- Hasuo+, IEEE Trans. Intell. Vehicles, early access https://doi.org/10.1109/TIV.2022.3169762
- https://arxiv.org/abs/2207.02387

Goal-Aware RSS for Complex Scenarios via Program Logic

Ichiro Hasuo[1,7,*], Clovis Eberhart[1,8,*], James Haydon[1,*], Jérémy Dubut[1,8], Rose Bohrer[2], Tsutomu Kobayashi[1], Sasinee Pruekprasert[1], Xiao-Yi Zhang[1], Erik André Pallas[3], Akihisa Yamada[4,1], Kohei Suenaga[5,1], Fuyuki Ishikawa[1], Kenji Kamijo[6], Yoshiyuki Shinya[6], and Takamasa Suetomi[6]

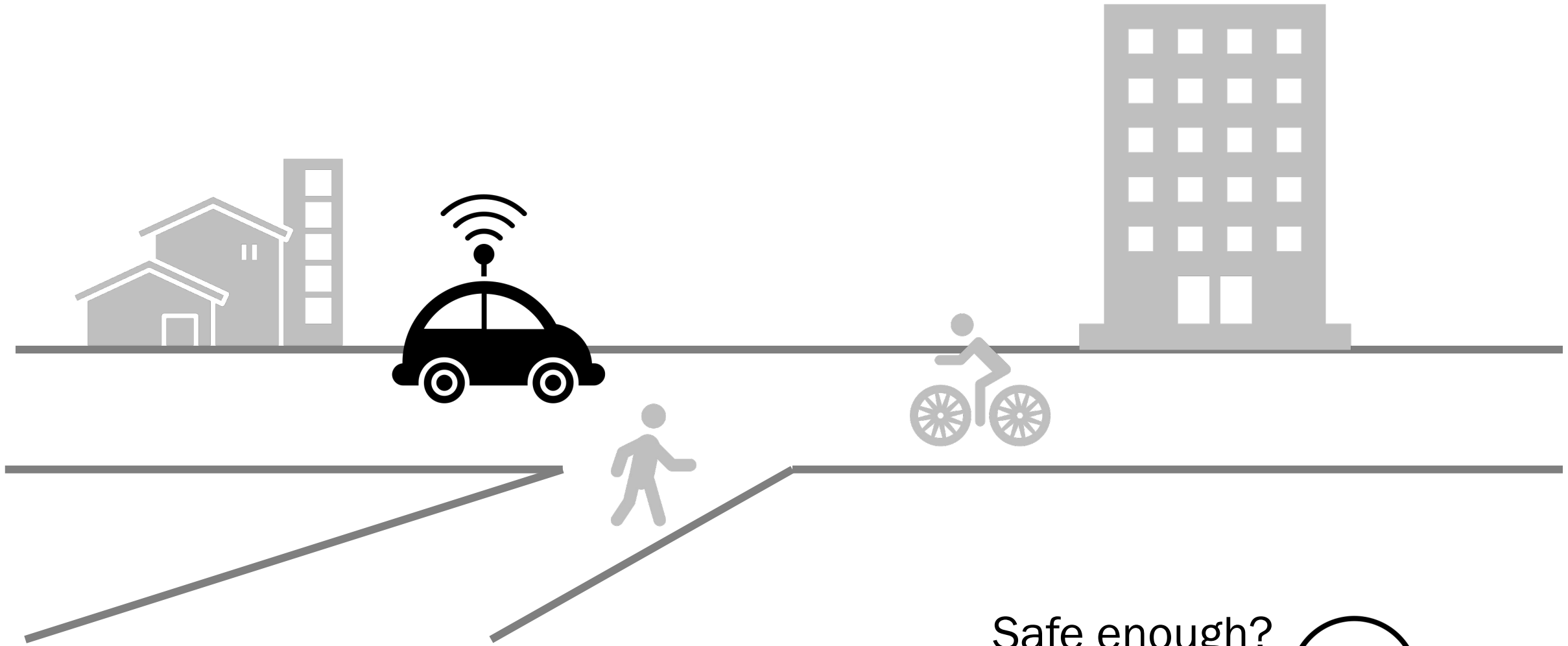
1: National Institute of Informatics, Tokyo, Japan
2: Worcester Polytechnique Institute, USA (work done at NII) 3: University of Augsburg, Germany (work done at NII)
4: AIST, Japan 5: Kyoto University, Japan 6: Mazda Motor Corporation, Japan
7: SOKENDAI (The Graduate University for Advanced Studies), Japan
8: Japanese-French Laboratory for Informatics (IRL 3527), Tokyo, Japan

Supported by ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603) and ACT-I (No. JPMJPR17UA), JST; and Grants-in-aid No. 19K20215 & 19K20249, JSPS.

*: equal contribution

Outline

- A non-technical overview
- The modeling problem
- The RSS answer to the modeling problem
- Technical contributions: the logic dFHL
- Perspectives, practical & theoretical

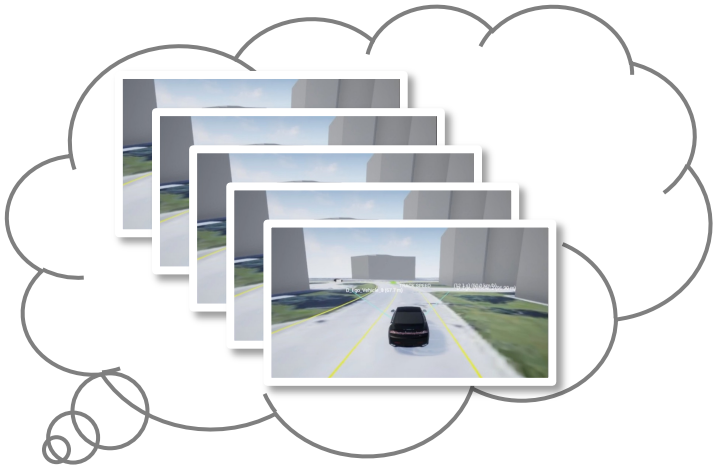


Safe enough?

Guarantee
by statistical data

9.1 crashes
per million miles
driven

Guarantee
by testing and simulation

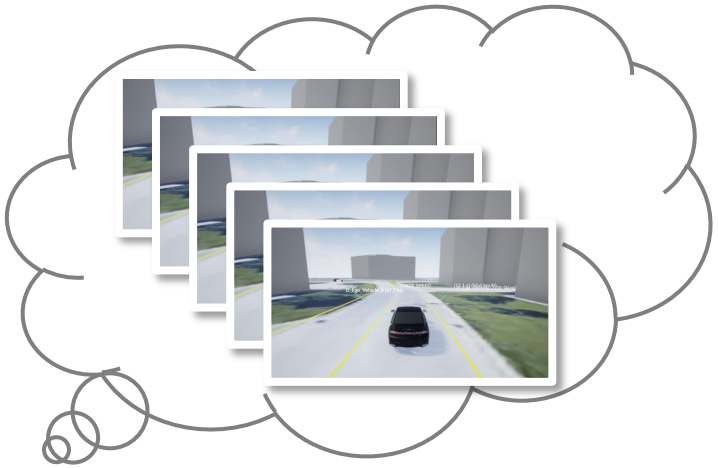


Guarantee strong enough?

Guarantee by statistical data

9.1 crashes per million miles driven

Guarantee by testing and simulation



Explainability?

Proof.

We prove the first statement. The rest is shown symm

Let $S \subseteq L$ be an arbitrary subset. We let $S^{\downarrow}$ be th
that is,

$$S^{\downarrow} \coloneqq \{y \in L \mid y \sqsubseteq s \text{ for each } s \in$$

Since $S^{\downarrow} \subseteq L$ is a subset of L, it has its supremum $\bigsqcup S^{\downarrow}$ in the
semilattice $(L, \sqsubseteq)$. We claim that $\bigsqcup S^{\downarrow}$ is the infimum

To prove the claim, it suffices to show the two-way
acterization in (2.1), that is, we need to show

$$\frac{y \sqsubseteq s \quad \text{for each } s \in S}{y \sqsubseteq \bigsqcup S^{\downarrow}}.$$

For the downward implication in **??**,

$$\begin{aligned} & y \sqsubseteq s \quad \text{for each } s \in S \\ \Longrightarrow \quad & y \in S^{\downarrow} && \text{by def. of } S^{\downarrow} \\ \Longrightarrow \quad & y \sqsubseteq \bigsqcup S^{\downarrow} && \text{since } \bigsqcup S^{\downarrow} \text{ is an u} \end{aligned}$$

For the upward implication in **??**, we first observe

$$\bigsqcup S^{\downarrow} \sqsubseteq s \quad \text{for each } s \in S.$$


12 : 00 PM
SPEED - 0 km/h
ENGINE STATUS - OFF
New Users -2.27%
Season +3.16%
Comparison of sessions
Views -4.1%
Unlock
Child lock
7.1 gal
1.3 gal
0.9 gal
84 psi
95%

Responsibility-Sensitive Safety (RSS)

[Shalev-Shwartz et al., arXiv preprint, 2017]

Lemma. (Conditional safety)
If all cars comply with **RSS rules**, then there is no collision

mathematically proved

+

Assumption. (Rule compliance)
All cars comply with **RSS rules**

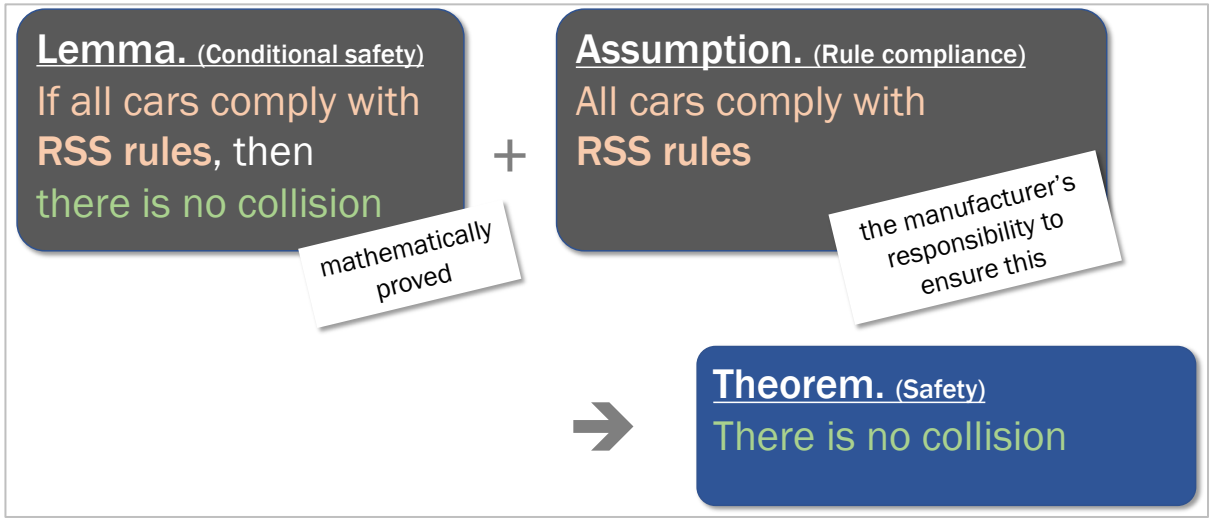
the manufacturer's responsibility to ensure this

→

Theorem. (Safety)
There is no collision

Responsibility-Sensitive Safety (RSS)

[Shalev-Shwartz et al., arXiv preprint, 2017]



- "Let's put all dirty details in an assumption"…
  Isn't this cheating?
  Isn't the assumption too big?

- → No!
  - RSS rules are rigorous,
    their compliance is verifiable by the third party
  - RSS rules can be enforced by the safety architecture (later)
  - Overall, RSS rules have the right granularity to impose as social contracts

- (Fresh view on proofs for us logicians…)

RSS Rule, an Example

[Shalev-Shwartz et al., arXiv preprint, 2017]

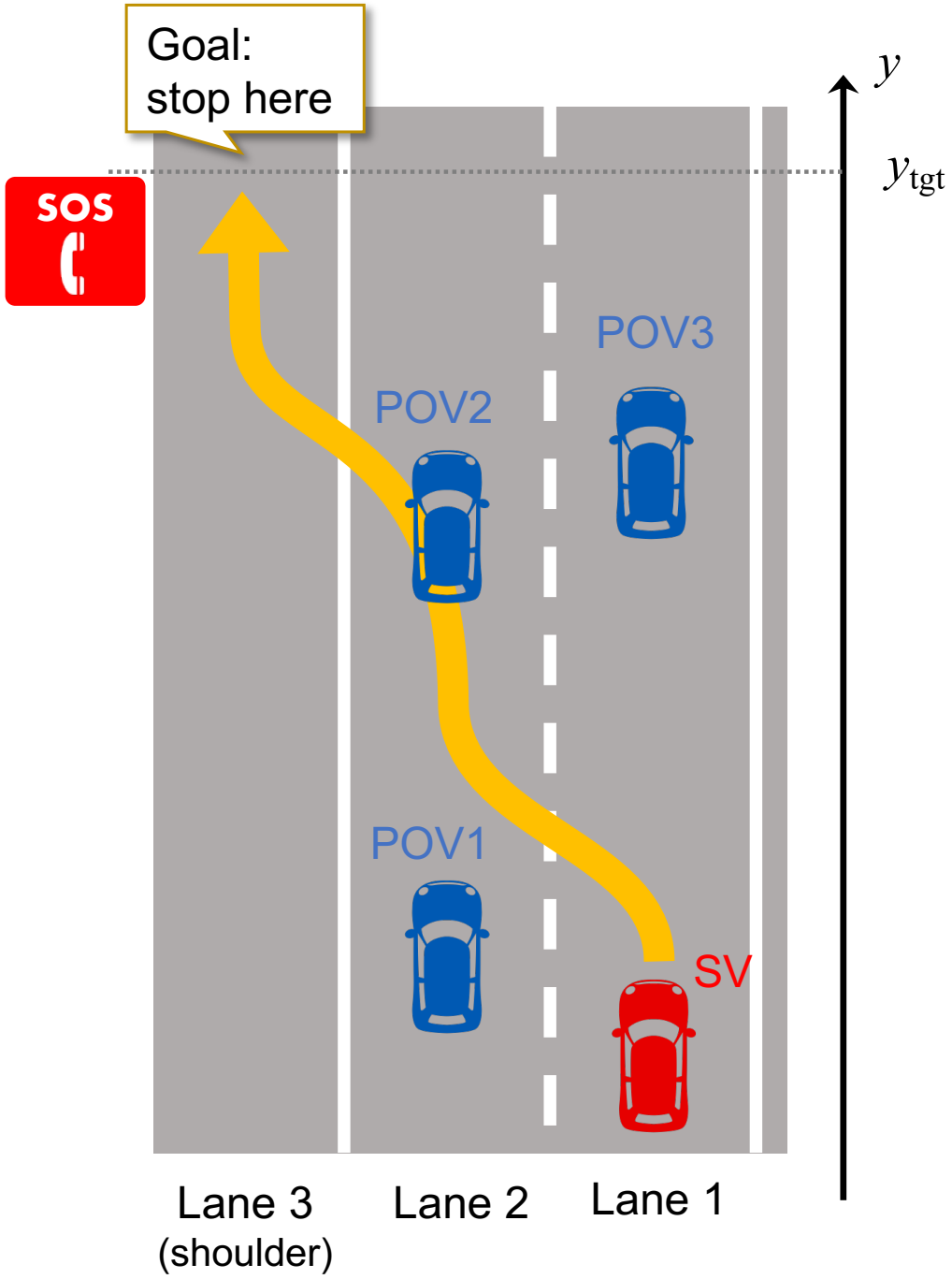
- An RSS rule is a pair (A, α) of an *RSS condition* A and a *proper response* α

RSS condition A:
Maintain an inter-vehicle distance at least

$$d_{\min} = \left[v_r\,\rho + \frac{1}{2} a_{\max,\text{accel}}\,\rho^2 + \frac{(v_r + \rho\, a_{\max,\text{accel}})^2}{2a_{\min,\text{brake}}} - \frac{v_f^2}{2a_{\max,\text{brake}}} \right]_+$$

Proper response α:
If A is about to be violated, brake at rate $a_{\text{min, brake}}$ within ρ seconds

Conditional safety lemma:
Any execution of α, from a state that satisfies A, is collision-free.



- Now what about this pull over scenario?
- Essential for eyes-off ADVs to hand the control over to human drivers
- Requires complex decision making
  - Merge before POV1, or after?
  - Accelerate to pass POV1…
    - ➔ Risk of overrun?

Our Contribution: Logical Formalization of RSS ➔ More Scenarios

RSS

Responsibility-Sensitive Safety,
Shalev-Shwartz et al., 2017

- Basic methodology of logical safety rules
- Standardization (IEEE 2846)
- Lack of formal implemantion
  ➔ <u>appl. to complex scenarios is hard</u>
- Guarantees only collision-freedom so far

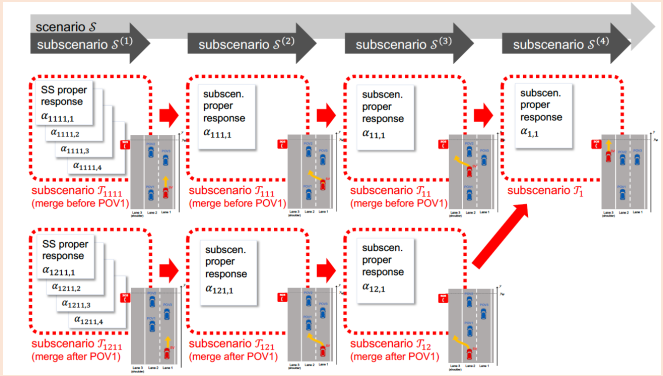
↓ Software science research

differential program logic dFHL (our contribution)

$$\frac{\begin{array}{ll} \text{inv}: & A \Rightarrow e_{\text{inv}} \sim 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{inv}} \simeq 0 \\ \text{var}: & A \Rightarrow e_{\text{var}} \geq 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{var}} \leq e_{\text{ter}} \\ \text{ter}: & A \Rightarrow e_{\text{ter}} < 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{ter}} \leq 0 \end{array}}{\{A\}\ \text{dwhile}\,(e_{\text{var}} > 0)\,\dot{\mathbf{x}} = \mathbf{f}\ \ \{e_{\text{var}} = 0 \wedge e_{\text{inv}} \sim 0\} : e_{\text{inv}} \sim 0 \wedge e_{\text{var}} \geq 0}\ (\text{DW}_{\text{H}})$$

- A logical system for deriving and proving safety rules

Compositional rule derivation workflow by dFHL (our contribution)

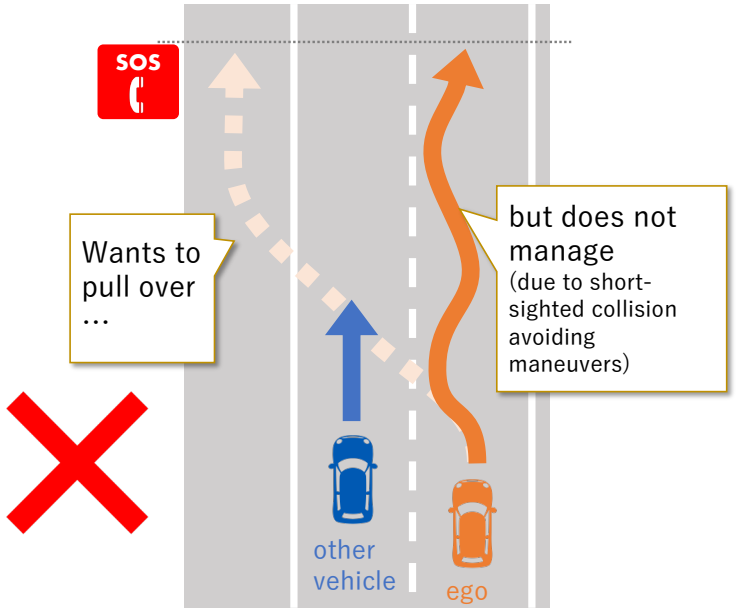


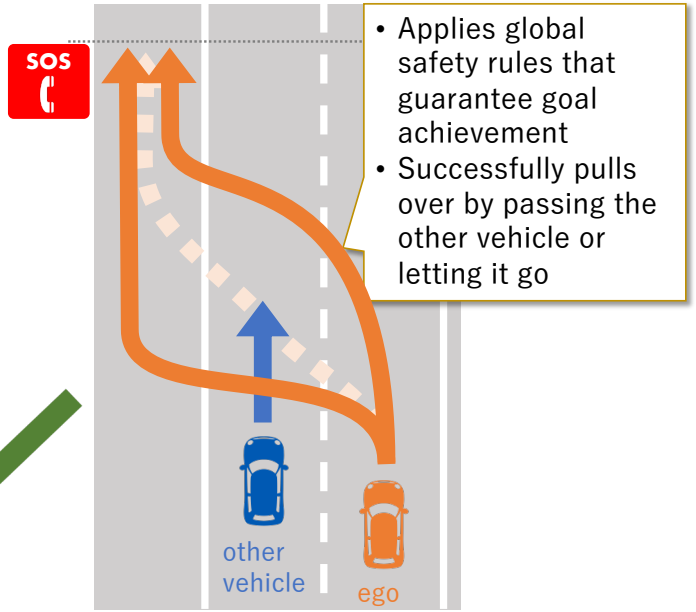
- ”Divide and Conquer” complex driving scenarios
- Tool support by autom. reasoning

GA-RSS (our contribution)

Goal-Aware
Responsibility-Sensitive Safety
[Hasuo+, IEEE T-IV, to appear]

- Guarantees <u>goal achievement</u> (e.g. successful pull over) and collision-freedom
- Global safety rules that combine mult. maneuvers
- Necessary for real-world complex driving scenarios





What is Formalization?

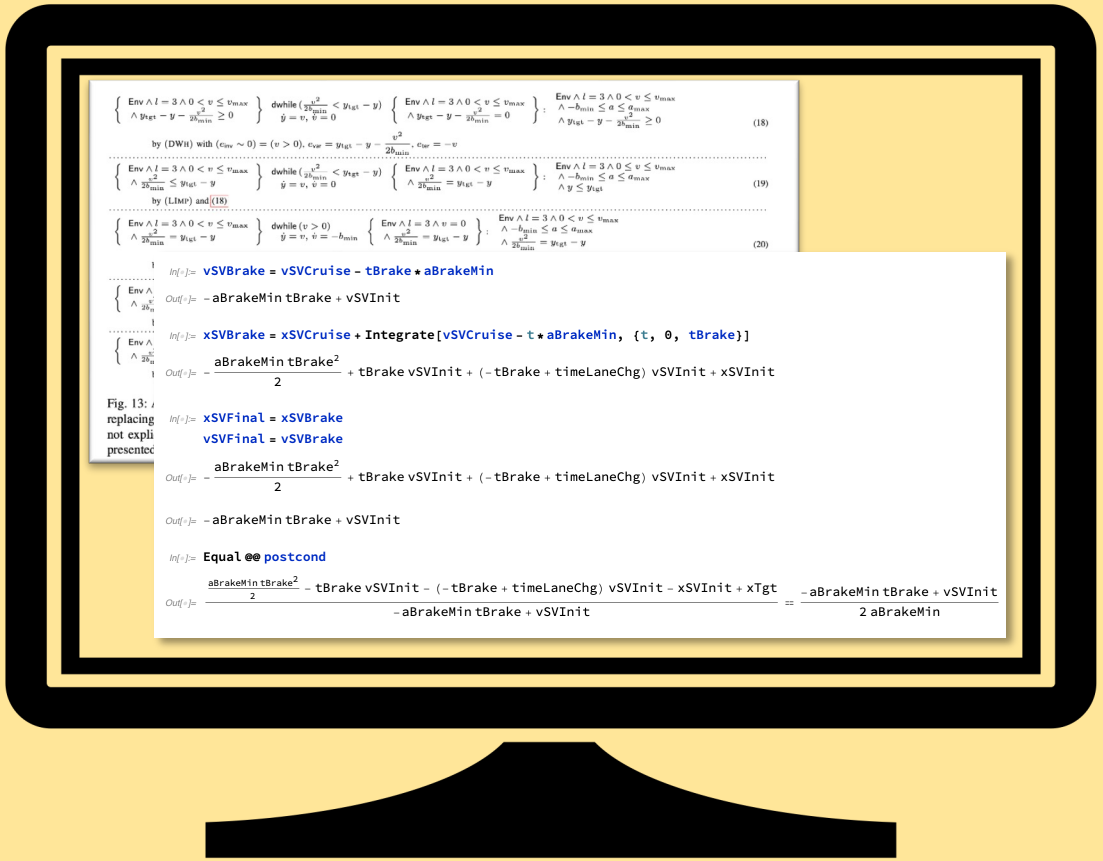
Informal

pen-and-paper proofs

- Error-prone
- Poor traceability

Formal

software-assisted proofs



- Symbolic proofs in our formal logical system dFHL
- Software tool checking the validity of each logical step of reasoning

Outline

- A non-technical overview
- The modeling problem
- The RSS answer to the modeling problem
- Technical contributions: the logic dFHL
- Perspectives, practical & theoretical

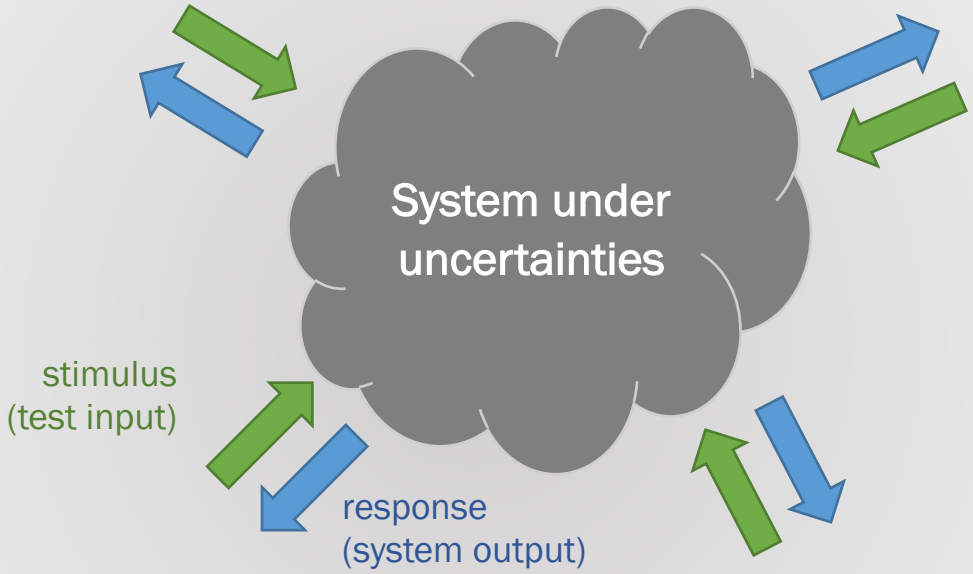
The Modeling Problem

- Theorems need *definitions*;
  formal verification needs *modeling*

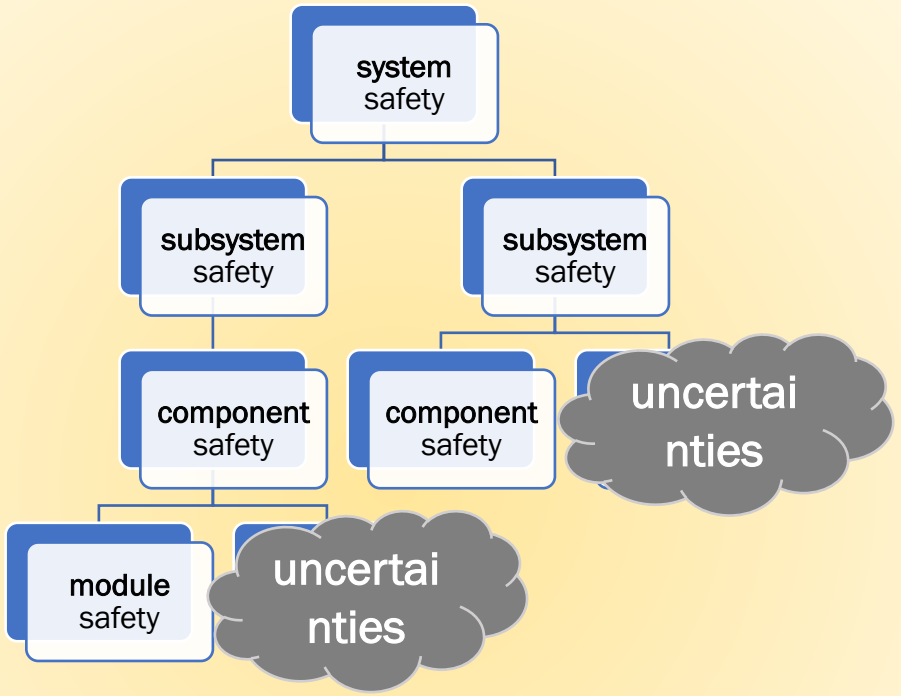
- Automated driving systems (ADS) are assively complex system
  - Hundreds of chips, millions of LoC
  - Physical components. Internal combustion
  - ML components, especially for perception
  - Unpredictable road conditions
  - Other cars
  - Pedestrians
  - …

- Modeling is hard (a grand challenge for us)

Logical Confinement of Uncertainties



- The whole system as a monolithic blackbox
- Analyzed by statistical and empirical means
- E.g. automated driving:
  "1 fatality per XXX miles driven"
  ➜ Doesn't exclude a scenario that is always fatal



- Logical argumentation of safety cases
- Impose rules/contracts on uncertain components
  ➜ runtime monitoring, accountability, identifying causes of accident
- Finding a good "logical angle" is crucial, which takes theoretical insights and experience

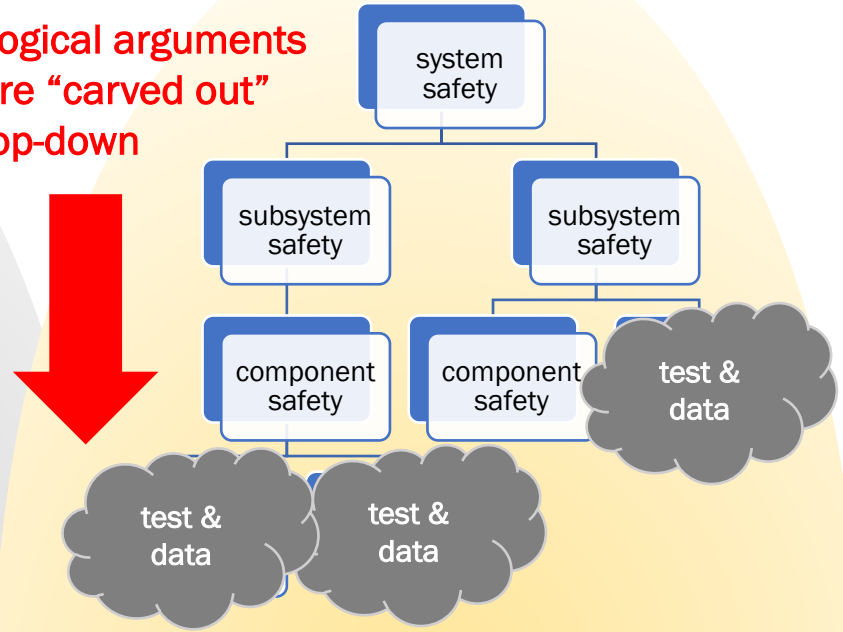
Purely data-driven approach to safety assurance

e.g. “one derailment every 10,000 miles” in automated driving

✓ Scalability, automation by efficient processing of big data

✗ Accountability. Hard to convince the customer/public of safety, or that duties of care have been fulfilled

Logical arguments are “carved out” top-down



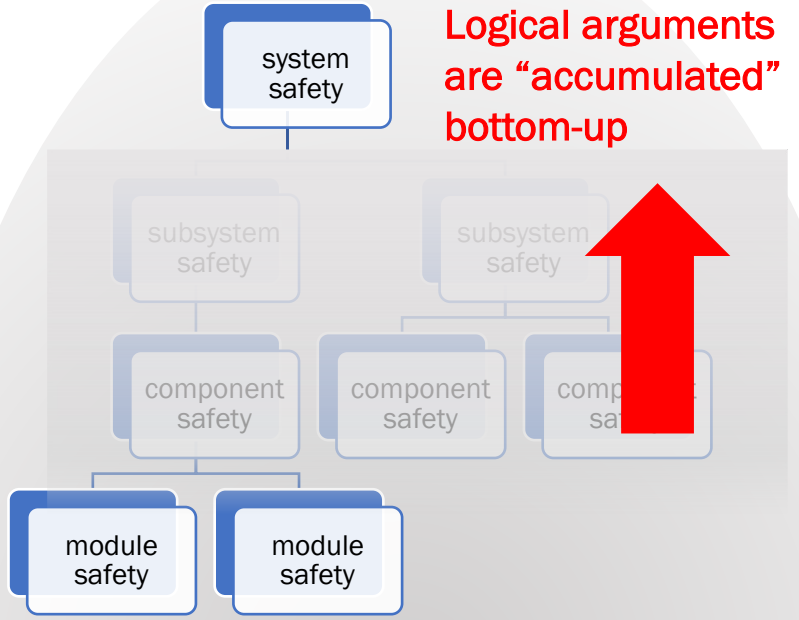
(Our approach) Logical confinement of uncertainties

- Start from the conclusion (system safety), and carve out logical arguments that lead to it
- Use test & data once the limit of logical arguments is reached

✓ Best-effort logical guarantee
Smaller resources/efforts yield non-zero assurance (if smaller)

✓ Explainability by logic.
Crucial for public acceptance of new ICT paradigms (such as automated driving)

Logical arguments are “accumulated” bottom-up



Purely logical approach to safety assurance

- Formal verification, a software science tradition
- Start with mathematical modeling of the target system, and build up logical consequences

✓ Traceability. Accountability. Trust.
Every deduction step is explicit and rule-based.

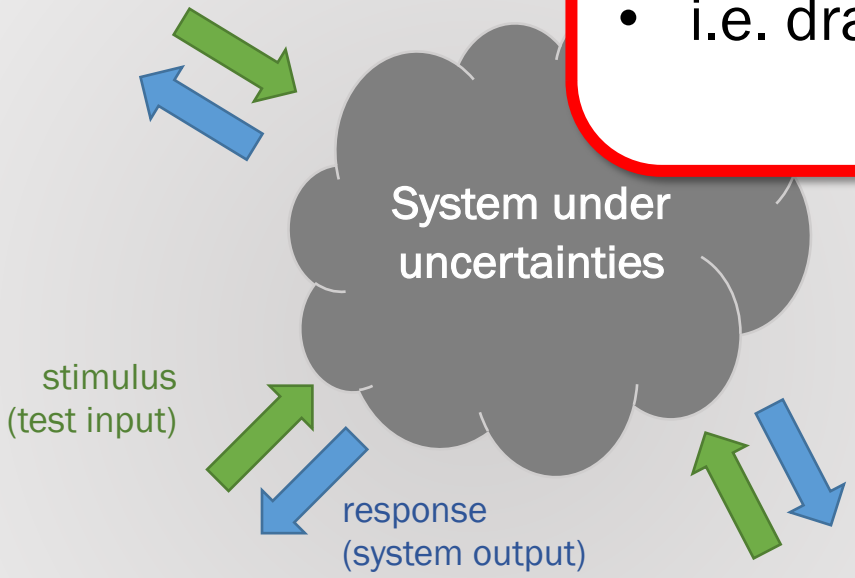
✗ Complexity of modern ICT systems
→ Bottom-up efforts might never reach the final goal (namely the system safety)

✗ Moreover, an incomplete proof is totally useless. Huge cost until a non-zero value is produced

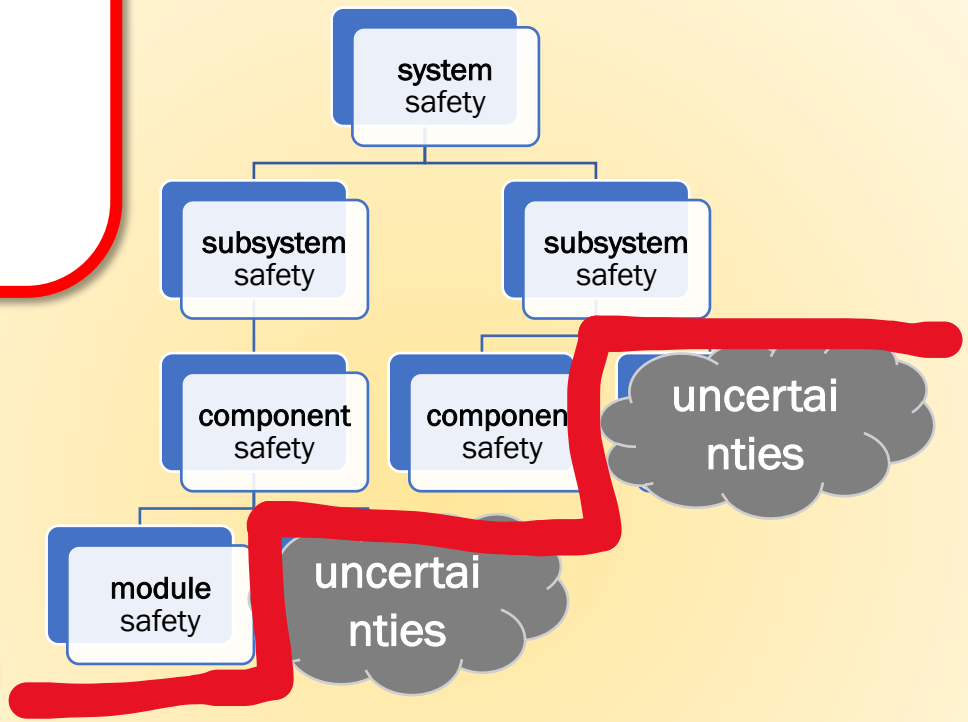
Logical Con inties

The modeling problem:

- Deciding what to model and what not to model
- i.e. drawing a good border



- The whole system as a monolithic blackbox
- Analyzed by statistical and empirical means
- E.g. automated driving:
  "1 fatality per XXX miles driven"
  ➜ Doesn't exclude a scenario that is always fatal



- Logical argumentation of safety cases
- Impose rules/contracts on uncertain components
  ➜ runtime monitoring, accountability, identifying causes of accident
- Finding a good "logical angle" is crucial, which takes theoretical insights and experience

Outline

- A non-technical overview
- The modeling problem
- The RSS answer to the modeling problem
- Technical contributions: the logic dFHL
- Perspectives, practical & theoretical

Logical Con inties

The modeling problem:

- Deciding what to model and what not to model
- i.e. drawing a good border

System under

RSS's answer

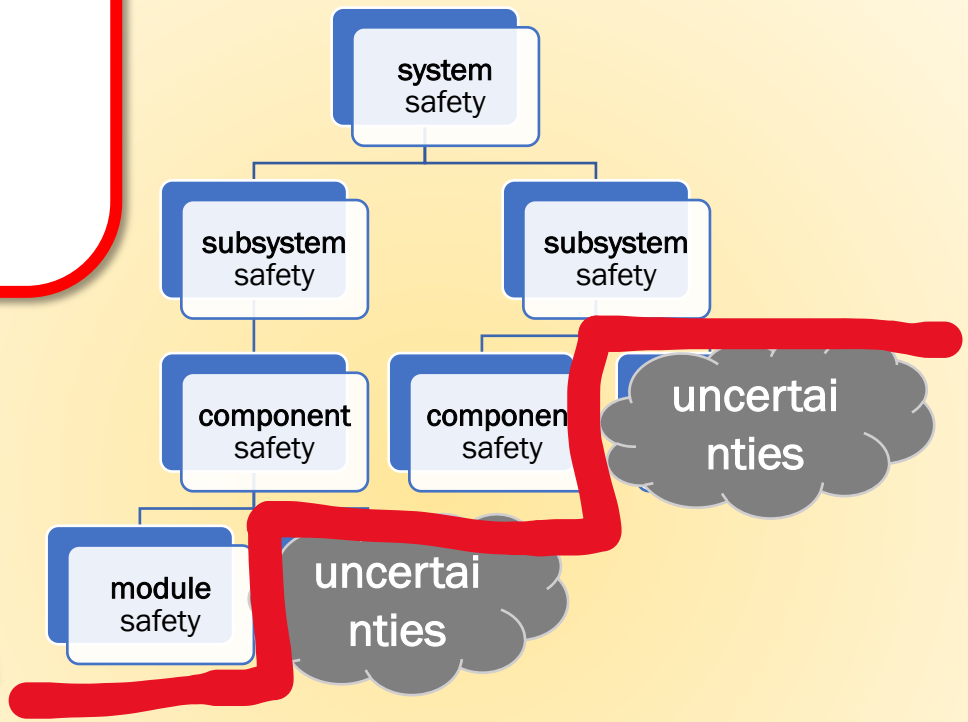
Responsibility-Sensitive Safety (RSS)
[Shalev-Shwartz et al., arXiv preprint, 2017]

Lemma. If all cars comply with **RSS rules**, then there is no collision
(mathematically proved)

\+

Lemma. All cars comply with **RSS rules**
(guaranteed by the manufacturer (testing, proofs, etc.))

→ **Theorem.** There is no collision



- Logical argumentation of safety cases
- Impose rules/contracts on uncertain components
  → runtime monitoring, accountability, identifying causes of accident
- Finding a good "logical angle" is crucial, which takes theoretical insights and experience

Safety Guarantee for Automated Driving via Logical Safety Rules and Mathematical Proofs

- Decompose **safety** (a complex goal) into **logical safety rules** (explicit, easy to check and enforce)
- "Ultimate assurance" in the form of **mathematical proofs**. Logical explanation by following their reasoning steps
- Safety rules are generic and reusable
  → regulation, standard → social acceptance
- Attribution of liabilities
  (collision → someone must have broken the rules)

Regulation/Standardization Body

R_1
R_2
R_3

Safety Rule R_1

In the *same-lane same-direction* driving scenario,

- Maintain the safety distance

$$d_{\min} = \left[v_r\,\rho + \frac{1}{2} a_{\max,\mathrm{accel}}\,\rho^2 + \frac{(v_r + \rho\, a_{\max,\mathrm{accel}})^2}{2a_{\min,\mathrm{brake}}} - \frac{v_f^2}{2a_{\max,\mathrm{brake}}} \right]_+$$

  from the preceding car
- When that's hard, brake at acceleration $a_{\max,\mathrm{brake}}$

Theorem (Safety)

There is no collision attributed to the ego vehicle as long as the safety rule R_1 is respected

Proof (of the safety thm.)

⋮

The only non-obvious point is that $e_{\mathsf{inv},2}$ is preserved by the dynamics. We first observe

$$\mathcal{L}_{\delta_f,\delta_r^1} e_{\mathsf{inv},2} = \begin{cases} 0 & \text{if } \mathsf{dRSS}_{\pm}(v_f, v_r, \rho - t) \geq 0 \\ v_f - v_r & \text{otherwise,} \end{cases}$$

where $\mathsf{dRSS}_{\pm}(v_f, v_r, \rho)$ is given by

$$\mathsf{dRSS}_{\pm}(v_f, v_r, \rho) = v_r\rho + \frac{a_{\max}\rho^2}{2} + \frac{(v_r + a_{\max}\rho)^2}{2b_{\min}} - \frac{v_f^2}{2b_{\max}}.$$

Therefore, we can infer as follows.

$$\mathsf{dRSS}_{\pm}(v_f, v_r, \rho - t) < 0$$
$$\iff v_r(\rho - t) + \frac{a_{\max}(\rho - t)^2}{2} + \frac{(v_r + a_{\max}(\rho - t))^2}{2b_{\min}} - \frac{v_f^2}{2b_{\max}} < 0$$

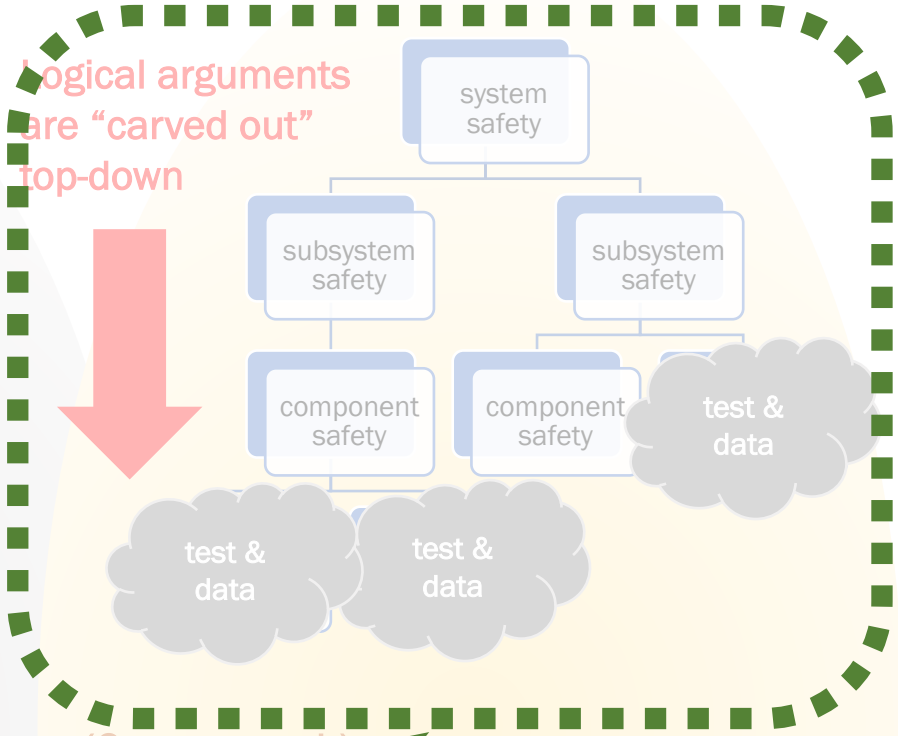
⋮

Purely data-driven approach to safety assurance

e.g. “one derailment every 10,000 miles” in automated driving

✓ Scalability, automation by efficient processing of big data

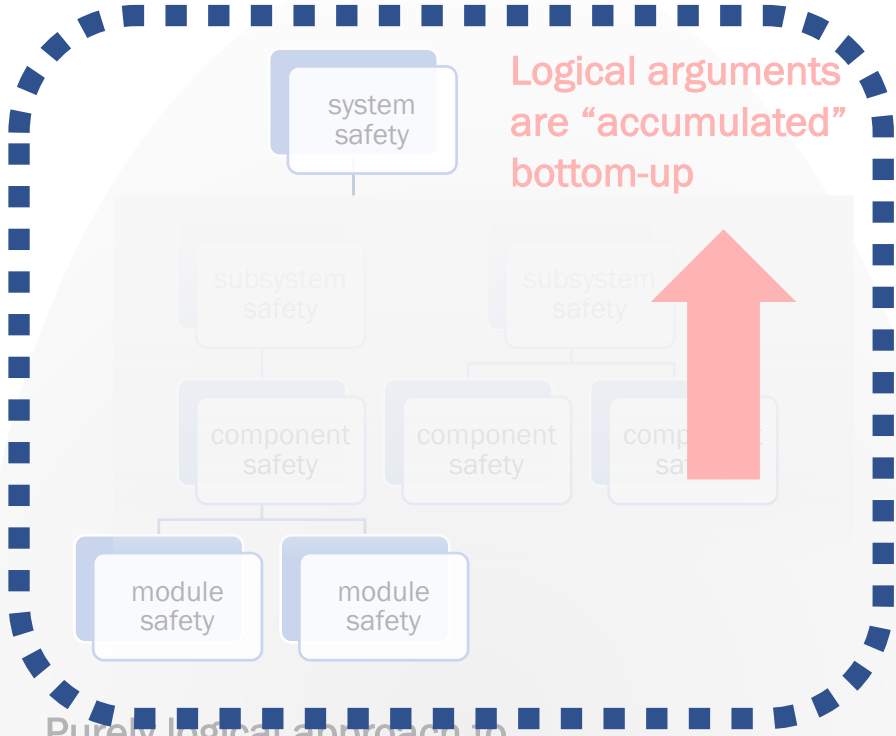
✗ Accountability. Hard to convince the customer/public of safety, or that duties of care have been fulfilled



(Our approach)

Logical safety rules for ADS

- Safety “theorems” are reduced to “axioms” (namely safety rules)
- The reduction is math. rigorous
- Rule compliance can be logically verified, but also be tested or monitored
- Not a full safety proof, but feasible. Enough for many usages



Purely logical approach to safety ass

(Full) formal verif. of ADS safety

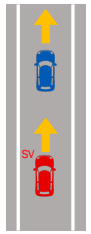
- Strong mathematical guarantee… if the proof is completed
- and completion is very unlikely (lack of models, budget limit, uncertainties in driving scenarios, etc.)
- (Full verif. is more viable in aerospace. No pedestrians, more budget)

RSS Framework

[Shalev-Shwartz et al., arXiv, 2017]
See also [Hasuo, arXiv 2206.03418]

Each rule consists of
a **condition** and a **proper response**

Scenario S_1



RSS safety rule
$R_1 = (C_1, P_1)$

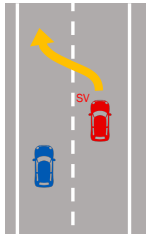
RSS safety condition C_1:
Distance is no smaller than
$\max\left(0, v_r\rho + \frac{1}{2}a_{\max}\rho^2 + \frac{(v_r + a_{\max}\rho)^2}{2a_{\text{brake,min}}} - \frac{v_f^2}{2a_{\text{brake,max}}}\right)$

Proper response P_1:
Full (non-emergency) braking

Theorem (safety).
No collision, if P_1 is engaged at a state satisfying C_1

Theorem (responsibility).
P_1, engaged at a state satisfying C_1, respects the RSS responsibility principles

Scenario S_2



RSS safety rule
$R_2 = (C_2, P_2)$

RSS safety condition C_2:
Distance is no smaller than …

Proper response P_2:
Cut in

Theorem (safety).
No collision, if P_2 is engaged at a state satisfying C_2

Theorem (responsibility).
P_2, engaged at a state satisfying C_2, respects the RSS responsibility principles

. . .

Usages
- Attribution of liability
- Safety metric
- Formal safety verification
- Safety architecture
- …

assume

guarantee

RSS responsibility principles
1. Don't hit the car in front of you
2. Don't cut in recklessly
3. Right of way is given, not taken
4. Be cautious in areas with limited visibility
5. If you can avoid a crash without causing another one, you must

Outline

- A non-technical overview
- The modeling problem
- The RSS answer to the modeling problem
- Technical contributions: the logic dFHL
- Perspectives, practical & theoretical

Our Contribution: Formal Logic Foundations of RSS ➜ More Scenarios

RSS

Responsibility-Sensitive Safety,
Shalev-Shwartz et al., 2017

- Basic methodology of logical safety rules
- Standardization (IEEE 2846)
- Lack of formal implemantion
  ➜ <u>appl. to complex scenarios is hard</u>
- Guarantees only collision-freedom so far

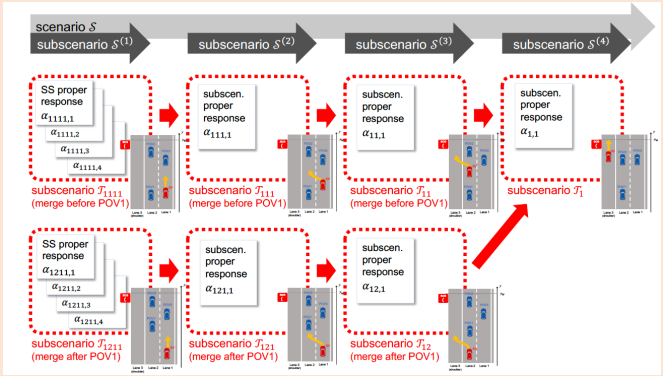
↓ Software science research

differential program logic dFHL (our contribution)

$$\frac{\begin{array}{ll}\text{inv}: & A \Rightarrow e_{\text{inv}} \sim 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{inv}} \simeq 0 \\ \text{var}: & A \Rightarrow e_{\text{var}} \geq 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{var}} \leq e_{\text{ter}} \\ \text{ter}: & A \Rightarrow e_{\text{ter}} < 0 \quad e_{\text{var}} \geq 0 \wedge e_{\text{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\text{ter}} \leq 0 \end{array}}{\{A\}\ \text{dwhile}\,(e_{\text{var}} > 0)\,\dot{\mathbf{x}} = \mathbf{f}\ \ \{e_{\text{var}} = 0 \wedge e_{\text{inv}} \sim 0\} : e_{\text{inv}} \sim 0 \wedge e_{\text{var}} \geq 0}\ (\text{DW}_{\text{H}})$$

- A logical system for deriving and proving safety rules

Compositional rule derivation workflow by dFHL (our contribution)

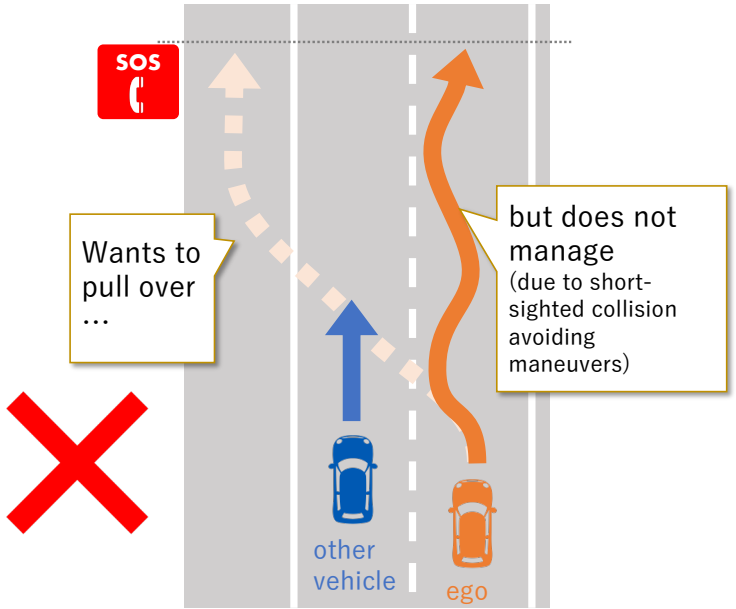


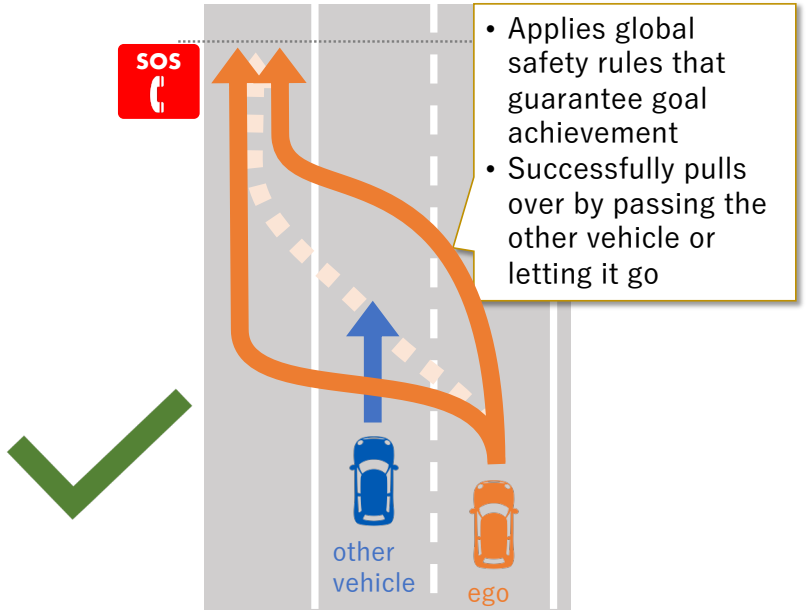
- ”Divide and Conquer” complex driving scenarios
- Tool support by autom. reasoning

GA-RSS (our contribution)

Goal-Aware
Responsibility-Sensitive Safety

- Guarantees <u>goal achievement</u> (e.g. successful pull over) and collision-freedom
- Global safety rules that combine mult. maneuvers
- Necessary for real-world complex driving scenarios





Differential program logic dFHL

- Hoare logic
  + ODEs (dwhile)
  + "safety condition"

$$\{A\}\ \alpha\ \{B\} : S$$

postcondition ↑ (true at the end of α)

"safety condition" ↑ (true throughout α)

- Reasoning about ODEs via differential invariants (barrier cert.) and ranking/Lyapunov functions
- Theoretically not so much different from Platzer's dL.
  Simplified, aiding proof engineers

Def. (dFHL programs)

$$\alpha, \beta \quad ::= \quad \mathsf{skip} \mid \alpha; \beta \mid x := e \mid \mathsf{if}\,(A)\,\alpha\,\mathsf{else}\,\beta \mid \mathsf{while}\,(A)\,\alpha \mid \mathsf{dwhile}\,(A)\,\{\,\dot{\mathbf{x}} = \mathbf{f}\,\}\,.$$

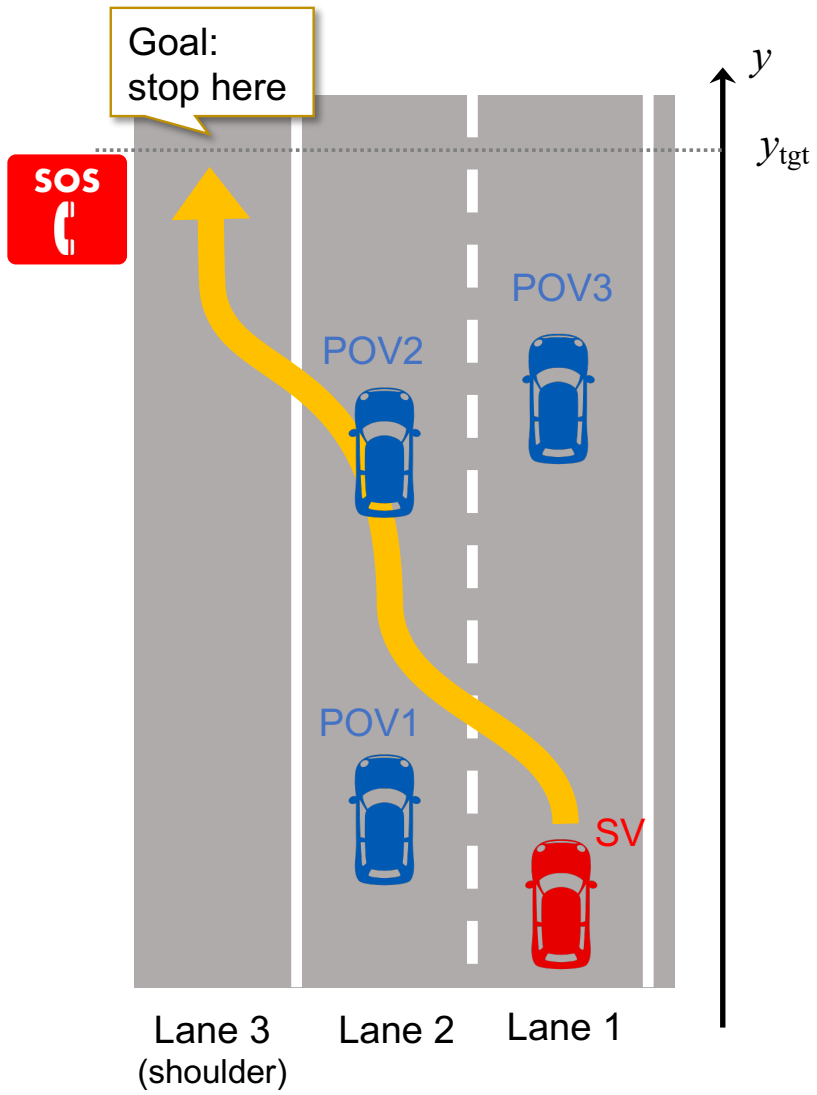
Def. (dFHL rules)

$$\frac{\{A\}\ \alpha\ \{B\} : S \qquad \{B\}\ \beta\ \{C\} : S}{\{A\}\ \alpha;\beta\ \{C\} : S}\ (\textsc{Seq})$$

$$\frac{\{A'\}\ \alpha\ \{B'\} : S' \qquad \begin{array}{c} A \Rightarrow A' \\ S' \wedge B' \Rightarrow B \\ S' \Rightarrow S \end{array}}{\{A\}\ \alpha\ \{B\} : S}\ (\textsc{Limp})$$

$$\frac{\begin{array}{ll} \mathsf{inv}: & A \Rightarrow e_{\mathsf{inv}} \sim 0 \quad e_{\mathsf{var}} \geq 0 \wedge e_{\mathsf{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\mathsf{inv}} \simeq 0 \\ \mathsf{var}: & A \Rightarrow e_{\mathsf{var}} \geq 0 \quad e_{\mathsf{var}} \geq 0 \wedge e_{\mathsf{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\mathsf{var}} \leq e_{\mathsf{ter}} \\ \mathsf{ter}: & A \Rightarrow e_{\mathsf{ter}} < 0 \quad e_{\mathsf{var}} \geq 0 \wedge e_{\mathsf{inv}} \sim 0 \Rightarrow \mathcal{L}_{\dot{\mathbf{x}}=\mathbf{f}}\, e_{\mathsf{ter}} \leq 0 \end{array}}{\{A\}\ \mathsf{dwhile}\,(e_{\mathsf{var}} > 0)\,\dot{\mathbf{x}} = \mathbf{f}\ \{e_{\mathsf{var}} = 0 \wedge e_{\mathsf{inv}} \sim 0\} : e_{\mathsf{inv}} \sim 0 \wedge e_{\mathsf{var}} \geq 0}\ (\textsc{DWh})^{\dagger}$$

Compositional Rule Derivation



- We shall derive

$$\{A\}\ \alpha\ \{B\} : S$$

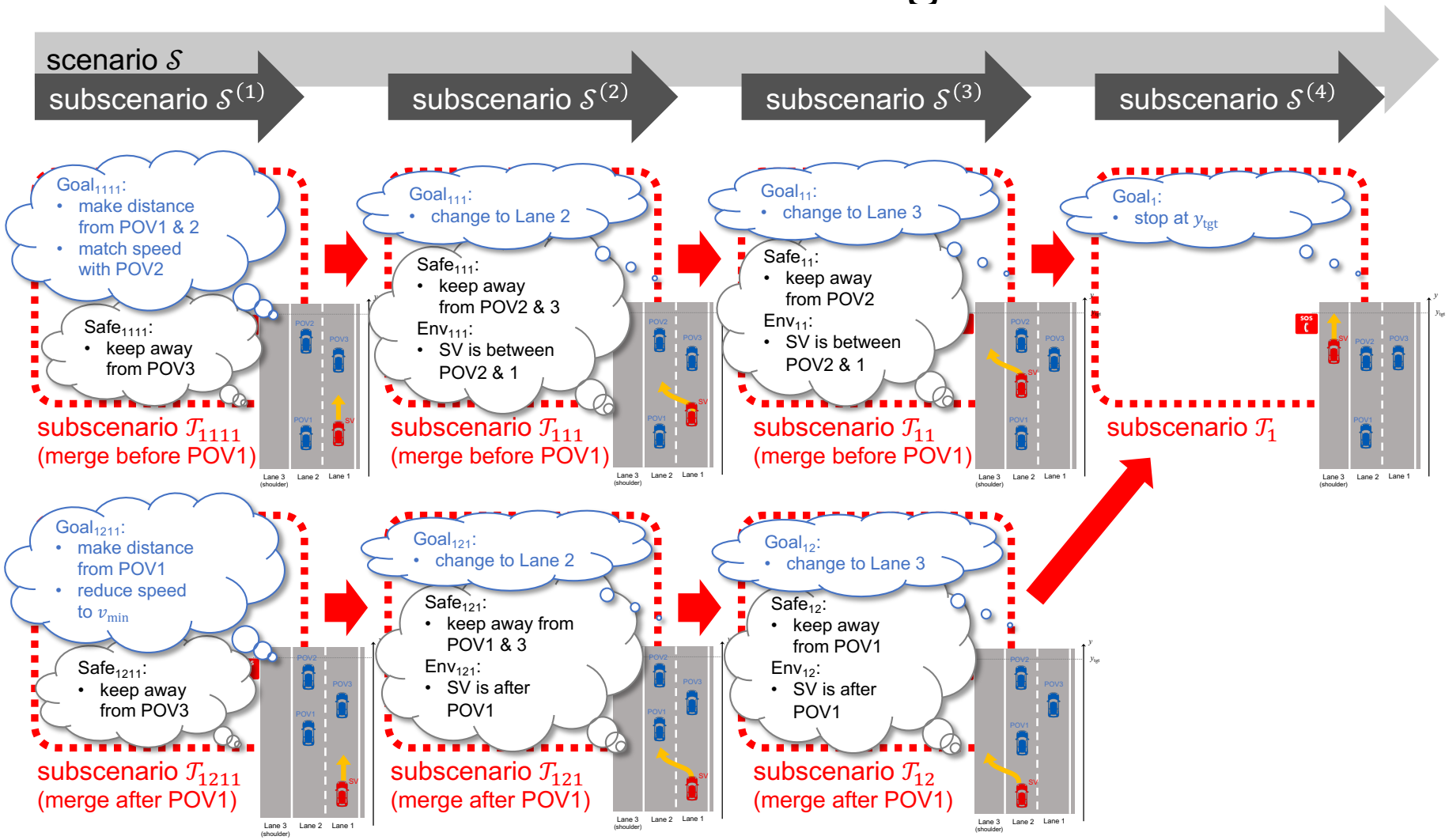
for the following given data

- $\boldsymbol{B}$ is the **goal**: "stoping on the shoulder at $\boldsymbol{y}_{\text{tgt}}$"
- $\boldsymbol{S}$ is the **safety**: "no collision," or better "securing RSS distance from every other car"

- We shall identify
  - α as an **RSS proper response**: "executing α will safely achieve the goal"
  - A as an **RSS condition**: "when A is true, B and S are guaranteed by executing α"

Compositional Rule Derivation

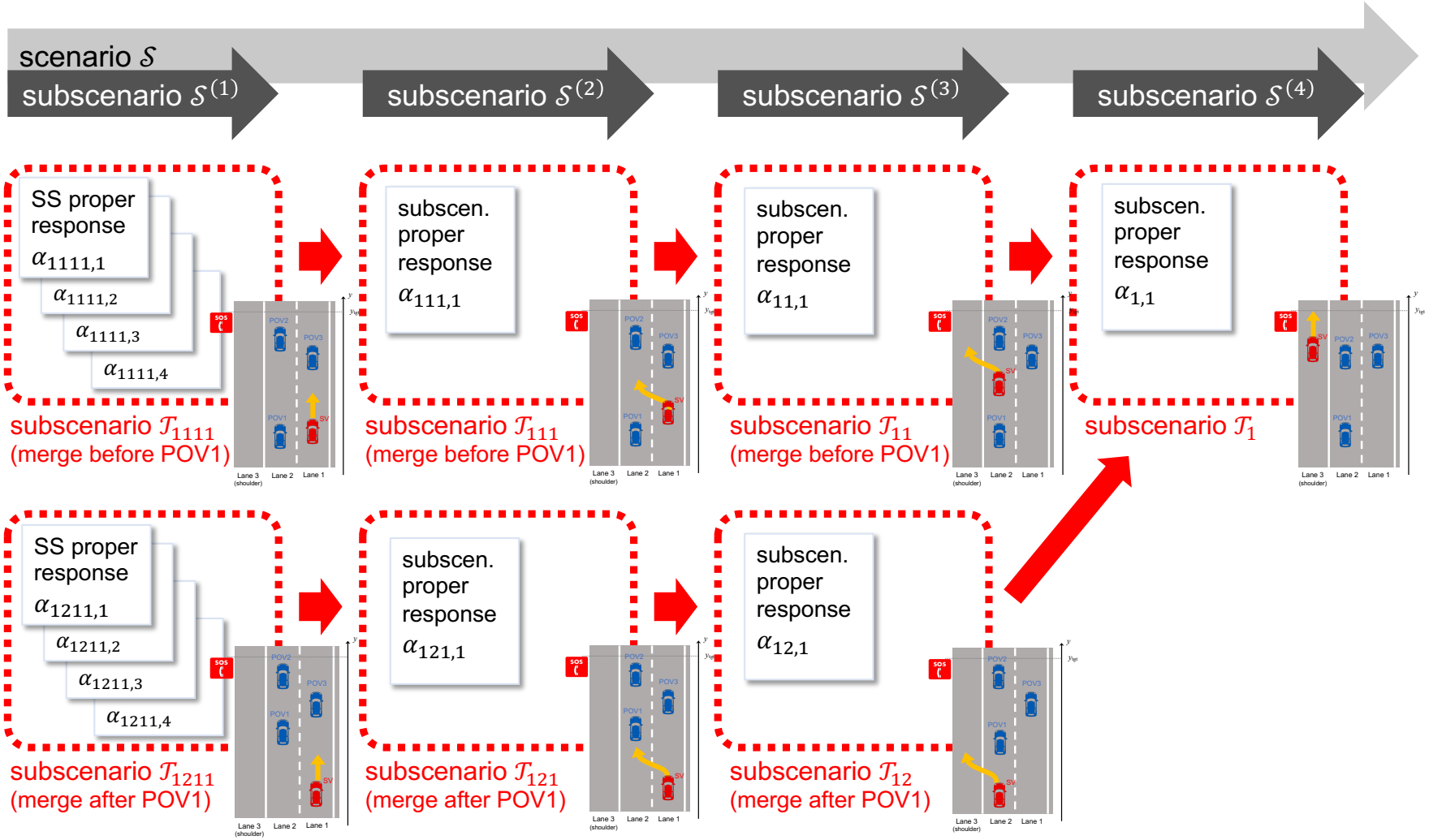
(1) Decompose the scenario into subscenarios, each of which has clearer focuses and goals



Compositional Rule Derivation

(2) Devise subscenario proper responses for each subscenario

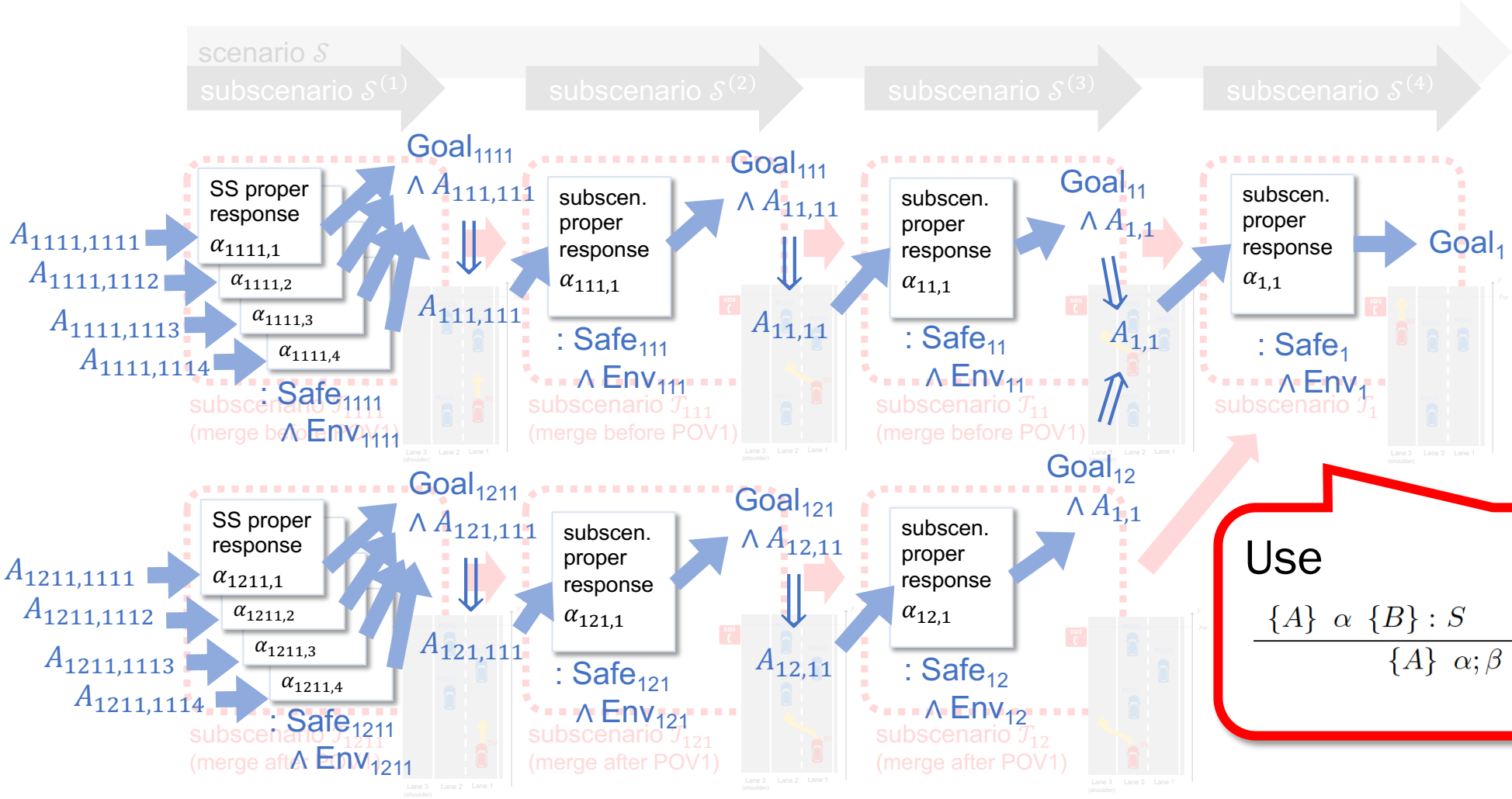
$$\{A\}\ \alpha\ \{B\} : S$$



Compositional Rule Derivation

(3) Backpropagate pre/postconditions,
leading to the scenario-wide precondition

$$\{A\}\ \alpha\ \{B\} : S$$



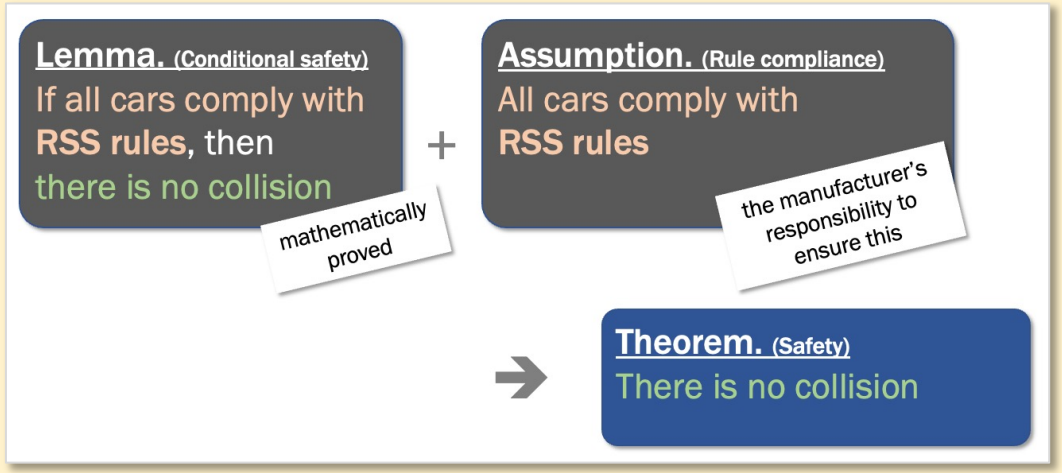
Use

$$\frac{\{A\}\ \alpha\ \{B\} : S \qquad \{B\}\ \beta\ \{C\} : S}{\{A\}\ \alpha;\beta\ \{C\} : S}\ (\textsc{Seq})$$

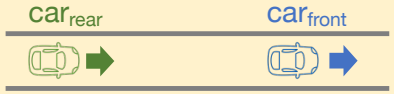
Outline

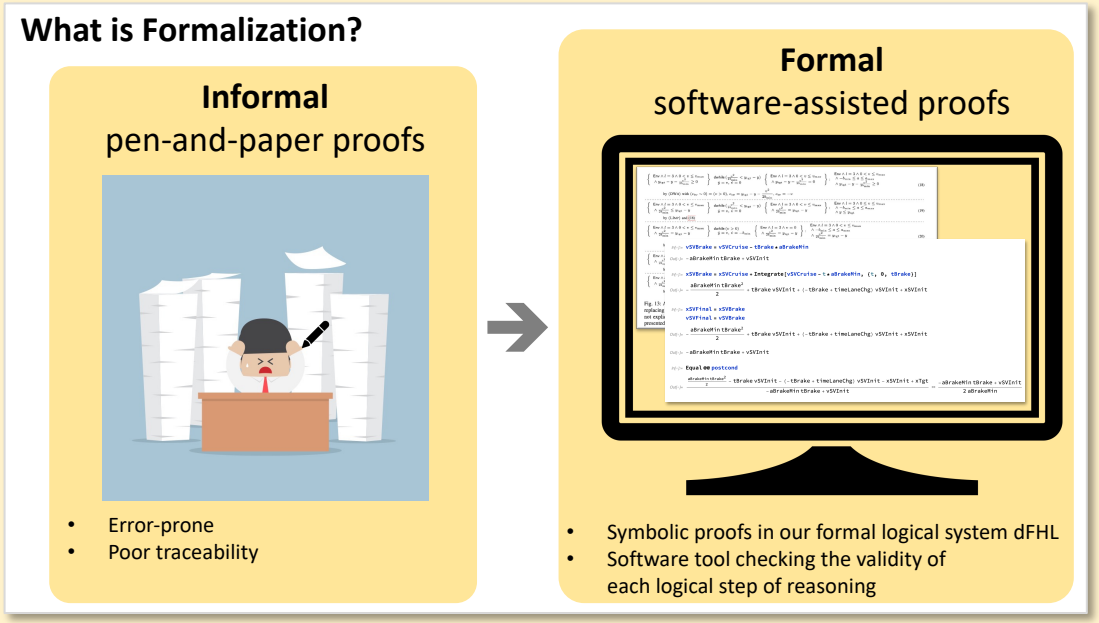
- A non-technical overview
- The modeling problem
- The RSS answer to the modeling problem
- Technical contributions: the logic dFHL
- Perspectives, practical & theoretical

Logical Formalization of RSS
Covering More Scenarios → Real-World Deployment

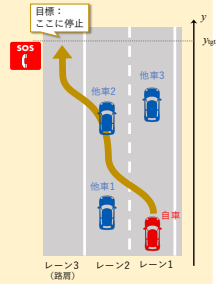


- RSS as in [Shalev-Shwartz et al., arXiv, 2017] is a **methodology**–
  it is sensible and promising,
  but came with no proof technologies
- thus application was limited to
  simple driving scenarios





- Our contribution
  [Hasuo+, IEEE T-IV, to appear]:
  Logical technologies to prove
  conditional safety lemmas for complex scenarios
- Compositional proofs,
  ensuring goal achievements, …
- Much more scenarios proved safety by RSS
  → RSS at work → social acceptance of ADV



Safety Guarantee for Automated Driving via Logical Safety Rules and Mathematical Proofs

- Decompose **safety** (a complex goal) into **logical safety rules** (explicit, easy to check and enforce)
- "Ultimate assurance" in the form of **mathematical proofs**. Logical explanation by following their reasoning steps
- Safety rules are generic and reusable
  → regulation, standard → social acceptance
- Attribution of liabilities
  (collision → someone must have broken the rules)

Regulation/Standardization Body

R_1
R_2
R_3

Safety Rule R_1

In the *same-lane same-direction* driving scenario,

- Maintain the safety distance

$$d_{\min} = \left[v_r\,\rho + \frac{1}{2} a_{\max,\mathrm{accel}}\,\rho^2 + \frac{(v_r + \rho\, a_{\max,\mathrm{accel}})^2}{2a_{\min,\mathrm{brake}}} - \frac{v_f^2}{2a_{\max,\mathrm{brake}}} \right]_+$$

  from the preceding car
- When that's hard, brake at acceleration $a_{\max,\mathrm{brake}}$

Theorem (Safety)

There is no collision attributed to the ego vehicle as long as the safety rule R_1 is respected

Proof (of the safety thm.)

⋮

The only non-obvious point is that $e_{\mathsf{inv},2}$ is preserved by the dynamics. We first observe

$$\mathcal{L}_{\delta_f,\delta_r^1} e_{\mathsf{inv},2} = \begin{cases} 0 & \text{if } \mathsf{dRSS}_{\pm}(v_f, v_r, \rho - t) \geq 0 \\ v_f - v_r & \text{otherwise,} \end{cases}$$

where $\mathsf{dRSS}_{\pm}(v_f, v_r, \rho)$ is given by

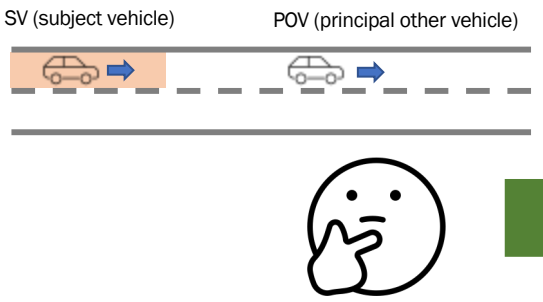
$$\mathsf{dRSS}_{\pm}(v_f, v_r, \rho) = v_r\rho + \frac{a_{\max}\rho^2}{2} + \frac{(v_r + a_{\max}\rho)^2}{2b_{\min}} - \frac{v_f^2}{2b_{\max}}.$$

Therefore, we can infer as follows.

$$\mathsf{dRSS}_{\pm}(v_f, v_r, \rho - t) < 0$$
$$\iff v_r(\rho - t) + \frac{a_{\max}(\rho - t)^2}{2} + \frac{(v_r + a_{\max}(\rho - t))^2}{2b_{\min}} - \frac{v_f^2}{2b_{\max}} < 0$$

⋮

RSS Rules as Social Contracts Impacts Everywhere in the ADV Ecosystem



Want an RSS rule C …
s.t. compliance with C guarantees collision-freedom

A **proof engineer** works on each driving scenario and derives an RSS rule C

- Correctness proof for C, at the same time

RSS rule C

+

Safety proof for C

Checkable by the public
…
proof-search is hard, proof-check is easy

- **Attribution of liabilities**
  (collision ➔ ∃ a party who did not comply)
- **Safety standards**
  (Show compliance before selling cars)
  e.g. IEEE P2846

  → standardization bodies, gov't, …

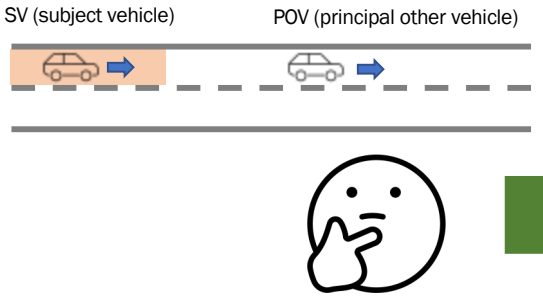
- **Safety verification**
  (the car complies with these rules. Thus safe)
- **Runtime monitoring**
  (more freedom in ADS design)

  → Car makers, ADS vendoes

- **Risk analysis in insurance**
  (Safety rule compliant ➔ lower risk ➔ cheaper insurance)

  → insurance companies

RSS Rules as Social Contracts Impacts Everywhere in the ADV Ecosystem



Want an RSS rule C …
s.t. compliance with C guarantees collision-freedom

A **proof engineer** works on each driving scenario and derives an RSS rule C

- Correctness proof for C, at the same time

RSS rule C

+

Safety proof for C

Checkable by the public
…
proof-search is hard, proof-check is easy

Attribution of liabilities
(collision
➔ ∃ a party who did not comply)

Safety standards
(Show compliance before selling cars)
e.g. IEEE P2846

standardization bodies, gov't, …

Safety verification
(the car complies with these rules. Thus safe)

Runtime monitoring
(more freedom in ADS design)

Car makers, ADS vendoes

Risk analysis in insurance
(Safety rule compliant ➔ lower risk
➔ cheaper insurance)

insurance companies

Safety Envelope by RSS Rules

Can Be Retrofit to Any ADV Controller
Monitor & Intervene ➜ Runtime Safety Guarantee

RSS Rule, an Example
[Shalev-Shwartz et al., arXiv preprint, 2017]

- An RSS rule is a pair (A, α) of an *RSS condition* A and a *proper response* α

RSS condition A:
Maintain an inter-vehicle distance at least

$$d_{\min} = \left[v_r \rho + \frac{1}{2} a_{\max,\text{accel}} \rho^2 + \frac{(v_r + \rho a_{\max,\text{accel}})^2}{2a_{\min,\text{brake}}} - \frac{v_f^2}{2a_{\max,\text{brake}}} \right]_+$$

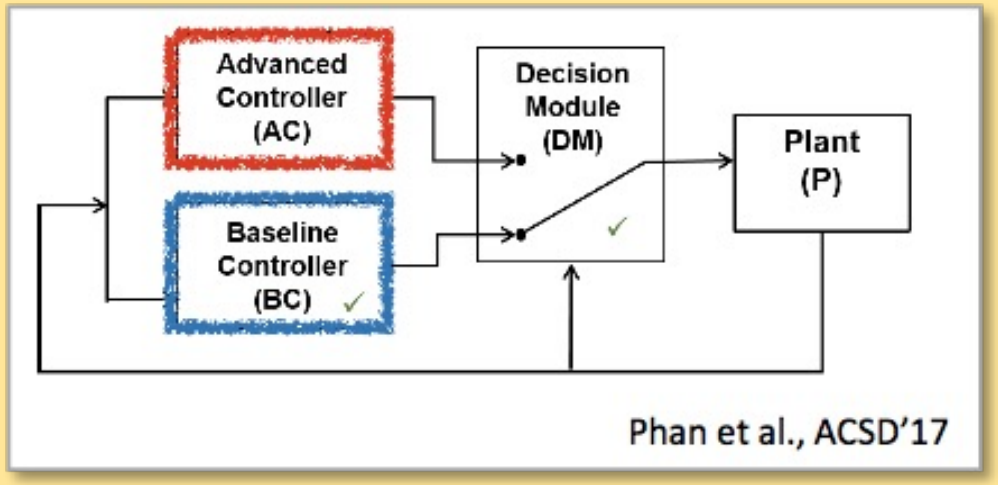
Proper response α:
If A is about to be violated, brake at rate $a_{\text{min, brake}}$ within ρ seconds

Conditional safety lemma:
Any execution of α, from a state that satisfies A, is collision-free.

escape = MRM (minimum risk maneuver)

Structure of an RSS rule

- RSS Condition A:
  "You can still *escape* if A is true"
- Proper response α:
  "control strategy to *escape*"



Simplex architecture

- AC pursues performance and safety
- BC pursues safety (only)
- DM (decision module) switches between them—
  "use BC to escape"

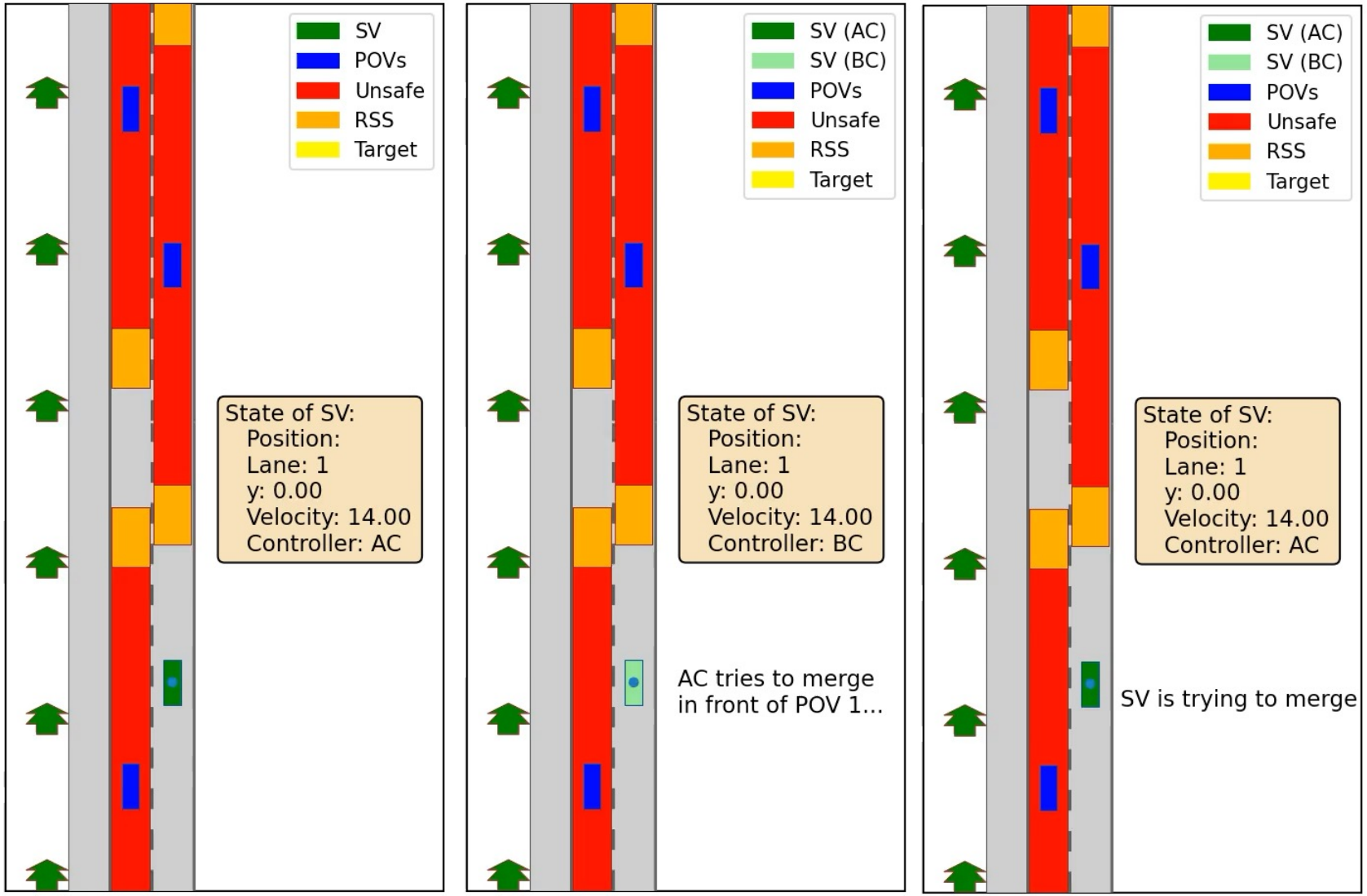
➜ RSS rules fit perfectly!

- AC: existing controller (optimization-based, ML, …)
- BC: executes a proper response
- DM: monitors an RSS condition.
  Violation foreseen ➜ switch to BC

RSS Safety Envelopes in Action, Scenario I

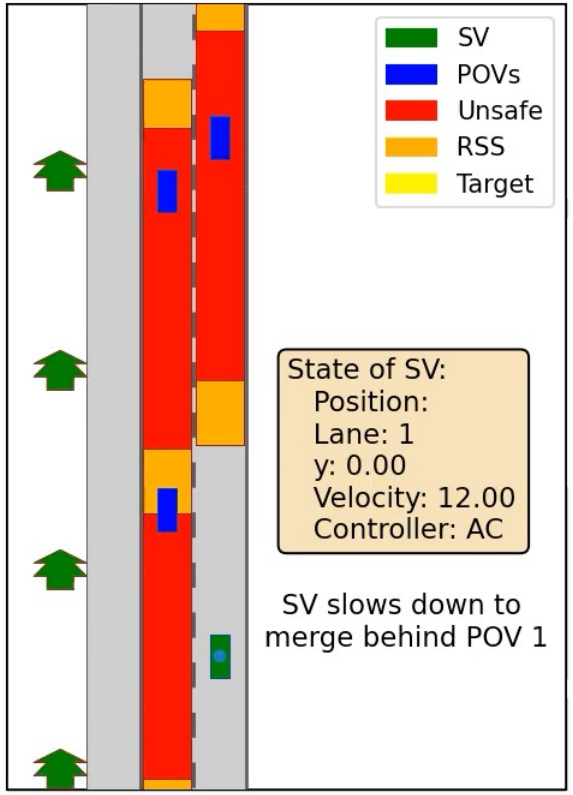
- AC: no safety envelope
- AC+RSS:
  Original RSS rule [Shalev-Shwartz et al., arXiv, 2017]
  as a safety envelope
  ("short-sighted" collision avoidance)
- AC+RSS^{GA} :
  Our RSS rule [Hasuo+, IEEE T-IV]
  as a safety envelope
  (goal achievement too with longer-term planning)

- AC is not safe (hazadous cut-in)
- AC+RSS does not reach the shoulder
- AC+RSS^{GA} successfully deployed the long term strategy of (brake → merge behind).
  Achieved both safety and the goal

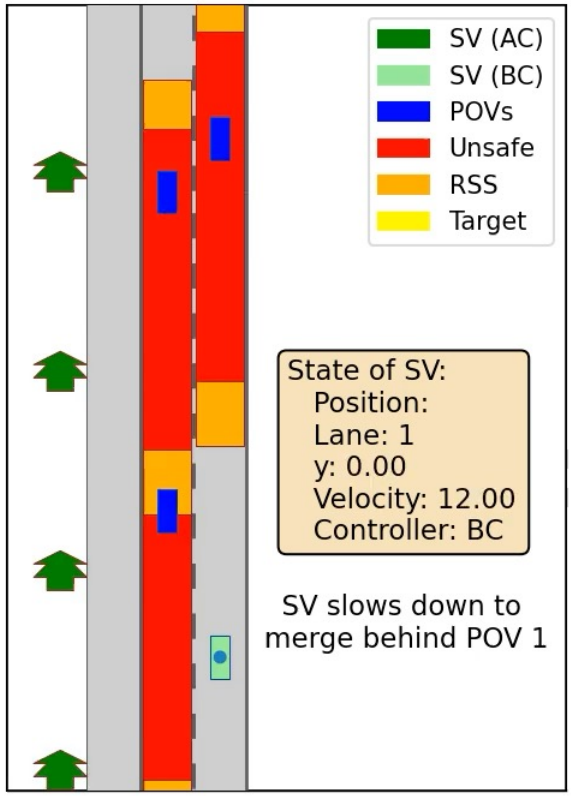


RSS Safety Envelopes in Action, Scenario II

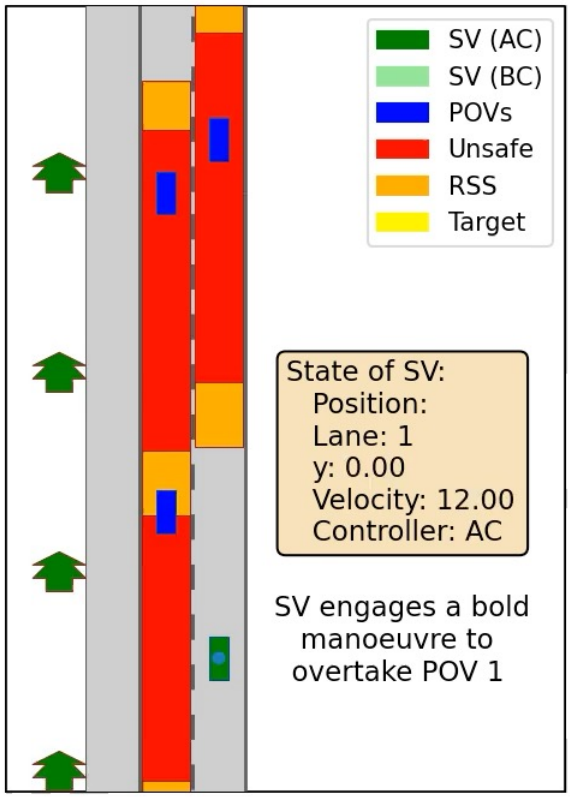
- AC: no safety envelope
- AC+RSS:
  Original RSS rule
  [Shalev-Shwartz et al., arXiv, 2017]
  as a safety envelope
  ("short-sighted" collision avoidance)
- AC+RSS^{GA} :
  Our RSS rule [Hasuo+, IEEE T-IV]
  as a safety envelope
  (goal achievement too
  with longer-term planning)



AC



AC+*RSS*

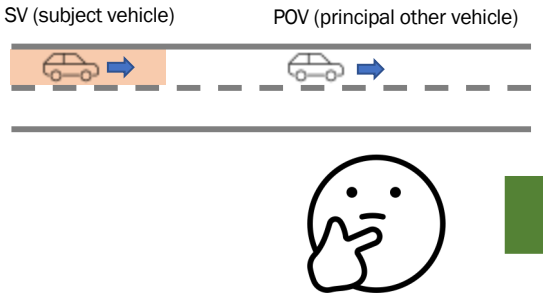


AC+*RSS*^{GA}

- AC & AC+RSS safety achieve the goal, but are **slow**
- AC+RSS^{GA},
  under mathematical safety guarantee,
  boldly accelerates and merge in front
  - ... who says safe ADVs are conservative and boring? ☺

replay

RSS Rules as Social Contracts Impacts Everywhere in the ADV Ecosystem



Want an RSS rule C …
s.t. compliance with C guarantees collision-freedom

A **proof engineer** works on each driving scenario and derives an RSS rule C

- Correctness proof for C, at the same time

RSS rule C

+

Safety proof for C

Checkable by the public
…
proof-search is hard, proof-check is easy

Attribution of liabilities
(collision
➔ ∃ a party who did not comply)

Safety standards
(Show compliance before selling cars)
e.g. IEEE P2846

standardization bodies, gov't, …

Safety verification
(the car complies with these rules. Thus safe)

Runtime monitoring
(more freedom in ADS design)

Car makers, ADS vendoes

Risk analysis in insurance
(Safety rule compliant ➔ lower risk
➔ cheaper insurance)

insurance companies

Two Different Approaches, with Different Business Models

Fixed-route bus, taxi, delivery service		*Consumer* ADV
remote	human intervention	on-site (human driver)
offers fixed-route mobility and delivery service	business model	sells consumer vehicles with ADV functionality
yes (the route is known)	geofencing	no (should drive on all public roads)
full ODD (automated driving in the entire route)	ODD operational design domain “Under which condition can the ADV take responsibility?”	partial ODD (automated driving only in prescribed situations, e.g. highway)

Two Different Approaches, with Different Business Models

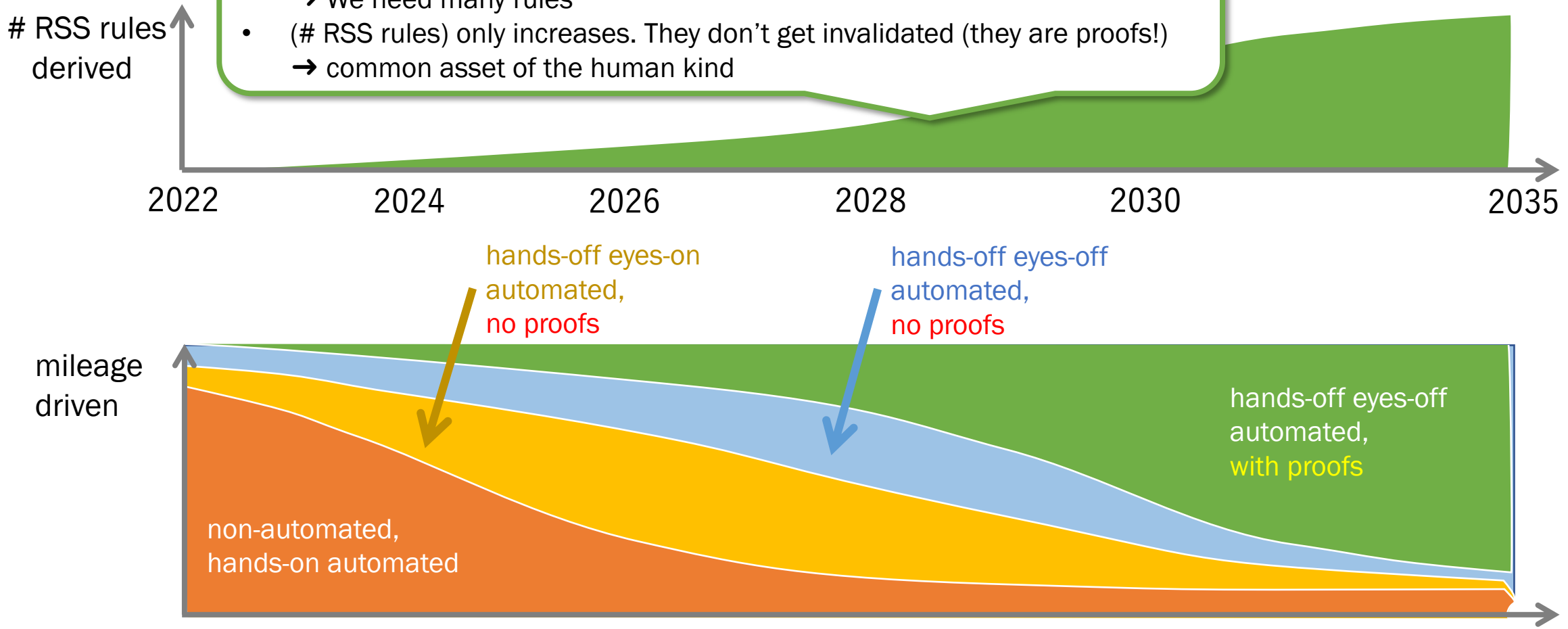
Either way, to be responsible,
we need to know driving scenarios
in advance

➜ We derive and verify RSS rules for
those driving scenarios, and
mathematically guarantee safety

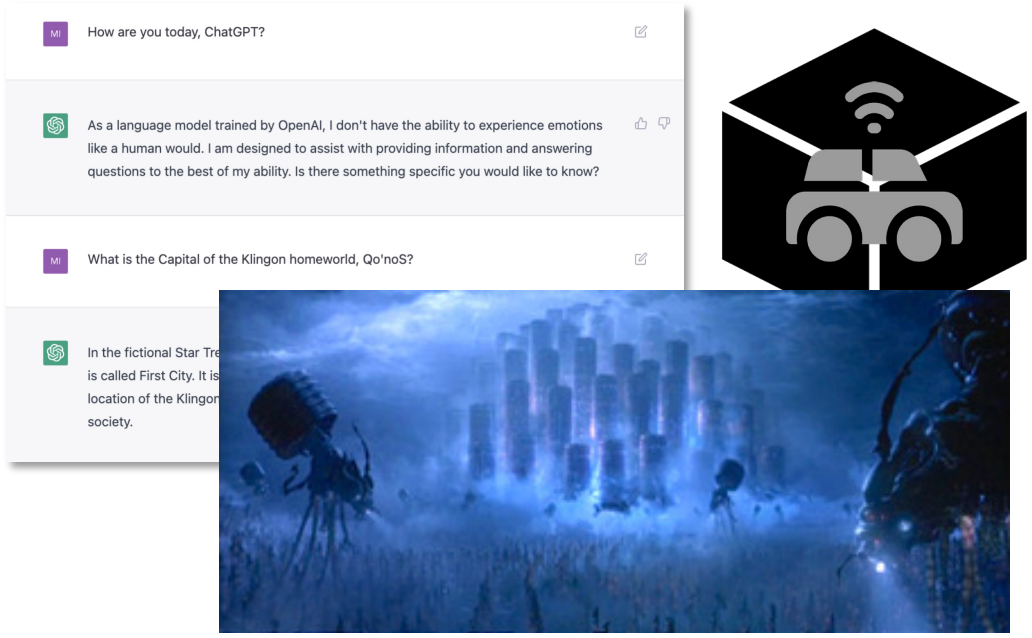
Fixed-route bus, taxi, delivery		*Consumer* ADV
remote	human intervention	on-site (human driver)
offers fixed-route mobility and delivery service	business model	sells consumer vehicles with ADV functionality
yes (the route is known)	geofencing	no (should drive on all public roads)
full ODD (automated driving in the entire route)	ODD operational design domain "Under which condition can the ADV take responsibility?"	partial ODD (automated driving only in prescribed situations, e.g. highway)

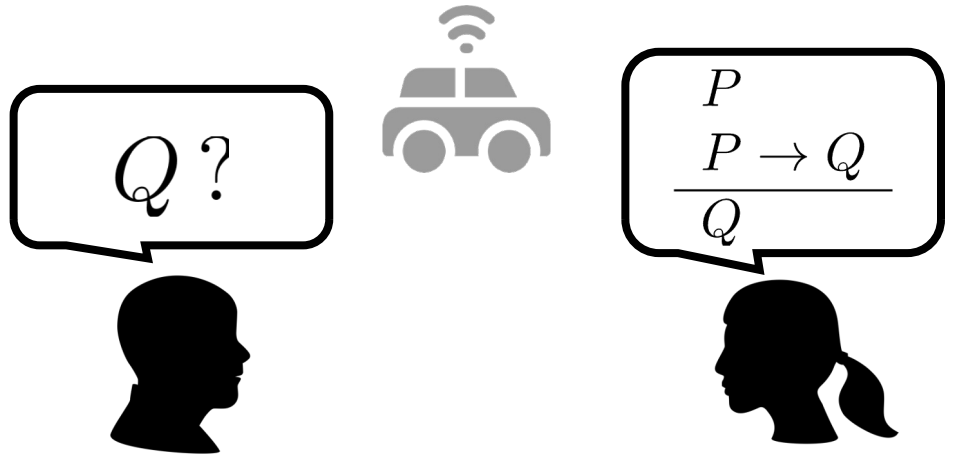
Incremental Accumulation of RSS Rules, Incremental ODD Expansion of “ADVs with Proofs”

- ~ 1 person.month for each scenario
- Different rules for different traffic laws & customs
  → We need many rules
- (# RSS rules) only increases. They don’t get invalidated (they are proofs!)
  → common asset of the human kind



Safety-Critical Systems Should Never be Blackbox
Proofs Explicate Assumptions, Contracts, ODDs, and Responsibilities





- Many emerging technologies are statistical and blackbox
- We shouldn't let them operate in safety-critical domains
- (… fight against the "lawyer up" approach towards safety!)

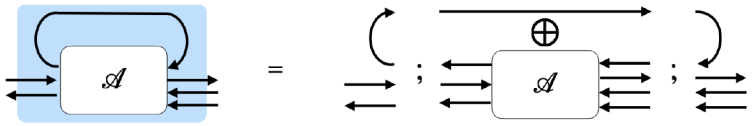
- Conventionally:
  Proofs are for establishing absolute truths
- New: proofs are communication media for
  - explicating assumptions and contracts,
  - showing who's responsible for what, and
  - writing and assessing safety cases
- Logiic as a social infrastructure for trust in ICT

Coming up...

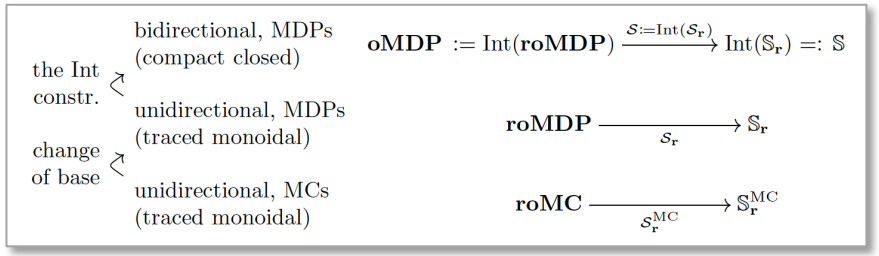
RSS verifying safety architecture
[Eberhart, Dubut, Haydon & Hasuo, IV'23]

Compositional MDP model checking by string diagrams
[Watanabe, Eberhart, Asada & Hasuo, CAV'23]

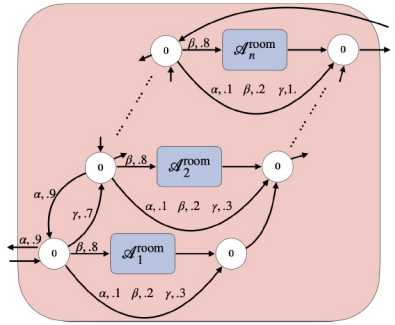
- MDP model checking can be **compositional** over string diagrams of MDPs



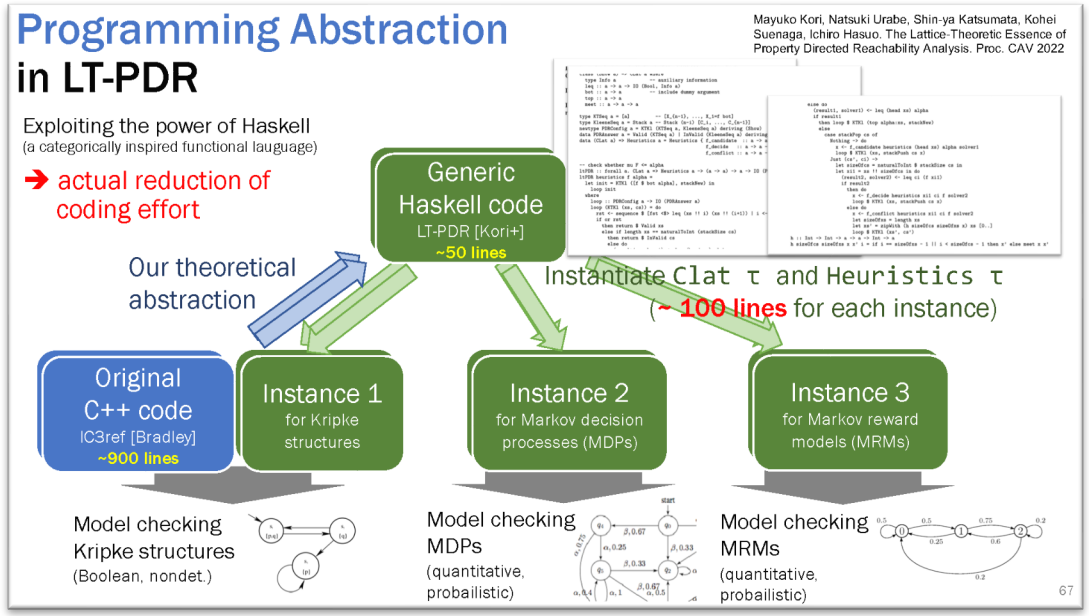
- Algorithm derived from the structural theory of monoidal categories



- … which can be *arbitrary faster* than existing (non-compositional) algorithms



From mathematical abstraction to programming abstraction
[Kori, Urabe, Katsumata, Suenaga & Hasuo, CAV'22]
[Kori, Ascari, Bonchi, Bruni, Gori & Hasuo, CAV'23]



- We can **literally code the abstract theory** thanks to Haskell
- Appl. to IC3/PDR (Bradley, Een, …):
  50 LOC (general) + ~100 LOC each (instant.)
  - vs. original IC3 impl., ~900 LOC in C++