Connected and Automated Driving:

Decentralised signal-free intersection management and cooperative routing

Claudio Roncoli

Associate Professor, Transportation Engineering

Aalto University
School of Engineering

University of Sydney, 03.11.2023

Outline

(Traffic) Control with Connected Automated Vehicles

Distributed optimisation for intersection management

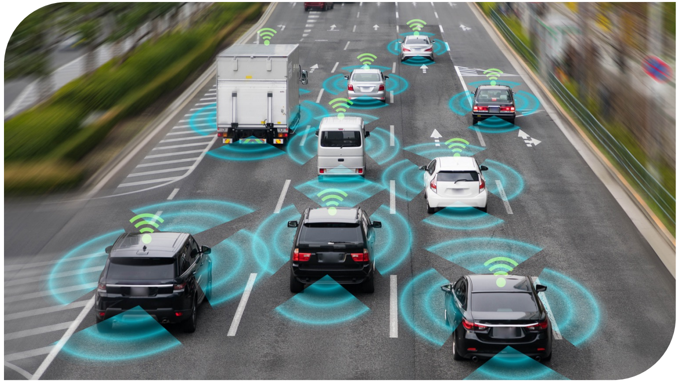
Cooperative rerouting in urban networks

Conclusions

Control with Connected Automated Vehicles

Introduction

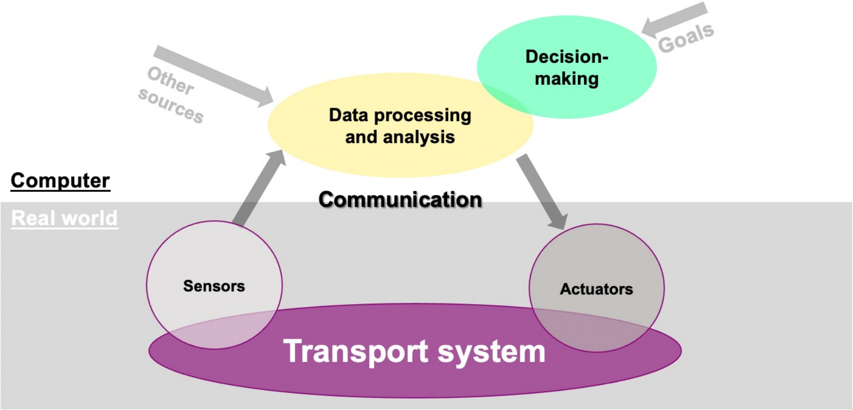
- Connected and automated vehicles (CAVs) have the potential to revolutionise traffic (and traffic management)
  - Vehicle-to-everything (V2X) communication
  - Actuation based on (optimisation) algorithms



- Resulting challenges in CAV management
  - Optimisation of CAVs driving behaviour (time, passenger comfort, energy consumption)
  - Coordination among CAVs (e.g., platooning)

- Among other challenges…
  - Integration with Traffic Management (TM)
  - Computational feasibility and scalability

Current TM Systems



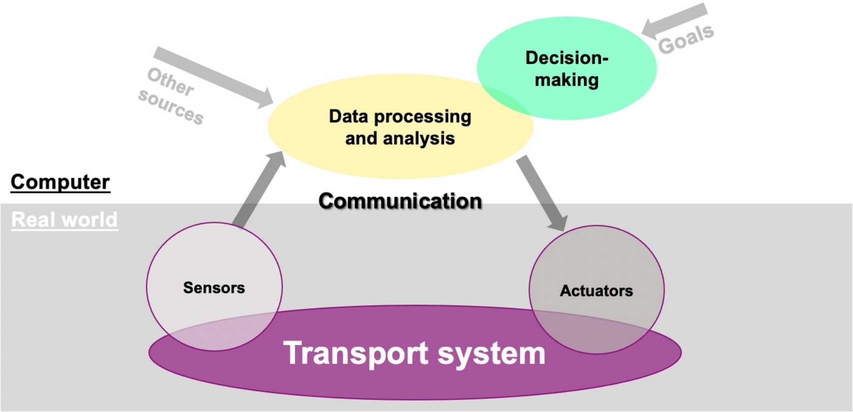
- Process: conventional vehicle flow
- Sensors: spot sensors

(loops, vision, magnetometers, radar, …)

- Communications: wired
- Computing: central, decentralized, hierarchical
- Actuators: road-side (traffic signals, ramp metering, VSL, VMS, …)

Future TM Systems

- Process: enhanced-capability vehicles
- Sensors: vehicle-based
- Communications: wireless, V2V, V2I, (V2X)
- Computing: central, massively decentralized, hierarchical
- Actuators: in-vehicle



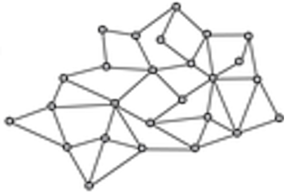
Implications/Exploitation for traffic flow efficiency?

Control strategies for CAVs

- Most management methods for CAVs are centralised
  - Computationally demanding
  - Single point of failure

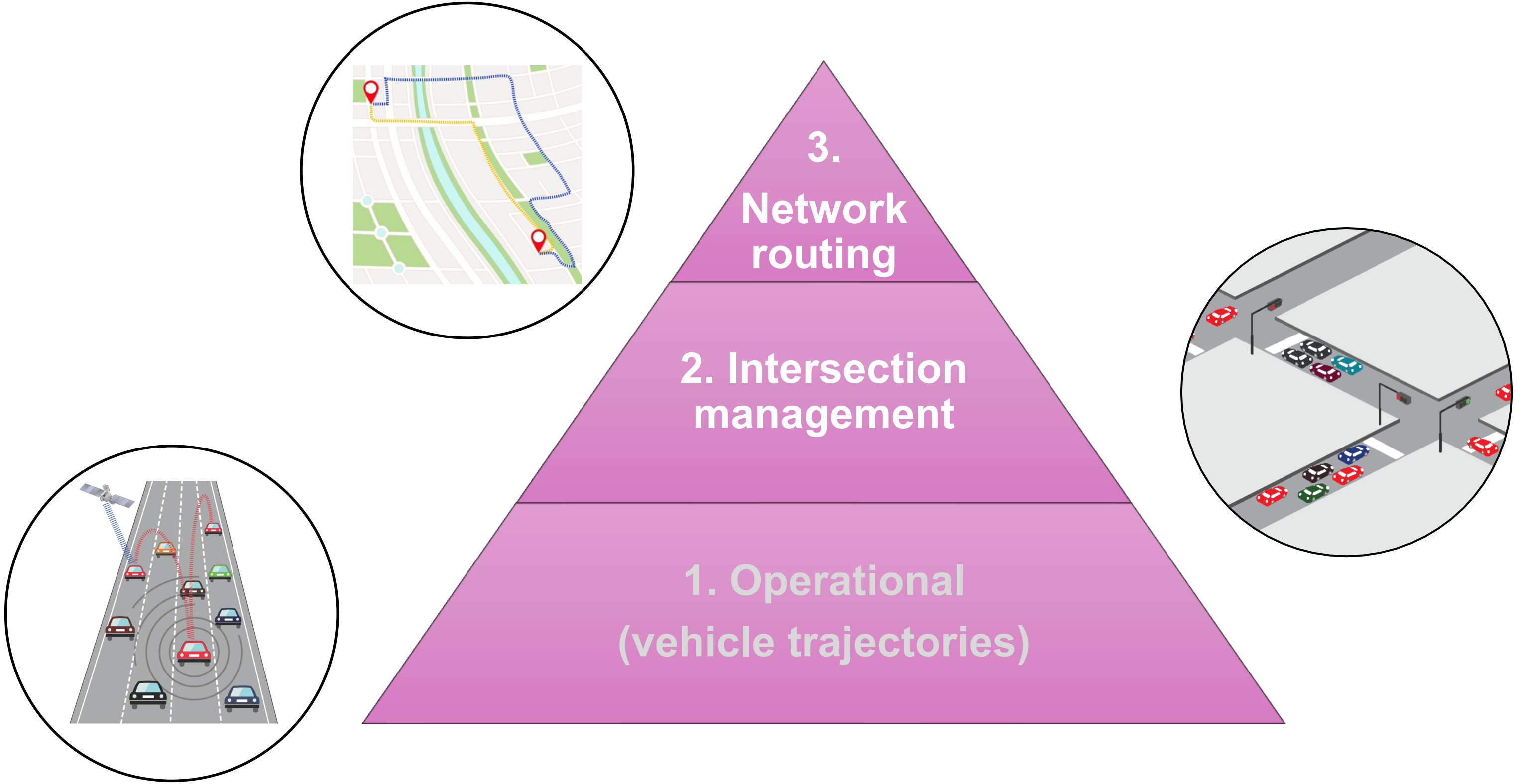
- Some are hierarchical or decentralised
  - Central units still need to broadcast information to some CAVs, with the latter forwarding to others

- Almost none are distributed
  - All CAVs (in a queue or road) would compute their solution and forward it to others



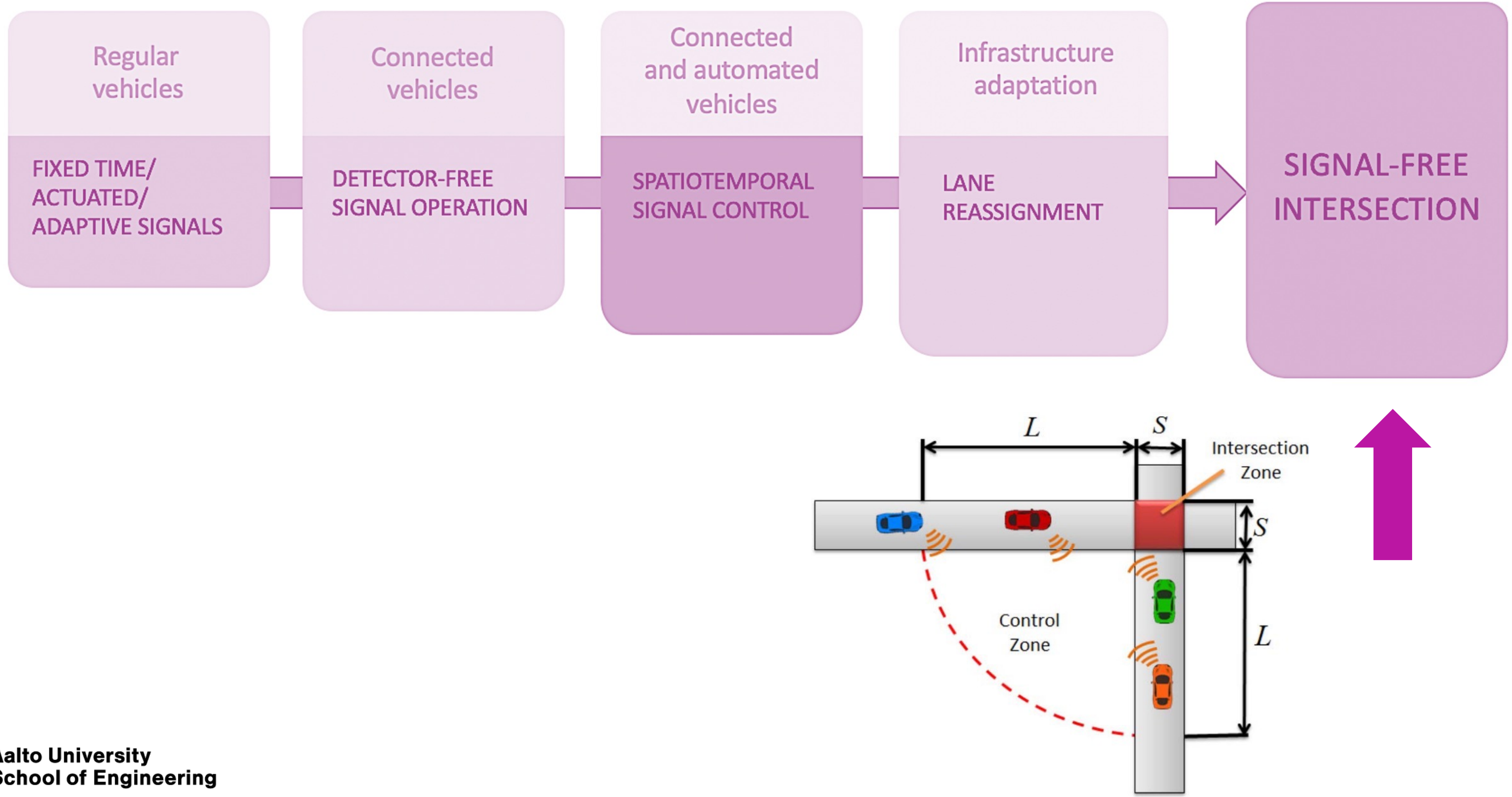
Aalto University
School of Engineering

Control strategies for CAVs



Distributed optimisation for intersection management

Motivation


Regular vehicles
FIXED TIME/ ACTUATED/ ADAPTIVE SIGNALS
Connected vehicles
DETECTOR-FREE SIGNAL OPERATION
Connected and automated vehicles
SPATIOTEMPORAL SIGNAL CONTROL
Infrastructure adaptation
LANE REASSIGNMENT
SIGNAL-FREE INTERSECTION
L
S
Intersection Zone
S
Control Zone
L

Current limitations of signal-free intersection methods

- Agreeing on intersection crossing time schedule often requires heuristics
  - No optimality
- Existing methods are typically designed assuming that vehicle pass an intersection at a given (maximum) speed
  - Useful but not always necessary
- Physical constraints mostly depend on space…
  - Speeds when driving in a leg are different from those when driving inside the intersection (e.g., different manoeuvres, …)
- … but current optimisation solutions mostly depend on time!
  - Heuristic algorithms play a crucial role in assigning the timing

Our solutions

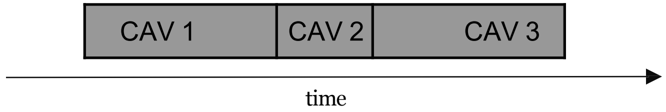
Two components:

1. A time scheduling distributed algorithm
    - Each CAV calculates its own timing for crossing the intersection
    - Exchange little information with neighbouring CAVs
        - No single point of failure
        - Computationally scalable
2. A space-dependent optimisation formulation
    - Quadratic cost function & linear constraints
        - Computationally inexpensive solution
        - Suitable for distributed optimisation, with little information exchange

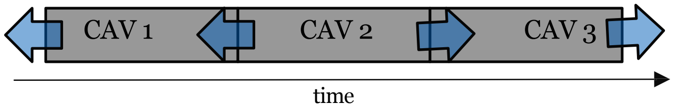
A time scheduling distributed algorithm

Inspired by algorithms for distributed scheduling tasks in robotic applications

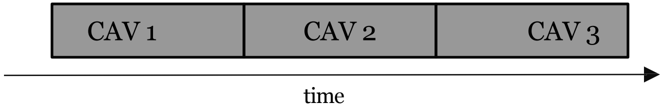
Exchange entering and exiting time



Adjust such a time period



Fixed time spent in the intersection…



Run at each iteration by each CAV

Calculate entering and exiting time in the intersection; check *danger* flag

↓

Identify previous (PV) and next (NV) vehicles at the intersection (based on current calculations)

↓

Check if there is overlapping with PV and/or NV

No overlapping, while PV XOR NV in *danger*: shift by δ towards the one not in danger

Overlapping with one only, while the other not in *danger*: shift by δ away from the overlapped one

Overlapping with both: do not shift and raise *danger* flag

Else: do not shift

Time-dependent optimisation formulation

$$\min \sum_{i=1}^{N}\sum_{t=1}^{T}\frac{1}{2}\left\|x_i^t - {x_{\mathrm{des}}}_i^t\right\|_Q^2 + \sum_{i=1}^{N}\sum_{t=1}^{T}\frac{1}{2}\left\|a_i^t\right\|_R^2$$

$$\text{s.t.}\quad x_i^{t+1} = x_i^t + \Delta_T v_i^t \qquad i=1,\dots,N$$

$$v_i^{t+1} = v_i^t + \Delta_T a_i^t \qquad t = 1,\dots,T$$

$${x_{\min}}_i^t \le x_i^t \le {x_{\max}}_i^t$$

$$0 \le v_i^t \le {v_{\max}}_i^t$$

$${a_{\max}}_i^t \le a_i^t \le {a_{\max}}_i^t$$

N: number of CAVs

T: time horizon

x_i^t, v_i^t, a_i^t: longitudinal position, speed, and acceleration

${x_{\mathrm{des}}}_i^t$: desired longitudinal position

${v_{\max}}_i^t$: maximum speed

${a_{\min}}_i^t, {a_{\max}}_i^t$: minimum and maximum acceleration

Q and R: weights matrices

Δ_T time sample

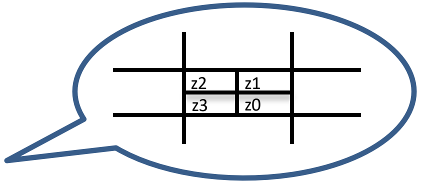
Bounds ${x_{\min}}_i^t$ and ${x_{\max}}_i^t$ are defined according to the entering and exiting times resulting from the distributed algorithm

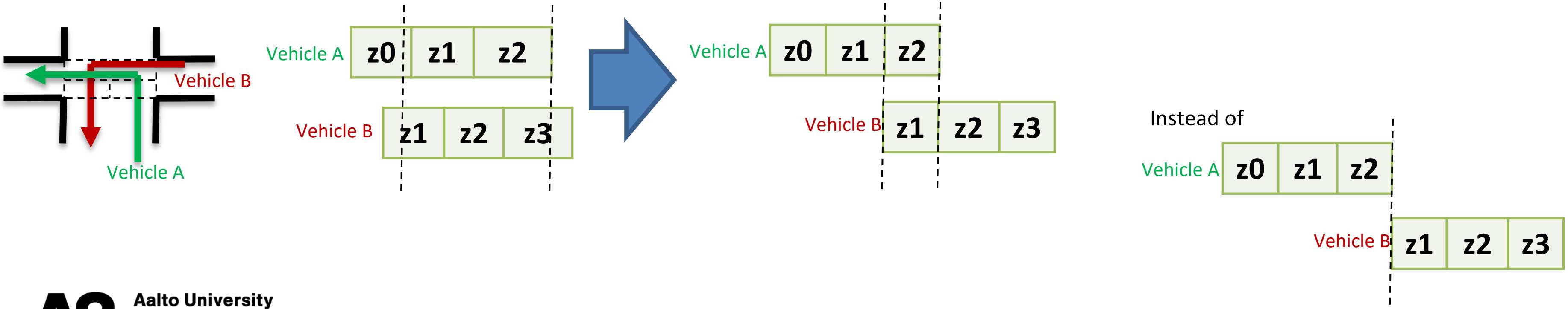
Multiple vehicles in the intersection

Problem: Only one vehicle at a time is allowed to cross the intersection

Solution:

1. Split the intersection into zones
2. Modify the previous algorithm so that overlapping is considered only if it occurs in the same zone (requires the exchange of starting and ending time for each zone)

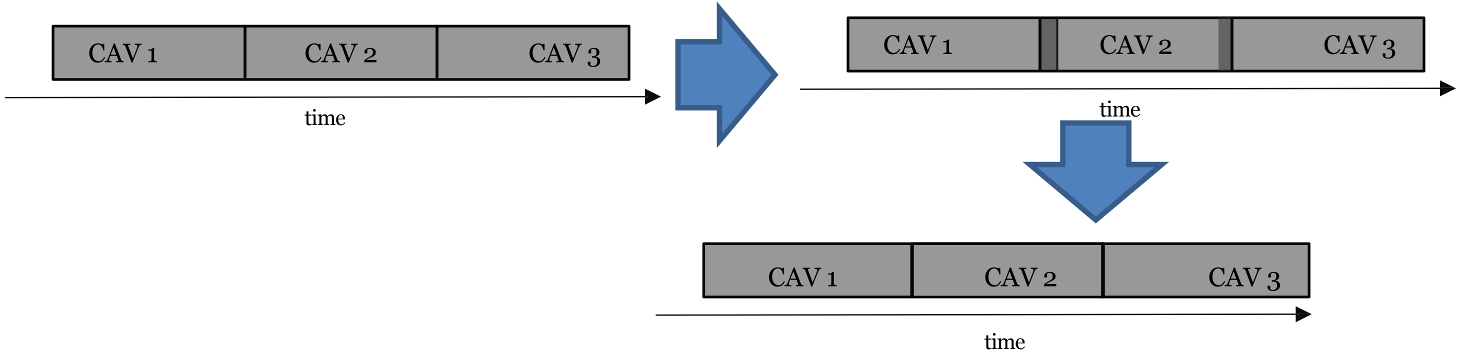




Reshaping time slots

Problem: The whole procedure does not allow the time slots to be reshaped, leading to possible increased control efforts to comply with them

Solution: Iteratively solve the optimisation problem, allowing each time a violation of the time slot constraints. Such violation tends to 0



From time- to space-dependent formulation

- The time-dependent optimisation formulation does NOT exhibit coupling constraints
    - Each CAV can only solve its own part of the problem

- Entering and exiting times are determined by the distributed algorithm
    - Not readily adjustable by optimisation

- How to allow such an adjustment…
    - …so that the solution is optimal and not affected by the (still heuristic) distributed algorithm?

Space-dependent optimisation formulation

$$\min \sum_{i=1}^{N}\sum_{\sigma=1}^{S}\frac{1}{2}\left\|t_i^\sigma - t_{\mathrm{des}}{}_i^\sigma\right\|_{\bar{Q}}^2 + \sum_{i=1}^{N}\sum_{\sigma=1}^{S-1}\frac{1}{2}\left\|\pi_i^\sigma - \pi_i^{\sigma+1}\right\|_{\bar{R}}^2$$

$$\text{s.t.}\quad t_i^{\sigma+1} = t_i^\sigma + \Delta_S \pi_i^\sigma \qquad i=1,\dots,N \quad \sigma = 1,\dots,S$$

$$t_i^\sigma \geq 0$$

$$\pi_i^\sigma \geq \pi_{\mathrm{min}}{}_i^\sigma$$

$$t_i^{\mathrm{sI}} - t_{i-1}^{\mathrm{eI}} \geq \tau$$

N: number of CAVs

S: discrete locations along the road

t_i^σ: time when vehicle i is at location σ

π_i^σ: *pace*, i.e., the inverse of the speed

$t_{\mathrm{des}}{}_i^\sigma$: desired time

$\pi_{\mathrm{min}}{}_i^\sigma$: minimum pace (i.e., $\frac{1}{v_{\mathrm{max}}{}_i^t}$)

τ: time gap

$\bar{Q}$ and $\bar{R}$: weights matrices

Δ_S: space sample

Bounds t_i^{sI} and t_i^{eI} are the starting and ending time in the intersection

Remarks on the space-dependent formulation

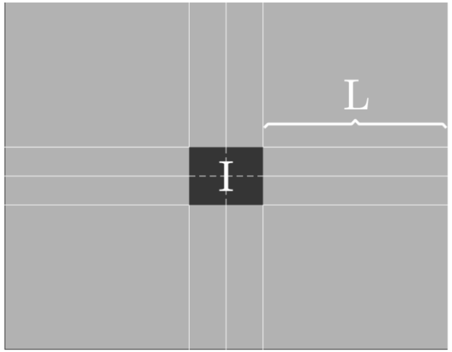
- The previous formulation DOES exhibit coupling constraints
    - However, it can be readily solved via a distributed optimisation algorithm (e.g., DPD[1]), where each CAV only solves its own part of the problem

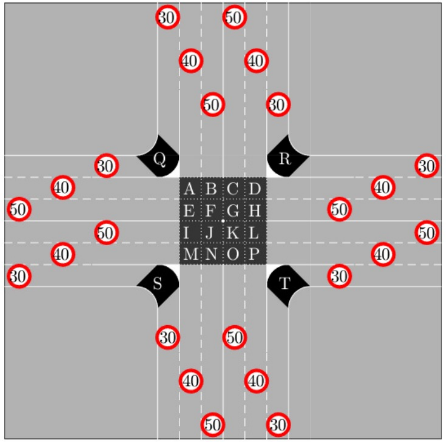
- Entering and exiting times can be obtained via optimization
    - The solution is optimal and not deteriorated by heuristics

[1]A. Camisa, F. Farina, I. Notarnicola, G. Notarstefano, Distributed constraint-coupled optimization via primal decomposition over random time-varying graphs, *Automatica*, vol. 131, 2021,

Simulation set-up

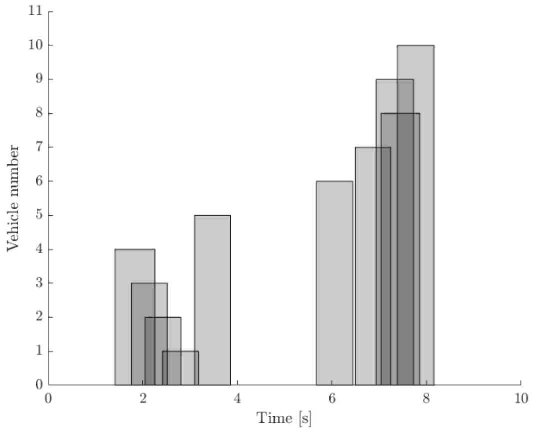
- A four-leg isolated intersection
- Ten CAVs approaching the intersection from different legs
- Without loss of generality, CAVs are assumed to go straight
- Only one CAV at a time is allowed to occupy the intersection
- The method can be extended to more complex scenarios



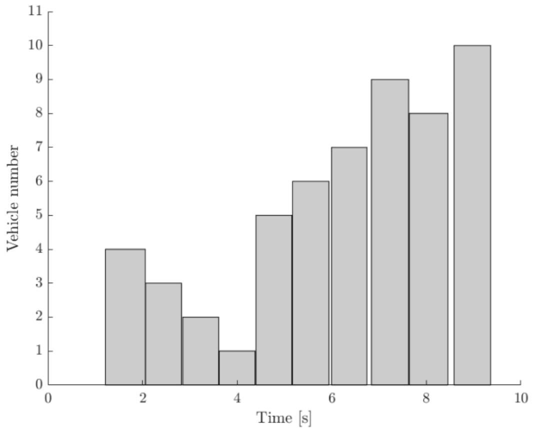


Results: Distributed Algorithm

Initial



Final



The bar width represents the time spent by CAV i, where i is the bar height, in the intersection

A value of $\delta = 0.1$ s is used.

No overlapping times in the intersection at the end of the algorithm run

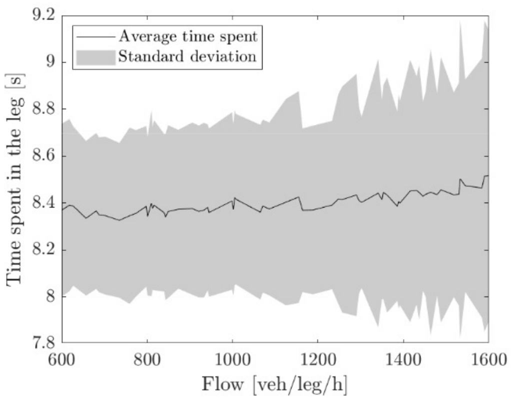
Time- vs Space- dependent optimisation

- Time-dependent solution
  - The CAVs are required to optimize a sufficiently long time-horizon (due to the unknown arrival time at the intersection)
  - Only the optimisation results up to the intersection area are shown

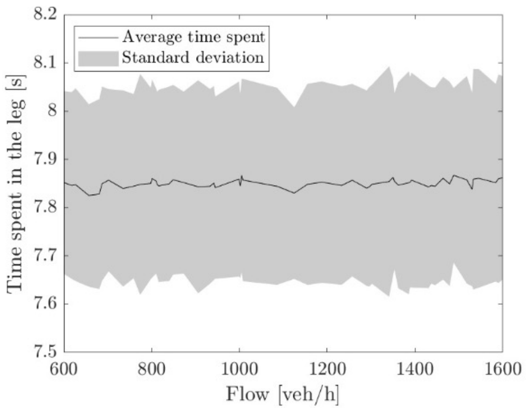
- Space-dependent solution
  - The CAVs are required to optimise along the path up to the intersection area
  - Results for whole optimisation horizon are shown

Time- vs Space- dependent optimisation

TDA



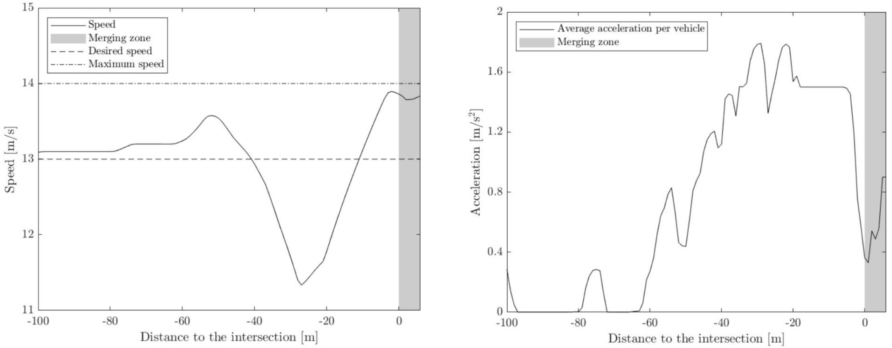
SDA



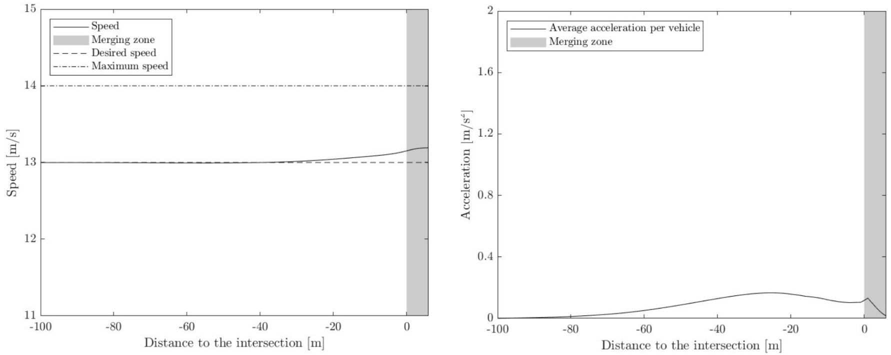
- Due to the combined heuristic and time-dependent optimisation necessary approximations, there is a loss of time at the intersection
- The space-dependent optimisation allows to rearrange optimally time periods and time gaps in the intersection.

Time- vs Space- dependent optimisation

time-dependent optimisation



space-dependent optimisation



- Moreover, the space-dependent optimisation results in smoother control actions
    - Reduced fuel/energy consumption
    - Increased comfort

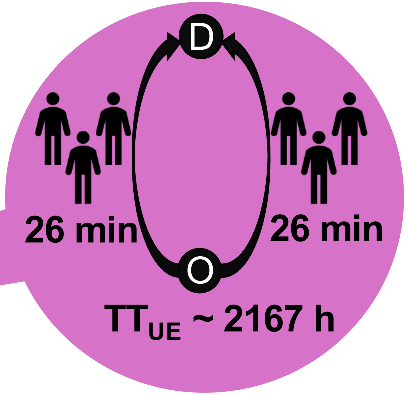
Cooperative rerouting in urban networks

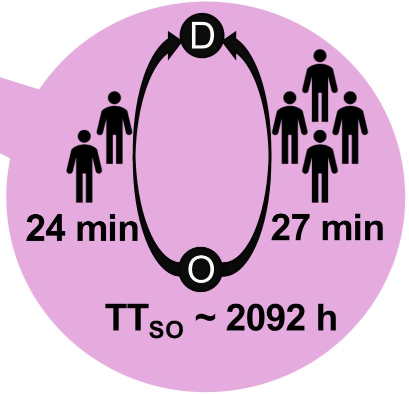
Goal

- Design of a cooperative real-time rerouting algorithm for CAVs
  - Congestion-aware redistribution of the flows at nodes (intersections)
- Based on a new cost function for routing algorithms
  - Promotes UE for smaller loads of vehicles
  - Prioritizes SO for larger loads of vehicles
- Distributed
  - Intersection units compute routes for CAVs within their range

Motivation

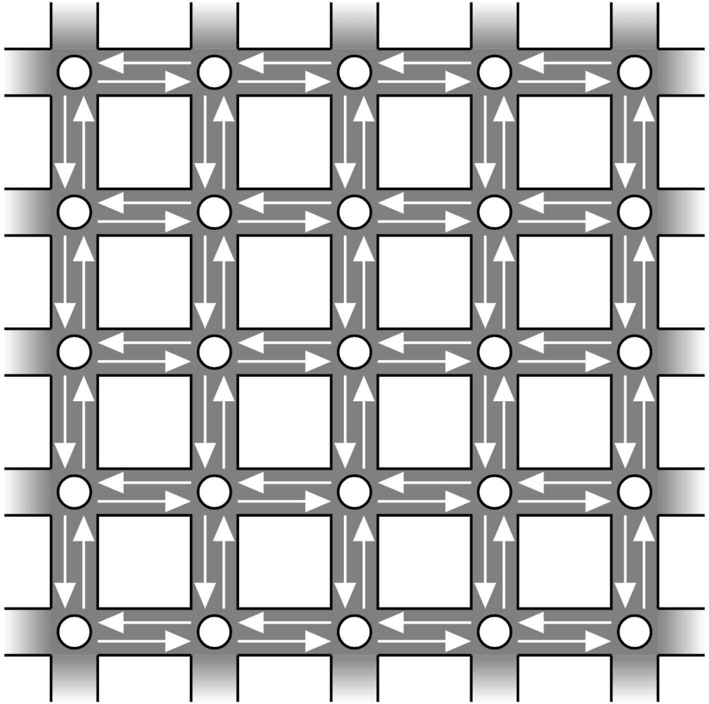
- Route planning is a complex problem
  - Connected and Automated Vehicles (CAVs)
- **Struggle** for optimal solutions
  - User Equilibrium (UE): Fairness
  - System Optimum (SO): Optimality
- **Scarcity** of congestion-aware approaches
  - Mostly with common fundamental diagrams
  - Based on centralised calculation of SO

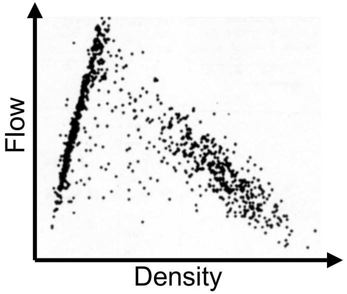




Assumptions

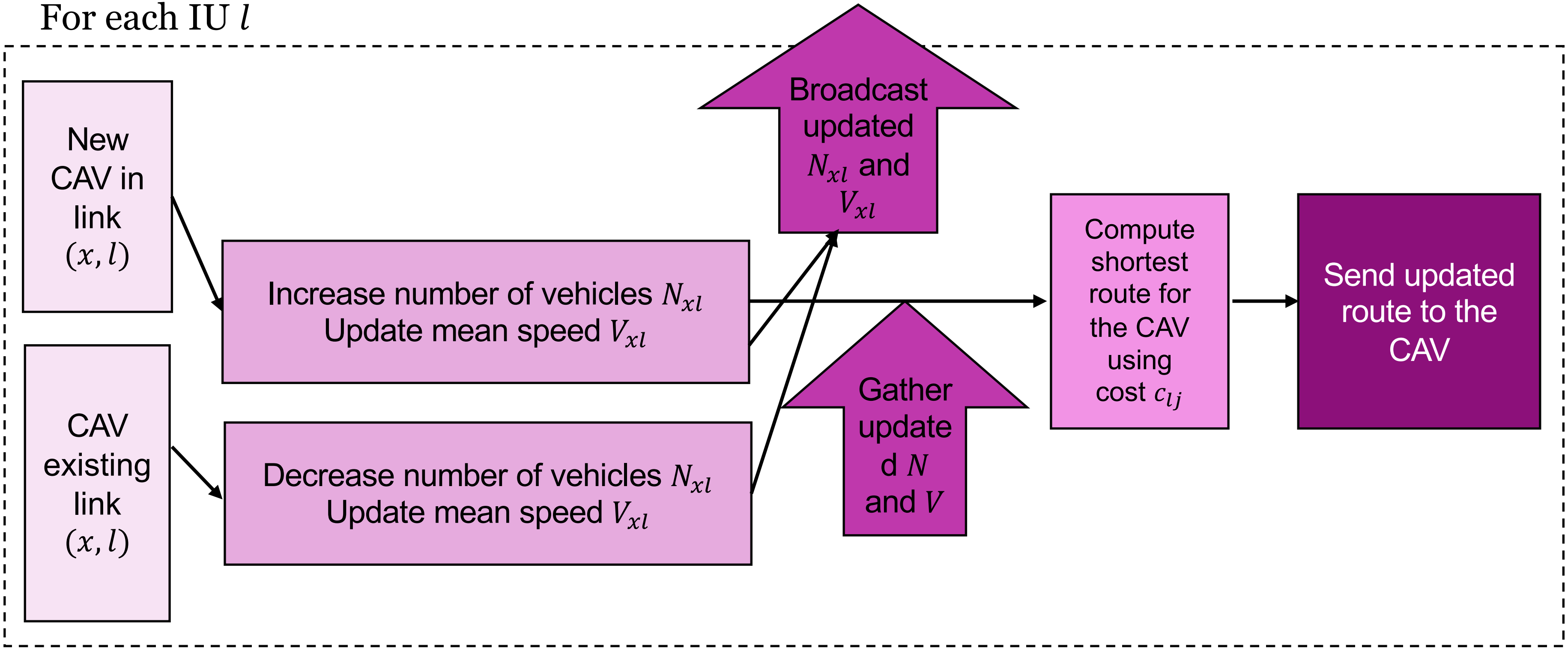
- Intersection Units (IUs) located at intersections
- Communication (V2I & I2I) between each CAV and the next IU
- CAV's origin and destination intersections known
- Fundamental diagram for each link known to IUs





Algorithm

For each IU l



Modified routing cost function

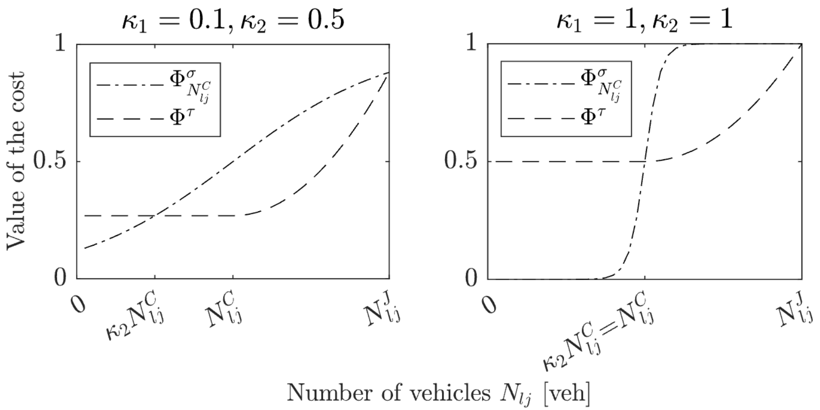
The total cost is

$$c_{lj} = d_l$$
$$+\Phi^{\sigma}_{N^C_{lj}}(\kappa_1, N_{lj}, N^C_{lj})$$
$$+\Phi^{\tau}\left(V_{lj}, \Phi^{\sigma}_{N^C_{lj}}(\kappa_1, \kappa_2 N^C_{lj}, N^C_{lj}), \Phi^{\sigma}_{N^C_{lj}}(\kappa_1, N_{lj}, N^C_{lj})\right)$$

Cost (distance) to IU l (affected by the current traffic situation)

sigmoid function centred at capacity N^C_{lj} whose skew is determined by κ_1

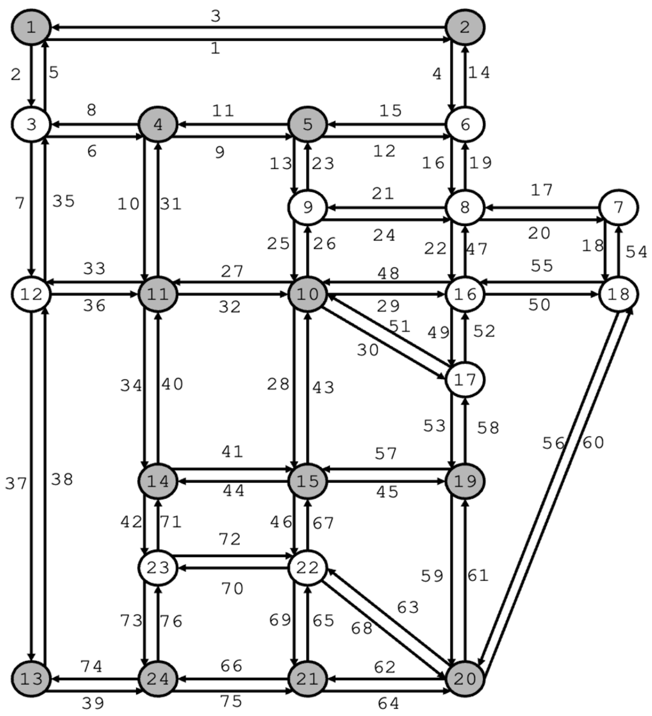
function scaling the cost of link (l, j) by κ_2 to give more or less importance to $\Phi^{\sigma}_{N^C_{lj}}$



Aalto University
School of Engineering

Numerical experiments

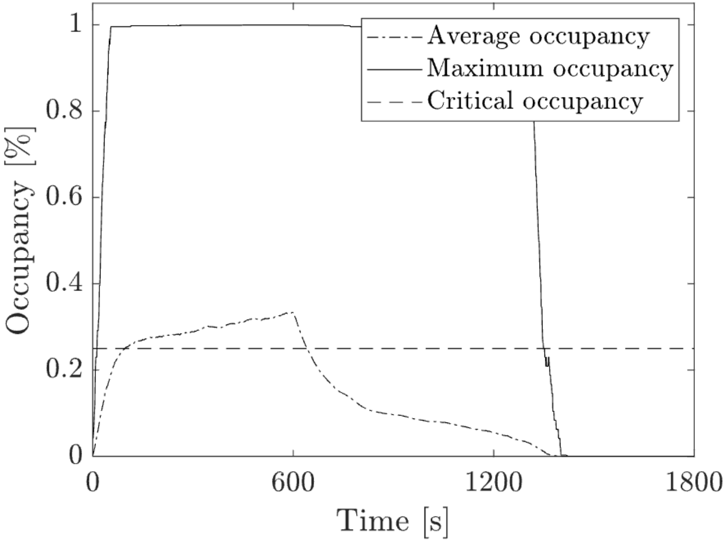
- Sioux Falls network
- 24 nodes (IUs) and 76 links
- Triangular FDs (different for each link)
- 30-minute simulation
- Demand of 360000 veh/h for the first 10 minutes, with vehicle arrivals randomly assigned
- Full compliance by CAVs
- Baseline: all-or-nothing assignment



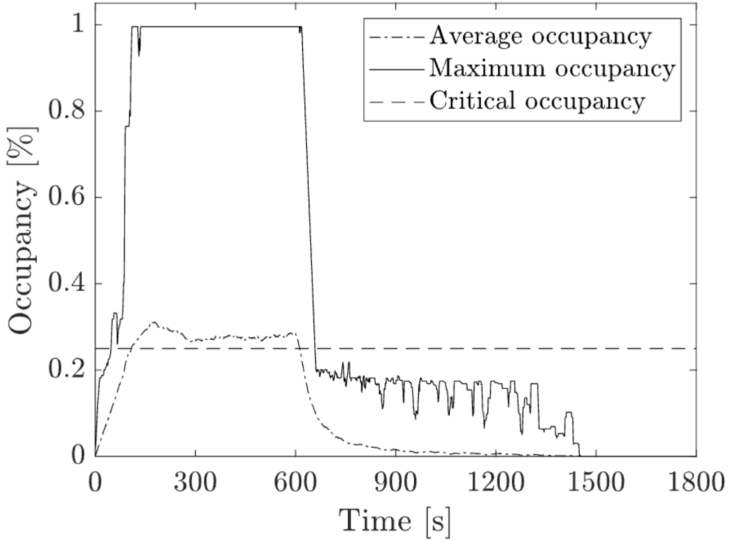
Results

Occupancy N_{lj}/N_{lj}^J

Baseline



Proposed



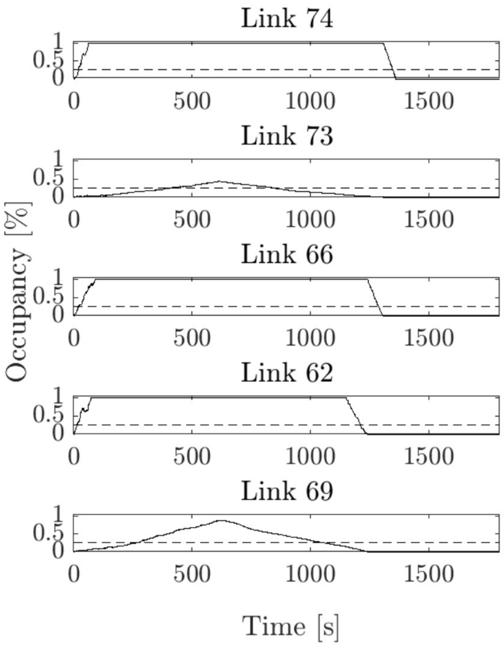
	Free flow	Baseline	Proposed	Improvement
Total travel time	3,162,540 s	6,234,352 s	5,427,291 s	12.9 %
Total delay	—	3,071,812 s	2,264,751 s	26.3 %

Aalto University
School of Engineering

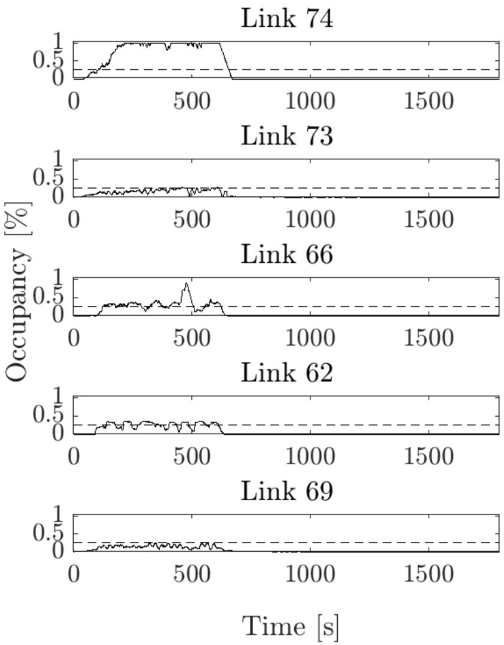
Results

Occupancy N_{lj}/N_{lj}^{J}

Baseline

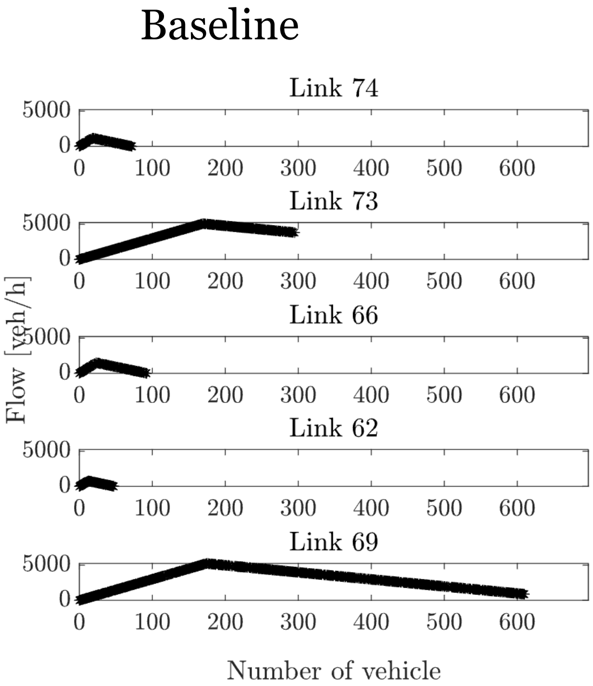


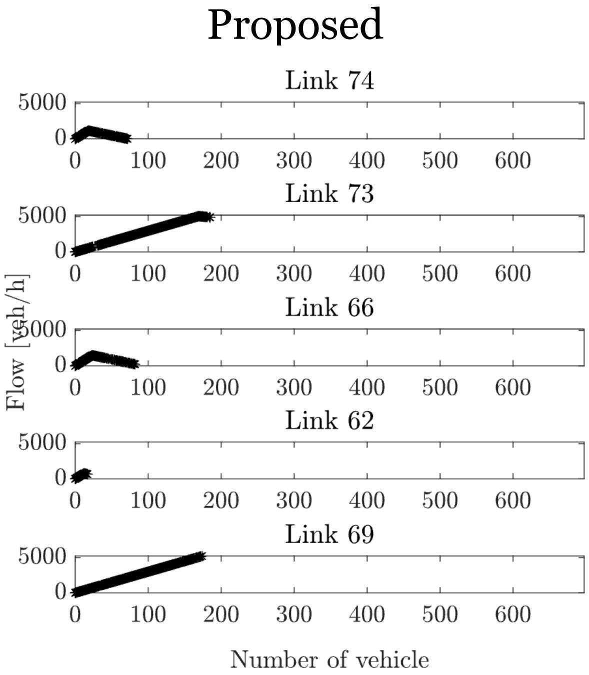
Proposed



Results

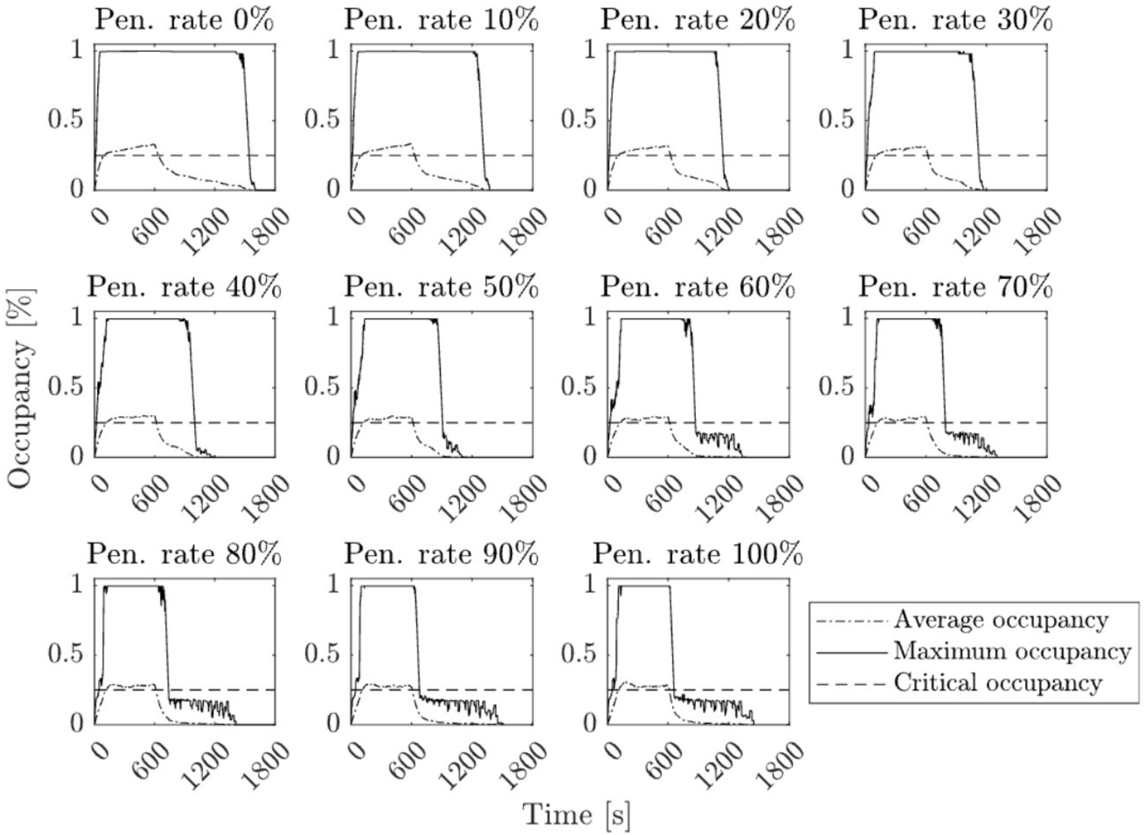
Fundamental diagrams





Results

Penetration rates



Vitale, F. and Roncoli, C. Cooperative rerouting to redistribute the load of Connected and Automated Vehicles in urban networks. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC).*

Conclusions

Conclusions

- Methods for distributed control of CAVs

- Reduced data exchange and limited computation time

- Distributed optimisation for signal-free intersection management
    - Time-dependent formulation
    - Space-dependent formulation

- Distributed algorithm for network routing, leading to
    - User-related objectives (**delay reduction**)
    - System-related objectives (**traffic dispersion**)

THANK YOU!

claudio.roncoli@aalto.fi

Aalto University
School of Engineering