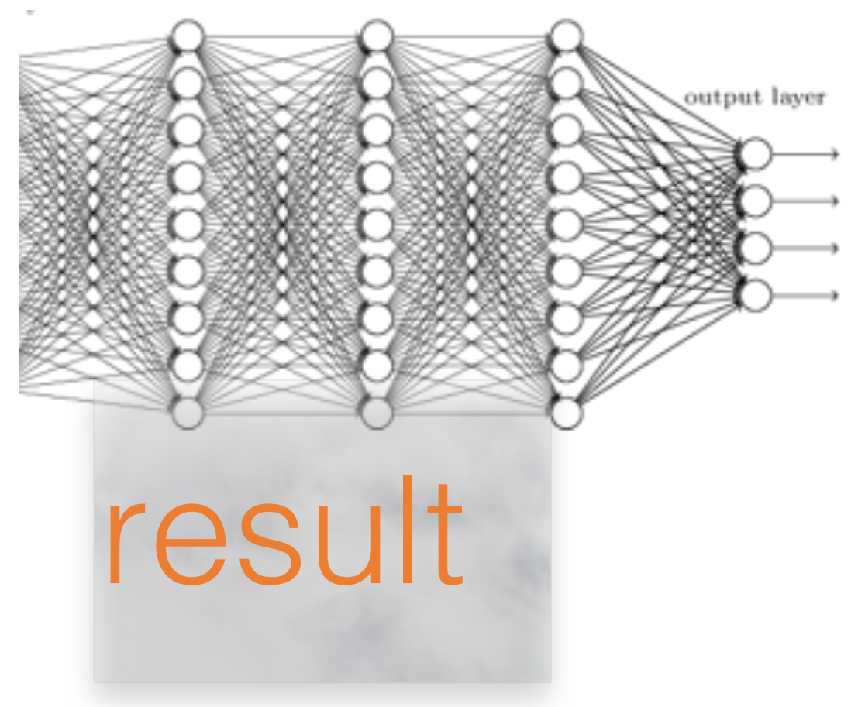
Anomaly Detection Through Explanations

Leilani H. Gilpin
lhg@mit.edu

MIT CSAIL
Massachusetts Institute of Technology

Problem

Complex machines and autonomous agents cannot provide insights into their behavior and thought processes.



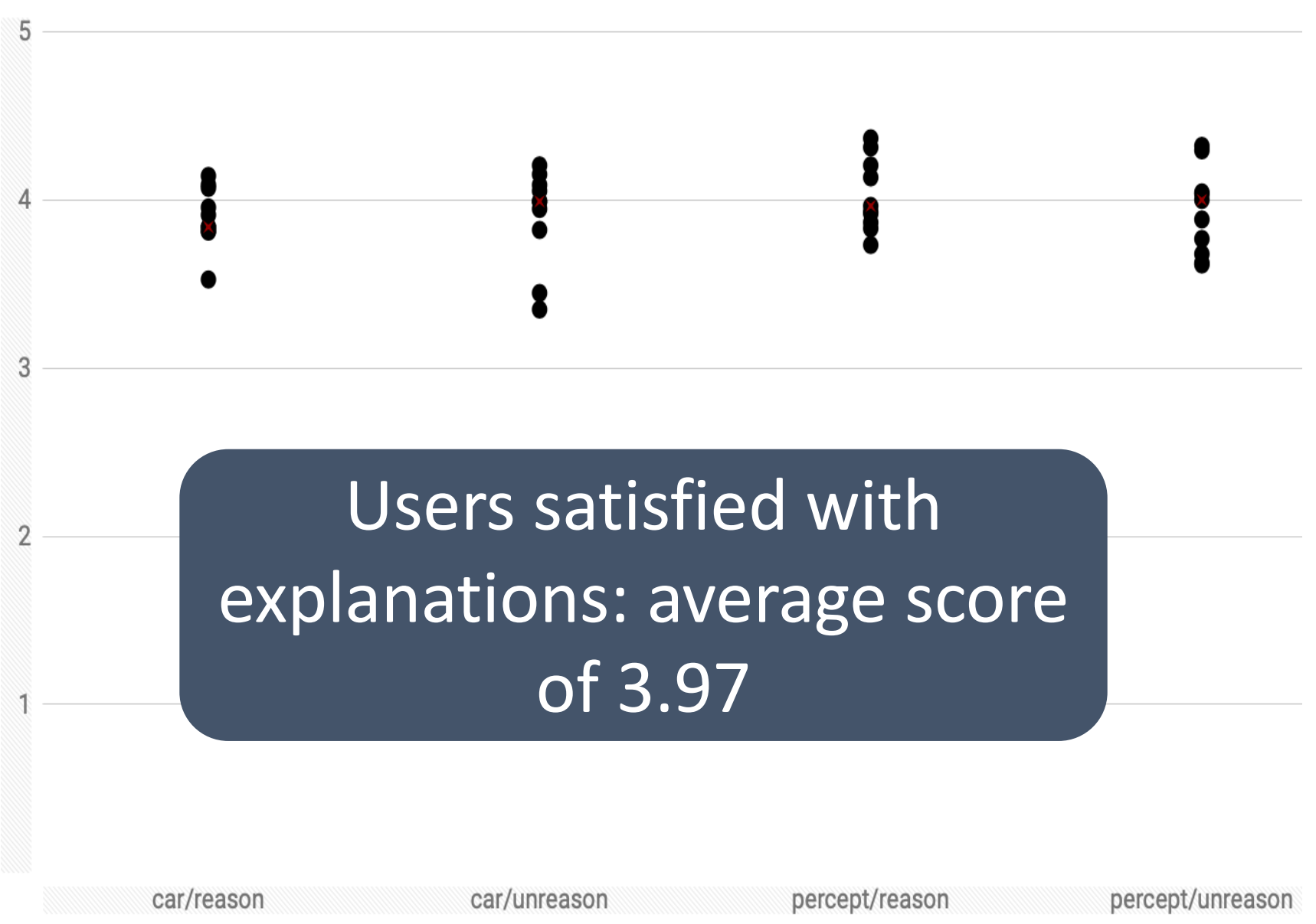
No explanation at all

Communication to non-expert

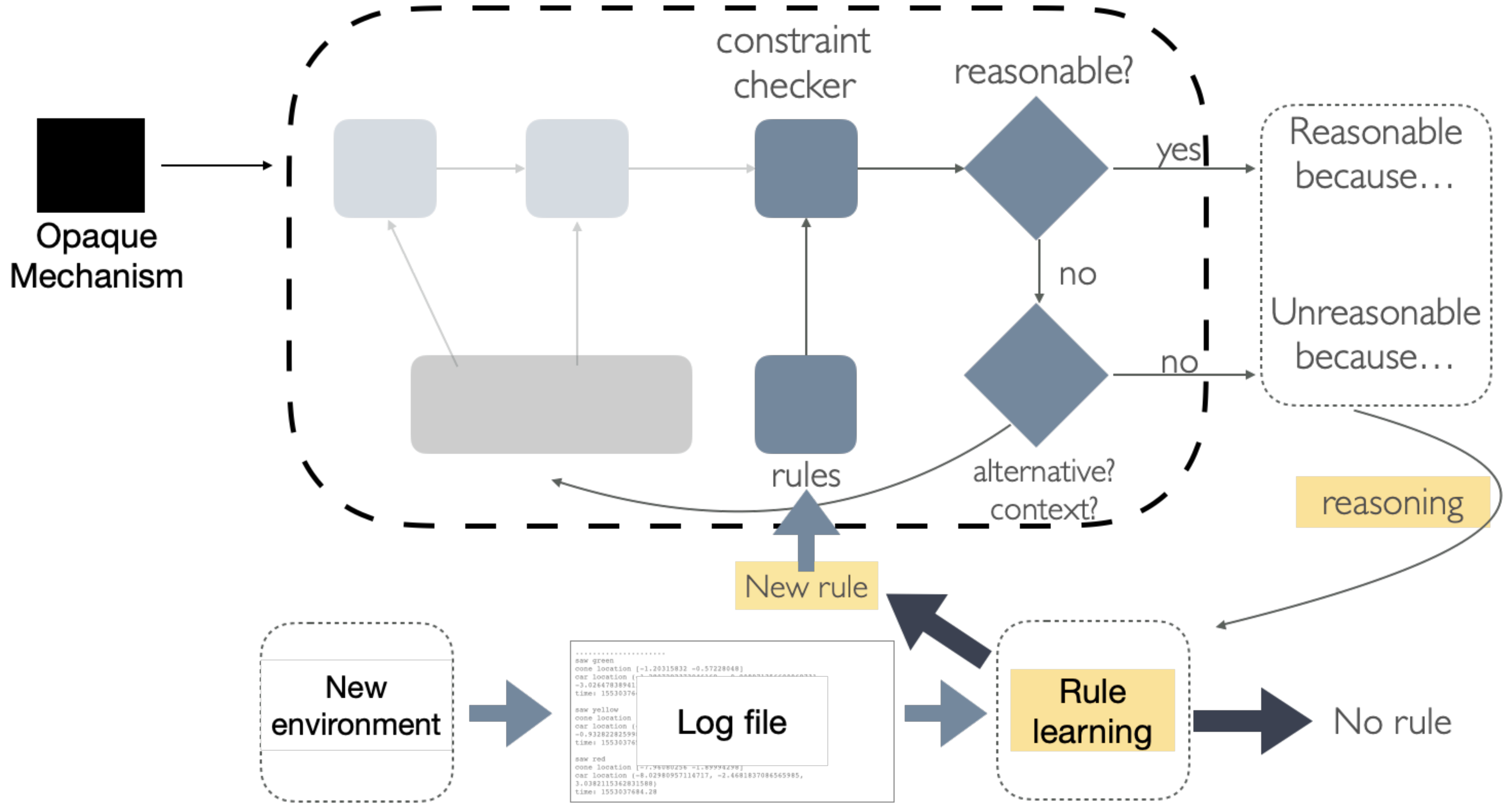
Explanation to human expert

Justify and Explain Local Decisions

- Supplement opaque decision-making systems with commonsense knowledge.
- Represent input and rules in reusable web-standards for adaptability and extension to multiple applications.
- Some rules can be *learned* by examining the justifications..

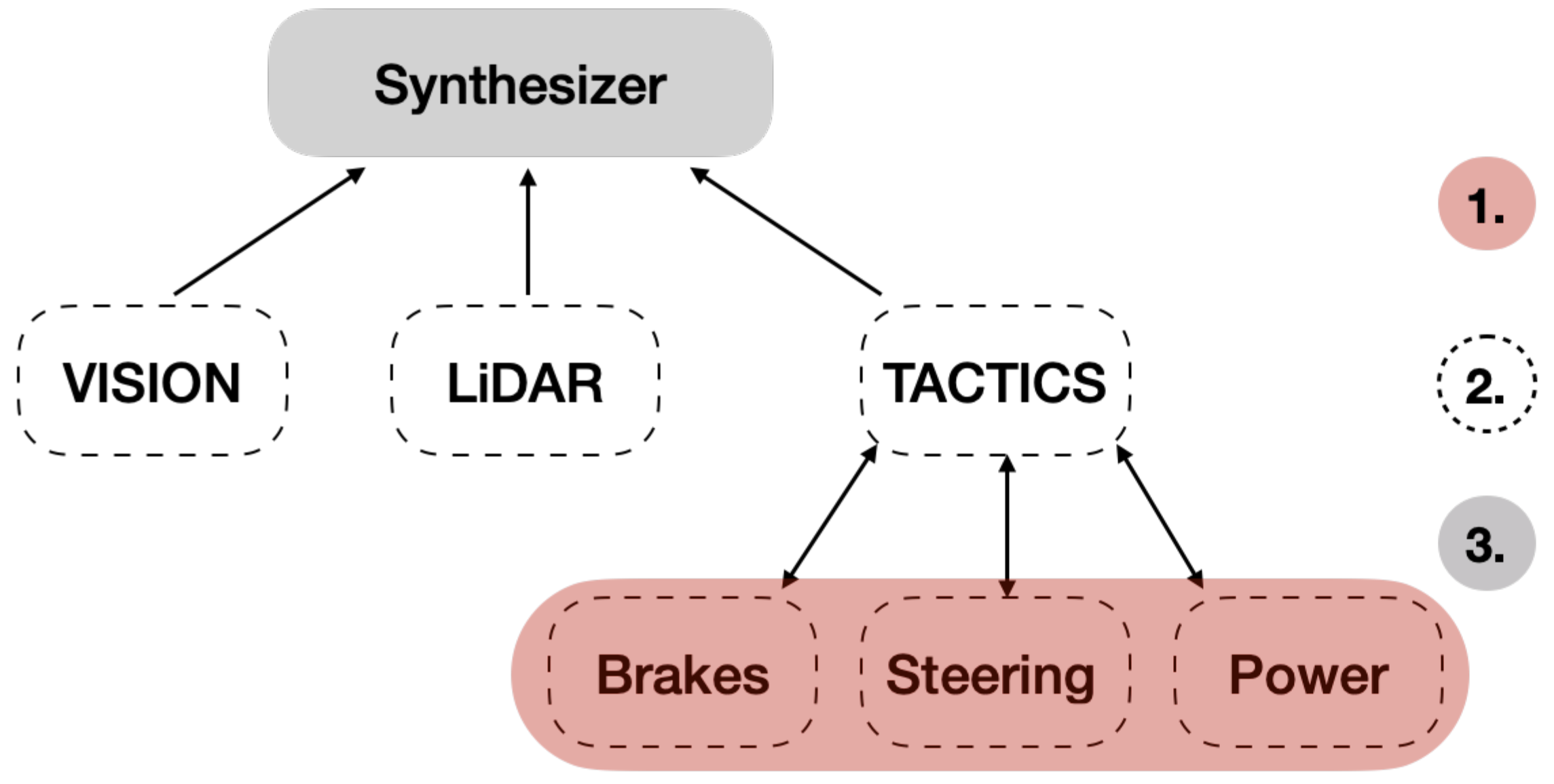


Users satisfied with explanations: average score of 3.97



Explain Failed Cooperations

- Reasonableness monitor around each component (including the planner).
- A reasoner that processes the component explanations.
- Priority hierarchy to reconcile conflicting interests.



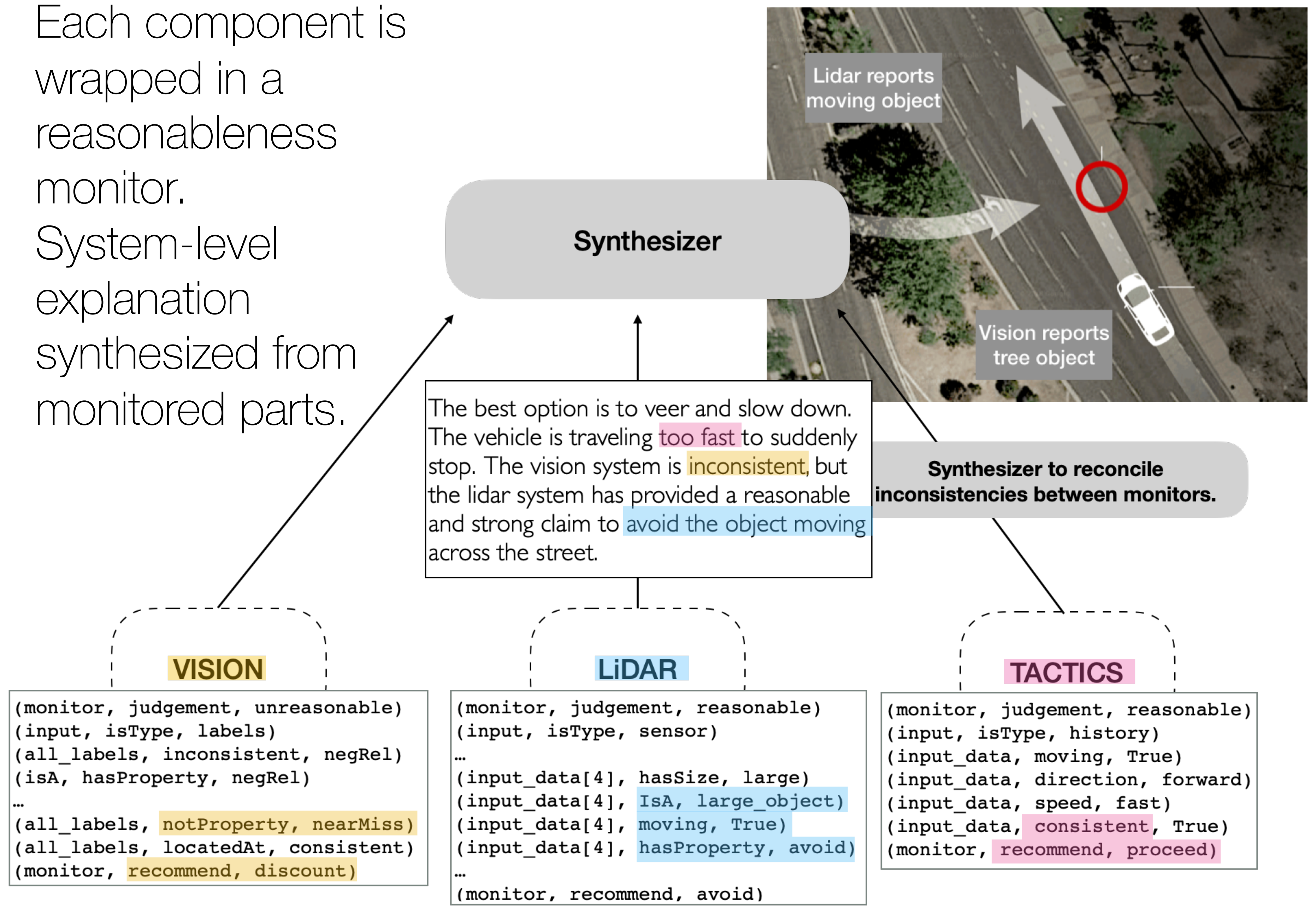
1. Generate Symbolic Qualitative Descriptions for each committee.
2. Input qualitative descriptions into local "reasonableness" monitors.
3. Use a synthesizer to reconcile inconsistencies between monitors.

Evaluation on

1. Real-world inspired scenarios: autonomous driving failures in Carla (an autonomous vehicle simulated).
2. Added errors to an autonomous driving dataset: NuScenes.

Results: Explain the Uber Fatality

Each component is wrapped in a reasonableness monitor. System-level explanation synthesized from monitored parts.


Synthesizer
Lidar reports moving object
Vision reports tree object
Synthesizer to reconcile inconsistencies between monitors.
VISION
LiDAR
TACTICS

The best option is to veer and slow down. The vehicle is traveling too fast to suddenly stop. The vision system is inconsistent, but the lidar system has provided a reasonable and strong claim to avoid the object moving across the street.

VISION

```
(monitor, judgement, unreasonable)
(input, isType, labels)
(all_labels, inconsistent, negRel)
(isA, hasProperty, negRel)
…
(all_labels, notProperty, nearMiss)
(all_labels, locatedAt, consistent)
(monitor, recommend, discount)
```

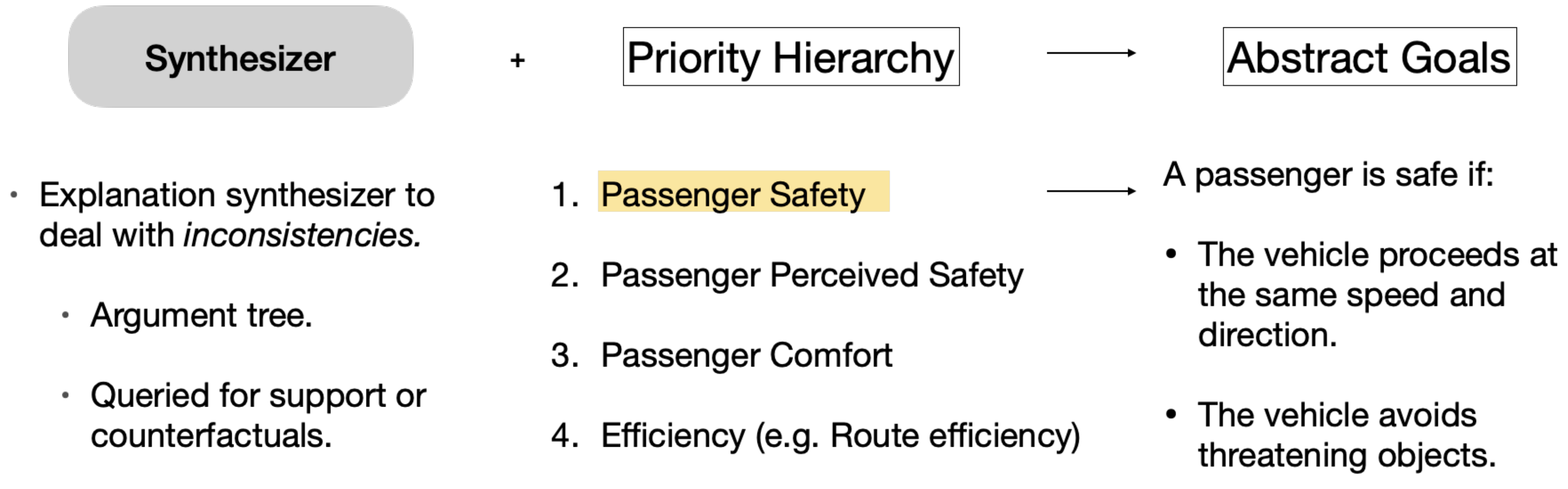
LiDAR

```
(monitor, judgement, reasonable)
(input, isType, sensor)
…
(input_data[4], hasSize, large)
(input_data[4], IsA, large_object)
(input_data[4], moving, True)
(input_data[4], hasProperty, avoid)
…
(monitor, recommend, avoid)
```

TACTICS

```
(monitor, judgement, reasonable)
(input, isType, history)
(input_data, moving, True)
(input_data, direction, forward)
(input_data, speed, fast)
(input_data, consistent, True)
(monitor, recommend, proceed)
```

Priority Hierarchy



Synthesizer + Priority Hierarchy → Abstract Goals

- Explanation synthesizer to deal with *inconsistencies.*
  - Argument tree.
  - Queried for support or counterfactuals.

1. Passenger Safety
2. Passenger Perceived Safety
3. Passenger Comfort
4. Efficiency (e.g. Route efficiency)

A passenger is safe if:

- The vehicle proceeds at the same speed and direction.
- The vehicle avoids threatening objects.

- Goals are represented in rules:

$$\forall s,t \in STATE, v \in VELOCITY$$
$$\big((self, moving, v), \textbf{state}, s\big) \land (t, \textbf{isSuccesorState}, s) \land \big((self, moving, v), \textbf{state}, t\big) \land$$
$$\nexists x \in OBJECTS \ \textbf{s.t.} \ \big((x, isA, threat), \textbf{state}, s\big) \lor \big((x, isA, threat), \textbf{state}, t\big)$$
$$\Rightarrow \big(\textbf{passenger, hasProperty, safe}\big)$$

$$\forall s \in STATE, x \in OBJECT, v \in VELOCITY$$
$$\big((x, moving, v), \textbf{state}, s\big) \land \big((x, locatedNear, self), \textbf{state}, s\big) \land$$
$$\big((x, isA, large\_object), \textbf{state}, s\big) \Leftrightarrow \big((x, isA, threat), \textbf{state}, s\big)$$

Results: Added Errors to NuScenes

- Added errors to a self-driving car data set: NuScenes
  - Scrambled image labels.
  - Added noise to the bounding box dimensions.
- First data set of *multi-modal* errors.
- Results show the *synthesizer* results in less false positives and false negatives.

Priority	Correctness	False Positives	False Negatives
No synthesizer	85.6%	7.1%	7.3%
Single subsystem	88.9%	7.9%	3.2%
Safety	93.5%	4.8%	1.7%

Contributions

- After-the-fact explanations for legal and liability analysis.
- Reasonableness monitoring for opaque systems.
- A methodology for explanatory error detection in complex systems: using introspection and explanation as an internal language for robust, explainable decisions.