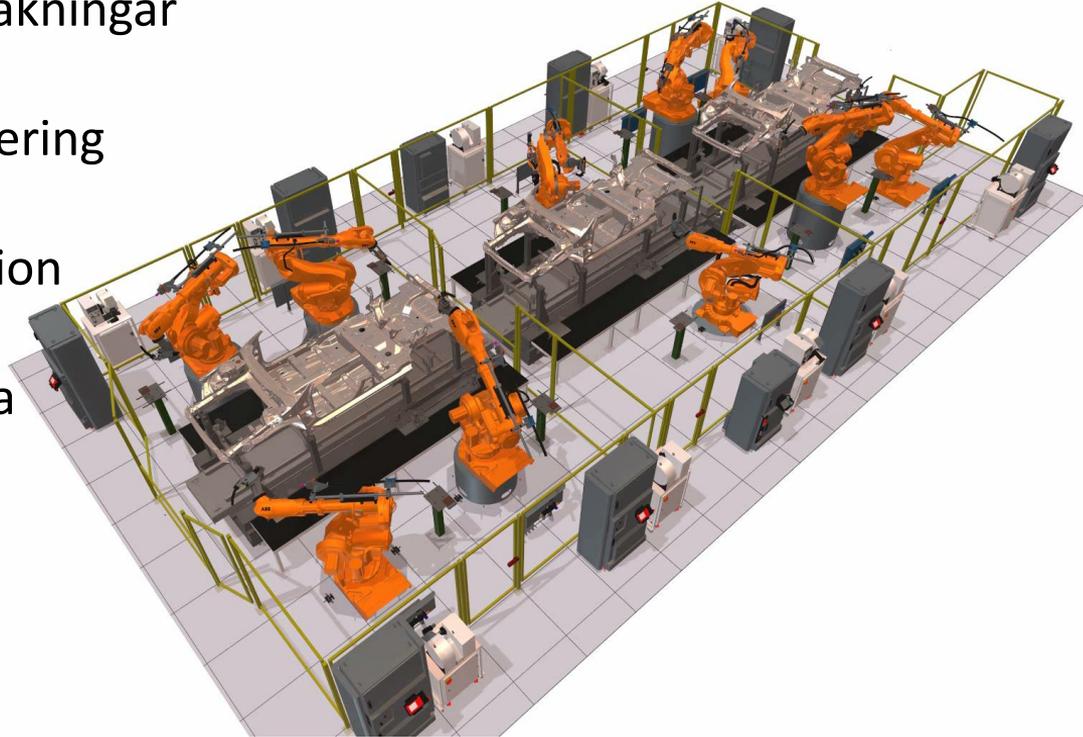
# Fraunhofer Chalmers Centre (FCC)

- ➔ Verksamhet i olika former
  - Samarbeten med Chalmers
  - Industriella projekt
  - Samarbete med andra Fraunhofer institut
  
- ➔ Avdelningar:
  - Geometri och banplanering
  - Beräkningsgruppen
  - System och data analys

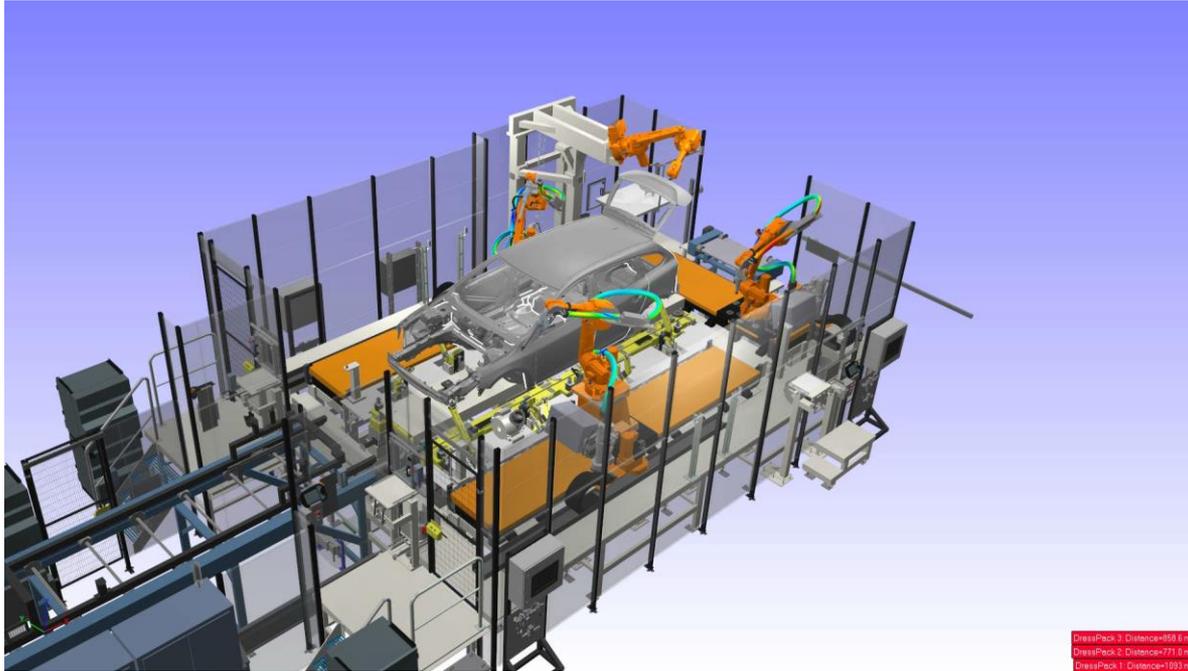


# Geometri och banplanering

- ➔ Avståndsberäkningar
- ➔ Resursoptimering
- ➔ Undvik kollision
- ➔ Automatisera



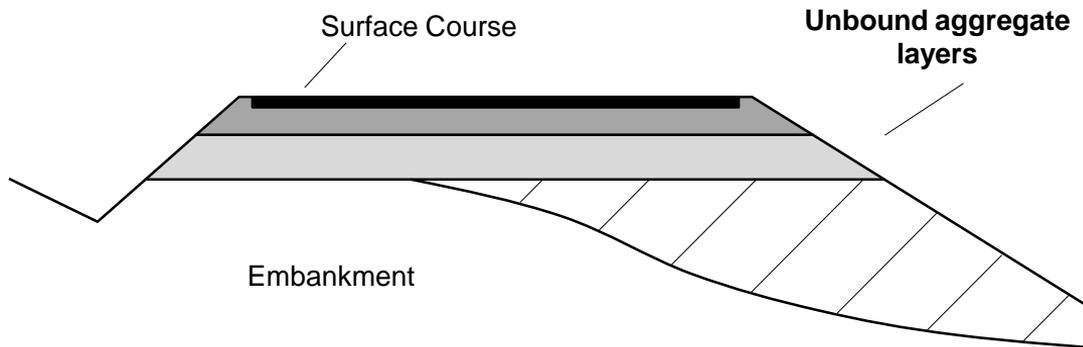
# Optimering av tätningstation och simulering av slangar



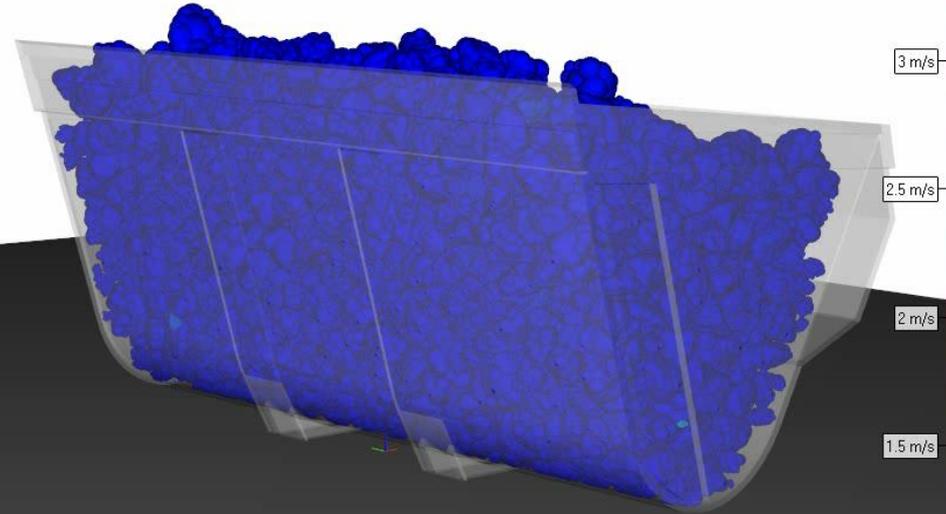
DressPack 3: Distance=658.8 mm  
DressPack 2: Distance=271.6 mm  
DressPack 1: Distance=109.5 mm

# DIGIROAD | Road body

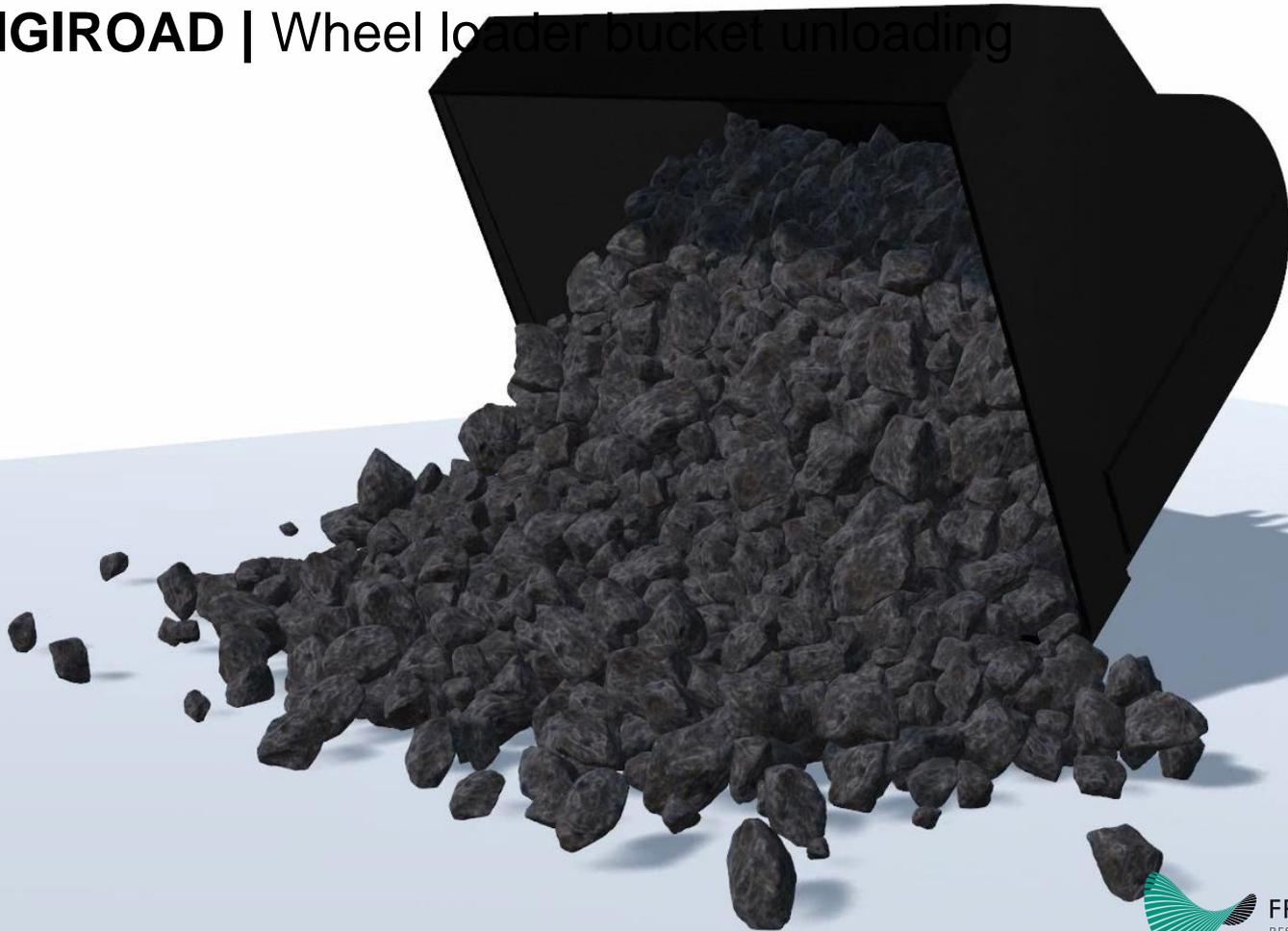
Ett project i beräkningsgruppen



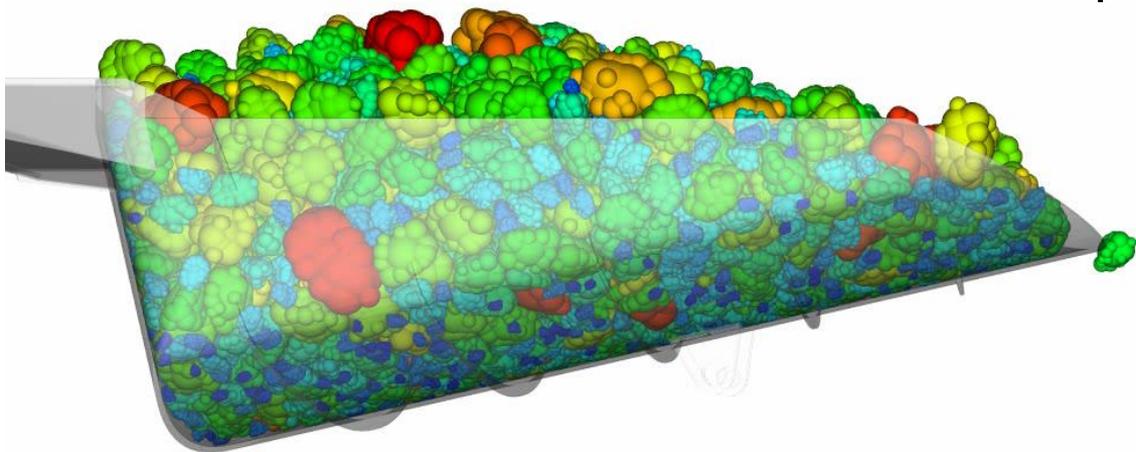
# DIGIROAD | Wheel loader bucket unloading



# DIGIROAD | Wheel loader bucket unloading



# DIGIROAD | Unloading and Size Segregation



Fine

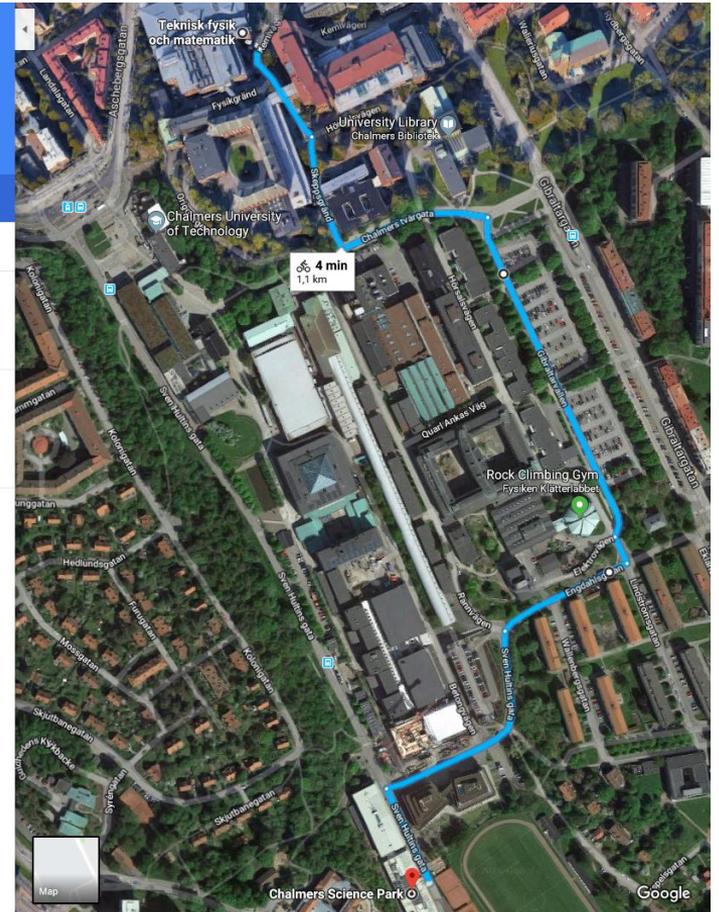
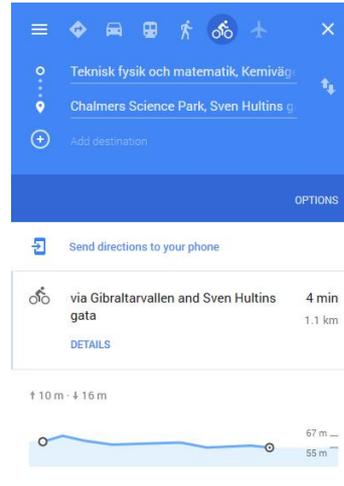


Coarse



# Hur jag hamnade på FCC

- ➔ Räkneövningsledare
- ➔ Optimeringskurser
- ➔ Lunchföreläsning
- ➔ Studentanställd på FCC 10%
- ➔ Exjobb på FCC

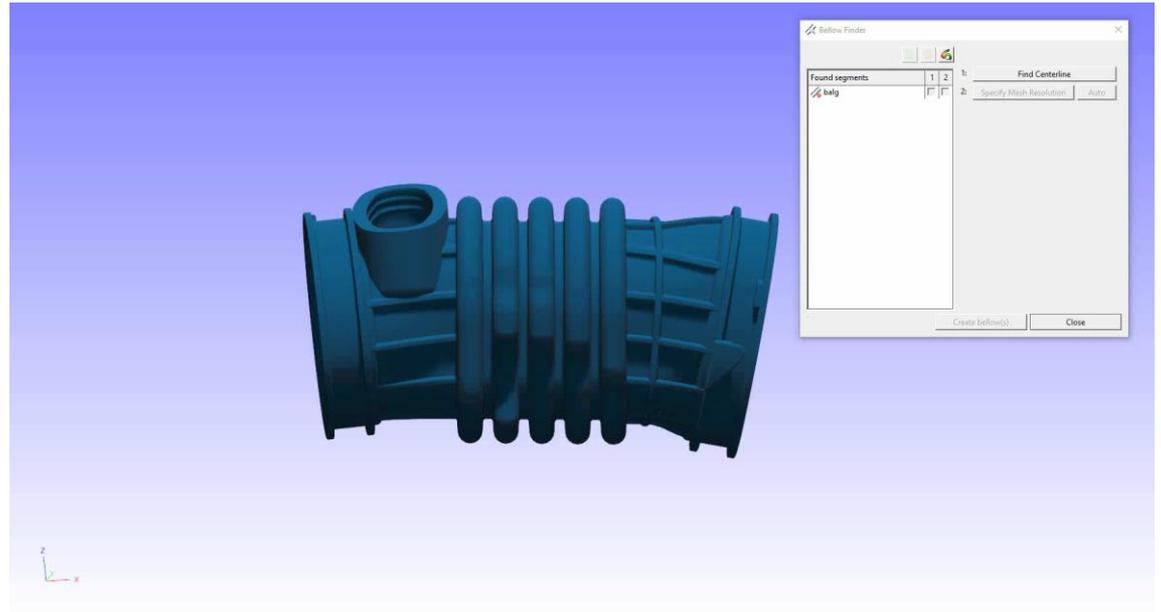


# Simulering av kablar

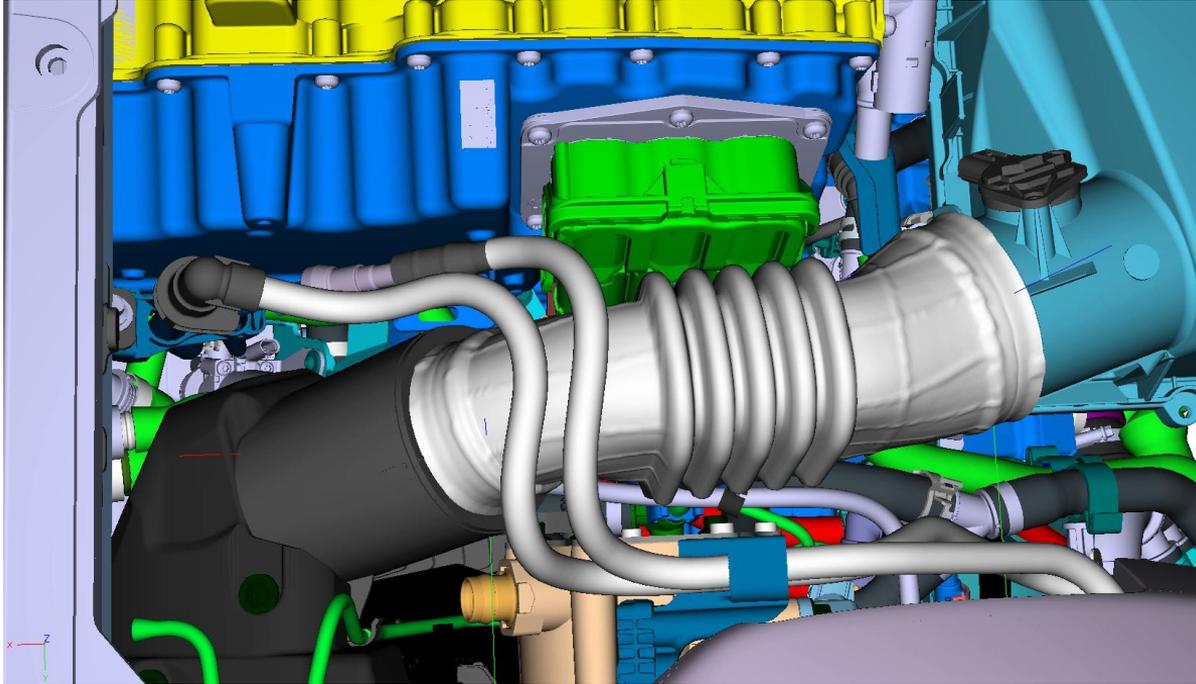
- ➔ Steg ett är att analysera en triangulerad geometri
- ➔ Linjär algebra: Skärning mellan plan och trianglar

# Simulering av kablar: parameterisera bälgar

- ➔ Hitta centrum kurva
- ➔ Hitta inner och ytter trianglar
- ➔ Använd cylindriska koordinater

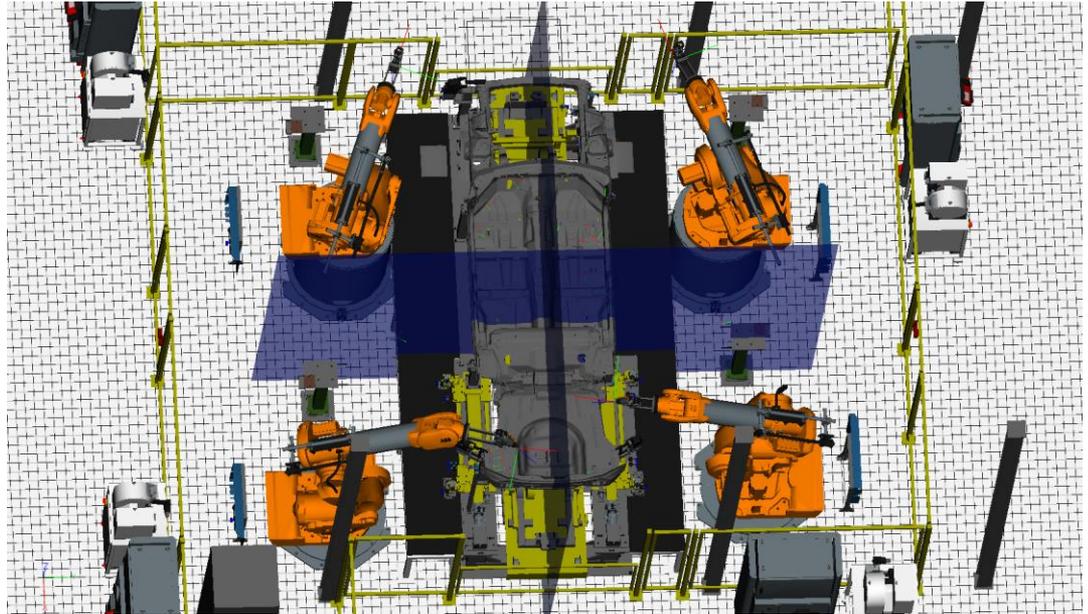


# Simulering av kablar: parameterisera bälgar

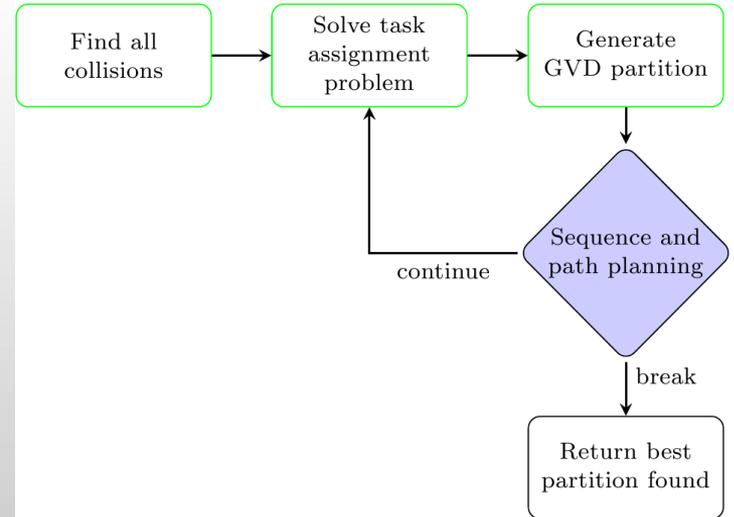
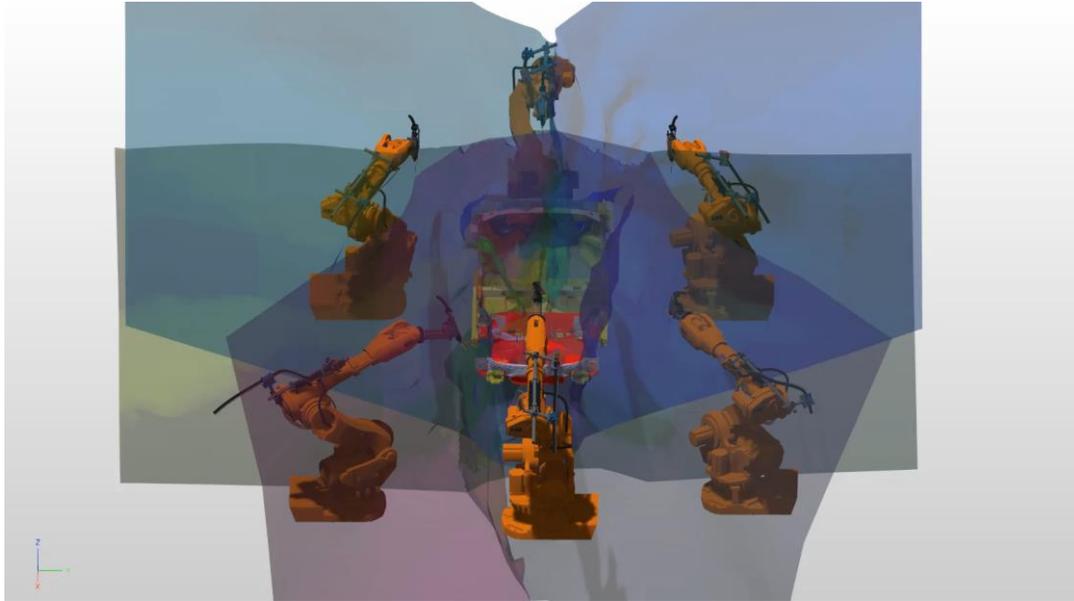


# Mitt exjobb på FCC: Disjunkta robotprogram

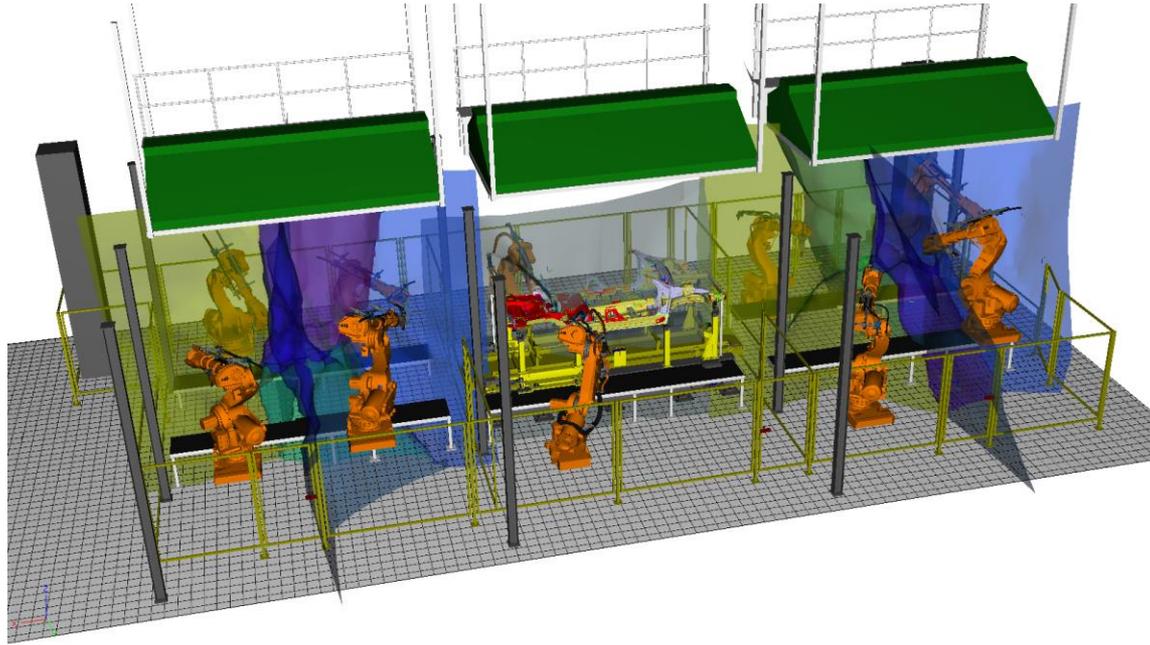
- ➔ **Fördel:** Koordinering behövs inte för att undvika kollision.
- ➔ **Nackdel:** Kanske ökar cykeltiden
- ➔ **Mitt exjobb:** Hur skapas dessa ytor och är det en bra approach?



# Mitt exjobb på FCC: Disjunkta robotprogram

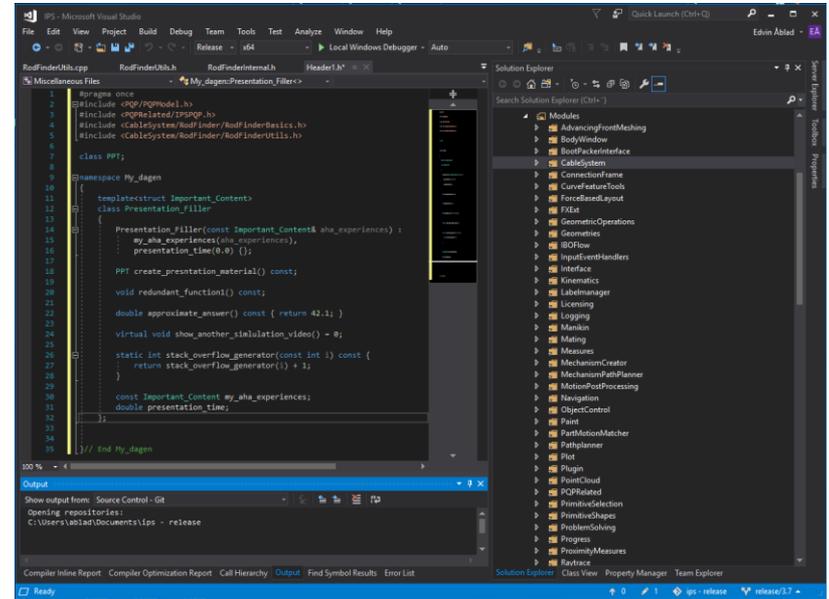


# Mitt exjobb på FCC: Disjunkta robotprogram



# Nyttigaste kurserna / lärdomarna från Chalmers

- ➔ Optimering
- ➔ Matematisk analys
- ➔ Statistisk databehandling
- ➔ Högprestanda beräkning
- ➔ Take home message:
  - ➔ Fokusera på Matematik men programmera lite också.
  - ➔ Börja brett, dvs. läs inte bara optimering =)



```
1 #pragma once
2 #include <PQR/PQModel.h>
3 #include <QP/qpModelStorage.h>
4 #include <CableSystem/RodFinder/RodFinderBasics.h>
5 #include <CableSystem/RodFinder/RodFinderUtils.h>
6
7
8 class PPT;
9
10 namespace My_dagen
11 {
12     template<struct Important_Content>
13     class Presentation_Filler
14     {
15     public:
16         Presentation_Filler(const Important_Content& aha_experiences) :
17             my_aha_experiences(aha_experiences),
18             presentation_time(0.0) {}
19
20         PPT create_presentation_material() const;
21
22         void redundant_function() const;
23
24         double approximate_answer() const { return 42.1; }
25
26         virtual void show_another_simulation_video() = 0;
27
28         static int stack_overflow_generator(const int i) const {
29             return stack_overflow_generator(i) + 1;
30         }
31
32         const Important_Content my_aha_experiences;
33         double presentation_time;
34     };
35 } // End My_dagen
```

