Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences



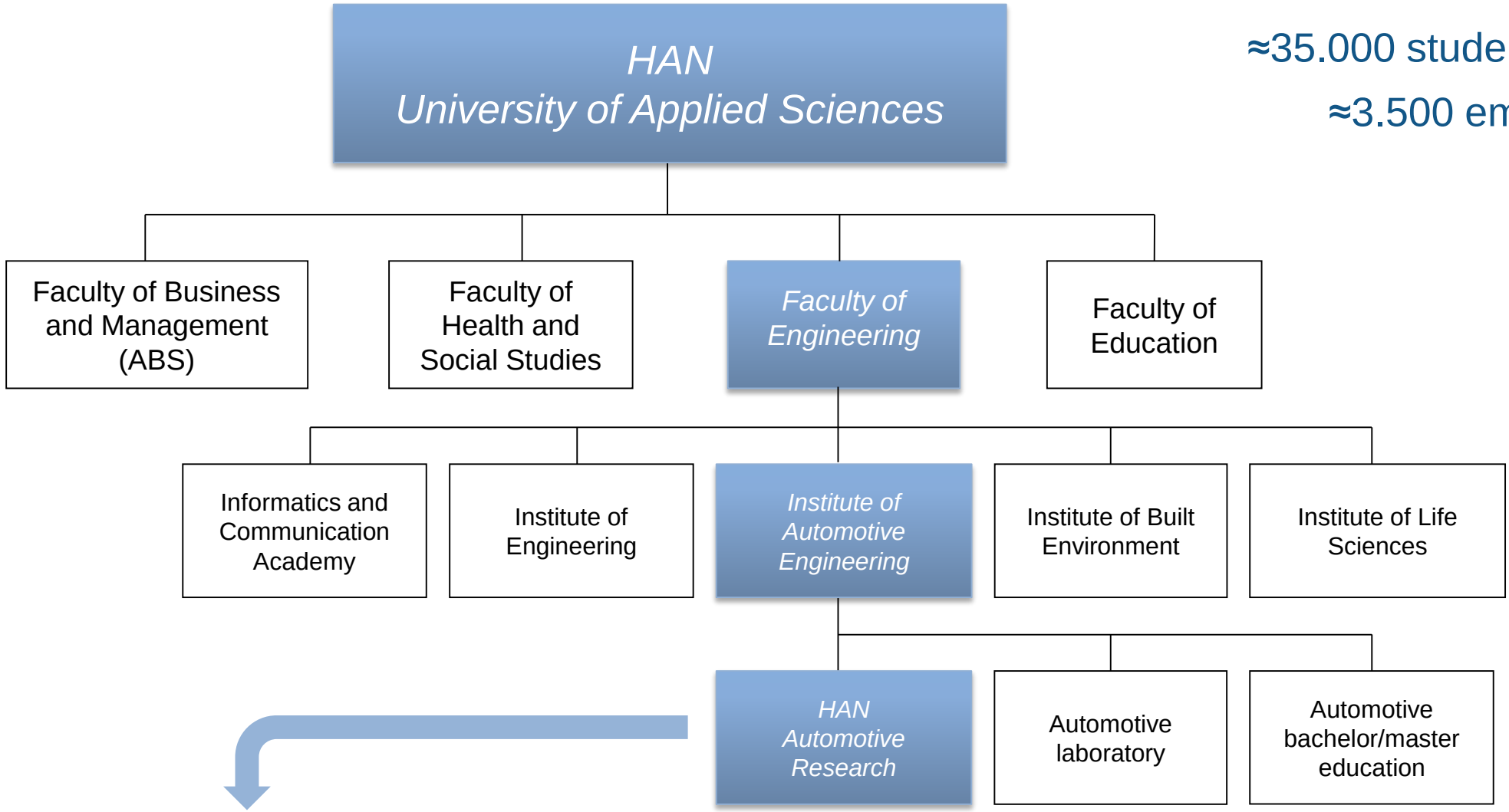
Drone assisted truck navigation

using model based development

Karel Kural & Jan Benders
HAN Automotive Research
Jan.Benders@han.nl
Karel.Kural@han.nl

HAN www.han.nl

HAN organization



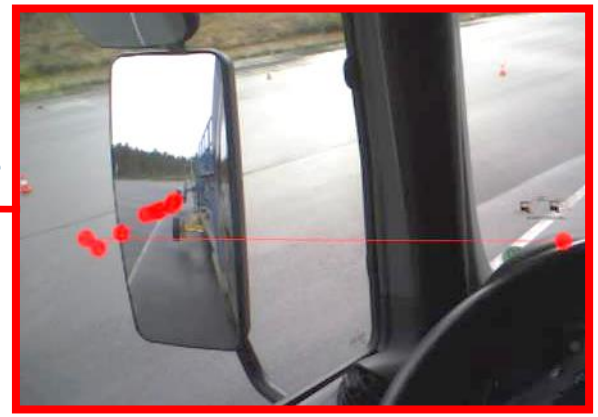
≈35.000 students

≈3.500 employees

≈20 FTE practice oriented researchers together with many students

Driver challenges while maneuvering/reversing

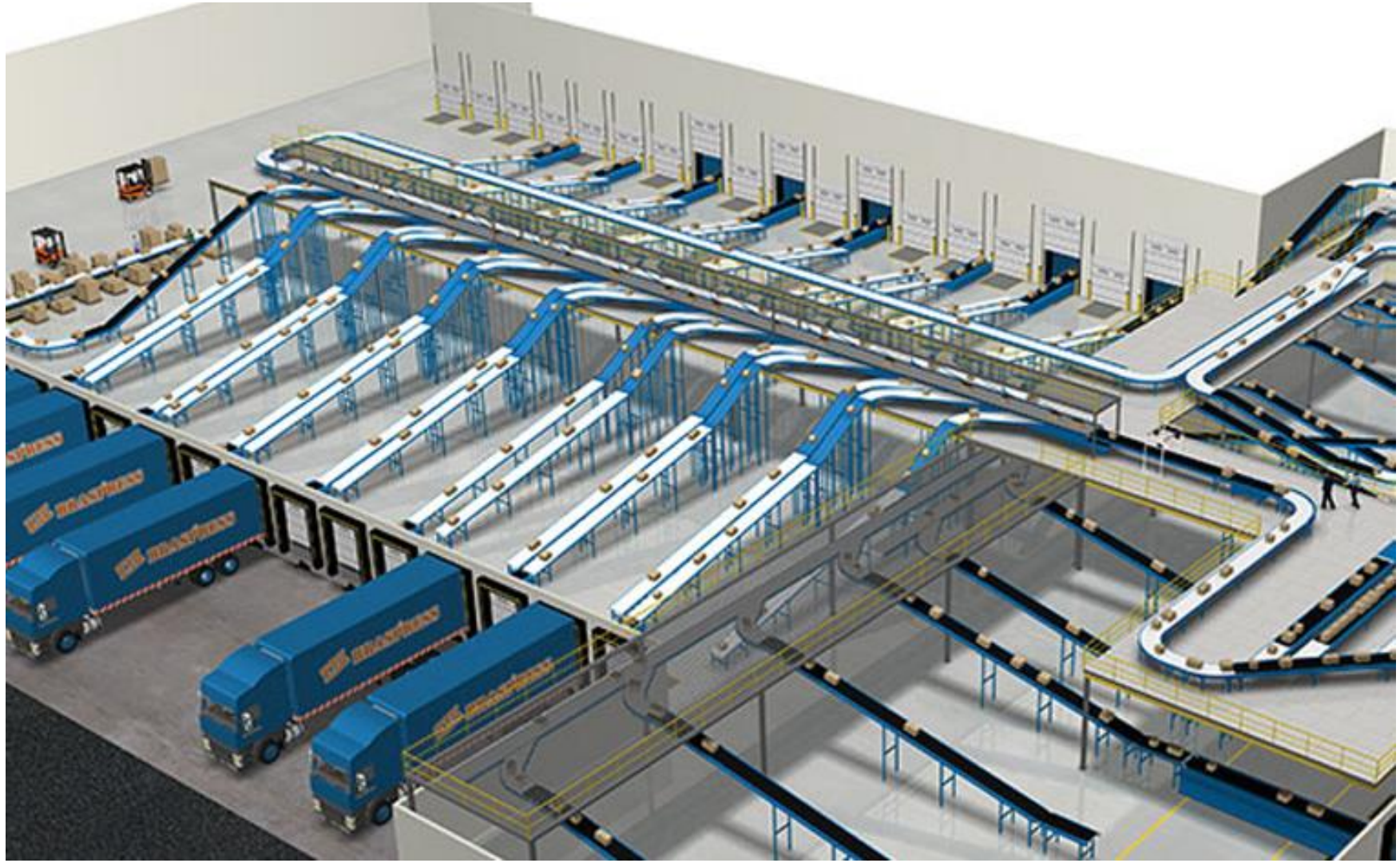
Limited field of view confirmed by eye tracking experiments



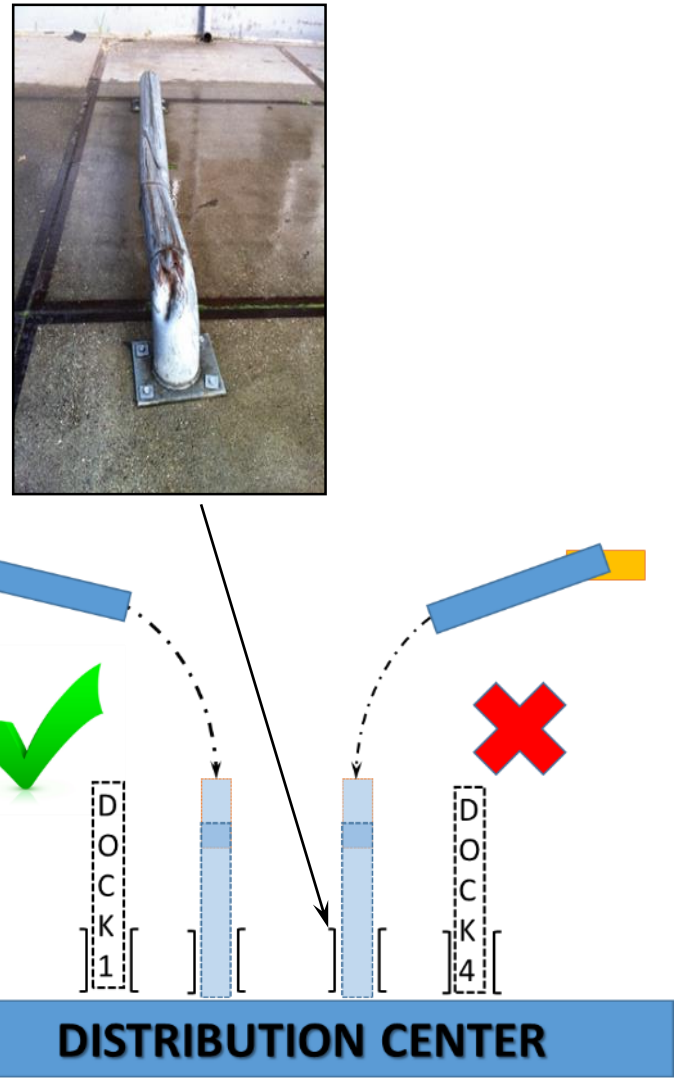
Divergent directional instability when reversing

Each vehicle combination behaves differently

Crucial maneuvering scenario: Docking



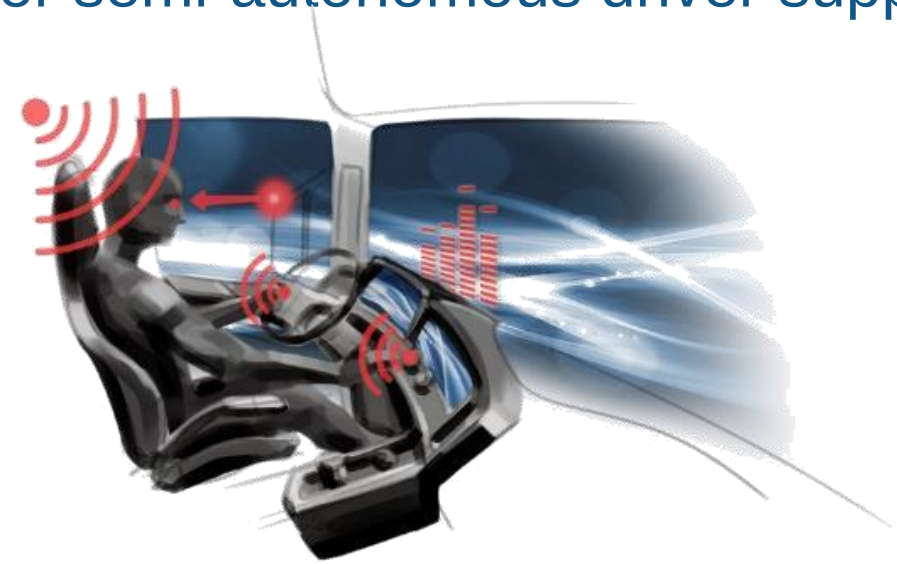
Docking: Challenges for the driver



Possible solutions

Autonomous or semi-autonomous driver support systems

but….



…we need an accurate referencing system (GPS is not sufficient)

Some alternatives:

- Magnets/transponders in the pavement
- Laser scanners at docking gates
- Drone with camera up in the air

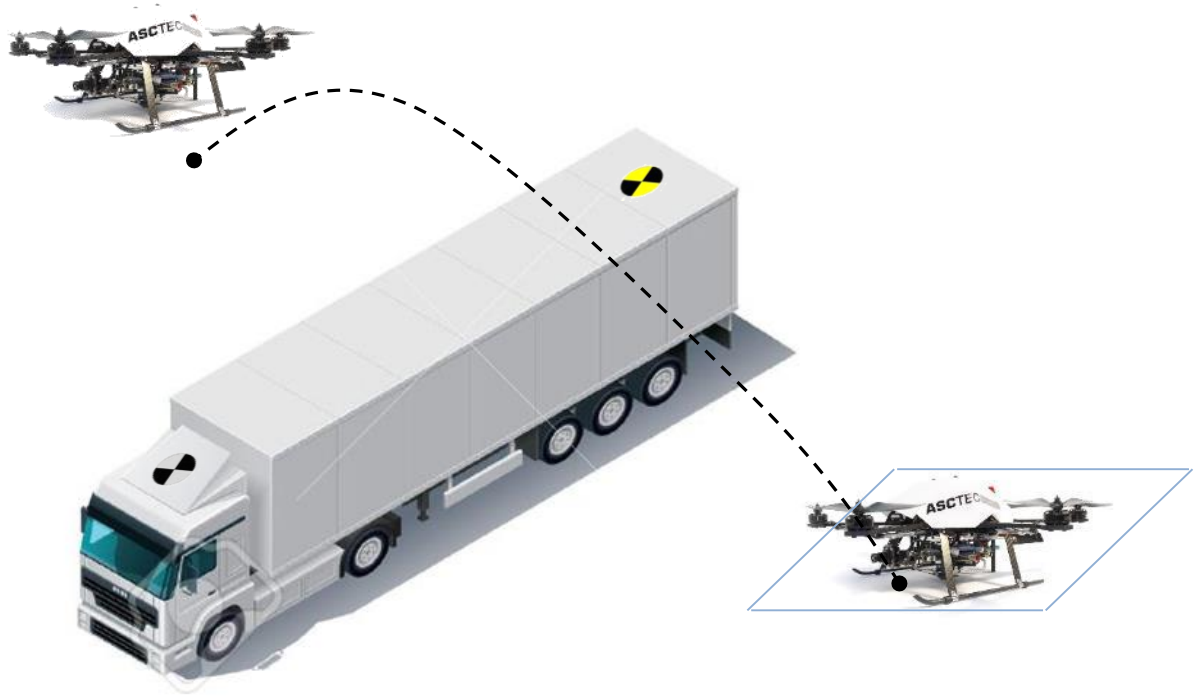
Drone functionality

- Provide the driver with top view of the situation
- Continuously monitor the position of the vehicle w.r.t. destination
- ‘Last mile’ navigation with precise backwards docking support

Conceptual function

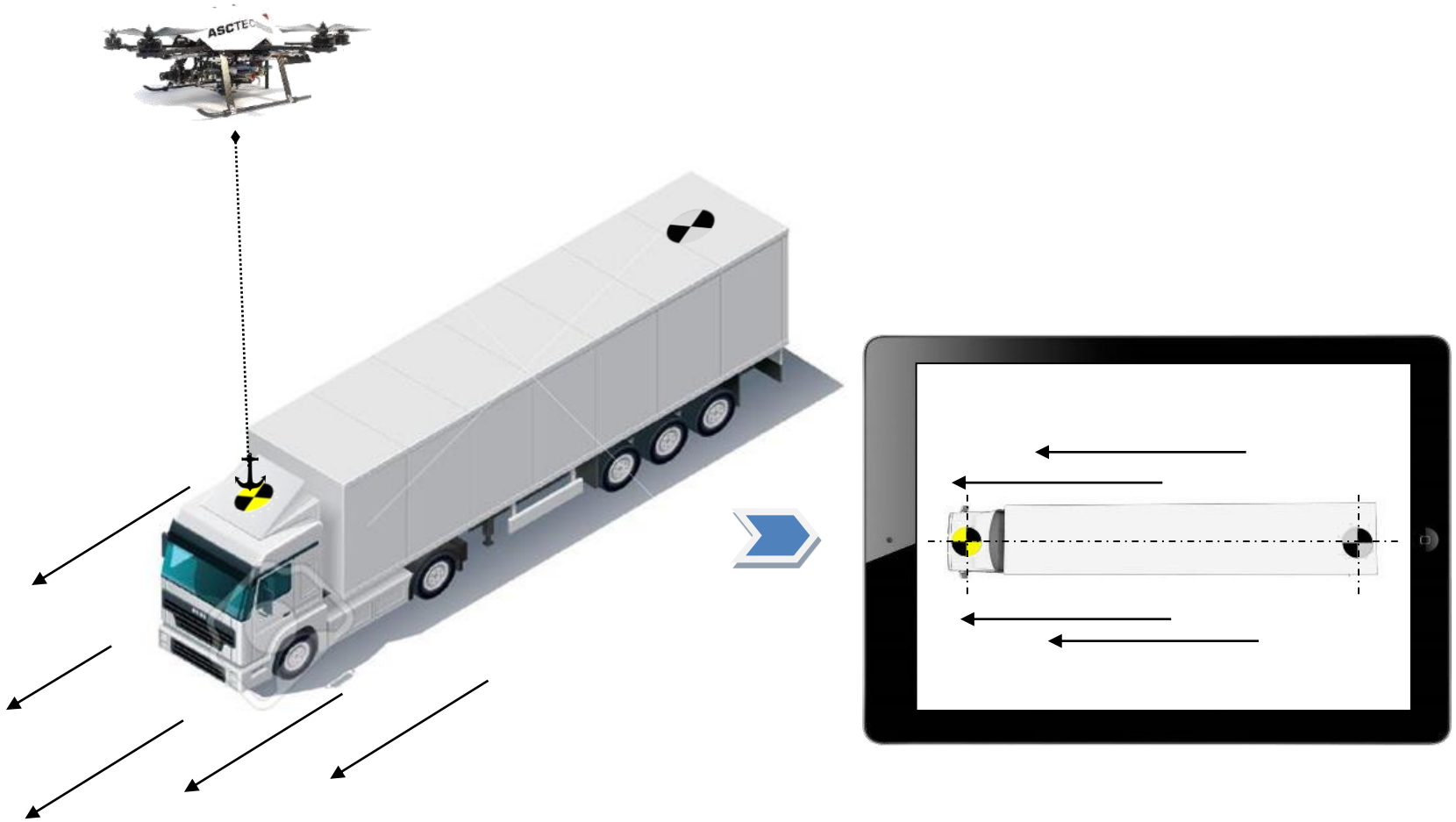
Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

Find the vehicle combination nearby DC while connected to WMS



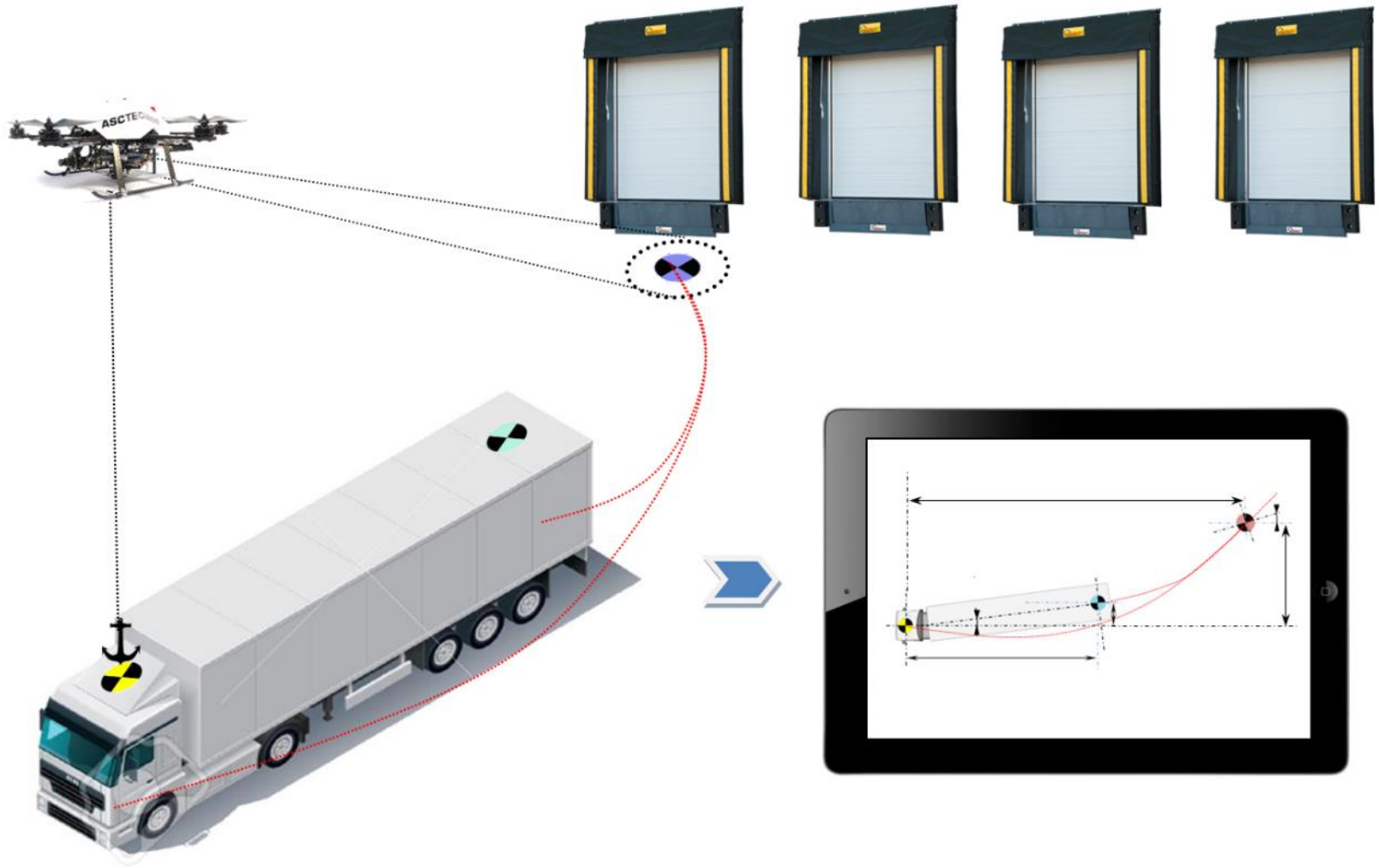
Conceptual function

Anchor & navigate the truck nearby the gate (GPS)



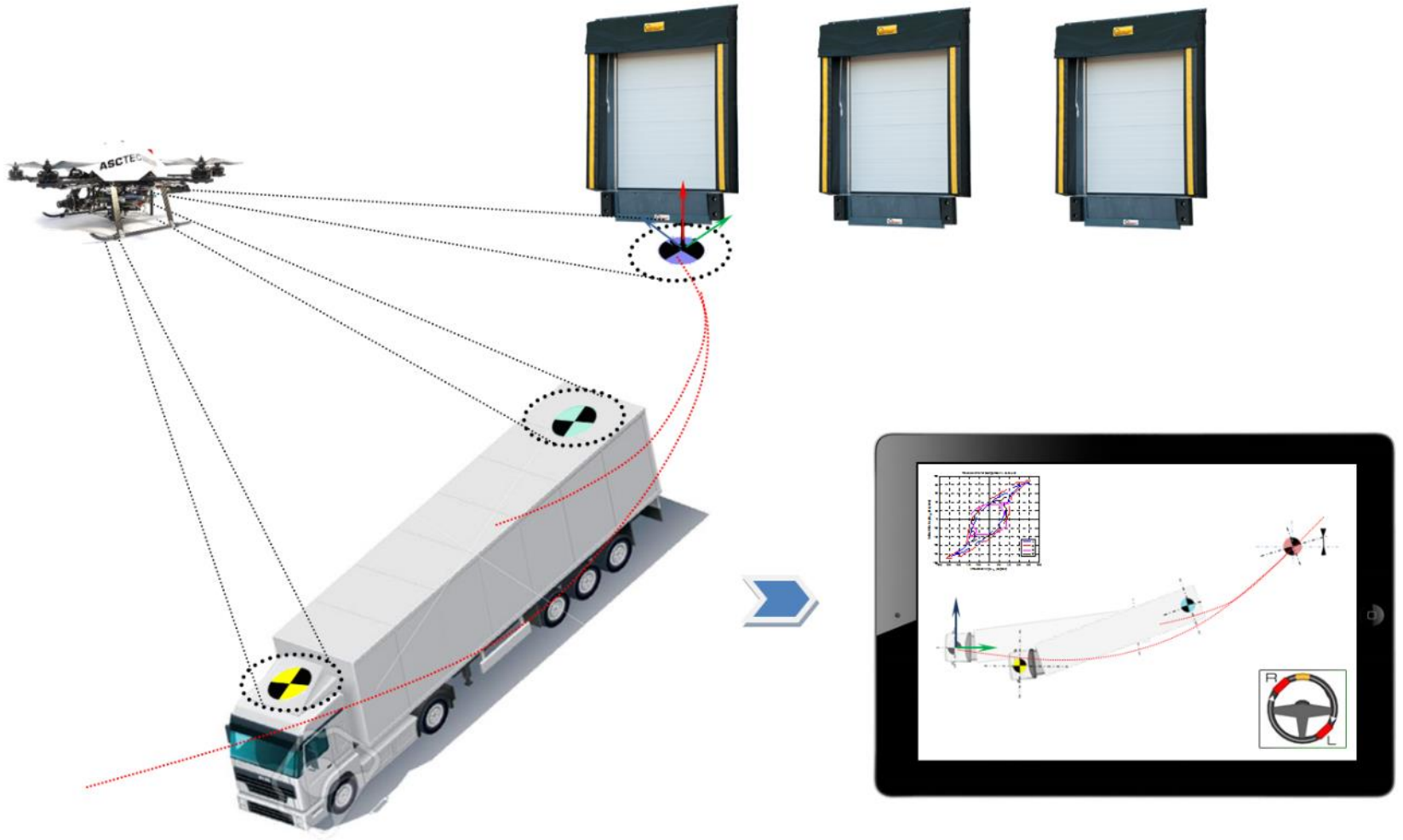
Conceptual function

Detect the gate → Measure mutual position → Plan path



Conceptual function

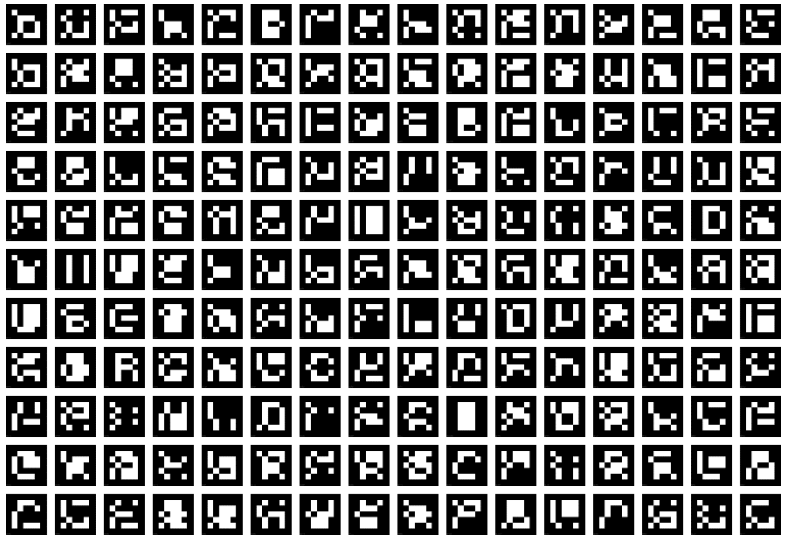
(Supported semi-) Autonomous path-following

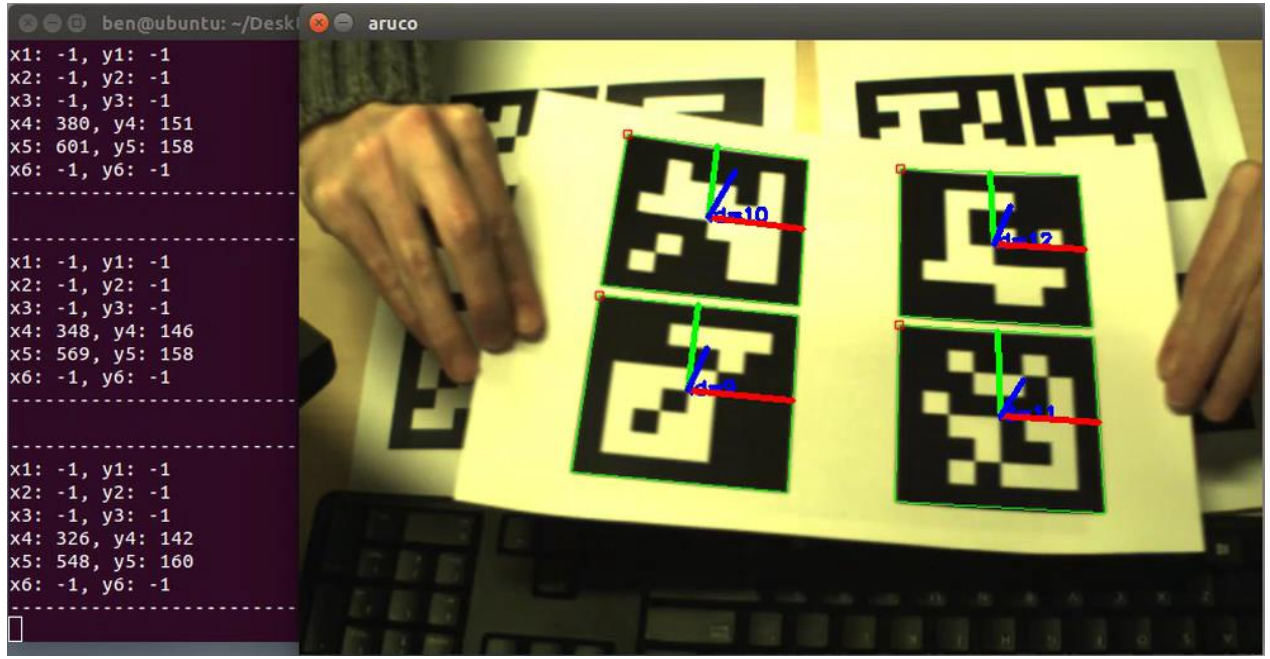


Drone functions: Vehicle localization

- Using computer vision

- Temporary solution: ArUco* markers
  - Identification
  - Localization

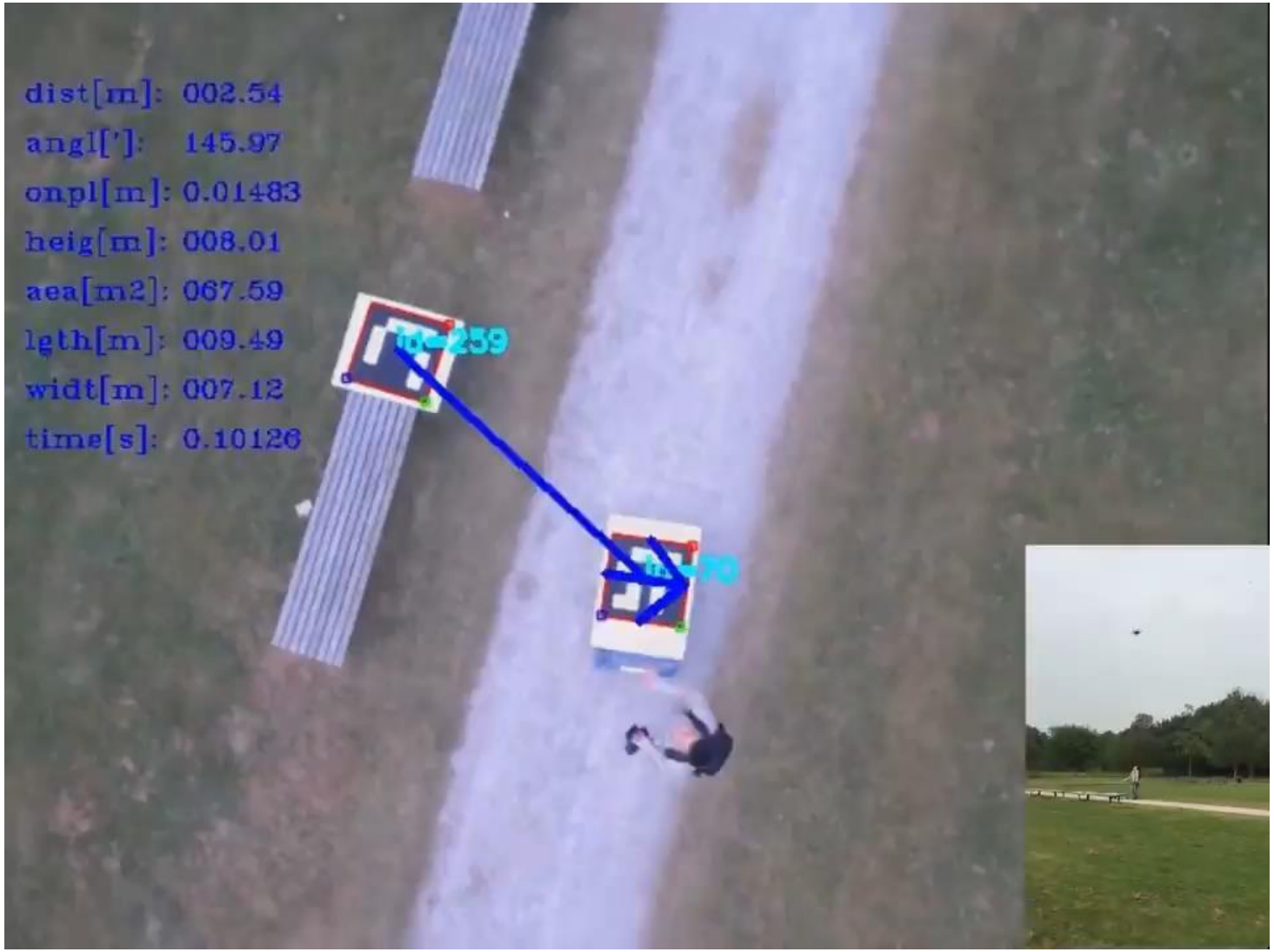




Drone functions: Vehicle localization

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

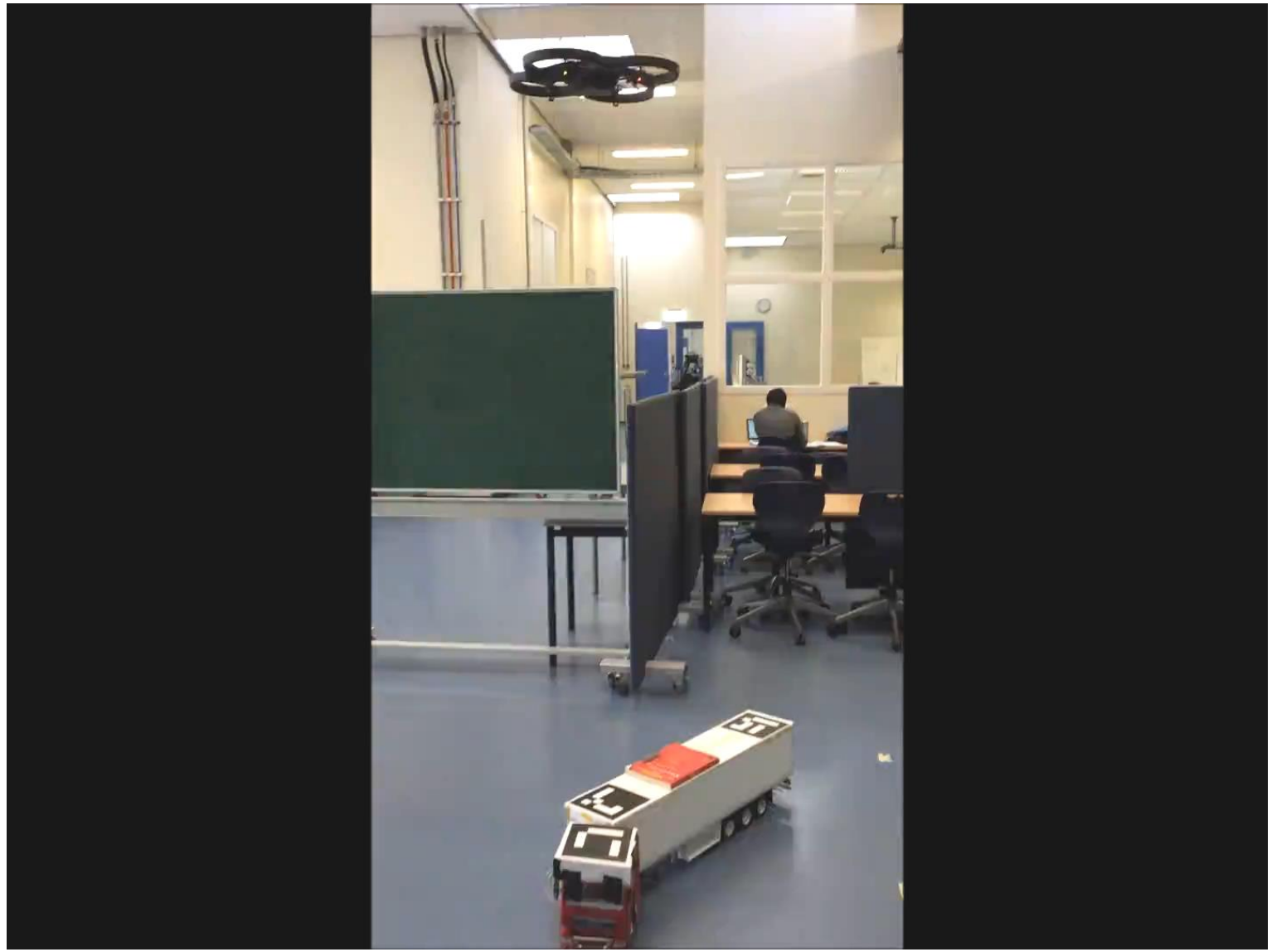
[Demo ArUco detection and following by drone]



Drone functions: Vehicle localization

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

Demo of drone following a truck



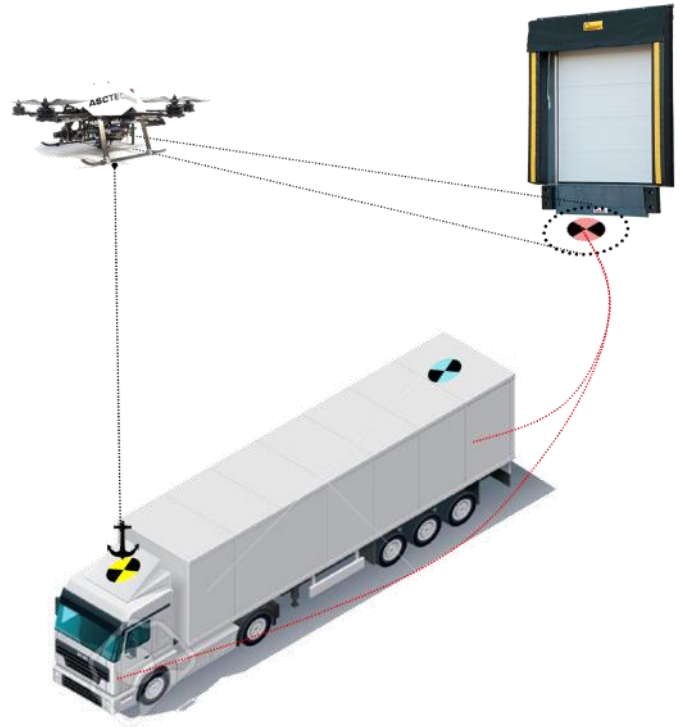
Drone functions: Docking path planner & controller

- Planner role:

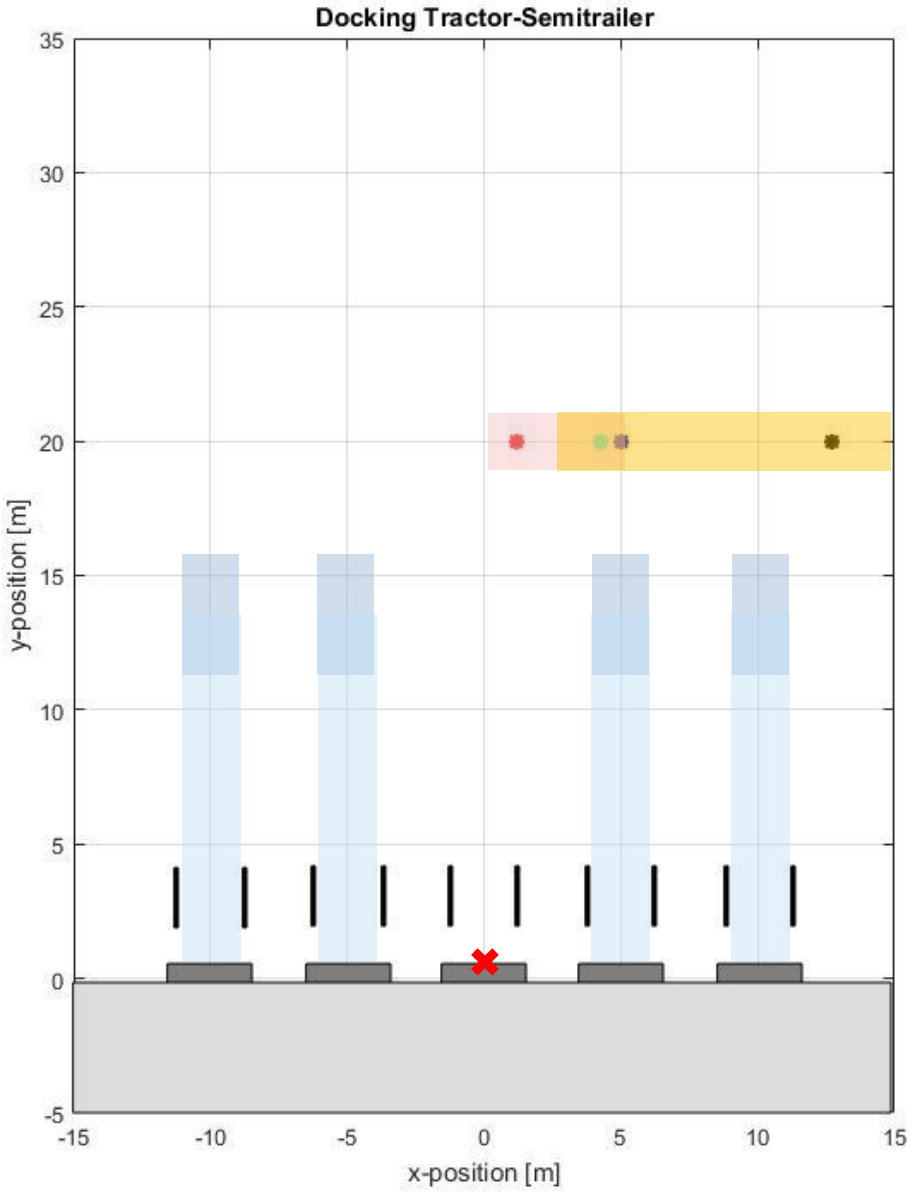
  To generate a path between initial position and the destination that is:
  - Continuous
  - Kinematically possible
  - Efficient (minimize maneuvering space)

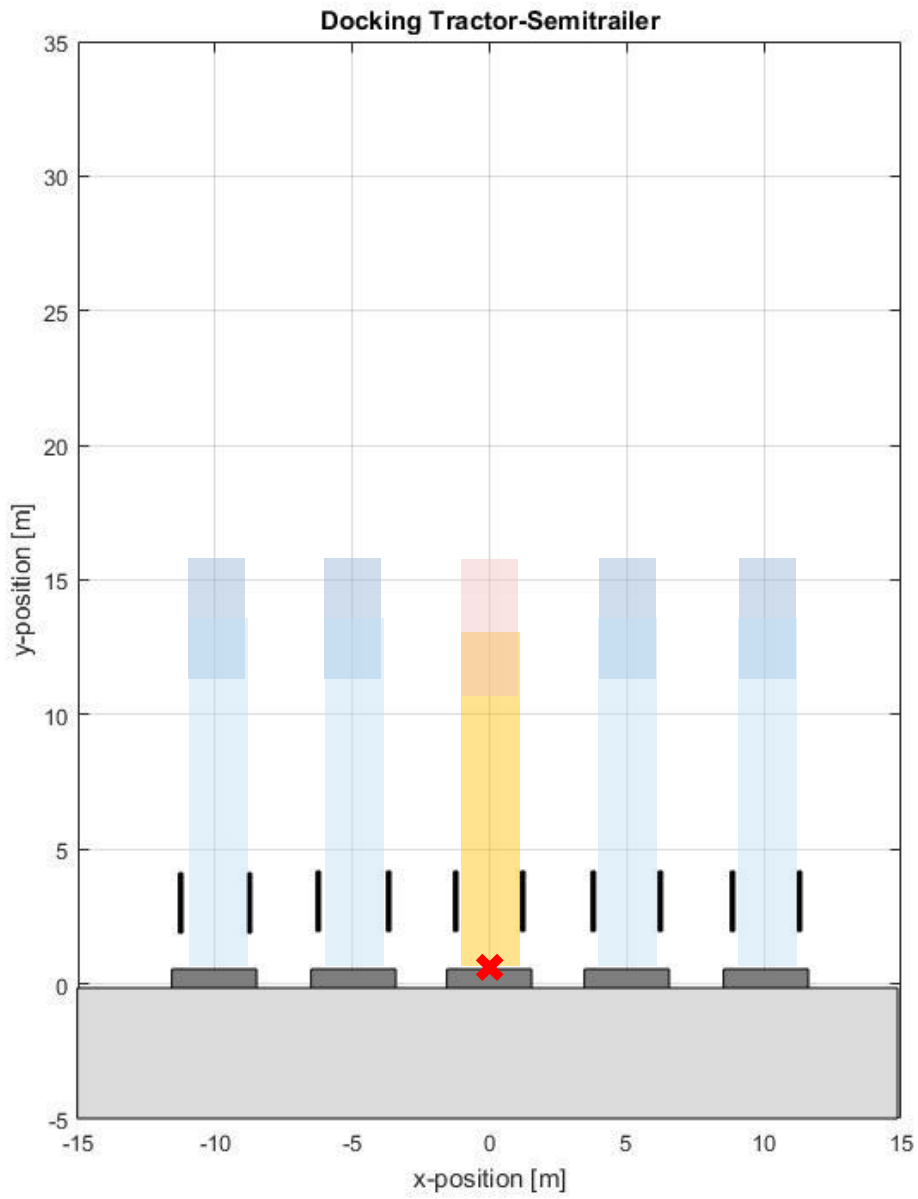
- Controller role:

  To navigate the vehicle over predefined path:
  - Forward and reverse
  - Minimize tracking error
  - Stability

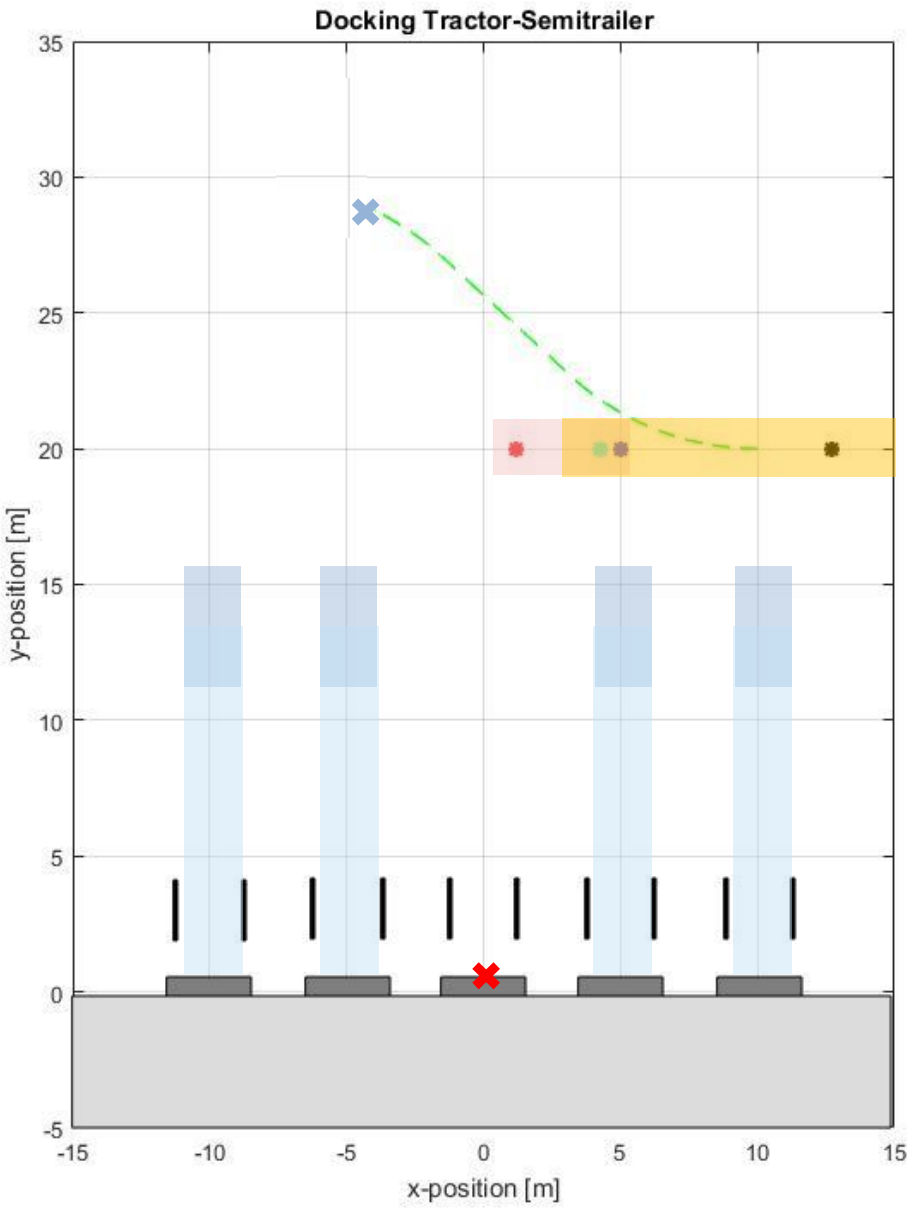


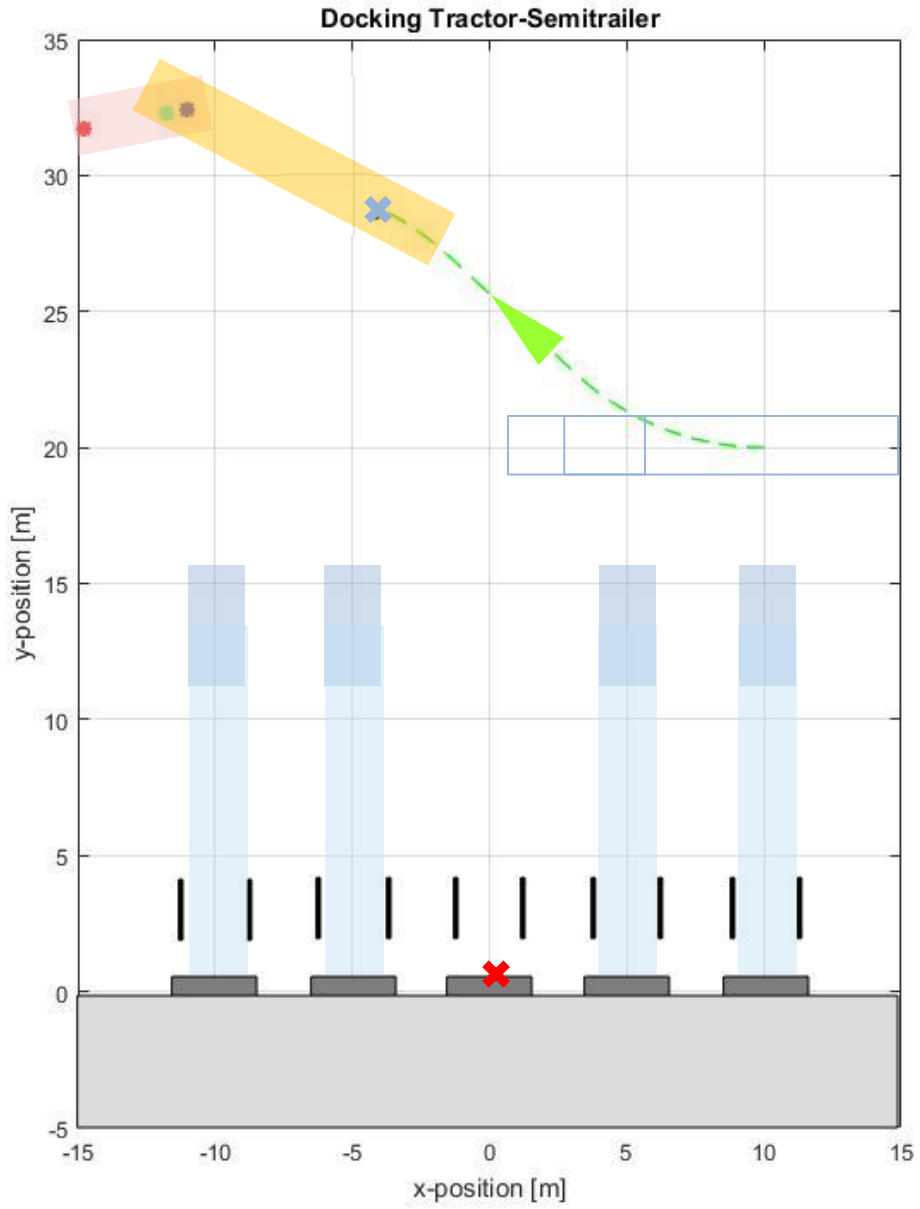
Path planner



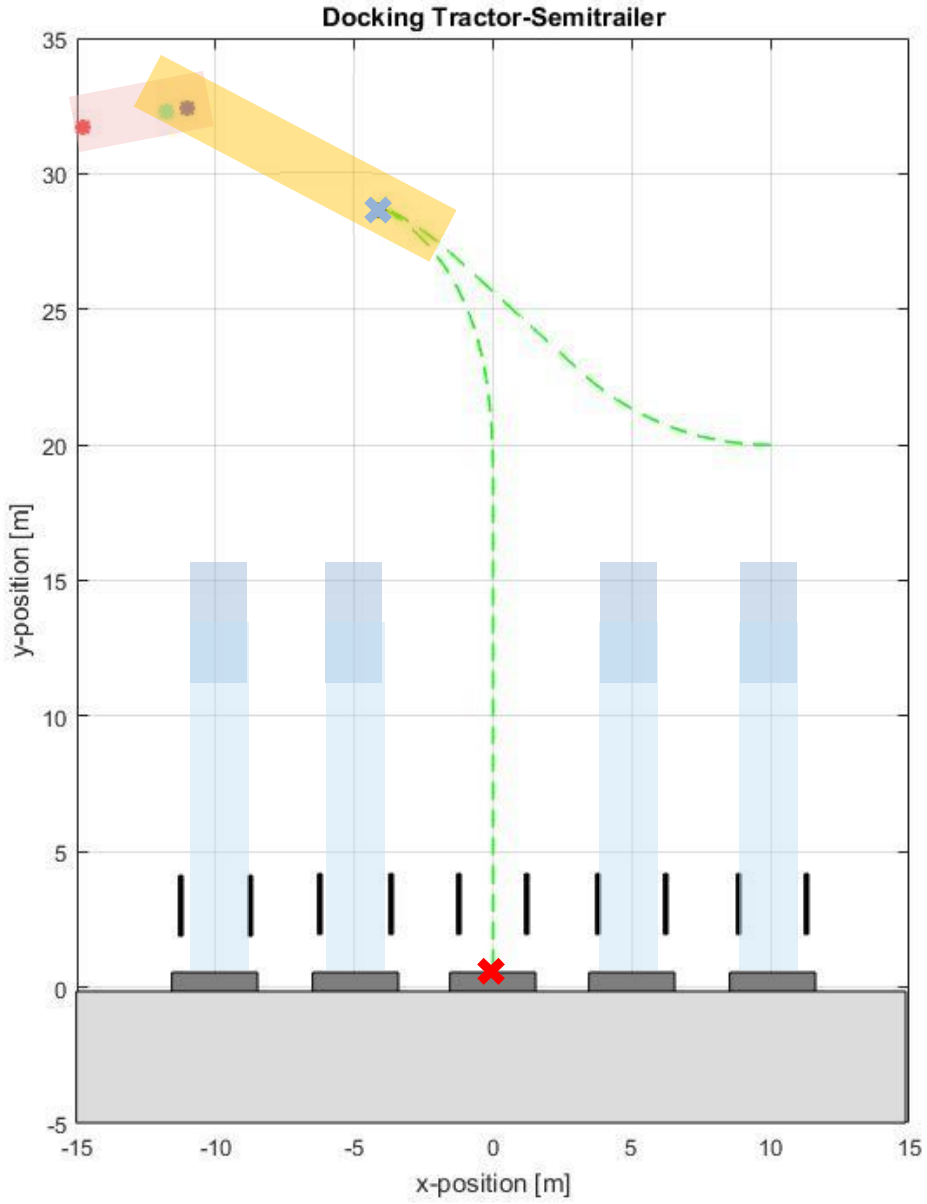


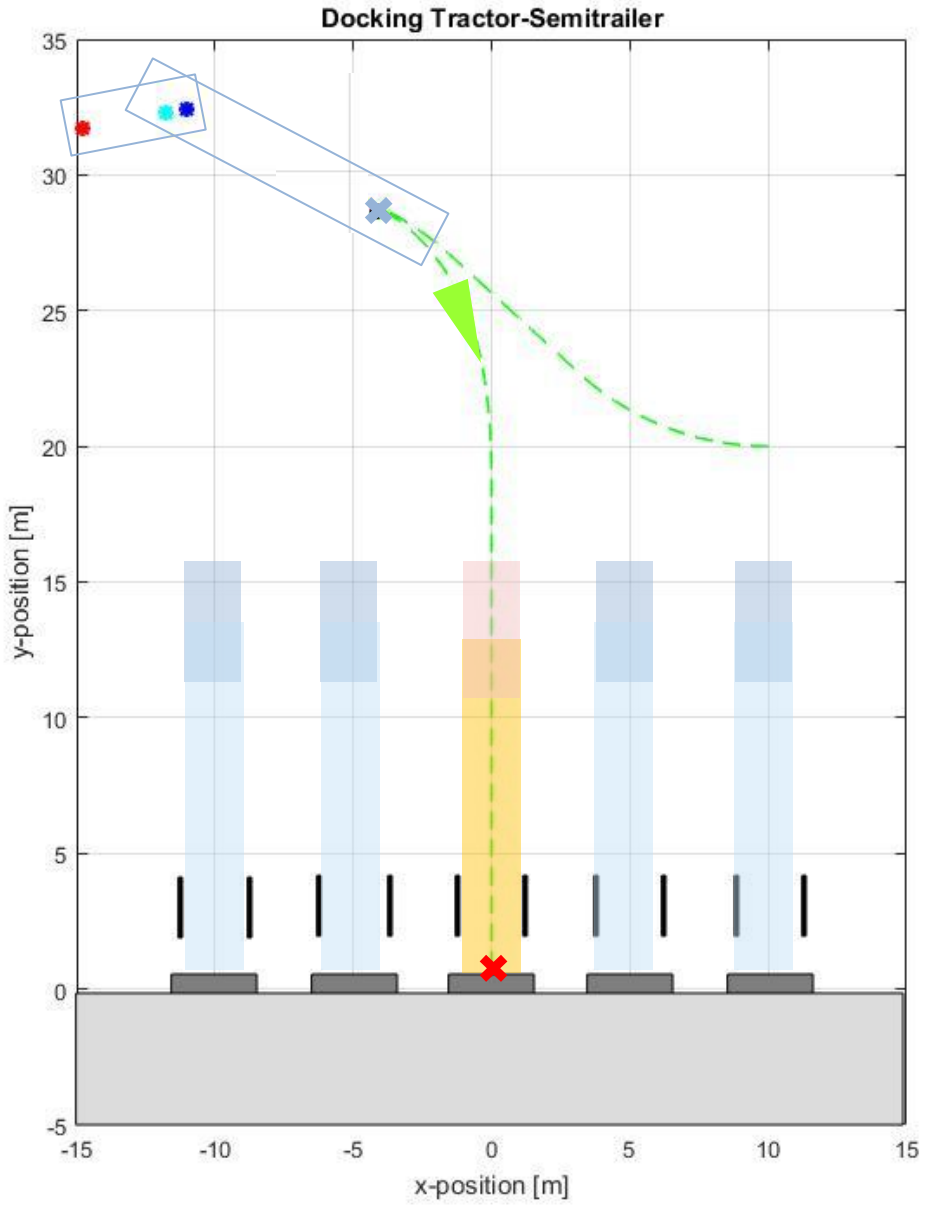
Path planner



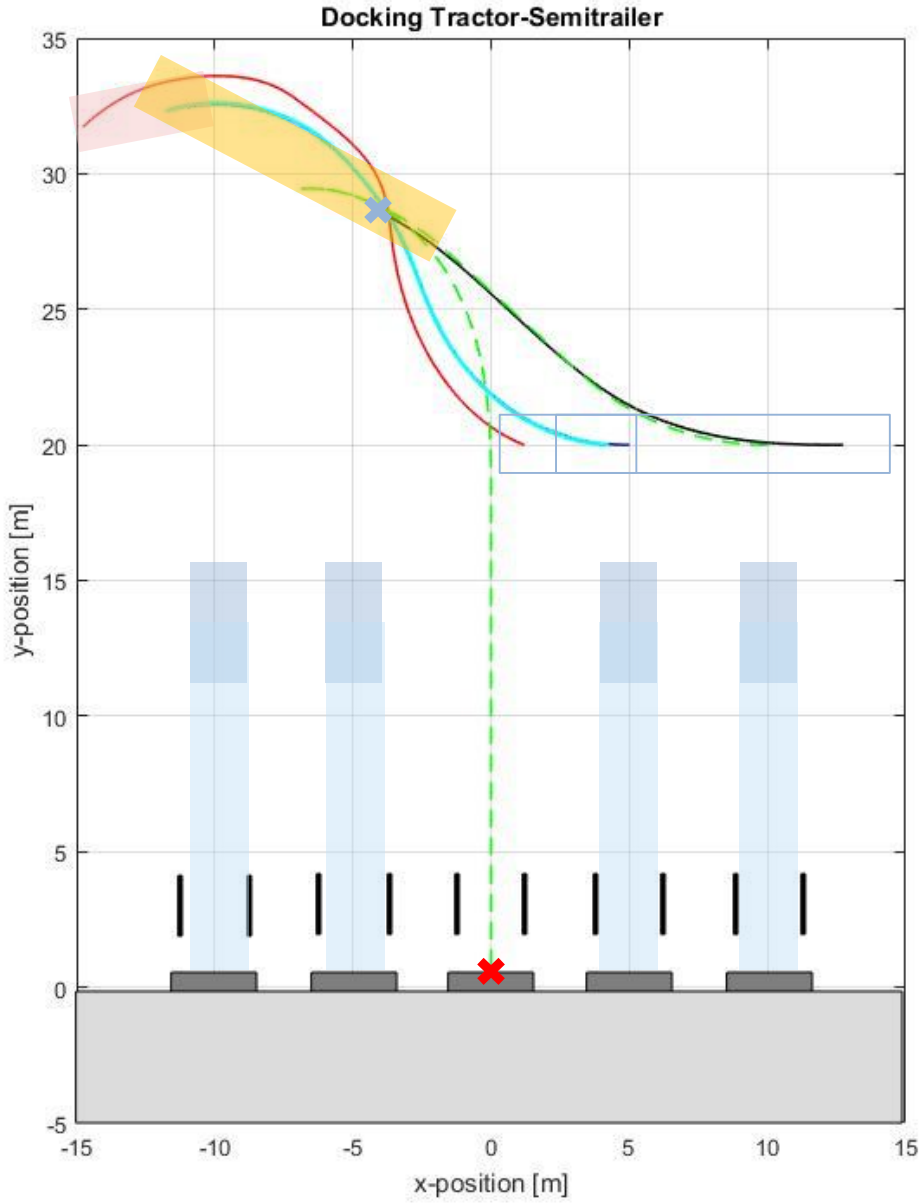


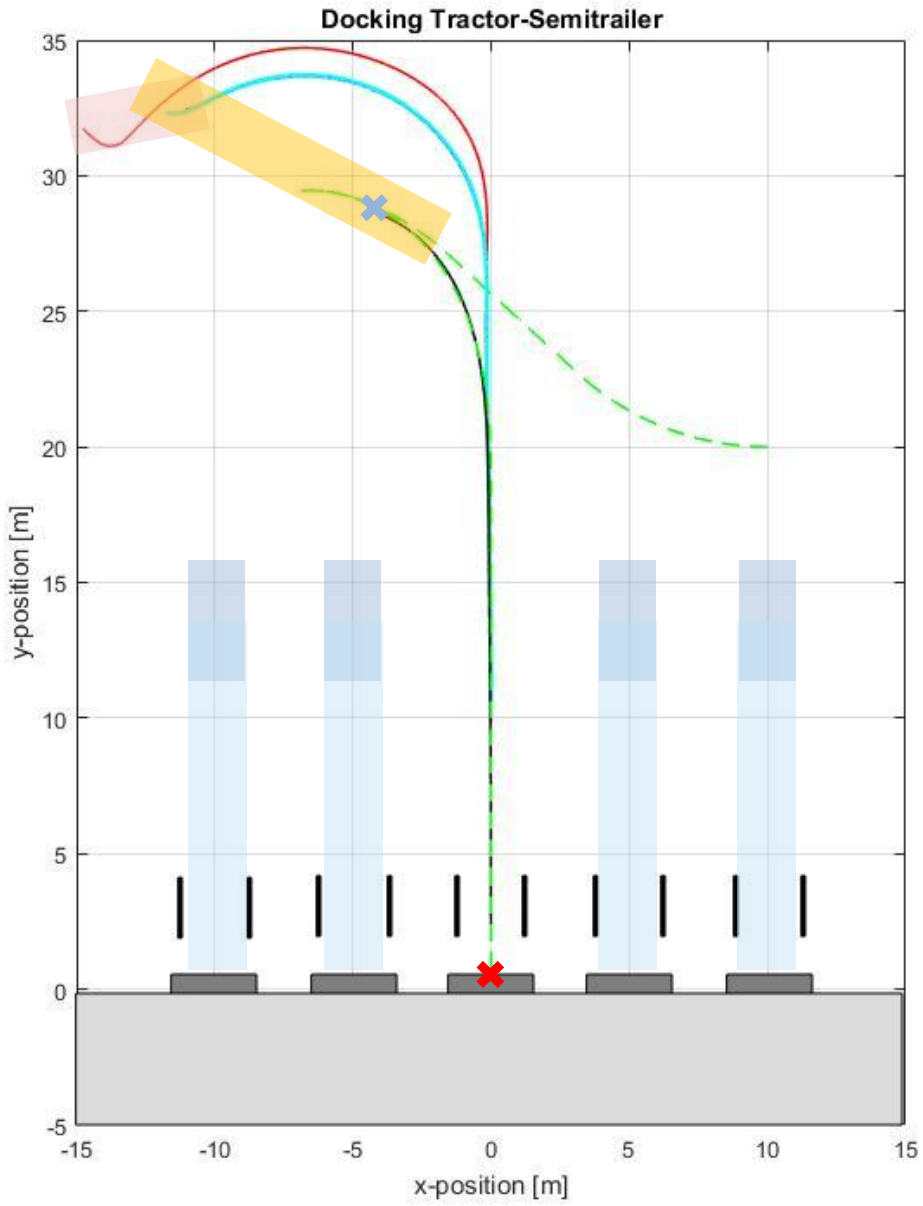
Path planner





Controller function

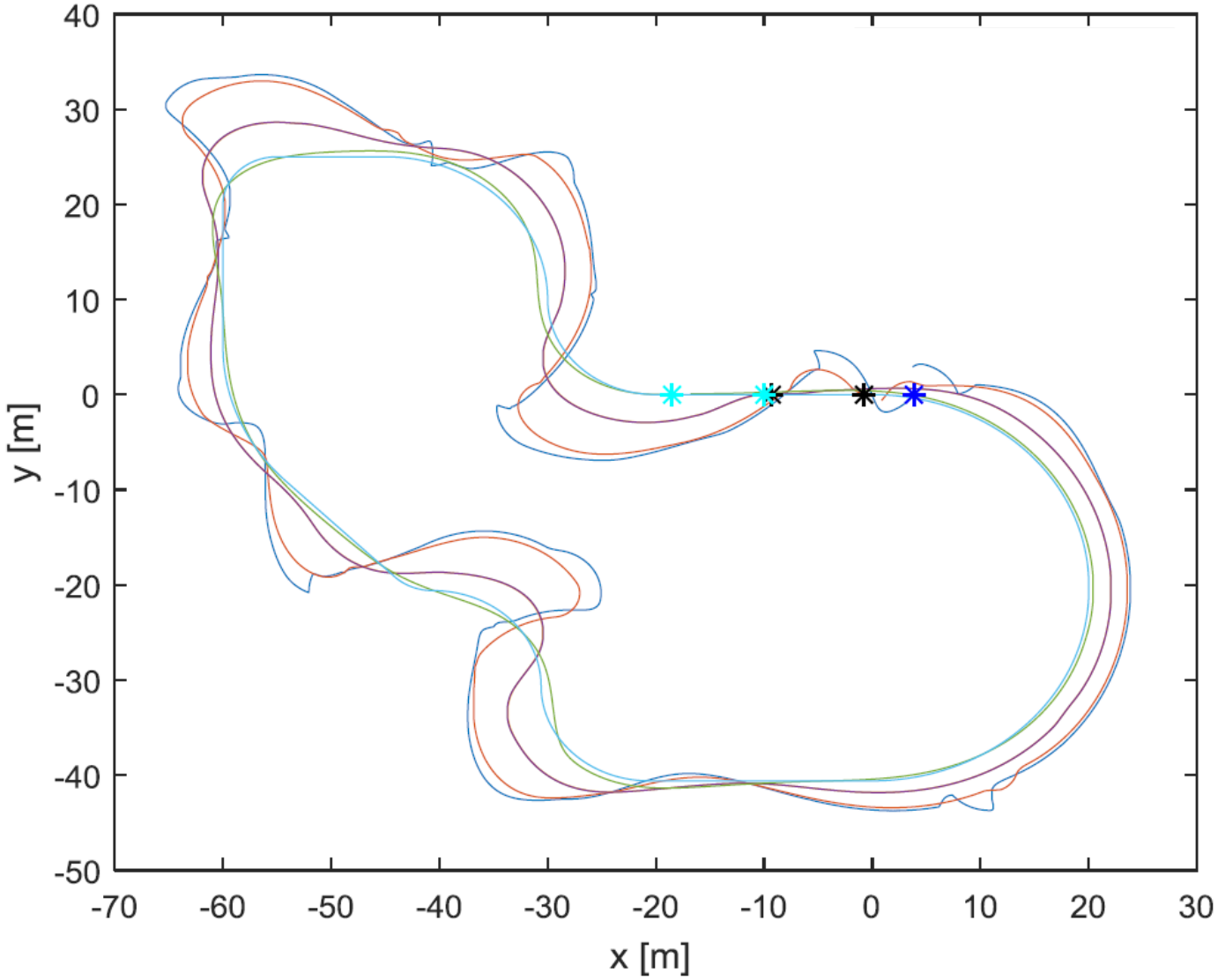


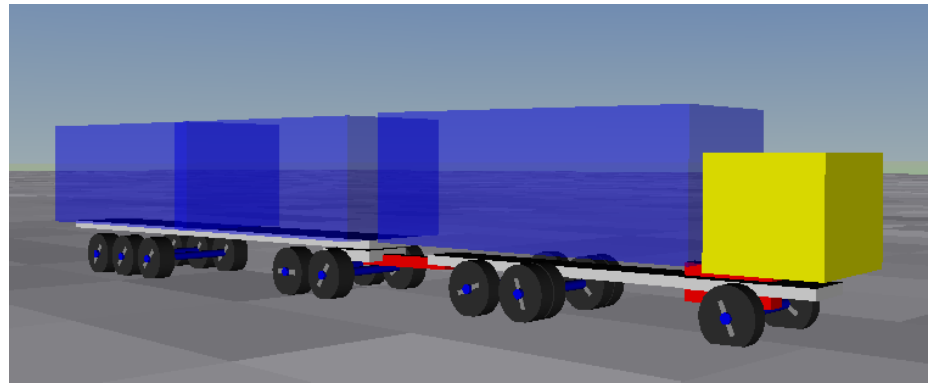


Next step: double articulated vehicle

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

Verification on reversing of double articulated vehicle (LZV)





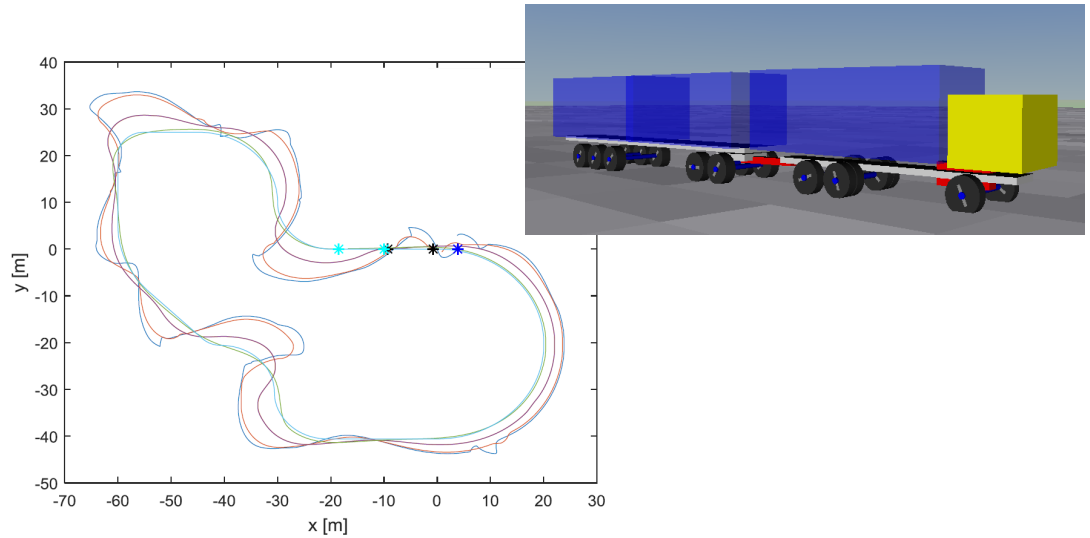
Scaled truck demonstration

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

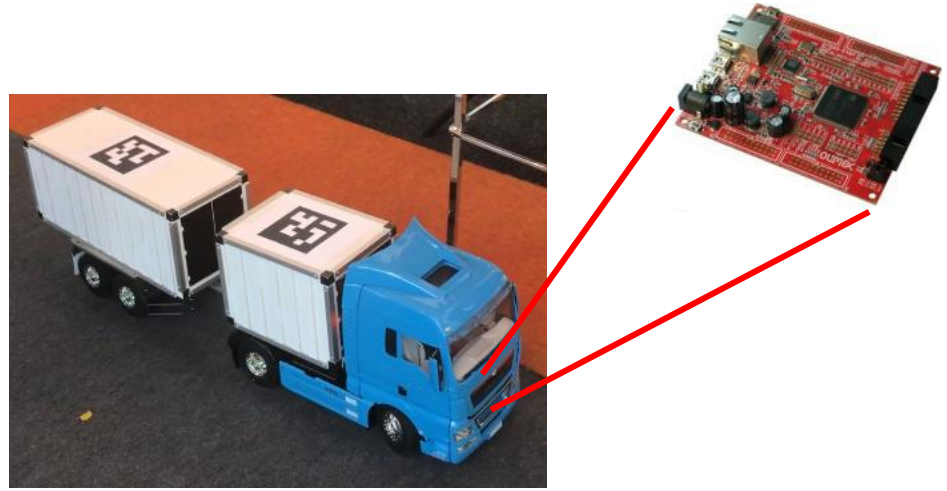
Autodocking demonstration

Model Based Development (MBD)

- Allows development of planner and control algorithms via simulation
  - No hardware required

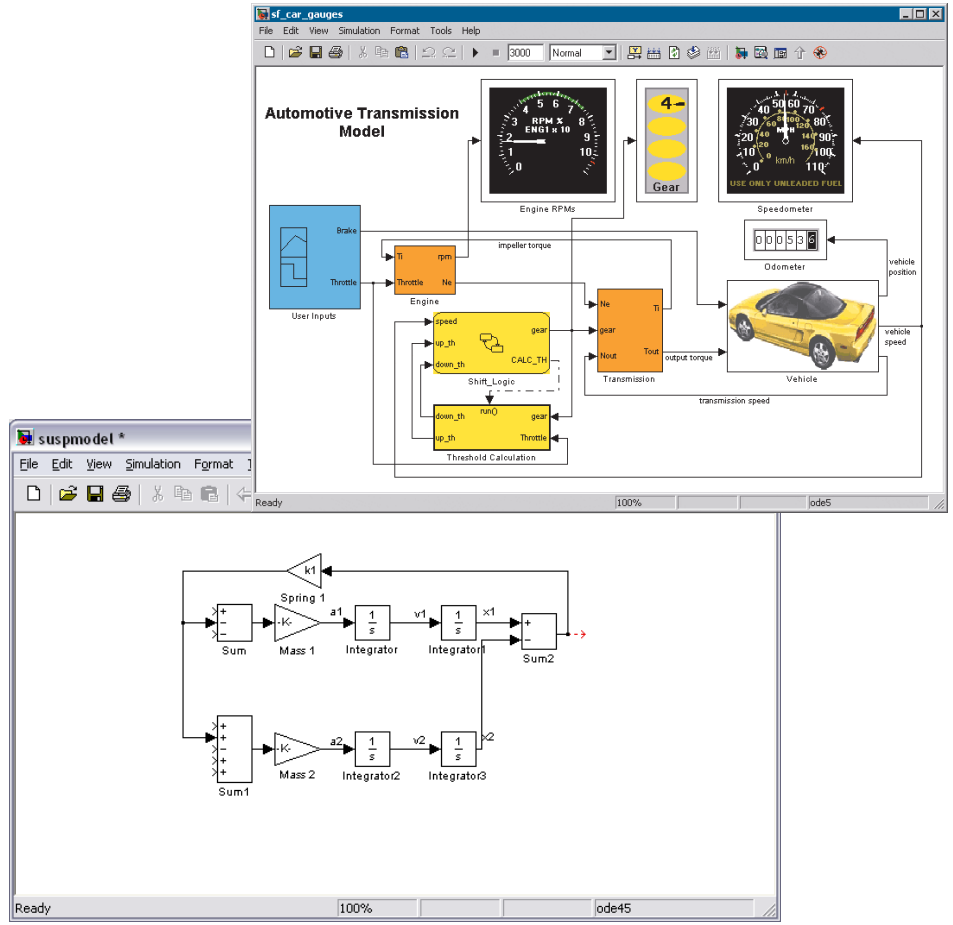


- Allows control engineers to program control system
  - No programming skills required



Programming for engineers


```
1    function ye = kalmanf(A,B,C,Q,R,u,t,yv) %#eml
2    P = B*Q*B';                    % Initial error covariance
3    x = zeros(size(B));            % State initial condition
4    ye = zeros(length(t),1);
5    errcov = zeros(length(t),1);
6    for i=1:length(t)
7      % Measurement update
8      Mn = P*C'/(C*P*C'+R);
9      x = x + Mn*(yv(i)-C*x);      % x[n|n]
10     P = (eye(size(A))-Mn*C)*P;   % P[n|n]
11     % Compute output
12     ye(i) = C*x;
13     errcov(i) = C*P*C';
14     % Time update
15     x = A*x + B*u(i);             % x[n+1|n]
16     P = A*P*A' + B*Q*B';          % P[n+1|n]
17   end
```



Source: Mathworks

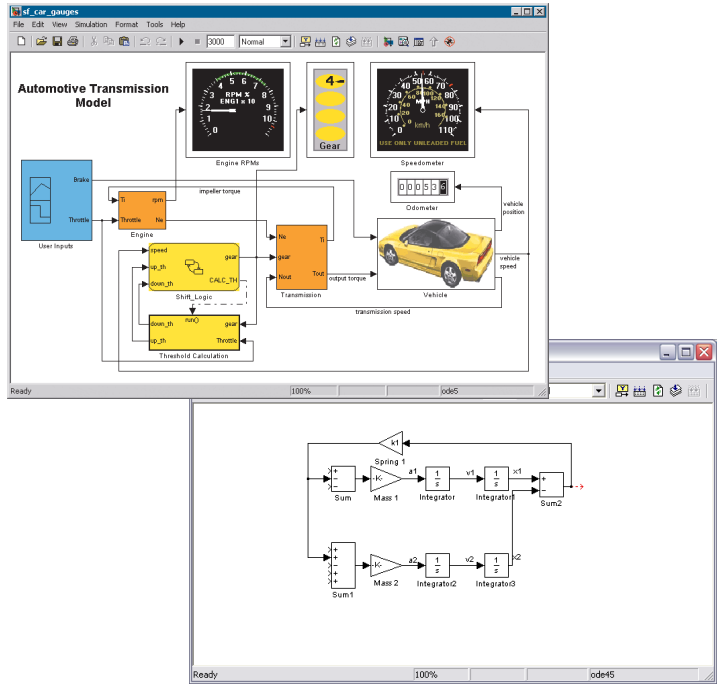
No complex code

but intuitive blocks

Automotive standard for control algorithm development

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

Model



Specification

clear communication with customer
and between engineers (in various disciplines)

Documentation

visual , self-documenting

Implementation

can be translated in an executable program

Basis for model based development methodology using HAN tools

HAN MBD tools

HAN coder

Various target hardware platforms

HAN tune

Supported hardware platforms

Hogeschool van Arnhem en Nijmegen
HAN University of Applied Sciences

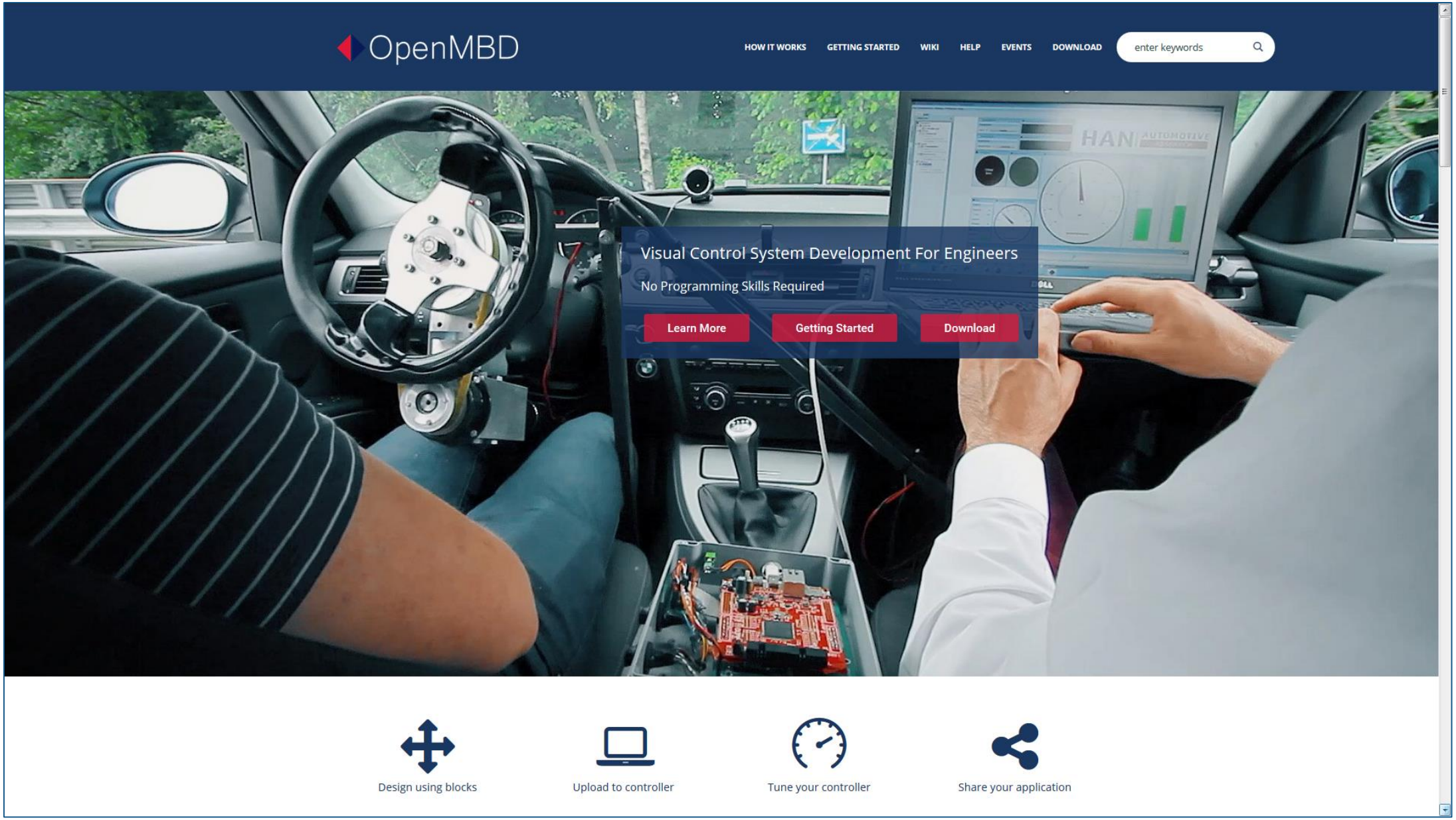
STM32 Target	RC30 Target	Prodrive GCU Target	Coming… Linux Target
			
Supported hardware Olimexino-STM32 STM32-E407 STM32-P405	Supported hardware RC36-20/30 RC28-14/30 RC12-10/30	Supported hardware Prodrive GCU2420	Hardware Lely GCB (NXP i.MX6)

From playground to robust and full scale

STM32 Target

RC30 Target

www.openMBD.com



Thank you for your attention