Why is it so hard to make self-driving cars?
(Trustworthy Autonomous Systems)

ESReDA Webinar
January 22, 2021

Joseph Sifakis

Autonomous systems – Main Characteristics

Autonomous systems are essential for reaching the Industrial IoT vision.

❑ They emerge from the needs to further automate existing organizations by progressive and incremental replacement of human operators by autonomous agents.

❑ They are very different from game-playing robots or intelligent personal assistants.

❑ They are often critical and should exhibit “broad intelligence” by handling knowledge in order to

- Manage dynamically changing sets of possibly conflicting goals – this reflects the trend of transitioning from “narrow” or “weak” AI to “strong” or “general” AI.
- Cope with uncertainty of complex, unpredictable cyber physical environments.
- Harmoniously collaborate with human agents e.g. “symbiotic” autonomy.

- Autonomous vehicles are a topical and emblematic case illustrating the obstacles to be overcome to meet the challenge for trustworthy autonomous systems.
- Building autonomous transport systems would be a big step toward closing the gap between machine and human intelligence

Why is it so hard to make self-driving cars?

Despite

- the enthusiastic involvement and the massive investment of big tech companies and car industry (*);

(*)"I almost view [autonomous cars] as a solved problem. We know what to do, and we'll be there in a few years." E. Musk, Nvidia Technology Conference, March 2015.

- the optimistic predictions about self-driving cars "being around the corner"

- AV manufacturers revise their ambitions because of technical problems and the erosion of public trust (*).

(*) "*I think both industry and media have been complicit in hyping this and not being open and honest enough about the realities of the technology*."
Jack Weast, vice president, autonomous vehicle standards, Intel, July 2019.

- Over-optimism led to misconceptions about what is technically achievable under the current state of the art and future evolution.

❑ Two different technical avenues both falling short of the autonomy challenge:

- traditional model-based critical systems engineering, successfully applied to aircraft and production systems, proves to be inadequate.
- industrial end-to-end AI-enabled solutions currently available e,g, "Waymo Driver", fail to provide the required strong trustworthiness guarantees.

❑ To meet the challenge we need to develop a new scientific and engineering foundation.

❑ Why is it so hard?

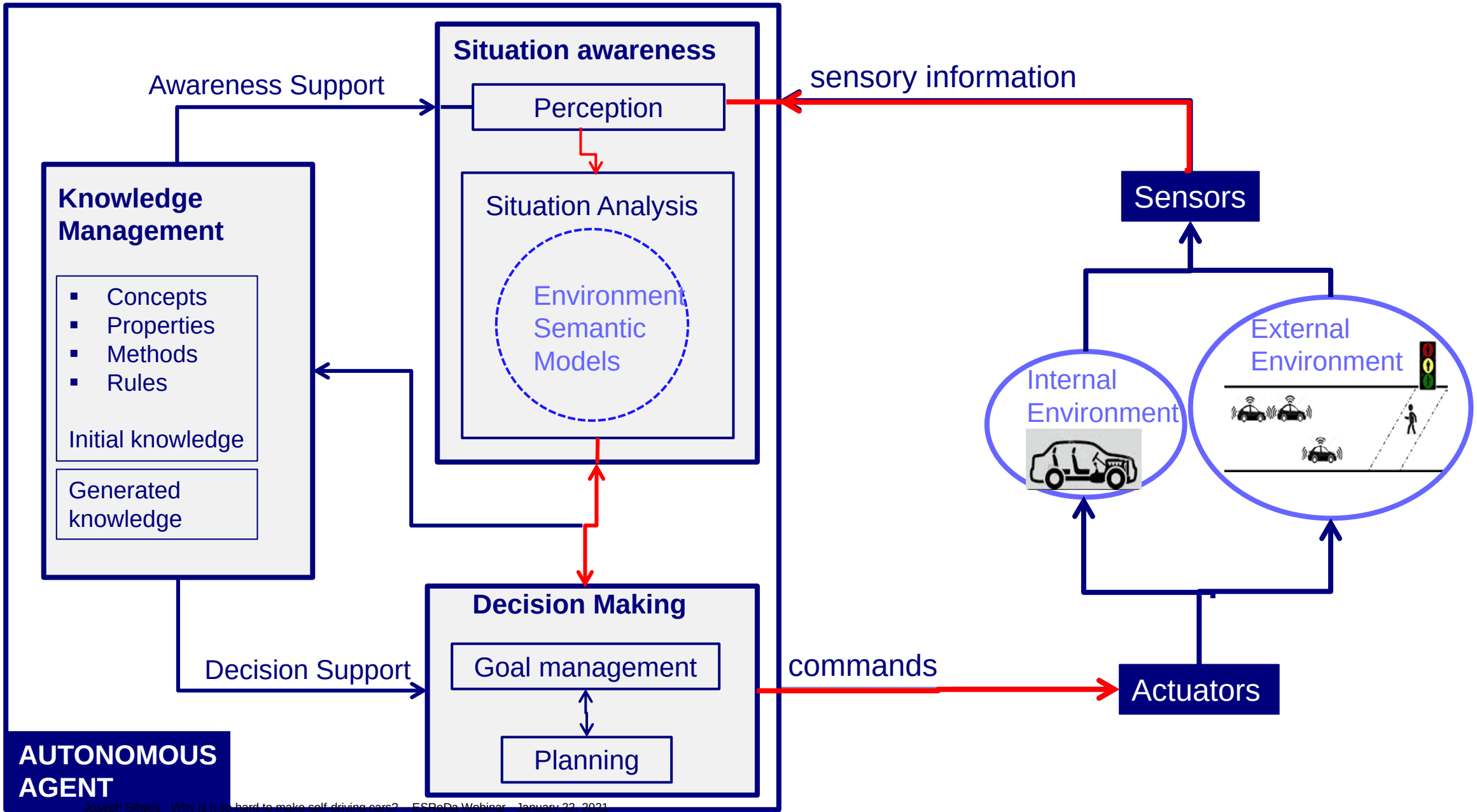
❑ Achieving Trustworthiness

- Trustworthiness vs. Criticality
- Trustworthy Autopilot Design
- When Self-driving Cars are Safe Enough?

❑ Discussion

OVERVIEW

Why is it so hard? – Autonomous Agent Architecture



Why is it so hard? – SAE Autonomy Levels

	SAE AUTONOMY LEVELS	
Level 0	No automation	AUTOMATION (ADAS)
Level 1	Driver assistance required	
	The driver still needs to maintain full situational awareness and control of the vehicle e.g. cruise control.	
Level 2	Partial automation options available	
	Autopilot manages both speed and steering under certain conditions, e.g. highway driving.	
Level 3	Supervised Autonomy	AUTONOMY
	The car, rather than the driver, takes over actively monitoring the environment when the system is engaged. However, human drivers must be prepared to respond to a "request to intervene"	
Level 4	Geofenced autonomy	
	Self driving is supported only in limited areas or under special circumstances, like traffic jams	
Level 5	Full autonomy	
	No human intervention is required e.g. a robotic taxi	

Why is it so hard? – Autonomic Complexity

- Complexity of perception characterizes the difficulty to interpret stimuli (cope with ambiguity, vagueness) and to timely generate corresponding inputs for the agent environment model.

- Uncertainty due to

  - situations involving imperfect or unknown information implying lack of predictability about the environment such as dynamic change caused by physical or human processes, rare events, critical events such as failures and attacks.
  - entirely new situations not anticipated at design time – requires self-learning and generation of new goals

- Complexity of decision reflected in the complexity of the agent's decision process (goal management and planning) and impacted by the following factors:

  - type of goals e.g. safety, reachability, security, optimization of resources
  - multiplicity of goals, especially long/mid/short goals, potentially conflicting
  - complexity of the space of solutions to be explored for plan generation and lack of controllability

Just building autonomous agents is less than half of the job!!!

Two important Systems Engineering issues :

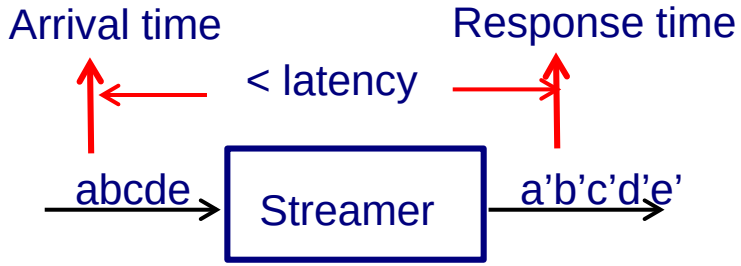
- agents should be integrated in complex cyber physical and human environments
- agents are subject to complex dynamic reconfigurable coordination rules

Why is it so hard? – Reactive Complexity

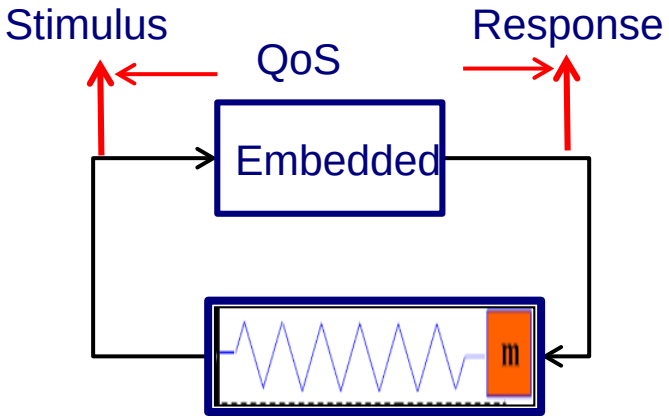
Reactive complexity of agents

- characterizes the intricacy of the interaction between an agent and its environment.
- is independent from space complexity or time complexity (related to resources needed)

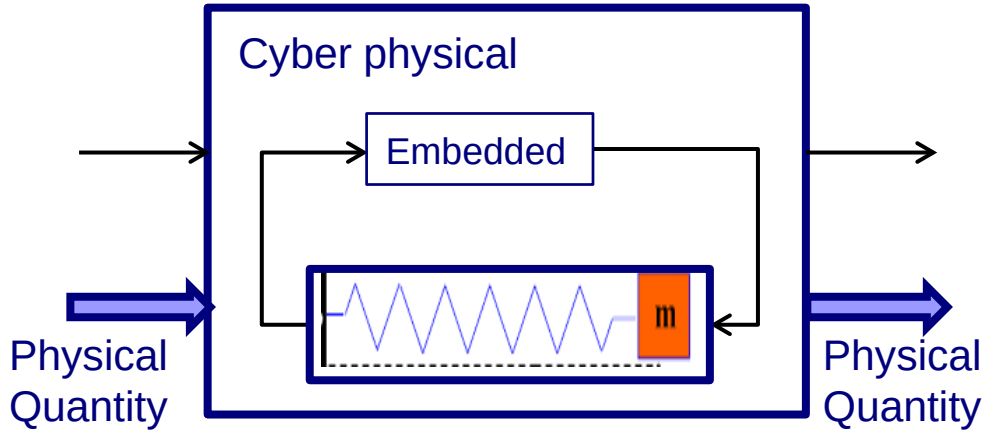
Transformational agent e.g.
Intelligent Personal Assistant



Streaming Agent
e.g. Encoder, Signal processor



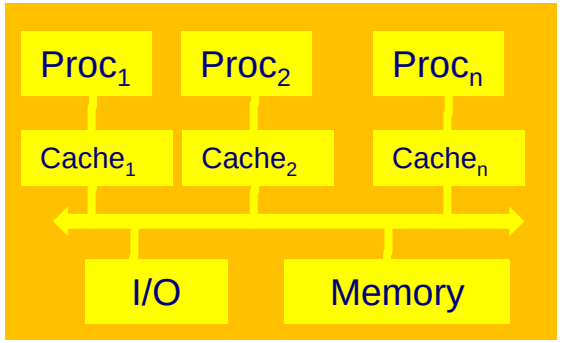
Embedded Agent
e.g. Flight controller



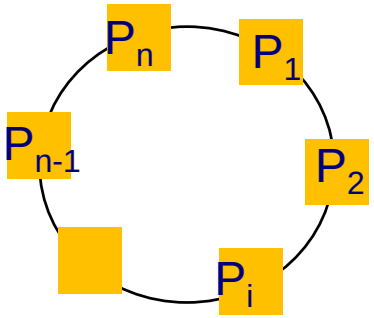
Cyber physical agent
e.g. Self-driving car

Why is it so hard? – Architectural Complexity

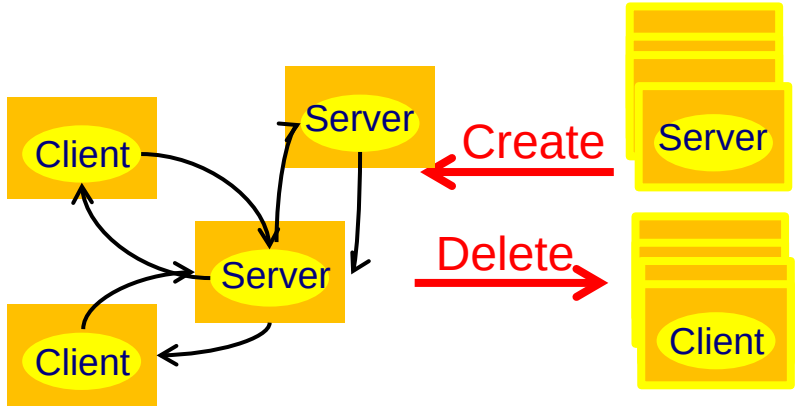
How much involved is the coordination between components?



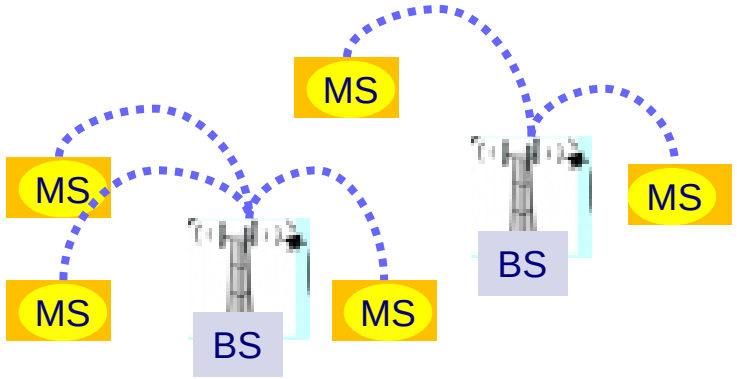
Static Architecture:
Multiprocessor System



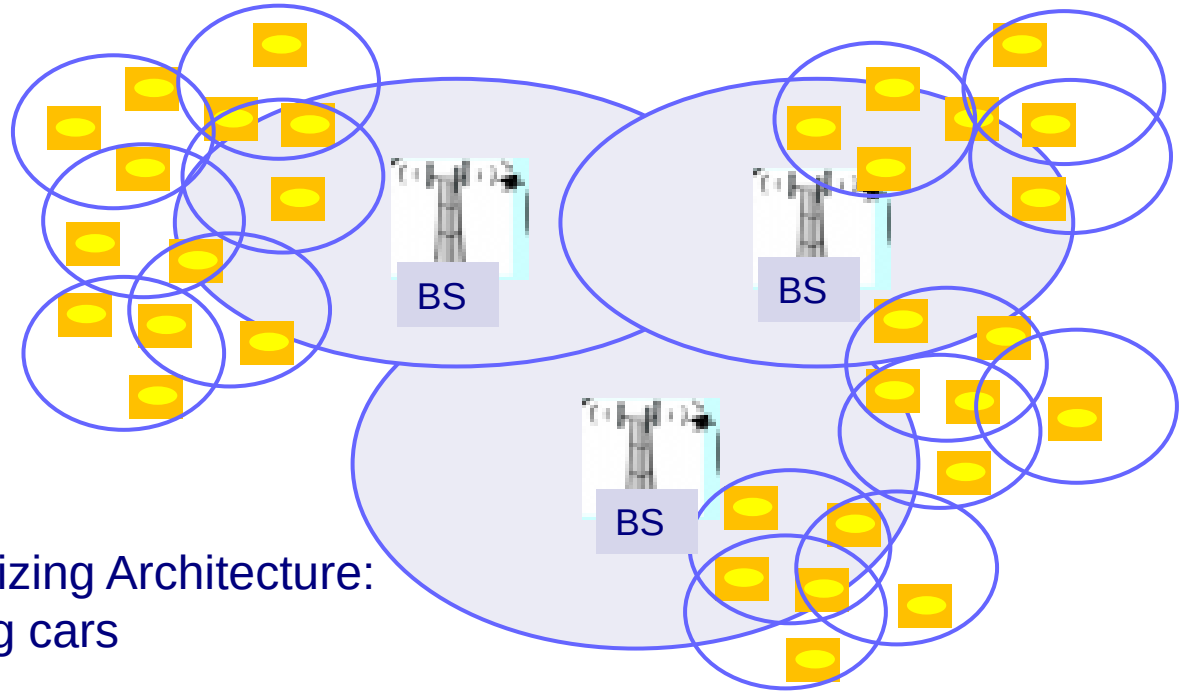
Parametric Architecture:
Ring Architecture



Dynamic Architecture:
Distributed System



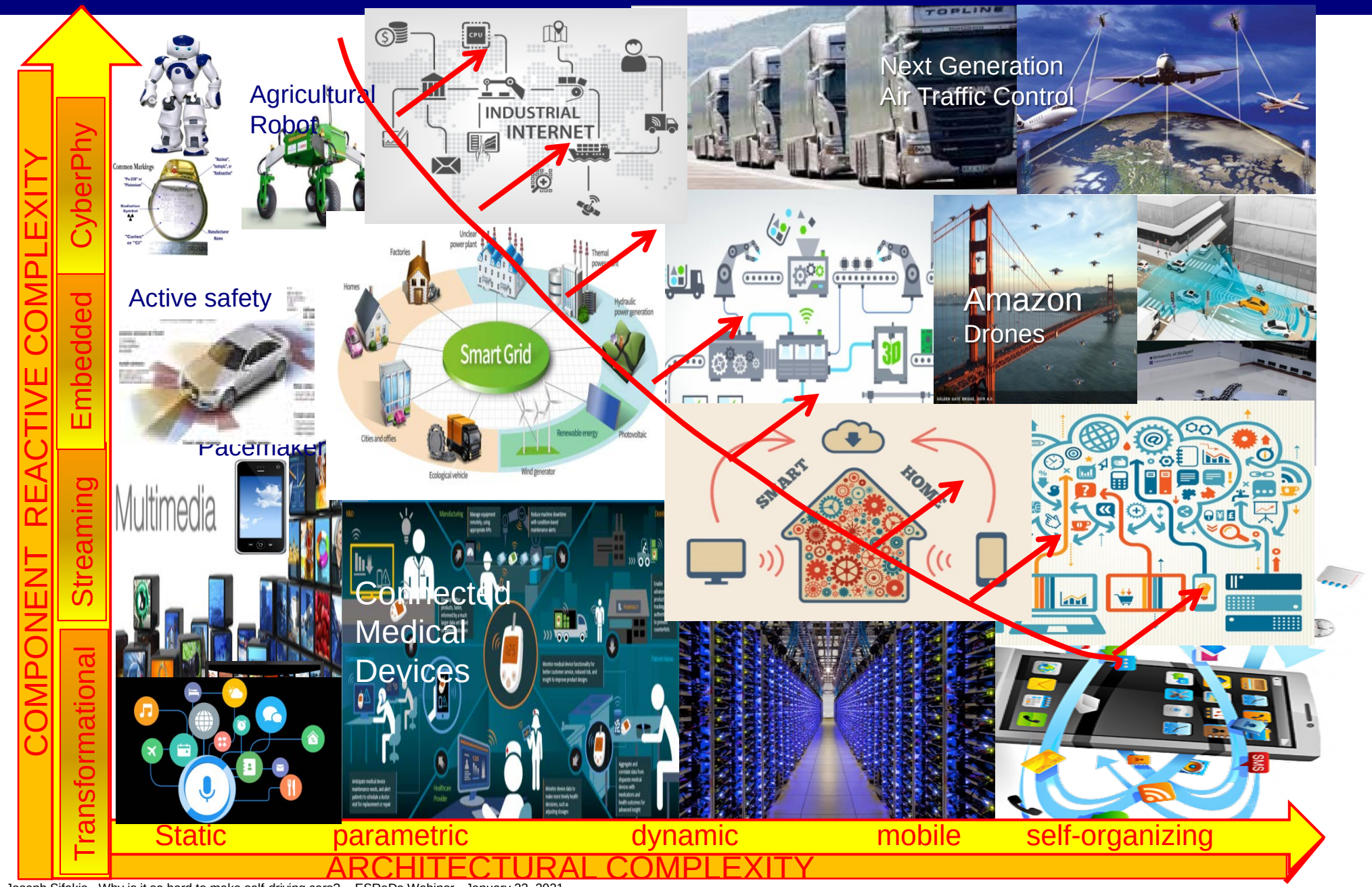
Mobile Architecture:
Mobile phones



Self-organizing Architecture:
Self-driving cars

Why is it so hard? – Systems Engineering Complexity

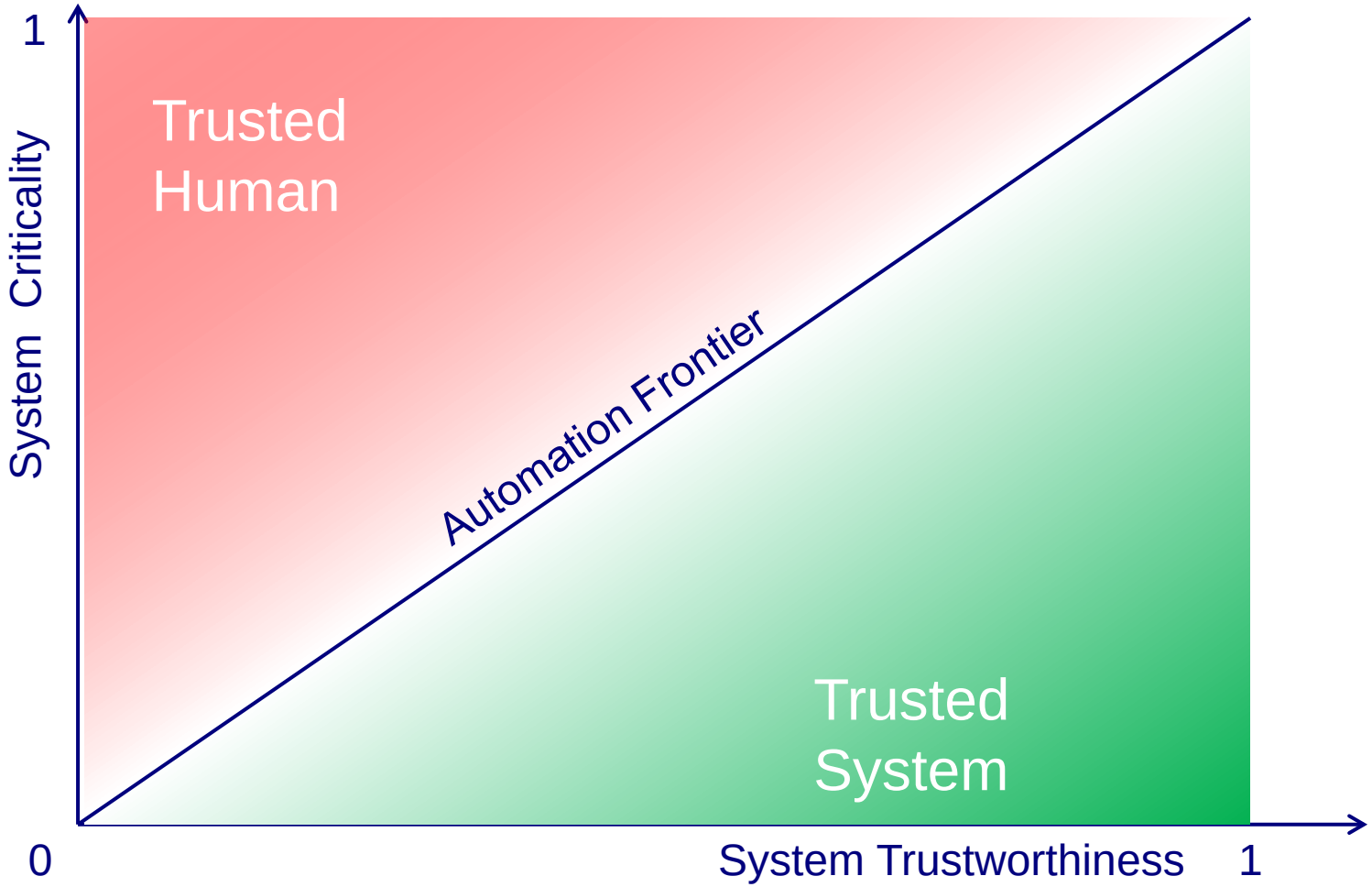
Why is it so hard? – Systems Engineering Complexity



OVERVIEW

- Why is it so hard?
- Achieving Trustworthiness
  - Trustworthiness vs. Criticality
  - Trustworthy Autopilot Design
  - When Self-driving Cars are Safe Enough?
- Discussion

Trustworthiness vs. Criticality – Automation Frontier

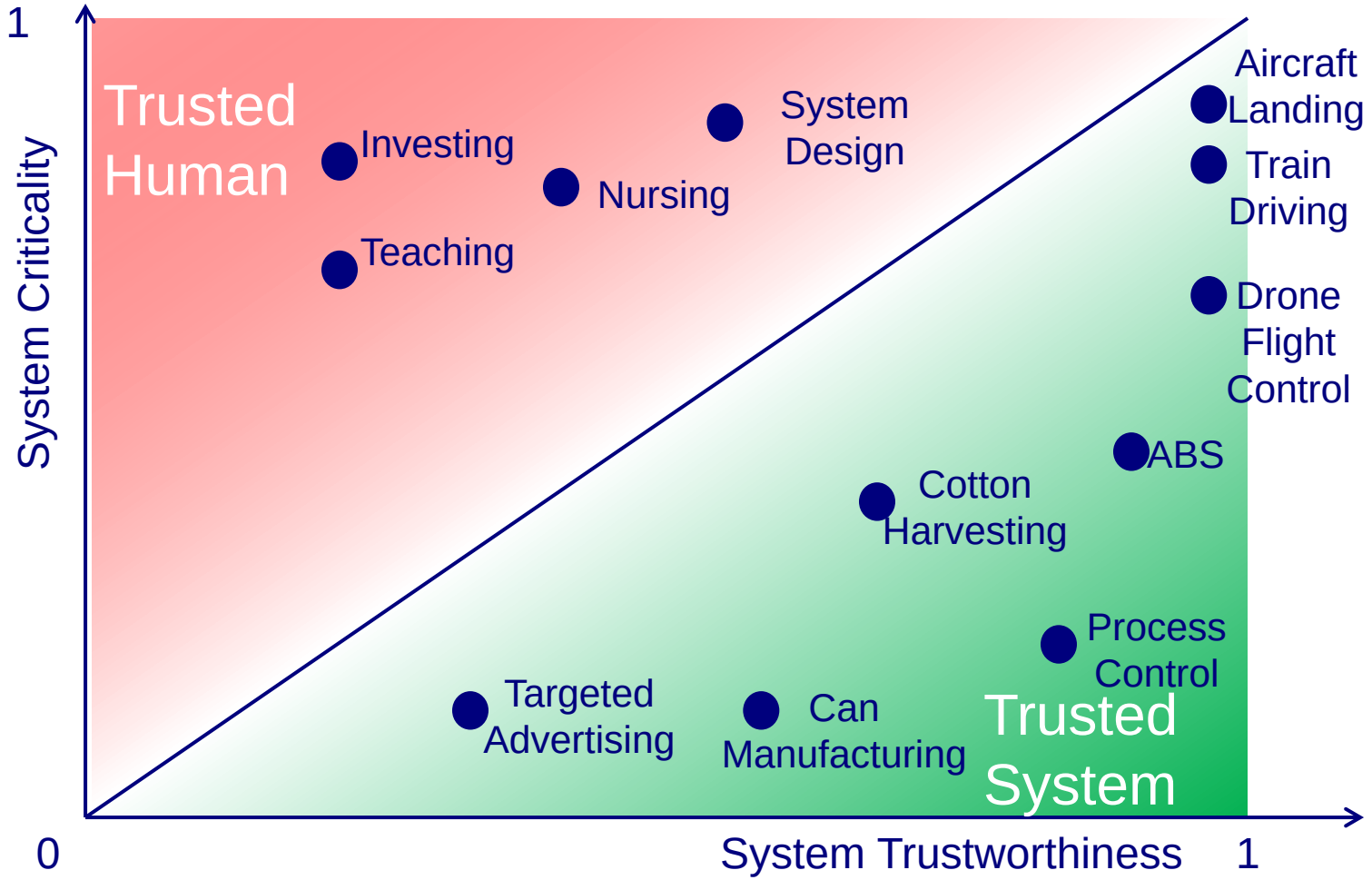


How we decide whether a System can be trusted?

- System Trustworthiness: the system will behave as expected despite any kind of hazards e.g. resilience to errors, failures, attacks – subsumes functional correctness.

- System Criticality: characterizes the severity of the impact of a system hazard e.g. driving a car, operating on a patient, nuclear plant control.

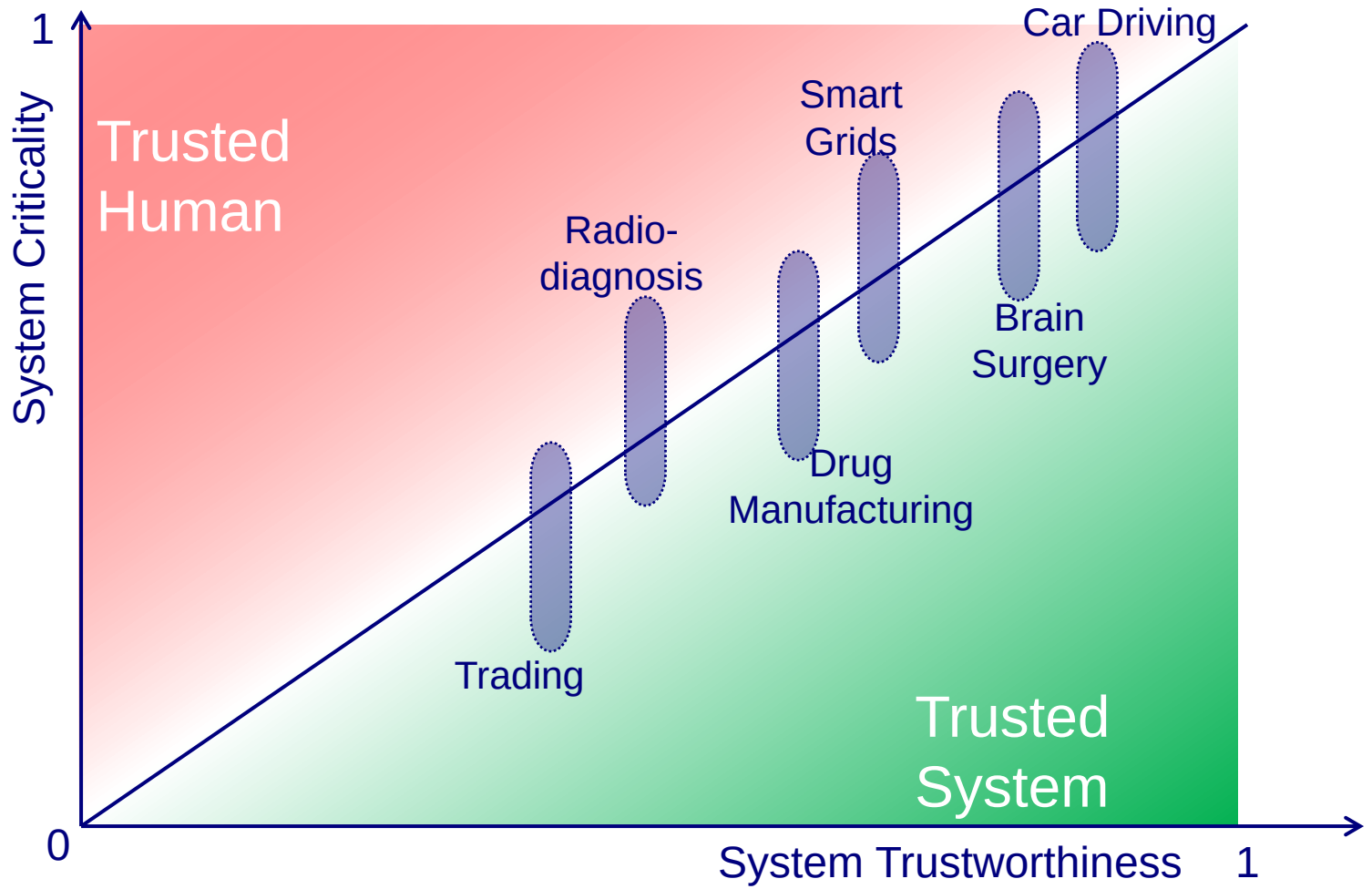
Trustworthiness vs. Criticality – Automated Systems



Automated systems: static decision process and/or small impact of failures.

Non-automated systems: require good situation awareness and multiple goal management.

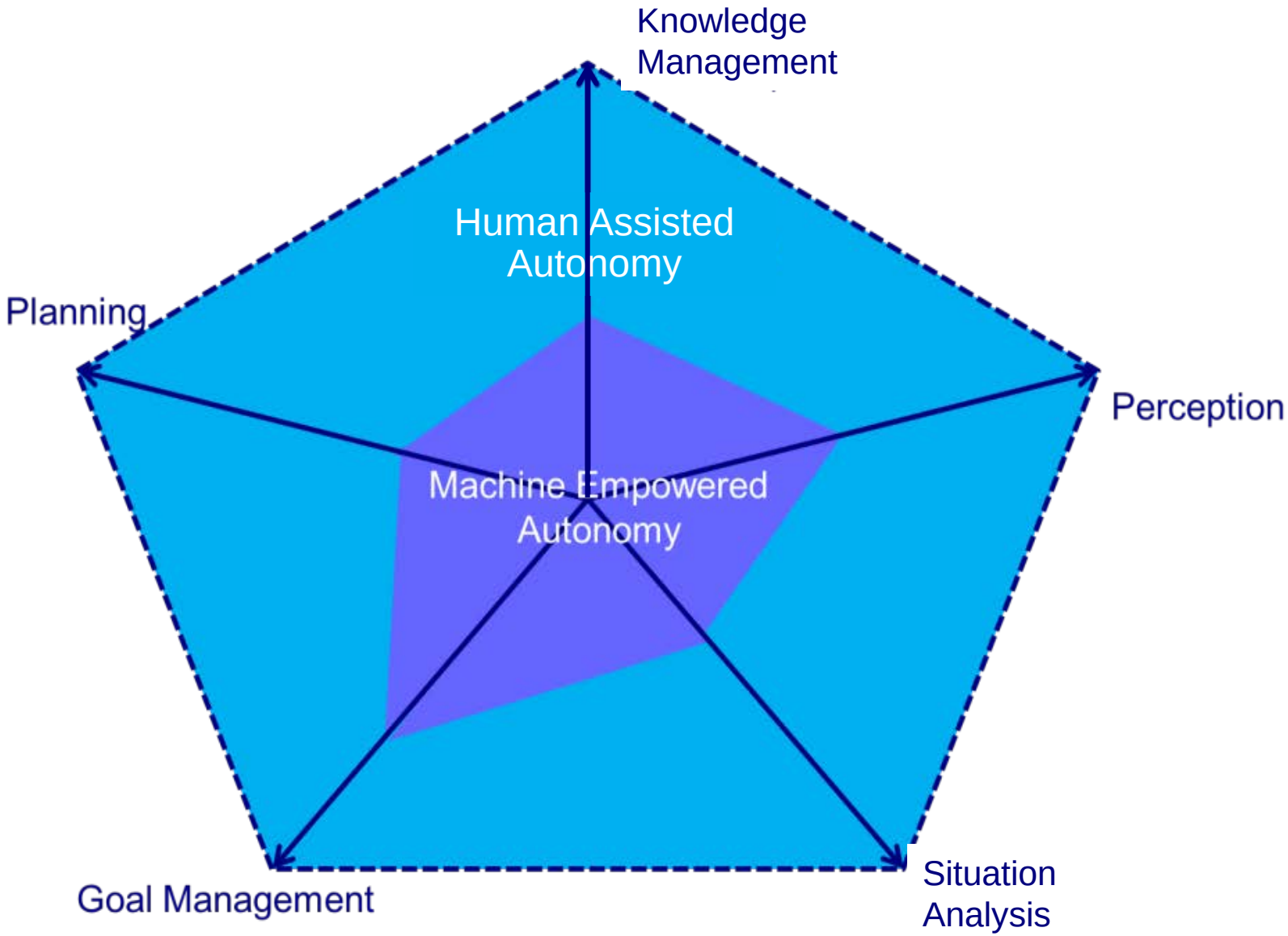
Trustworthiness vs. Criticality – Symbiotic Autonomy



Problem: Choose the appropriate autonomy level (protocol and rules) for the harmonious and safe collaboration between humans and machines allowing in particular

- a human agent to override the machine's decision(s);
- a machine to proactively solicit human agent's intervention.

Trustworthiness vs. Criticality – Symbiotic Autonomy (2)



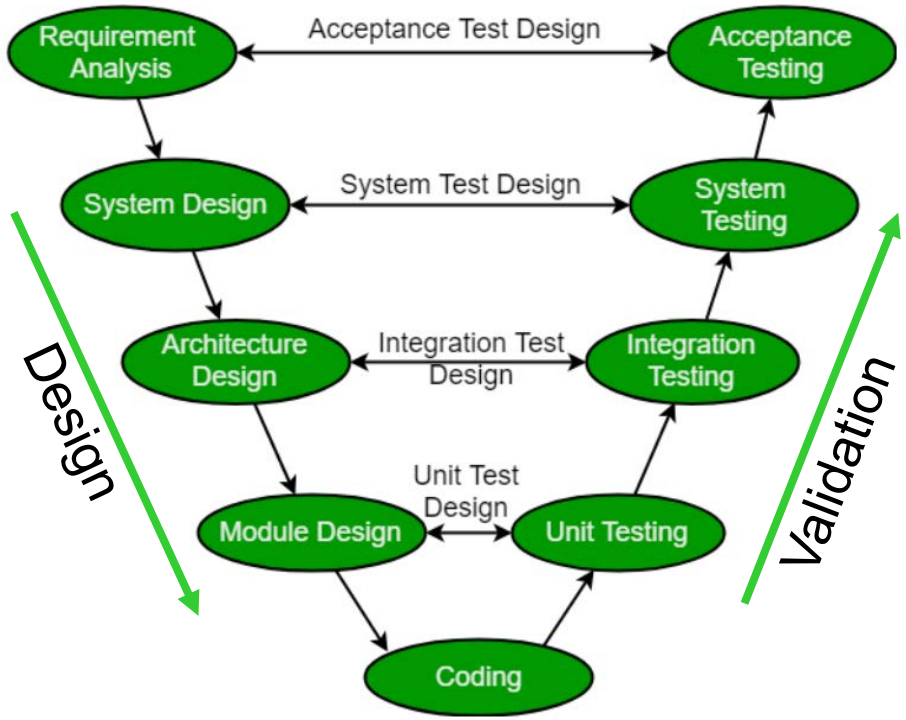
OVERVIEW

- Why is it so hard?
- Achieving Trustworthiness
  - Trustworthiness vs. Criticality
  - Trustworthy Autopilot Design
  - When Self-driving Cars are Safe Enough?
- Discussion

Autopilot Design – Critical Systems Engineering Limitations

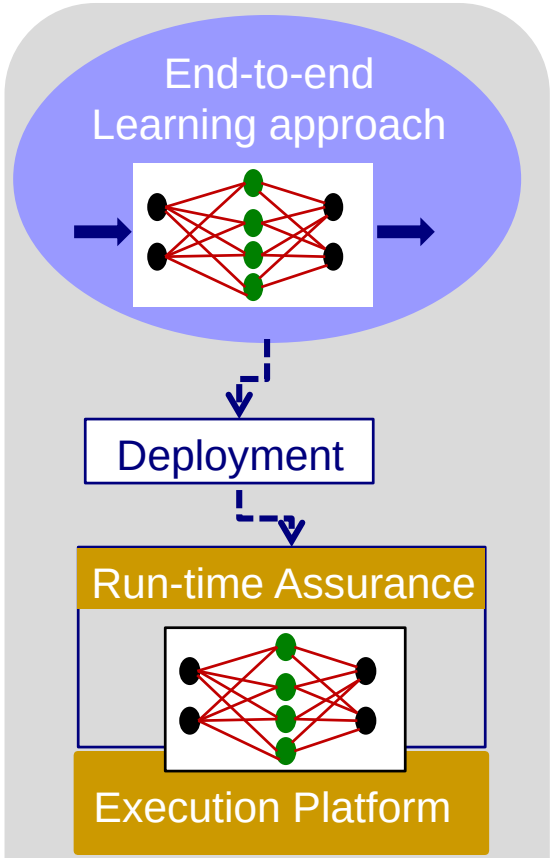
Critical systems design flows follow model-based prescriptive frameworks recommended by standards e.g. ISO26262

- Assume that system development is top-down and validation is bottom-up.
- Assume that all requirements are initially known, can be clearly formulated and understood.
- Consider that global system requirements can be broken down into requirements satisfied by system components.
- Focus on providing model-based conclusive evidence that the system is safe e.g.
  10^{-9} failures per hour of flight



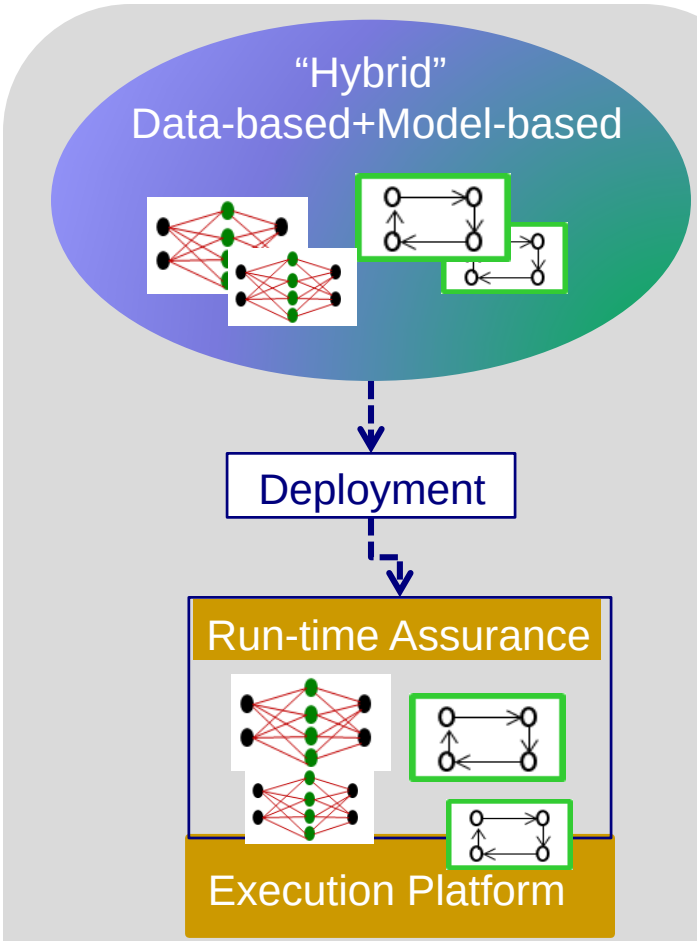
- The model-based paradigm is defeated by the overwhelming complexity and diversity of autonomous systems
- This explains the adoption by industry of end-to-end machine-learning-enabled techniques which however preclude conclusive safety guarantees

Autopilot Design –Taking the Best from Each

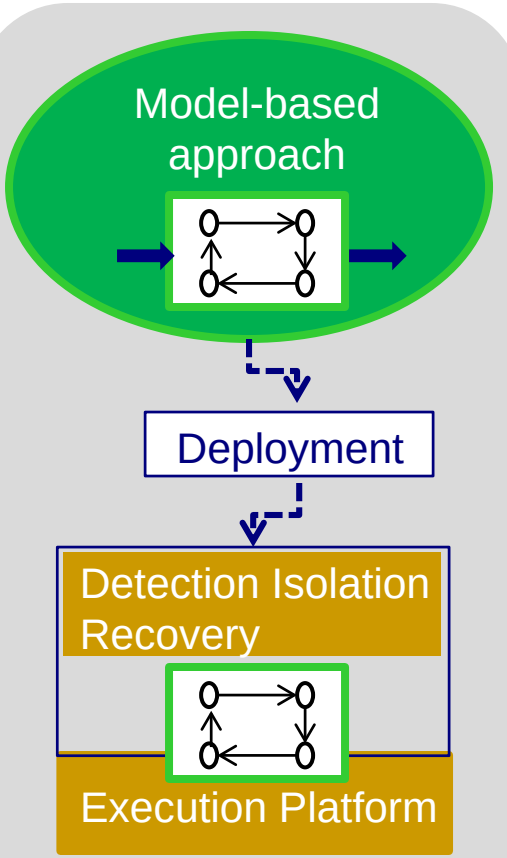


ML-ENABLED SYSTEM

(PERFORMANCE)



HYBRID AUTONOMOUS SYSTEM
(TRADE OFF: PERFORMANCE vs. TRUSTWORTHINESS)



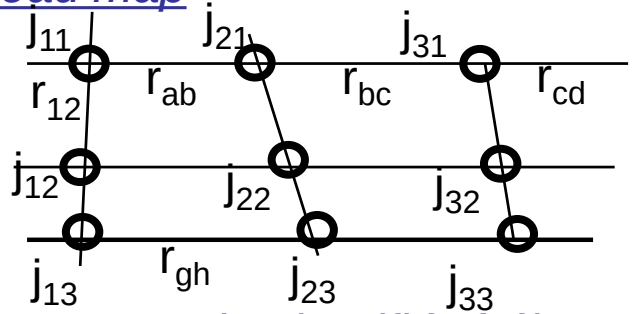
AUTOMATED SYSTEM

(TRUSTWORTHINESS)

Autopilot Design – Hierarchical Semantic Model

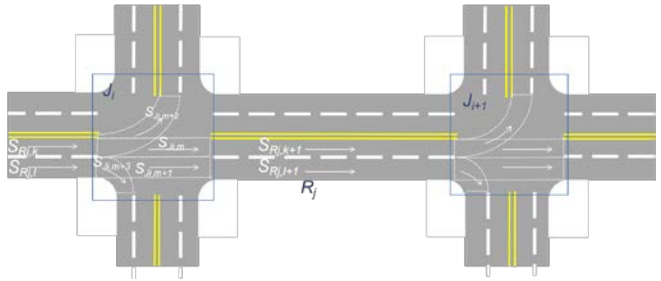
Maps

Road map



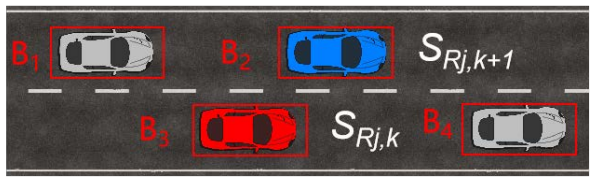
$RN = (J,R) = (\{j_k\}, \{r_s\})$

Lane map



$SN = (S,E)$

Synthesized local map



Hierarchical Autopilot

Mission Planning - Level 4

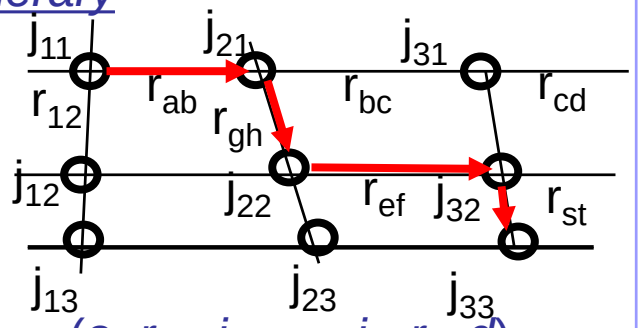
Path Planning - Level 3

Maneuver Planning & protocols - Level 2

Trajectory Planning - Level 1

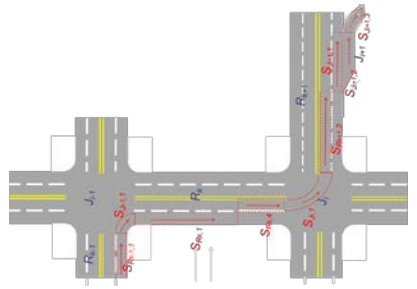
Goals

Itinerary



$(o, r_{ab}, j_{21}, \ldots j_k, r_k, d)$

Path

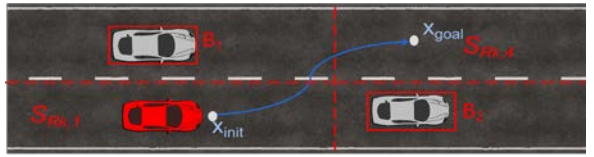


$(s_{Rk-1,1}, s_{Ji-1,1}, s_{Rk,1}, s_{Rk,4}, s_{Ji,1}, s_{Rk+1,2})$

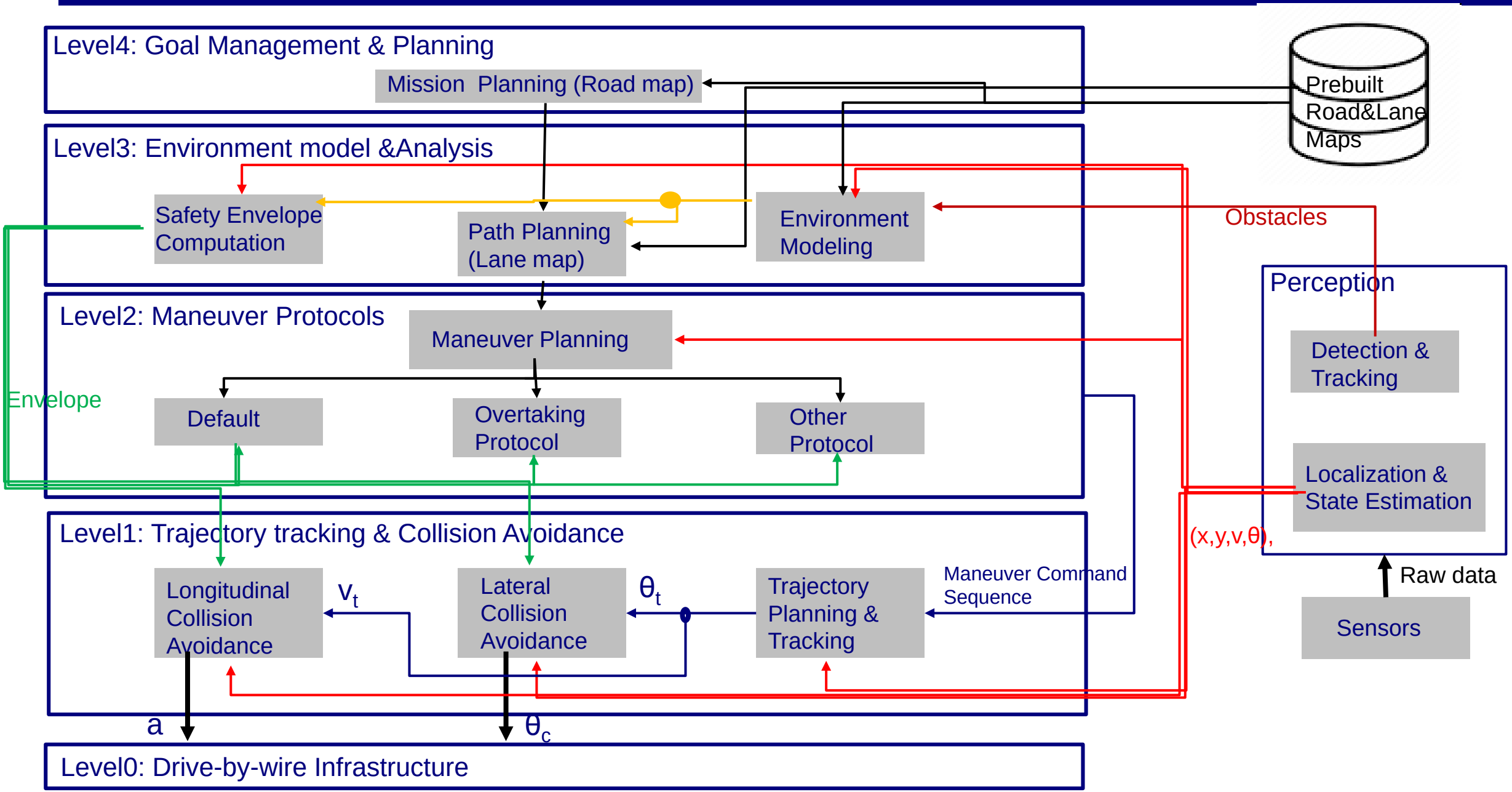
Maneuver sequences

$(lane\_change, s_{Rk,1}, s_{Rk,4})$

Trajectory



Autopilot Design – Hierarchical Architecture



OVERVIEW

- Why is it so hard?
- Achieving Trustworthiness
  - Trustworthiness vs. Criticality
  - Trustworthy Autopilot Design
  - When Self-driving Cars are Safe Enough?
- Discussion

When Self-driving Cars are Safe Enough? – The “Miles Argument”

Waymo has now driven 10 billion autonomous miles in simulation

Darrell Etherington @etherington / 11:17 pm CEST • July 10, 2019

Comment

It is possible to compute (*) the number of miles needed to drive without accident

- for given rate of accident type per mile driven
- for given confidence level

Accident type	Miles needed at confidence level	
	95%	50%
With fatality	291 million	67 million
With personal injury	5.4 million	1.2 million
Crash (=police-reported accident)	1.5 million	344.000
Any accident (estimated)	745.000	172.000

(*) https://www.rand.org/content/dam/rand/pubs/research_reports/RR1400/RR1478/RAND_RR1478.pdf, Rand report April 2016

The “miles argument” is not technically tenable e.g. 27 000 cars running 24/7, 10 million miles simulated per day, >10 Billion miles in simulation

When Self-driving Cars are Safe Enough? – Gaps in the State of the Art

- Global system validation is achievable only through simulation and testing, which nonetheless should take into account the following:

  - All the simulated miles are not equally efficacious - how a simulated mile is related to a “real mile” ?
  - Any technically sound safety evaluation should be model-based e.g. relies on criteria defined on an implicit or an explicit system model.
  - We need evidence that simulation covers a good deal of the many and diverse situations e.g. different types of roads, traffic conditions, weather conditions etc.

- We need validation theory based on the semantic simulation model.

  - Notions of coverage measuring the degree to which relevant system configurations have been explored, as for structural testing of software systems.
  - Scenario description languages to explore/detect corner cases and high risk situations, exactly as for functional testing software systems.
  - Verdicts and diagnostics about the relationship between failures and various risk factors (road structure, congestion level, weather) and violations of traffic regulations.

When Self-driving Cars are Safe Enough? – Simulation Key Issues

- Whatever design approach is taken, simulation is of paramount importance for validation – and raises a large variety of problems from purely technical to theoretical ones.
- Not only the appearance should be realistic but also it should be real: the execution mechanism should rely on a semantic model of the environment consistent with laws of Geometry and Physics.
- Note that realism and consistency with reality are hard to reconcile - simulation environments built on top of game engines lack semantic awareness.

1. Realism: agent behavior and environment look real in a way that is accurate or true to life.
2. Modeling: expressive modeling language e.g. DSL for the component-based description of mobile agents and their dynamic coordination.
3. Semantic awareness: the simulated system dynamics is rooted in transition system semantics.
   - Notion of state allowing controllability and repeatability of experiments.
   - Notion of execution sequence distinguishing between controllable and uncontrollable actions
   - Multiscale multigrain modeling of time scales and of their correlation with space scales
4. Performance: run-time infrastructure federating simulation engines e.g. HLA, FMI

- Why is it so hard?
- Achieving Trustworthiness
  - Trustworthiness vs. Criticality
  - Trustworthy Autopilot Design
  - When Self-driving Cars are Safe Enough?
- Discussion

OVERVIEW

Discussion – Human Situation Awareness Cannot be Matched

To match human-level performance, systems should be able to deal with knowledge of the common sense world.

- Our mind is equipped with a semantic model of the world
  - used to Interpret sensory information and natural language in particular;
  - progressively built and automatically updated though learning and reasoning;
  - integrating in a huge network knowledge acquired along lifespan and involving concepts, cognition rules and patterns.

ANDY GREENBERG SECURITY 10.11.2020 04:00 PM

Split-Second 'Phantom' Images Can Fool Tesla's Autopilot

Researchers found they could stop a Tesla by flashing a few frames of a stop sign for less than half a second on an internet-connected billboard.

Discussion – Human Situation Awareness Cannot be Matched

To match human-level performance, systems should be able to deal with knowledge of the common sense world.

- ❑ Our mind is equipped with a semantic model of the world
  - used to Interpret sensory information and natural language in particular;
  - progressively built and automatically updated though learning and reasoning;
- integrating in a huge network knowledge acquired along lifespan and involving concepts, cognition rulepatterns.

- ❑ Human understanding combines: 1) bottom-up reasoning from sensor level to the semantic model of the mind; and 2) top-down from the semantic network to perception.

- ❑ It is highly improbable that we could ever build such semantic models given their overwhelming complexity - as evidenced by the very little progress in semantic analysis of natural languages so far.

*"**Intelligence** is what you use when you don't know what to do." Jean Piaget*

Discussion – Some Conclusions

- The trustworthy autonomous systems challenge is not only about intelligent agents, it involves equally important systems engineering issues.

- End-to-end monolithic AI-enabled solutions take the "brute force way" that precludes safety guarantees – Hybrid design could leverage on a solid body of knowledge for safe and efficient decision making and thus enhance confidence.

- Global system validation is achievable only through simulation and testing.
  - Realistic and semantically sound modeling becomes of paramount importance for validation
  - Any technically sound safety evaluation should be model-based e.g. relies on criteria defined on an implicit or an explicit system model.

- There is a big gap between automated and autonomous systems – the transition cannot be progressive: ADAS cannot gradually evolve into self-driving systems!!

- Supervising autonomous cars on autopilot (SAE Level 3) turns out to be a very hazardous idea – safe collaboration between autonomous systems and humans goes much deeper than classical HMI.

- To reach the vision we need to develop a new scientific and engineering foundation. And this will take some time.

Thank you