CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

Future Mobility & Machine Learning

Self-Driving Cars

Intelligent Traffic Control

Kim G Larsen

Jakob H Taankvist, Marco Muniz, Andreas B Eriksen, Mikkel Færgeman, Marius Mikucions, ..

Self-Driving Cars

- Technologies have reached levels of sophistication that enable autonomous vehicles.
- By April 2014, Google's self-driving vehicles had logged more than 700,000 accident-free miles.
- Tesla cars come with an autopilot that is a pre-cursor to autonomous driving and with the necessary hardware.

DiCyPS
CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

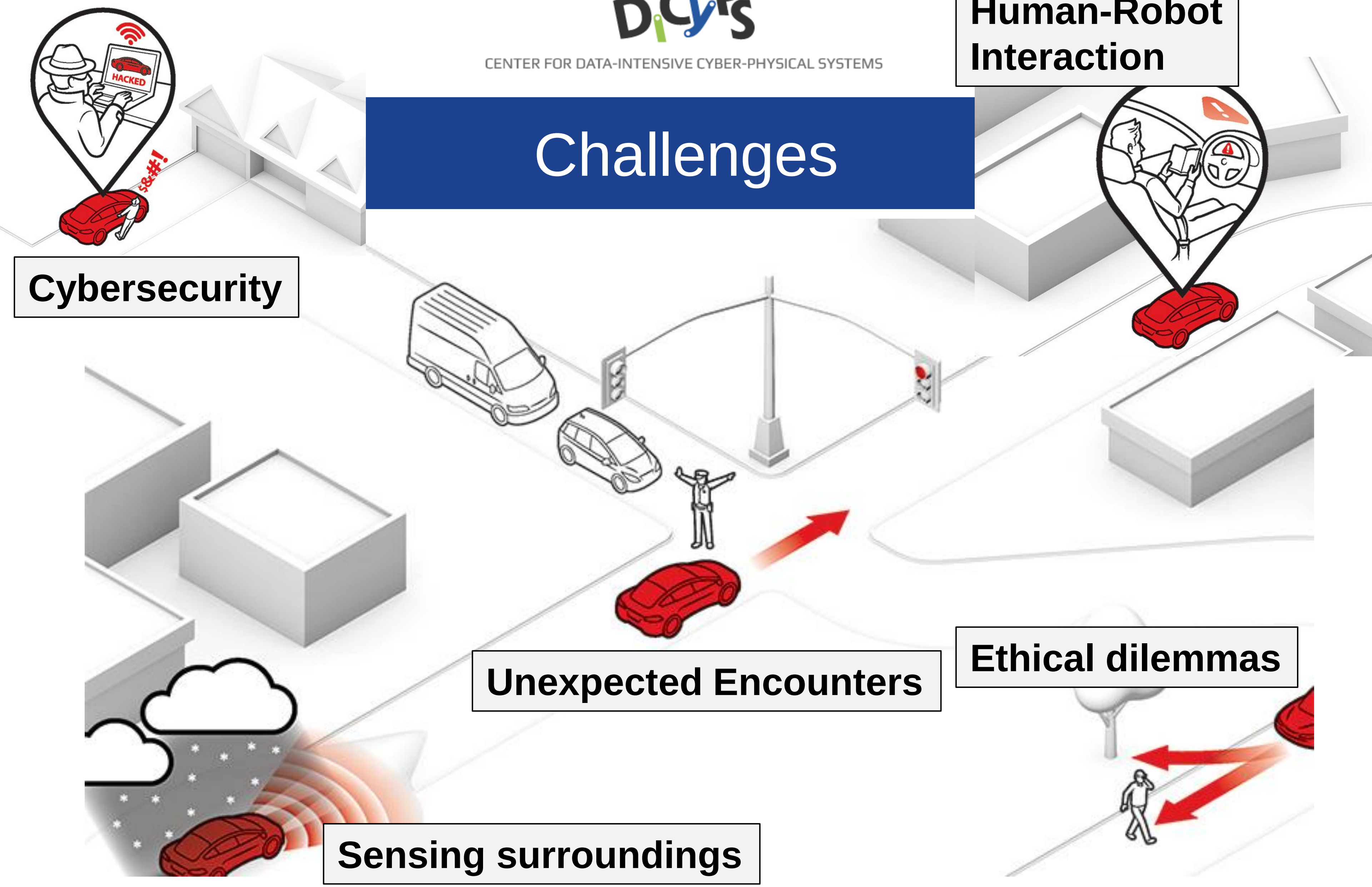
Self-Driving Cars

Self-Driving Cars – Pros

- Less accidents (81% car crash human errors)
- Travel time can be made useful
- Self-driving cars in large number participate in platooning. Reduction of time and pollution.
- Possible higher speed limits.
- Lots of cars have first stage of self-driving cars: autonomous braking, self-parking, obstacle sensors
- Less parking structure and parking headaches
- Drunk driving incidents should decrease.
- …

CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

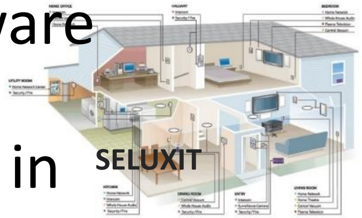
Challenges

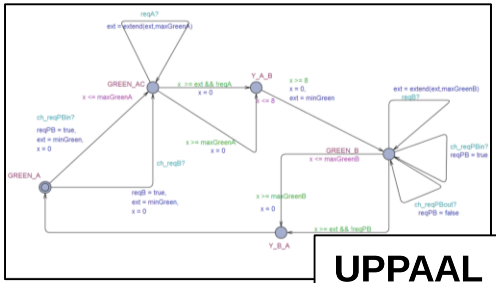


Control Synthesis using UPPAAL

- …using **symbolic methods** and **machine learning.**

- Zone-based climate control pig-stables
- Profit-optimal, energy-aware schedules for satelittes
- Personalized light control in home automation
- Energy- and comfort-optimal floor heating
- Adaptive cruise control

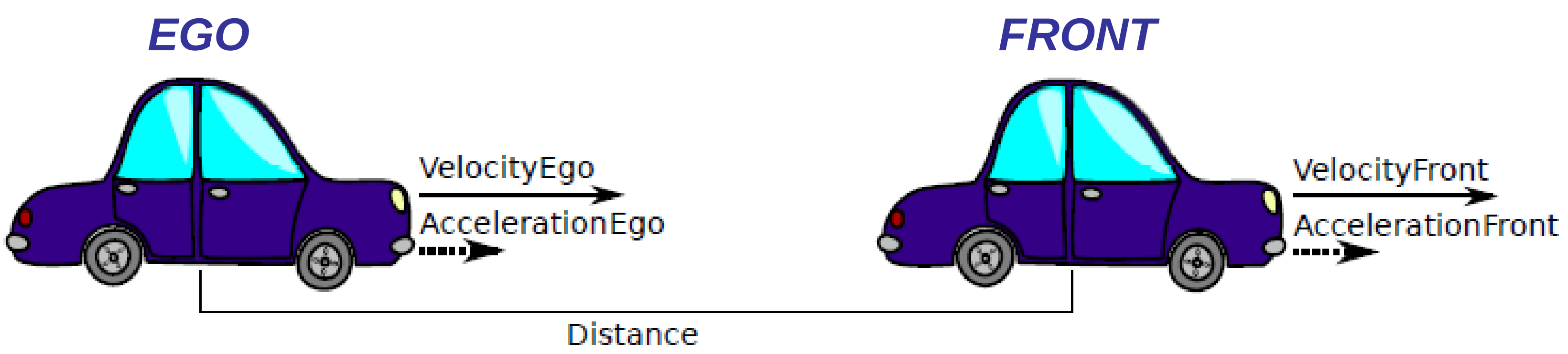




DiCyPS

CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

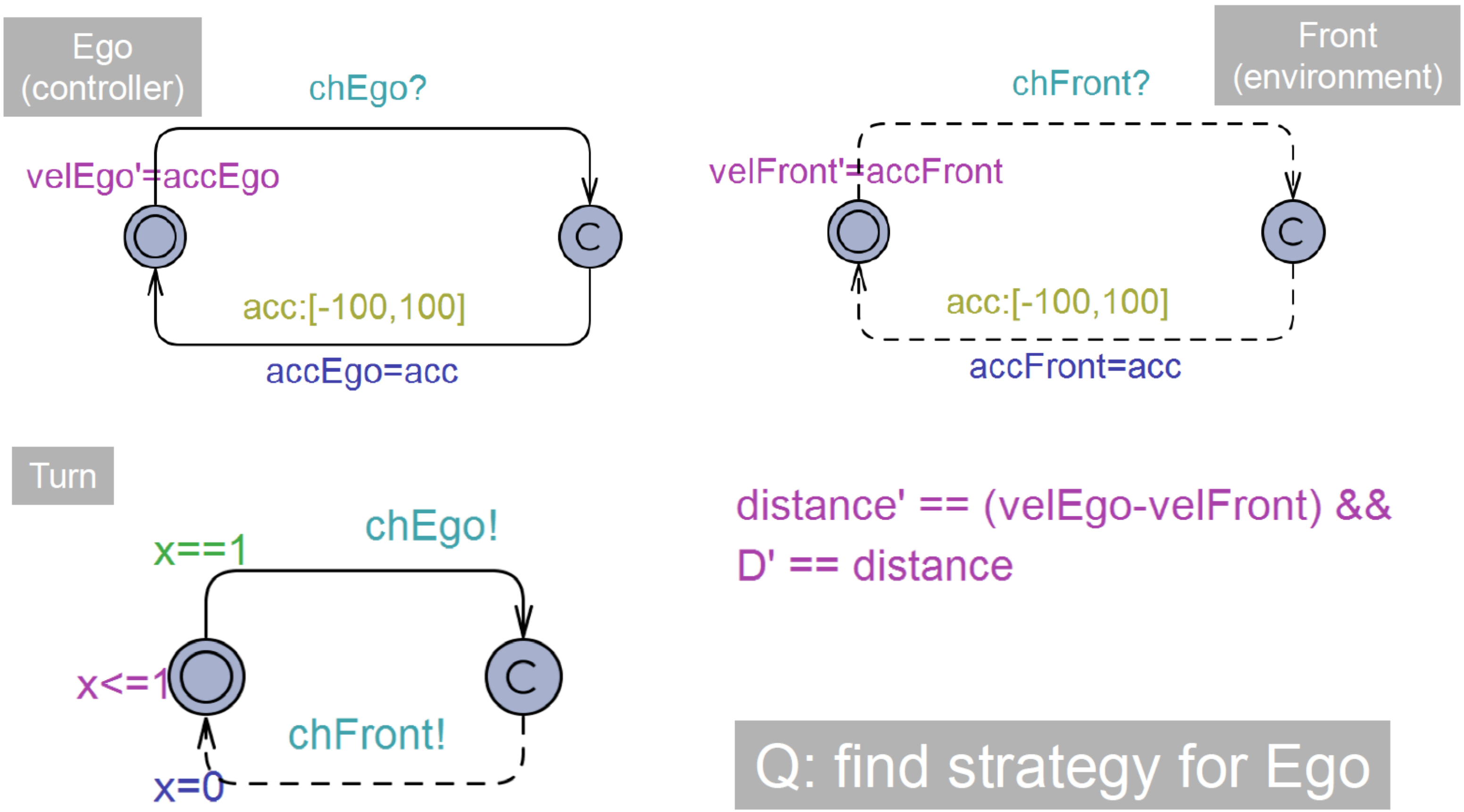
Synthesis of SAFE & Adaptive Cruice Control



Q1: Find a safety **strategy** for ***Ego*** such no crash will ever occur no matter what ***Front*** is doing.

Q2: Find the **optimal sub-strategy** that will allow *Ego* to go as far as possible (without overtaking).

Two Player Game (simplified)

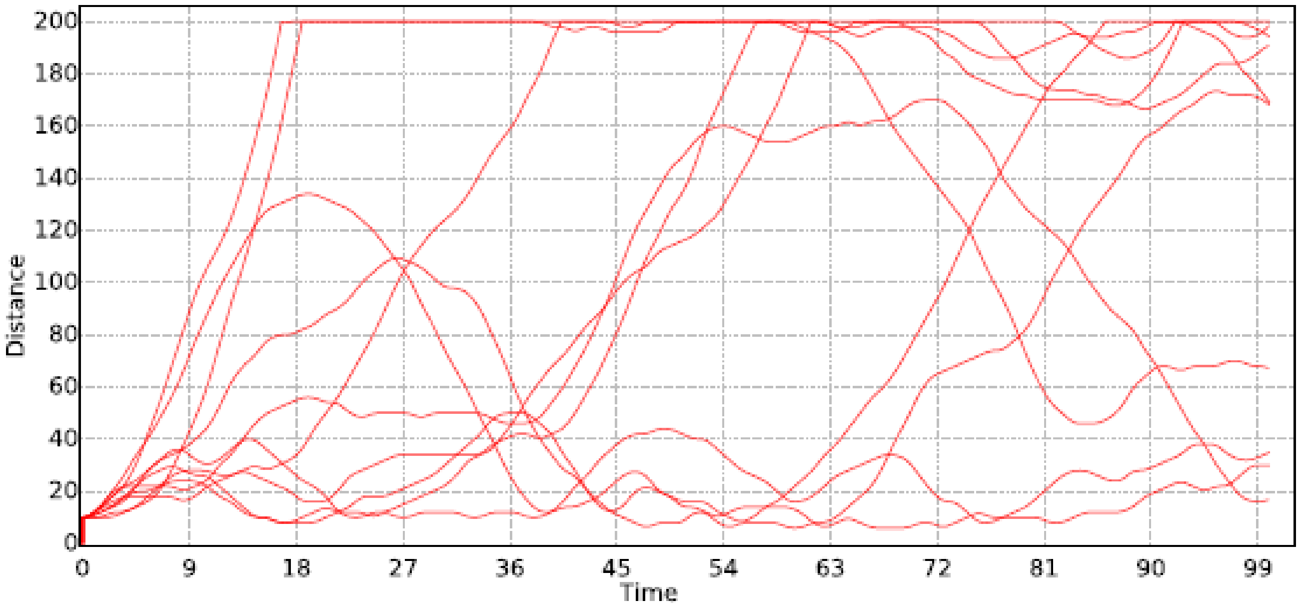


distance' == (velEgo-velFront) &&
D' == distance

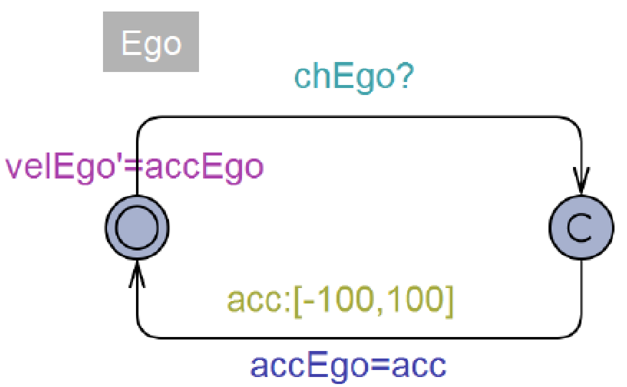
Q: find strategy for Ego

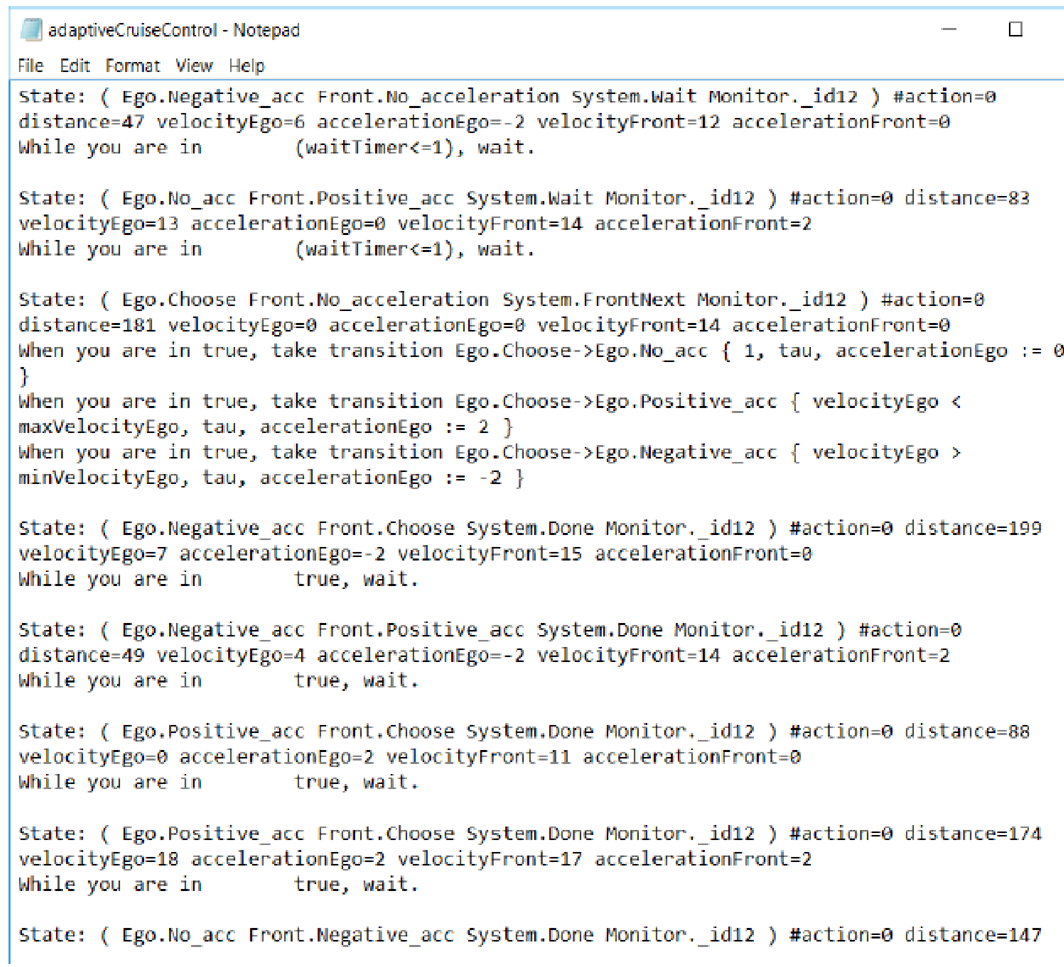
SAFE Strategy

```
strategy safe = control: A[] distance > 5
```



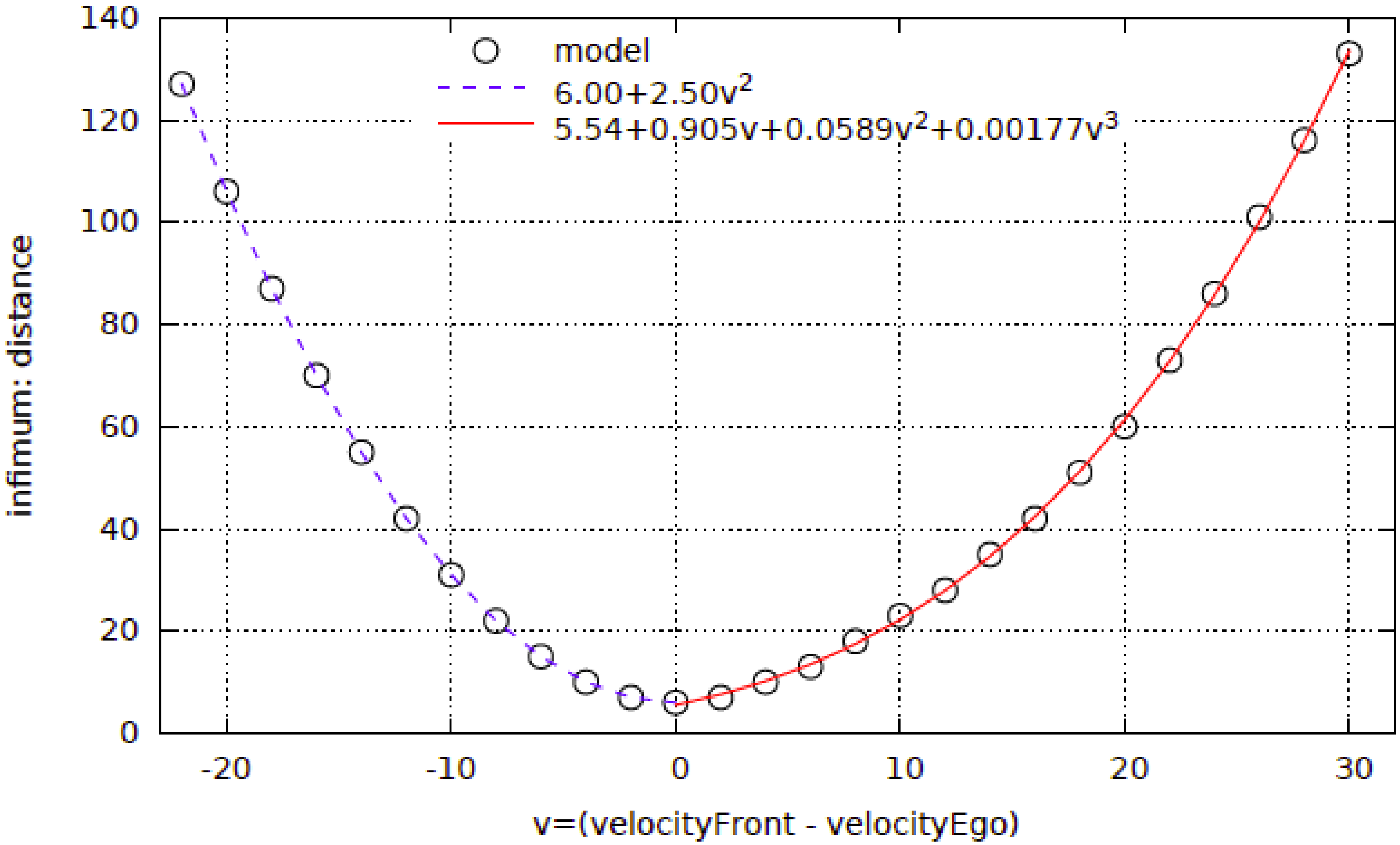
SAFE Strategy (Code)




adaptiveCruiseControl - Notepad
File Edit Format View Help

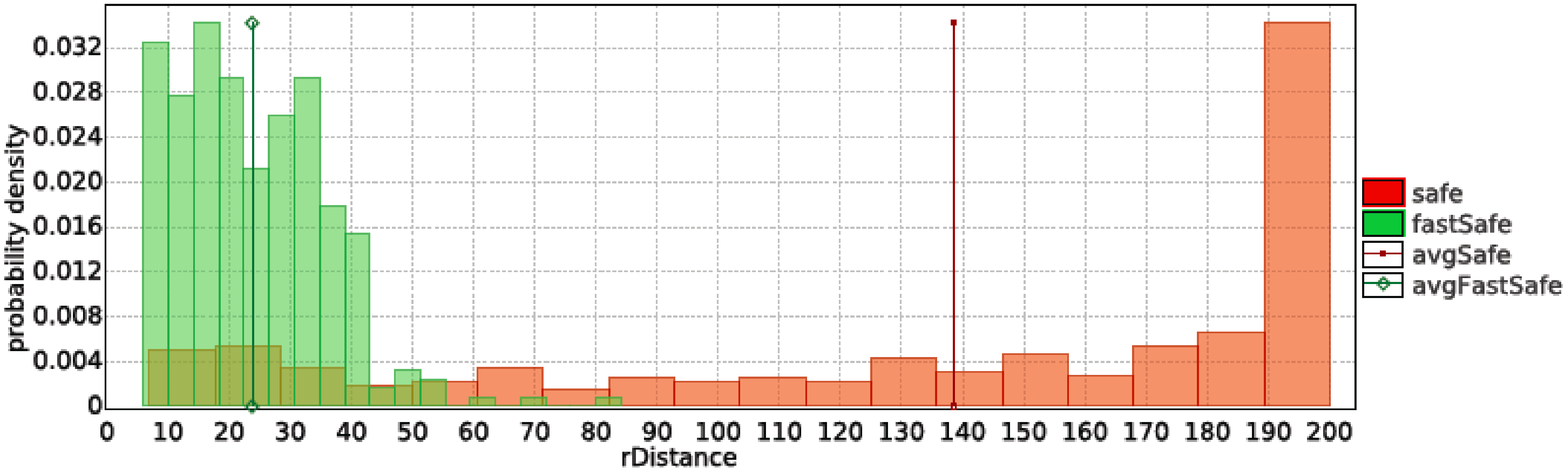
```
State: ( Ego.Negative_acc Front.No_acceleration System.Wait Monitor._id12 ) #action=0
distance=47 velocityEgo=6 accelerationEgo=-2 velocityFront=12 accelerationFront=0
While you are in        (waitTimer<=1), wait.

State: ( Ego.No_acc Front.Positive_acc System.Wait Monitor._id12 ) #action=0 distance=83
velocityEgo=13 accelerationEgo=0 velocityFront=14 accelerationFront=2
While you are in        (waitTimer<=1), wait.

State: ( Ego.Choose Front.No_acceleration System.FrontNext Monitor._id12 ) #action=0
distance=181 velocityEgo=0 accelerationEgo=0 velocityFront=14 accelerationFront=0
When you are in true, take transition Ego.Choose->Ego.No_acc { 1, tau, accelerationEgo := 0
}
When you are in true, take transition Ego.Choose->Ego.Positive_acc { velocityEgo <
maxVelocityEgo, tau, accelerationEgo := 2 }
When you are in true, take transition Ego.Choose->Ego.Negative_acc { velocityEgo >
minVelocityEgo, tau, accelerationEgo := -2 }

State: ( Ego.Negative_acc Front.Choose System.Done Monitor._id12 ) #action=0 distance=199
velocityEgo=7 accelerationEgo=-2 velocityFront=15 accelerationFront=0
While you are in        true, wait.

State: ( Ego.Negative_acc Front.Positive_acc System.Done Monitor._id12 ) #action=0
distance=49 velocityEgo=4 accelerationEgo=-2 velocityFront=14 accelerationFront=2
While you are in        true, wait.

State: ( Ego.Positive_acc Front.Choose System.Done Monitor._id12 ) #action=0 distance=88
velocityEgo=0 accelerationEgo=2 velocityFront=11 accelerationFront=0
While you are in        true, wait.

State: ( Ego.Positive_acc Front.Choose System.Done Monitor._id12 ) #action=0 distance=174
velocityEgo=18 accelerationEgo=2 velocityFront=17 accelerationFront=2
While you are in        true, wait.

State: ( Ego.No_acc Front.Negative_acc System.Done Monitor._id12 ) #action=0 distance=147
```

SAFE Strategy



OPTIMAL and SAFE Strategy

```
strategy safeFast = minE (D) [<=100]: <> time >= 100 under safe
```



Traffic Control

Tæt trafik koster milliarder

Tæt trafik på indfaldsvejene til de store danske byer koster samfundet milliarder. Det viser nye beregninger fra konsulentfirmaet Cowi, der for første gang analyserer de samlede økonomiske konsekvenser af trængslen på et sammenhængende vejnet, skriver Politiken.

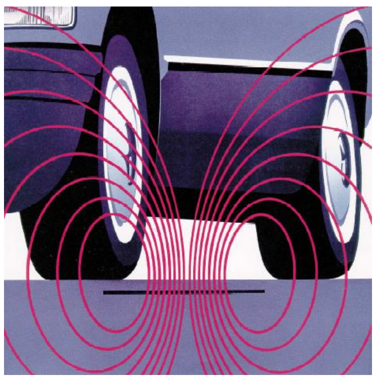
ONSDAG D. 10. JULI 2002 KL. 04:00

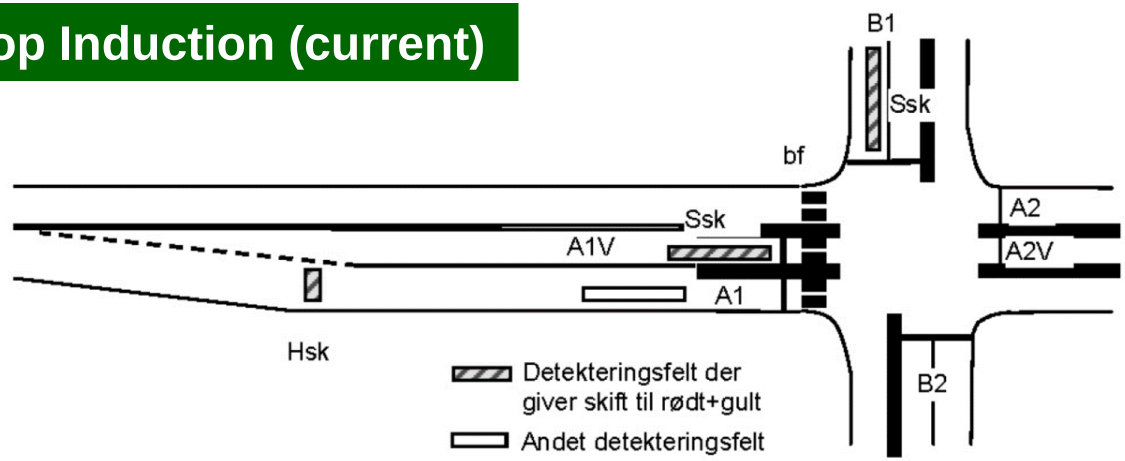
EU-kommissionen kommer om få måneder med et forslag, der skal begrænse biltrafikken ved at indføre kørselsafgifter og bruge pengene til blandt andet at styrke jernbanen og vandvejene. Ifølge kommissionen vil omkostningerne ved ventetid i bilkøer i de 15 medlemslande fordobles de næste ti år til 600 milliarder kroner.

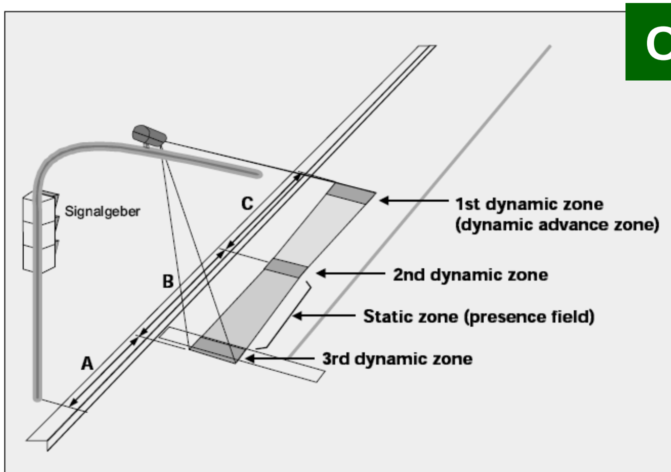
CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

Improved Detection

Loop Induction (current)

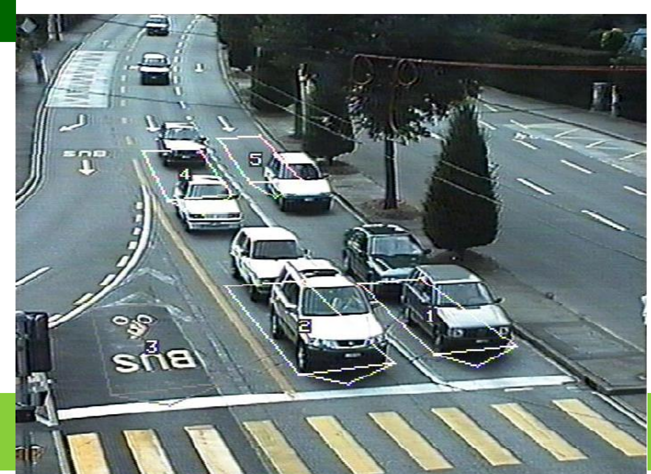






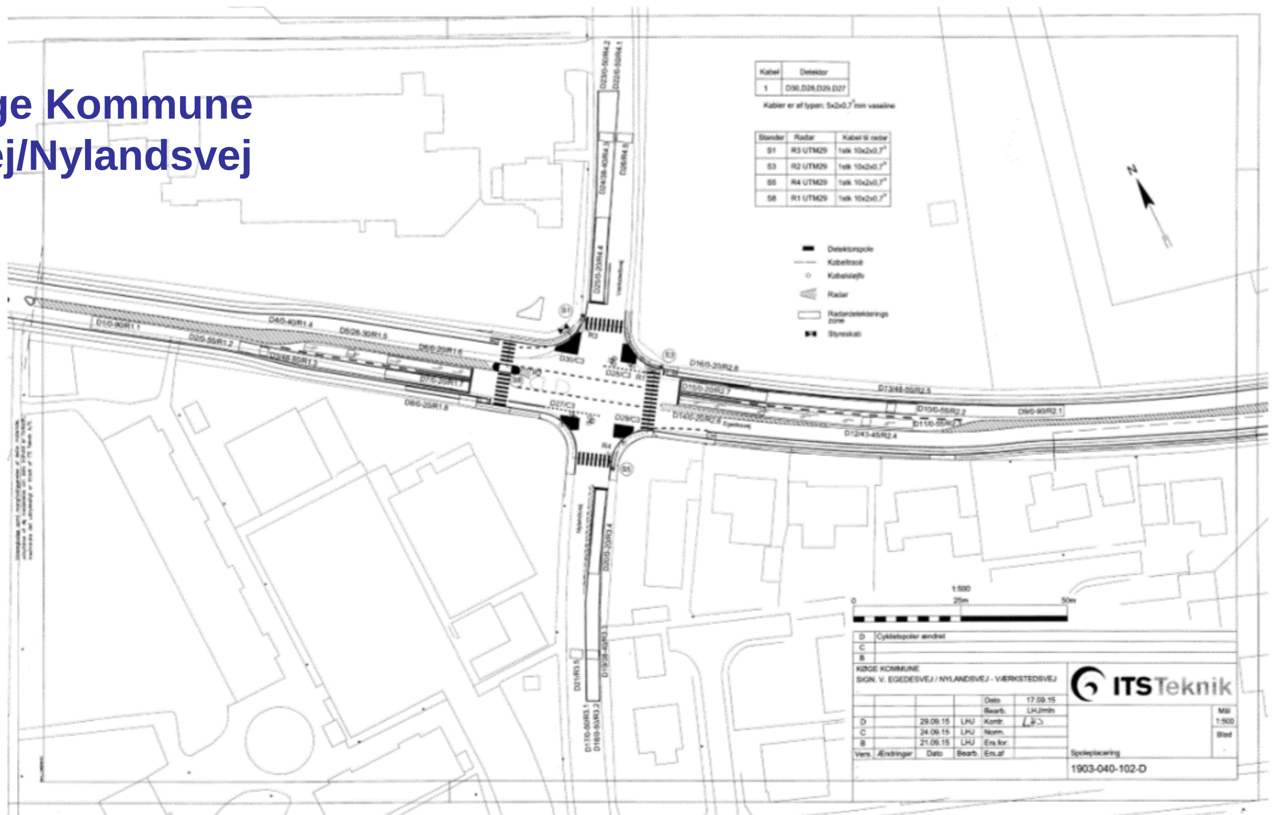
Detection fields

Camera (future)

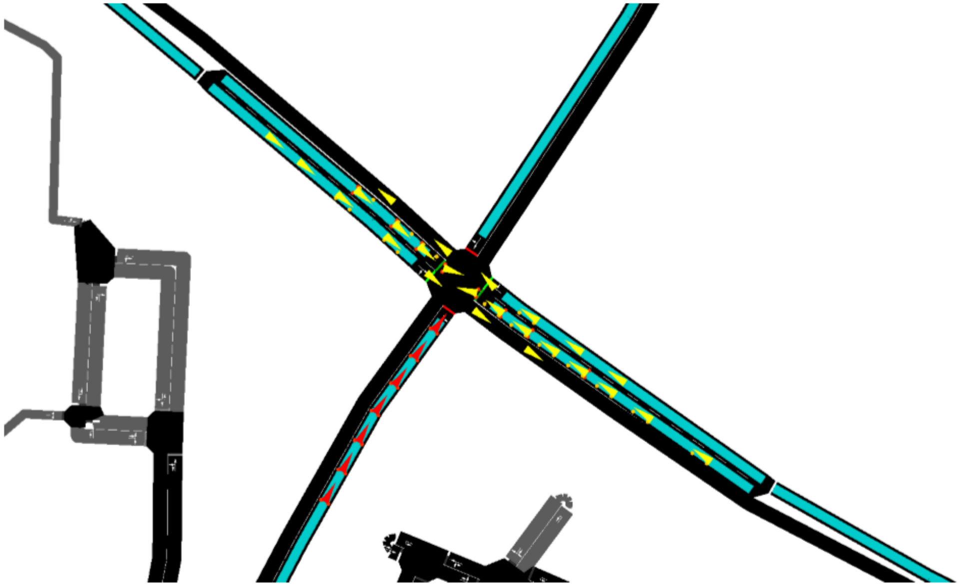


Intelligent Traffic Control

Køge Kommune
Egedesvej/Nylandsvej



SUMO Simulation of Urban MObility



- SUMO is a microscopic and continuous road traffic simulation package
- Detectors (areal, induction loops)
- Types of cars, types with different paramenters like speed ...
- Traffic lights

CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

Loop Induction vs UPPAAL

1. The signal has **two phases (A, B)**
2. The signal has an interval with **yellow** of **8 seconds** when switching between the two phases. A **green** phase must always be min. **10 seconds**,
3. The signal must always return to **green** in the direction A if there is no notification from direction B. (The signal has resting position in **green** in the direction A.).
4. The **crossing loops** in direction B notify/extend the **green** time with **3.2 second** when it is passed until a max extending time on **30 seconds** is reached.
5. The **presence loop** in direction B notify/extend the **green** time for direction B until a max extending time on **30 seconds** is reached.
6. If there is a notification from direction B the **crossing loops** in direction A will extend the **green** phase in direction A with **3.2 second** until a max **green** time of **60 second** is reached.

Loop Induction vs UPPAAL

```
Every 5 to 8 sec read sensor data
if Traffic Light in yellow phase then
    Run UPPAAL STRATEGO – decide next green phase
else if Traffic Light in green phase then
    Run UPPAAL STRATEGO – extend green phase or go to yellow
end if
```

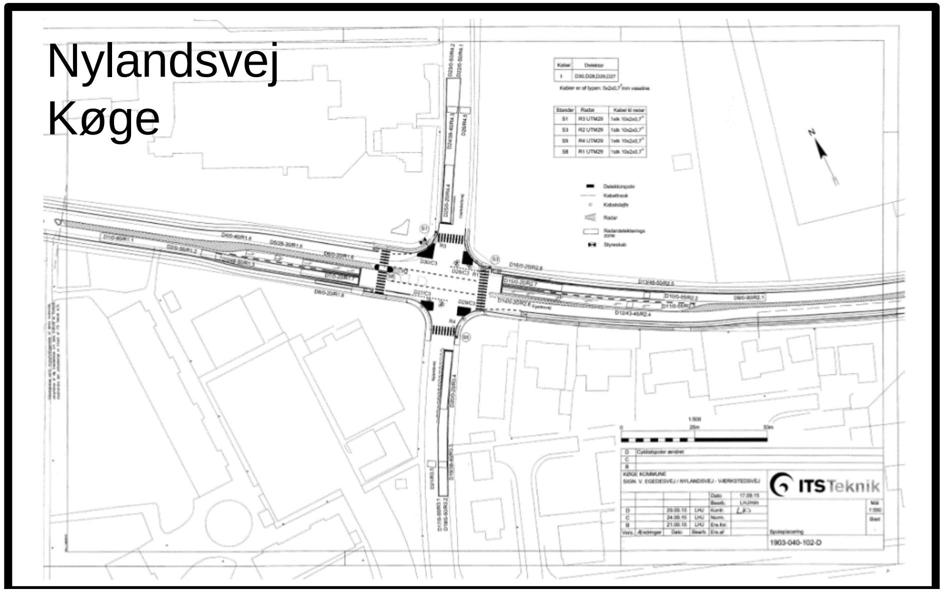
Number of cars waiting in each lane *(full information)*

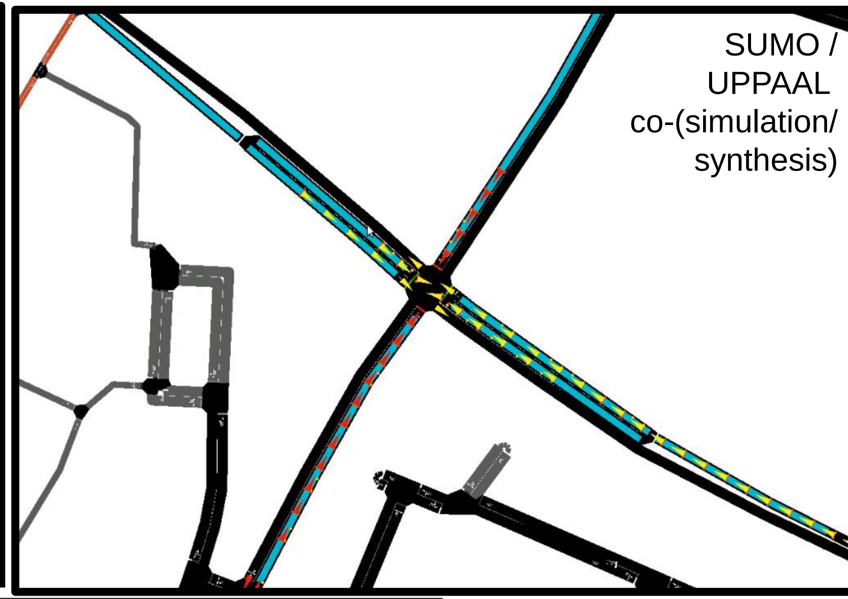
ONLINE Synthesis

- Identify optimal strategy up to horizon H=90sec.
  - Strategy changes phase (at least 5 sec).
  - Modelling of stochastic arrival of cars in different directions (from 60-850 cars/hour)
- Minimize waiting time or jam (# of waiting >2sec)

CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS

Preliminary Results





Scenario	Static		Loop Induction		Stratego		Imp W time over LI %
	Jam Km	W time s	Jam Km	W time s	Jam Km	W time s	
MAX	1451	191990	1185	157200	551	73001	53.5%
MID	456	60362	369	48936	331	43878	10.3%
LOW	138	18425	139	18566	101	13451	27.5%

Scenario:
2 hours traffic

Co-simulation
VISSEM / UPPAAL

SEE DEMO
Andreas & Mikkel

Co-simulation
VISSEM / UPPAAL

SEE DEMO
Andreas & Mikkel

THANKS!

Next Steps

- Validate experimental findings by co-simulation with VISSEM
- Look at a variety of intersections in Aalborg (Hasseris –Vesterbro, Sygehus Syd)
- Synthesis of controllers for Green Flow.

- THANKS!