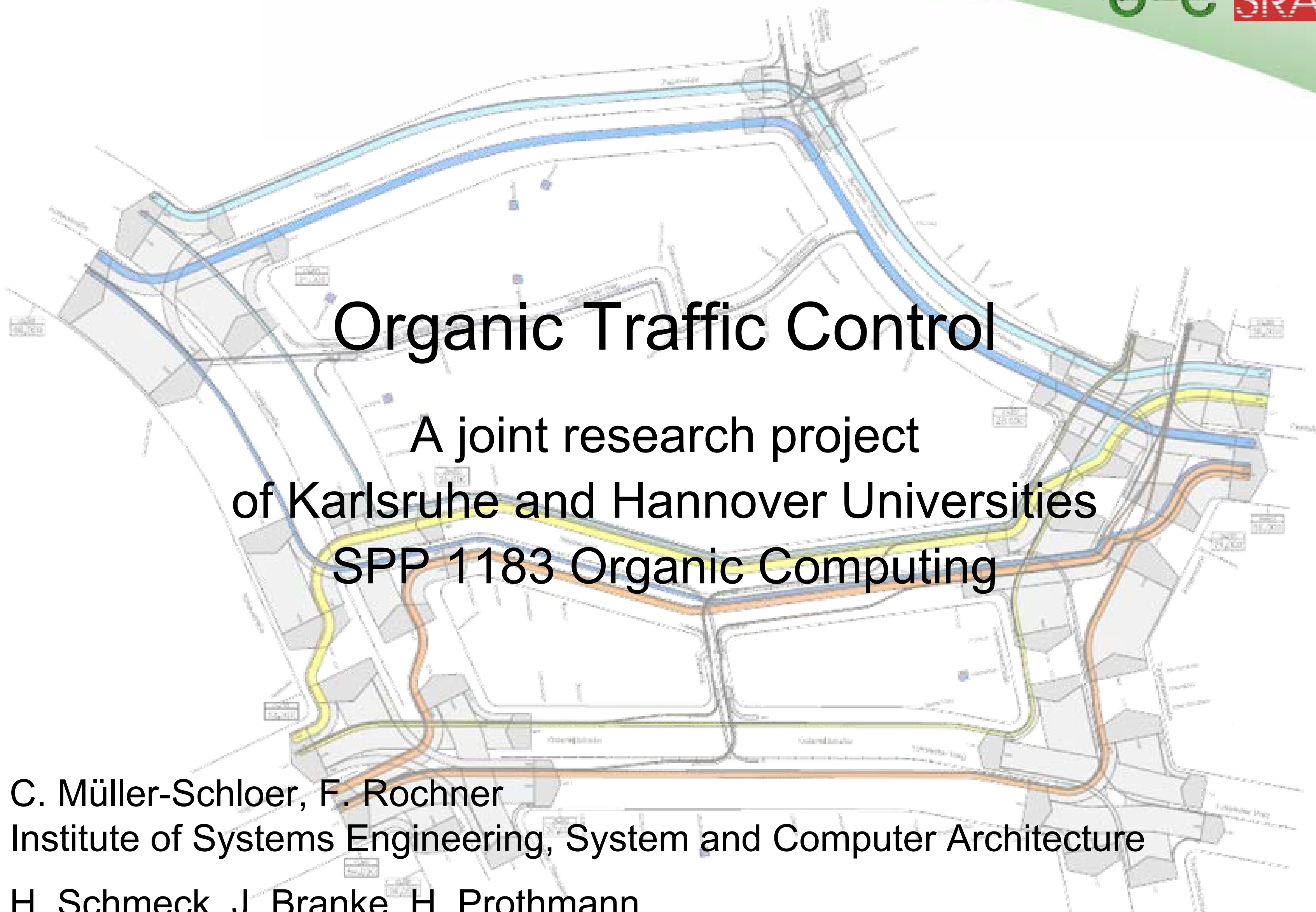


Organic Traffic Control

A joint research project
of Karlsruhe and Hannover Universities
SPP 1183 Organic Computing

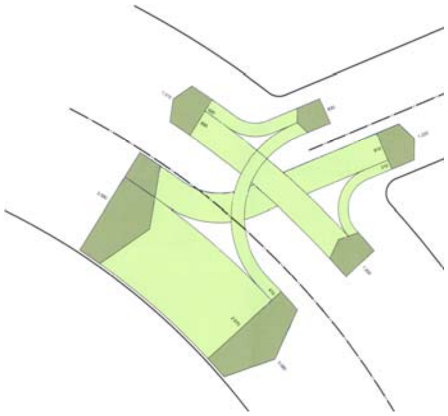
C. Müller-Schloer, F. Rochner
Institute of Systems Engineering, System and Computer Architecture

H. Schmeck, J. Branke, H. Prothmann
Institute of Applied Informatics and Formal Description Methods

Organic Traffic Control

Goals

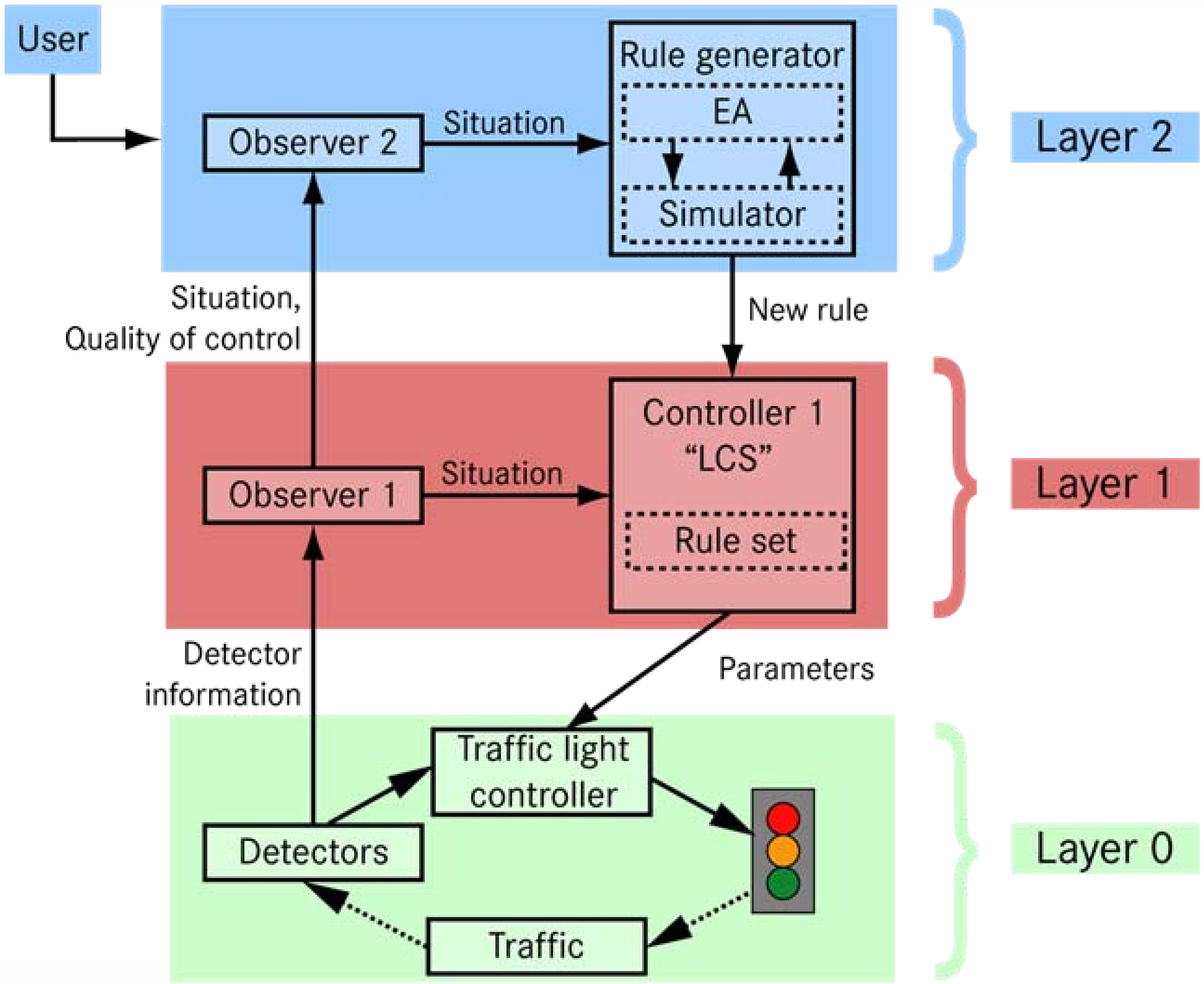
- Study a network of adaptive learning traffic light controllers (TLCs)
- TLCs learn with some limited sensory horizon
- TLCs cooperate to achieve a global goal (reduced avg. travel time)
- Explore possibilities/limitations of decentralized control systems



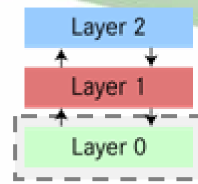
Goals for phase 1

- Design generic architecture
- Implement for single, isolated junction
- Test, optimize

Architecture



Layer 0
Direct Control of Junction



TLCs: Different levels of complexity

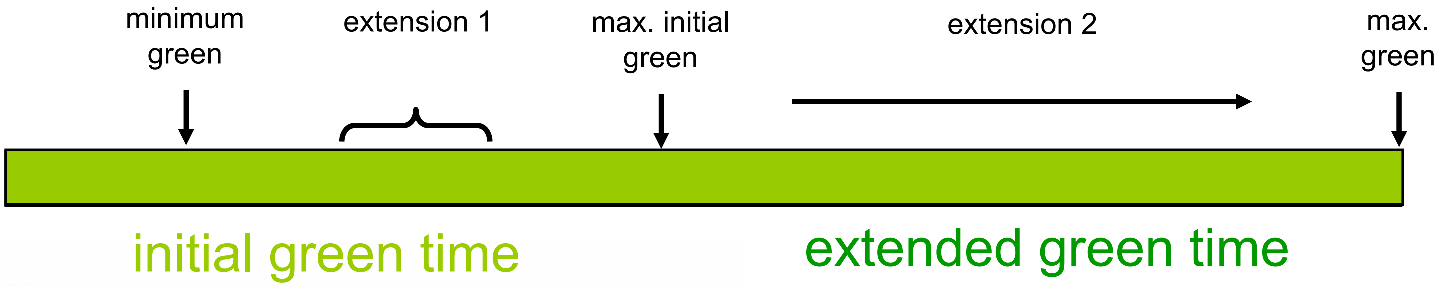
Fixed time control

- \+ Coordination of junctions easy
- \+ Simple parameter set
- \- No adjustment to traffic → external component needed to adjust controller

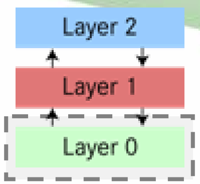
Traffic responsive control

- \+ React on sensor data
- \- Coordination of junctions difficult
- \- Numerous parameters → boundary conditions, no globally best parameter set

- Trade-off: Complexity on Layer 0, 1 and 2



Layer 0
Direct Control of Junction (2)

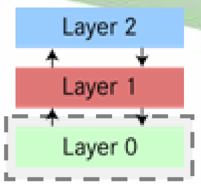


- Trade-off: Complexity on Layer 0, 1 and 2

Layer 0	Layer 1	Layer 2
Fixed time control	• Frequent intervention • Fine-grained traffic observation • Large population	• Small search space • Many parameter sets need to be computed.
Traffic responsive control	• Interventions less frequent • Coarse classification of traffic situations • Smaller population	• Large search space, many parameters. • Fewer parameter sets need to be computed.

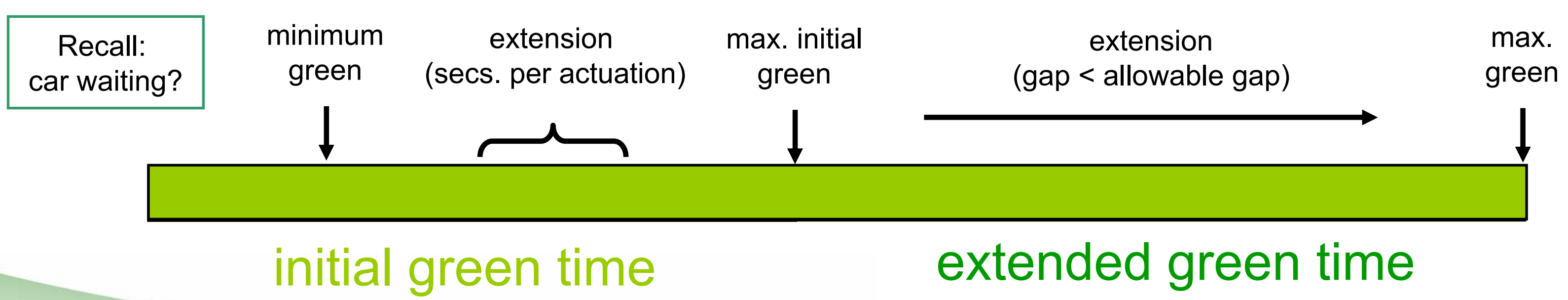
- Goal: Smooth transition → easy tuning, close to standards

Layer 0
NEMA-Controller

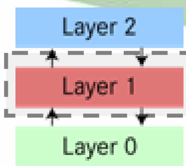


American standard for TLCs *National Electrical Manufacturers Association*

1.	Fixed time control	(1 parameter per phase)
2.	Recall	(2)
3.	Extend phase if gap is small	(4)
4.	Extend phase if queue is long	(6)
5.	Adjust gap	(9)
6.	Change sequence of phases	(>9)

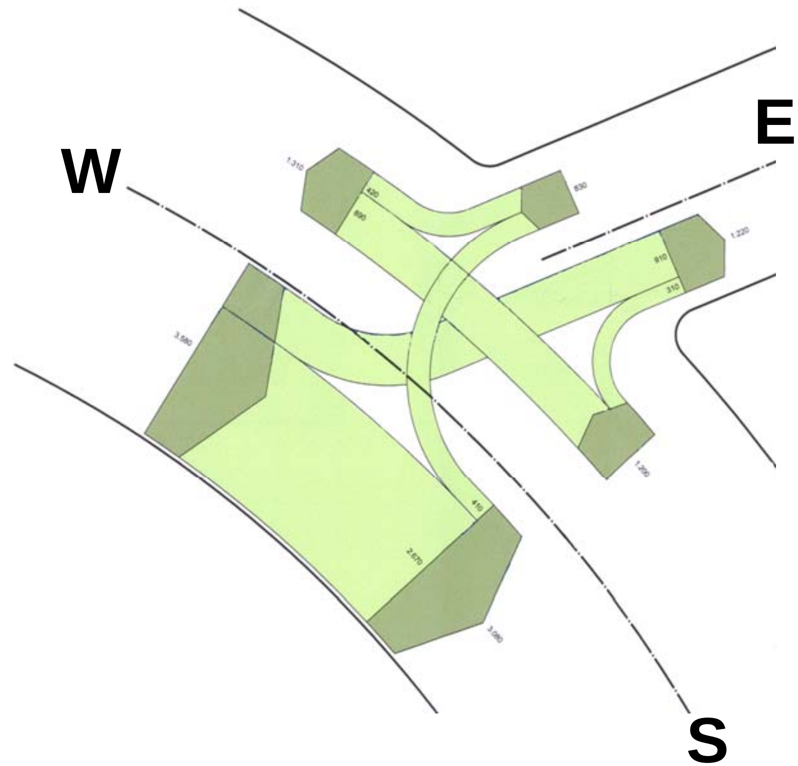


Layer 1
Online-Selection of TLC



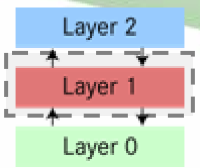
Input: Observer

- Available data: Demand on approaches, average speed, waiting times, up- and downstream information
- Aggregation: Single number per signal group (→ typically 6 to 12 numbers per junction)
- Numbers combined: “Traffic situation” as input for controller



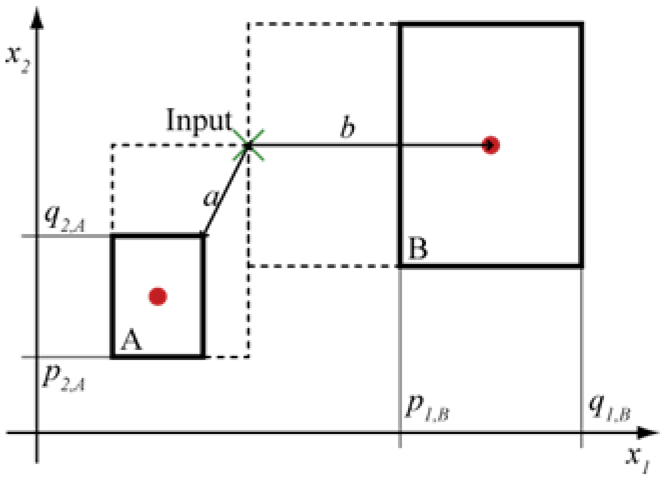
W→E	W→S	E→W	
910	2670	420	…

Layer 1
Online-Selection of TLC (2)



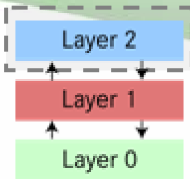
Output: Controller

- Classifier system, reduced functionality:
  - No rule generation
  - Refinement of classifier's values
- New rules: Matching situation precisely, no generalization
- Widening of condition
- Input likely not covered by existing classifier's condition



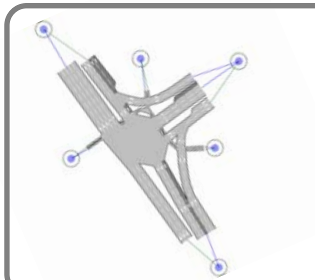
- “Closest” classifier duplicated, condition widened to match
- Widening > threshold → Trigger rule generation (Layer 2)
- Tune: “Closeness”, threshold

Layer 2
Offline-Generation of TLCs



- + Simulation-based evaluation of TLCs
- + No negative influences on real node
- + Fast TLC evaluation

Example
- Traffic node in Hamburg
- Simulated for 1 hour during morning peak period
- AMD Duron 1,6 Ghz, 512 MB, AIMSUN 5.0.13

	**fixed time controller (reference)**	**fixed time controller (generated)**	**NEMA controller (generated)**
	avg. delay: 61.2	10 minutes avg. delay: 59.5	15 minutes avg. delay: 36.3

- + Automatic generation of TLCs
- - Quality of generated rules depends on precision of simulated environment

Separation of tasks

2 parts, asynchronous operation

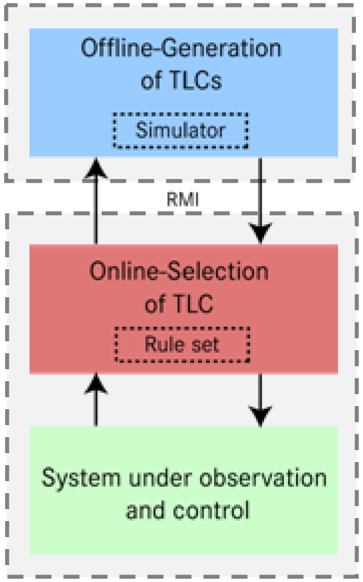
Part 1: Layer 0 and 1

- Observer: Collect data from network → traffic situation
- TLC to control single junction
- Controller: Choose TLC matching current traffic situation

Part 2: Layer 2

- Simulate (part of) network
- Adjust according to traffic situation received
- Find optimal TLC, send back

Benefit: Parts can use separate computers, “Simulation Server”, still decentralized structure



Summary

Architectural benefits

- Autonomous response to changing traffic situations
- Separation of learning and on-line control (speed, safety)
- Automatic generation of TLCs
- Separation of tasks that demand much resources
- Usage of industry-standard logic for TLCs

Architectural drawbacks

- Simulation needs calibrated model and significant computing power.
- Many different technologies, many interfaces

Status and Outlook

Status

- Concept development
  - F. Rochner, H. Prothmann, J. Branke, C. Müller-Schloer, and H. Schmeck. An organic architecture for traffic light controllers. In *Informatik 2006 – Informatik für Menschen*, pages 120–127. Köllen Verlag, 2006.
  - J. Branke, M. Mnif, C. Müller-Schloer, H. Prothmann, U. Richter, F. Rochner, and H. Schmeck. Organic Computing – Addressing Complexity by Controlled Self-organization. Accepted for ISoLA 2006.
- System implementation

Cooperation

- AutoNomos (Fekete/Fischer)
- ASoC (Rosenstiel/Herkersdorf)

Outlook …

- Evaluation based on real data
- Extensions and improvements

… and 2nd SPP-Phase

- Focus on networks
- Intensify QE-cooperation