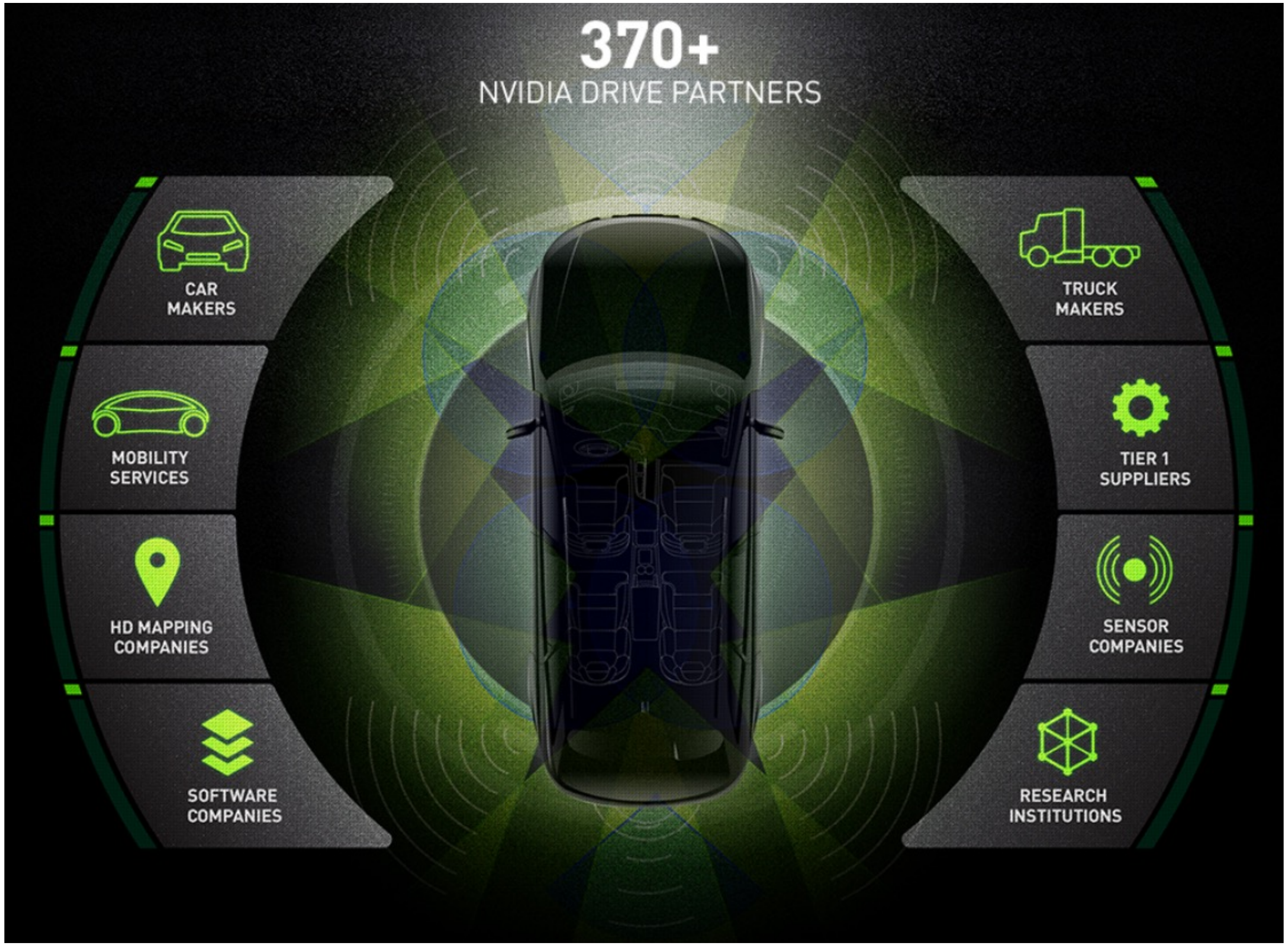
What Safety Challenges for Autonomous Systems Would Benefit from Research?

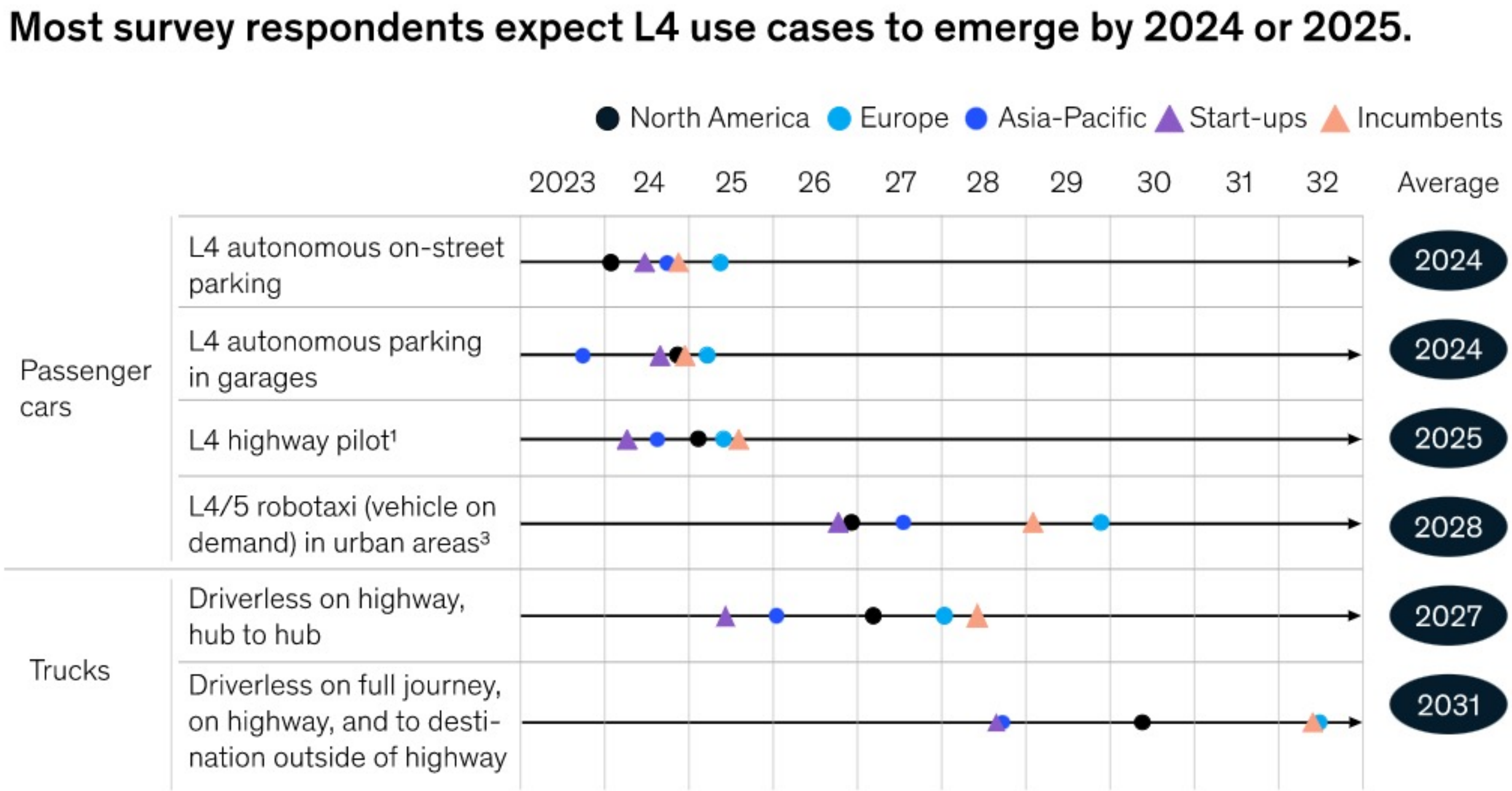
Timothy Tsai
2022-01-21 IFIP WG 10.4 Winter Workshop

NVIDIA

Tremendous Interest in Autonomous Vehicles

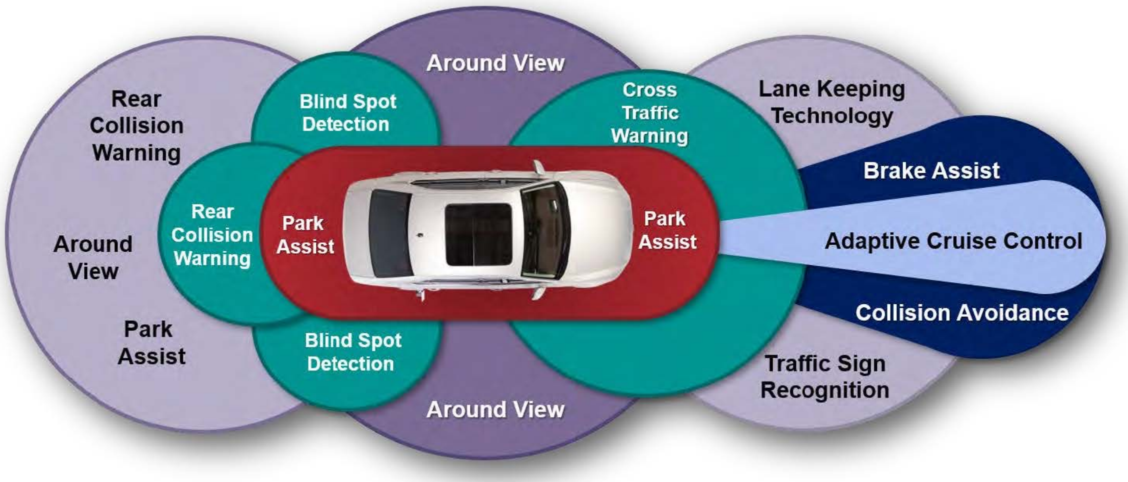


AVs Are Imminent



https://www.mckinsey.com/features/mckinsey-center-for-future-mobility/our-insights/whats-next-for-autonomous-vehicles

Functionality vs. Safety



- Functionality ("drive to there") and safety ("don't hit anything") are closely related
- Need correct perception and planning
- Full functionality is still lacking (e.g., for corner cases)

Today’s Commercial AVs

Level 2

Level 3

“Level 3”

Level 4

- Level 3 and beyond are starting to be sold
- Safety/liability is important!

Waymo Disengagement Reports

- From 2020 California DMV AV disengagement reports

Disengagement reason	#
A perception discrepancy for which a component of the vehicle's perception system failed to detect an object correctly	8
Adverse weather conditions experienced during testing	3
Incorrect behavior prediction of other traffic participants	1
A recklessly behaving road user	1
Unwanted maneuver of the vehicle that was undesirable under the circumstances	8

Perception is a challenge → **How can we improve perception?**

Disengagement-based HW FIT Estimate

- From 2020 California DMV AV disengagement reports

Disengagement reason	#
Hardware diagnostic caused software kickout	25
Hardware Issue: Smart camera stop working	3
Hardware diagnostic detected hardware health issue	3
Hardware Issue: Wrong GPS state	2
Hardware discrepancy or system fault	1

- HW-related Disengagements: 34 / 3695 over ~2e6 miles (assuming avg 30mph) = about 5e5 FIT (!!!)
- Need to read disparate logs with different methodologies cautiously!
  - All 34 reports from 3 companies representing ~1% of all miles

Random Hardware Faults

- ISO 26262 requirements
  - Single-Point Fault Metric (SPFM): Diagnostic coverage
  - Probabilistic Metric for random Hardware Failures (PMHF)

	ASIL-D	ASIL-C	ASIL-B	ASIL-A
SPFM	≥99%	≥97%	≥90%	
PMHF	<10 FIT	<100 FIT	<100 FIT	<1000 FIT

- Companies spent a lot of money and time on this
  - Vendors like Nvidia can’t assume specific SW when evaluating HW error propagation
  - FMEDA requires time and assumptions
    - What SW?
    - How to measure error propagation? Fidelity vs. efficiency trade-off

→ **<u>How can we find the expected error propagation for different modules?</u>**

Very High Error Masking for AVs

Low error propagation for …

- DNNs (SC’17)
  - Low propagation for LSBs and early layers
- AV perception (Internal FI on Nvidia DriveWorks)
  - Tolerance via smoothing and fusion
- Arch → actuators and car behavior (DSN’19)
  - Must corrupt many frames to make a difference
- Closed-loop control system (DSN’19)
  - Braking/throttle and steering compensate
- Typical scenarios
  - E.g., most drunk and texting drivers don’t have accidents (dumb luck – nothing to hit)

Do random hardware faults matter?

- Low error propagation through entire AV stack.
- Few reports of random hardware faults in disengagement reports.
  - Like HW faults on Windows, would we blame the SW because SW FIT rate is higher?
- Transient faults seem to largely get masked out.
- Permanent faults tend to result in DUEs. AVs are fail-safe with minimum risk maneuvers.

→ **Do random hardware faults matter?**

Do random hardware faults matter? (cont.)

→ **How can we demonstrate this (random HW faults don't matter)?**

- FMEDA for diagnostic coverage takes a lot of time, people, and assumptions
  - Don't know which SW runs
  - Full-system simulation is expensive
  - Low error propagation requires a lot of FI runs
- Can we …
  - Do importance sampling?
  - Use higher-level FI (e.g., PINFI, NVBitFI) by modeling lower-level propagation from the fault?

Do random hardware faults matter? (cont.)

- By avoiding low-level error detection and mitigation, can we
  - Save time and money?
  - Avoid unnecessary DUEs?
- Sales view is absolutely no, because we need certification. Especially true for vendors, like Nvidia.
- But what is the engineering view?

→ **<u>What modules should we focus on (biggest bang for the buck)?</u>**

Safety-Critical Scenarios

- Most scenarios are not safety critical

- Scenario coverage metric?
  - SOTIF is emerging but relies on a HARA enumeration of scenarios
    - Relies on engineering expertise → May not be repeatable
    - How do we know the HARA analysis is complete?

- Benchmark of safety-critical scenarios?
  - NCAP (list others) exist, but how comprehensive are they? I.e., what do they miss?
  - How about a scalable benchmark that yields a quantitative metric? E.g., if a system can handle one scenario, adjust scenario parameters to find breaking point.

→ **How can we find the safety critical scenarios?**

→ **Is there a metric for scenario coverage?**

→ **Can we produce a benchmark of safety-critical scenarios?**

Conclusion

- Safety will improve as functionality improves
- How do we figure out which random hardware faults matter and which don't?
- How do we figure out which scenarios matter and which don't (for safety)?