

自動運転システムの形式検証手法の確立に向けて

富田 堯

北陸先端科学技術大学院大学



自己紹介

- 富田 堯

- 所属

- 北陸先端科学技術大学院大学(JAIST) 情報社会基盤研究センター 准教授
 - (兼任) 先端科学技術研究科 次世代デジタル社会基盤研究領域

- 専門分野

- 形式手法

- モデル検査, 仕様検証, 自動合成, 及びそれらの定量的拡張
 - 実製品/開発現場への応用
 - 理論計算機科学, 数理論理学, 形式言語, オートマトン理論, 確率論・統計論

- 研究プロジェクト

- JST戦略的創造研究推進事業 CRESTプロジェクト「次世代車載基盤システムのための形式手法と検証ツールの創出 (2023年10月-2029年3月)」主たる共同研究者

概要

- これまでの取り組み
 - 現代的車載システムの形式的検証
- CRESTプロジェクト紹介
 - 次世代車載システムの形式的検証

形式手法

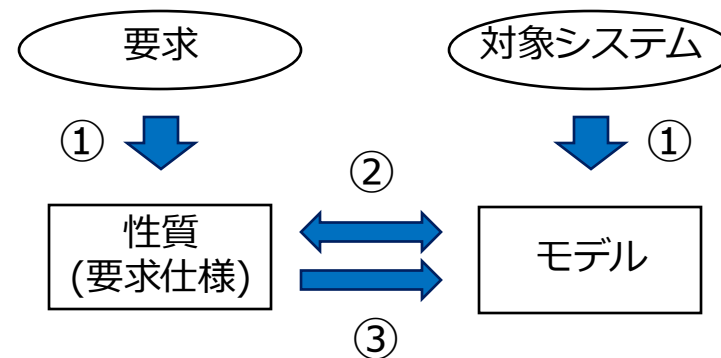
- 数学を基盤にした技法及びそれを実装したツールを活用して，主にSWやHWを検証・開発する手法群

① 形式的記述（仕様記述，モデル化）

② 検証

- 定理証明，モデル検査，制約解消 (SAT/SMT/...), ...

③ 合成 (& 最適化)



- 利点

- 安全性・信頼性を厳密に保証できる.
- 様々なツールが開発されている.
 - 定理証明支援系 (Coq, Lean, ...), モデル検査器 (SPIN, UPPAAL, PRISM, CBMC...), SAT/SMTソルバ, ...

- 難しさ

- 理論的限界/スケーラビリティ：計算不能性/計算困難性がある.
- 適用困難性：現実問題が既存技術の適用範囲に上手くハマらない. 高い専門知識とスキルが必要 (そうに見える).

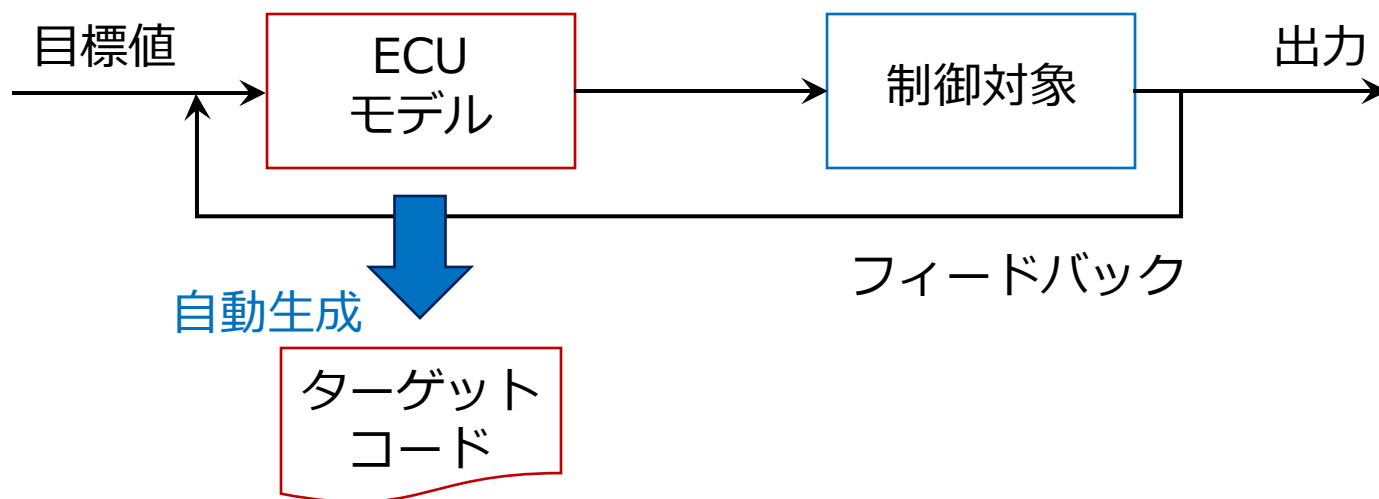
これまでの取り組み：
現代的車載システムの形式的検証

現代的車載システム

- ここでは、運転支援/自動運転がない車載システムを指す（ことにさせていただきます）。
- 現代の自動車は、内燃機関自動車（ICEV）であっても、構成する各コンポーネントは電子制御されている。
 - パワートレイン系：エンジン, トランスミッション, ...
 - シャシ系：アクセル, ブレーキ, パワーステアリング, サスペンション, ...
 - ボディ系：ドア, 照明, ワイパー, ...
 - マルチメディア系：カーナビ, バックモニタ, ...

モデルベース開発

- 車載システム等に搭載される高安全・高信頼性が求められる電子制御ユニット (ECU) 用ソフトウェアの開発では, MATLAB/Simulinkを用いたモデルベース開発 (MBD) が広く採用されている.
 - 制御の仕組み (と制御対象を) モデリング
 - モデルを実行可能な仕様とみなせる.
 - 高抽象度モデル-低抽象度モデル間, モデル-コード間の適合性はBack-to-Backテストで確認

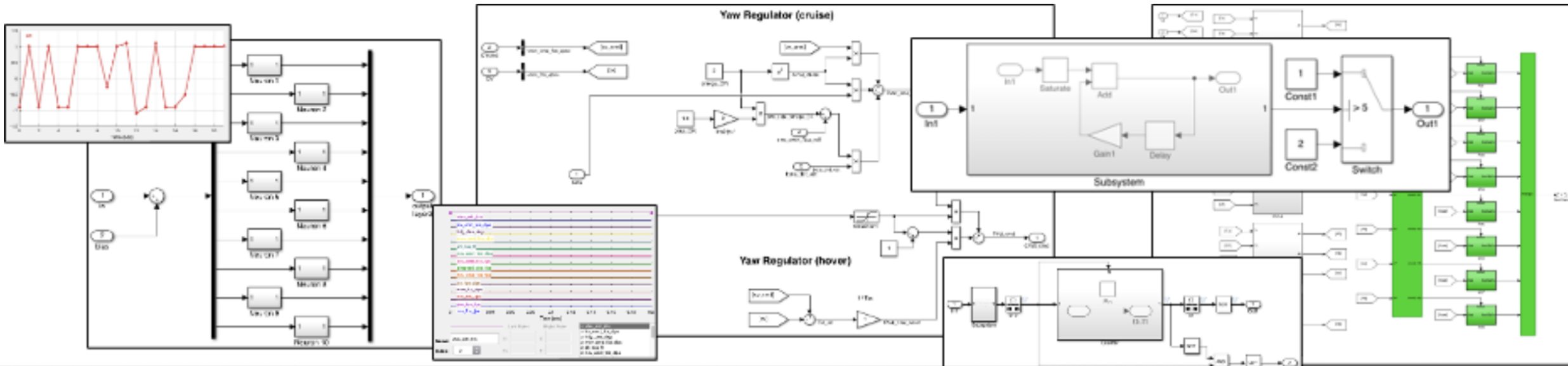


MATLAB/Simulink

- 車載ECU開発ではデファクトスタンダードツール
 - MATLAB：数値計算プラットフォーム
 - Simulink：ブロック線図モデリング&シミュレーションツール
 - Stateflow：フローチャート/状態遷移図/状態遷移表記述ツール

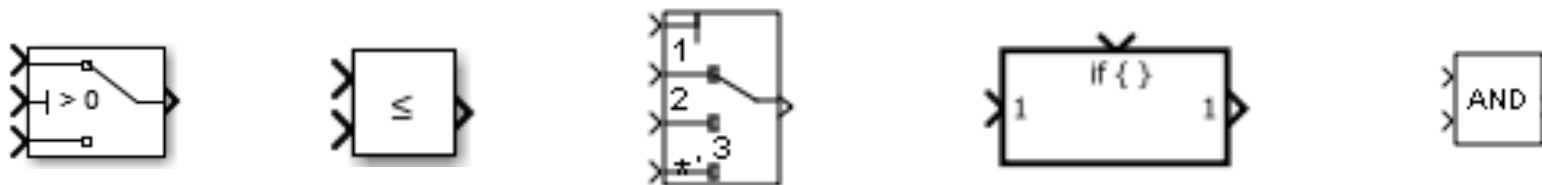


グラフィカルに制御システムを設計&実装



車載ECU向けカバレッジテスト

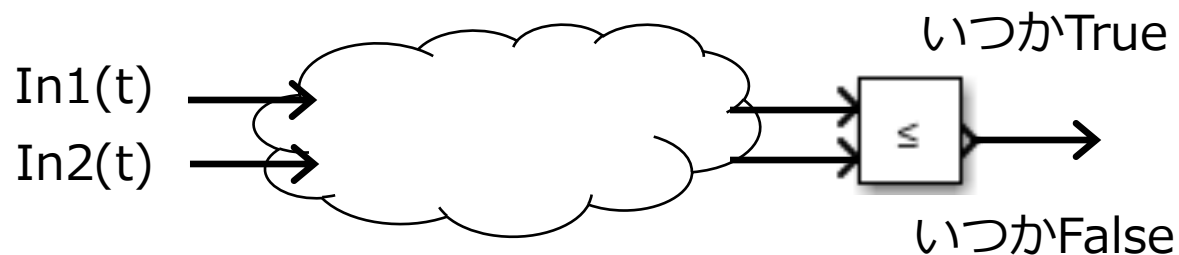
- ISO26262では、故障時のリスクが高いコンポーネントに対してカバレッジ基準を満たすテストを実施することが求められている。
 - カバレッジ：コード/モデル中の論理機構がどれだけ網羅的に動作するかを示す指標。
 - 判定カバレッジ (DC), 条件カバレッジ (CC), 修正条件・判定カバレッジ (MC/DC)など



- ECU用ソフトウェアの大規模・複雑化へ対応するため、上流工程での十分なモデルテストが強く求められている。
 - ⇒ テストの高品質化・効率化・自動化

車載ECUモデル向けカバレッジテストにおける課題

- MATLAB/Simulinkのテスト用ツールボックス (SLDV及びSLC) は,
 1. 実用的なモデルのテスト自動生成ができない（ことがある）.
 - 主に形式手法を用いているため.
 - 大規模 (ブロック数/入力数が多い, データ幅が広い, 長い時系列入力信号) $\Rightarrow$ 計算コストが莫大
 - 複雑 (非線形演算, 遅延回路, etc.) $\Rightarrow$ 決定困難/不能



2. カバレッジ計測が遅い.
 - テストを作成しても実施に多大な時間を要する.

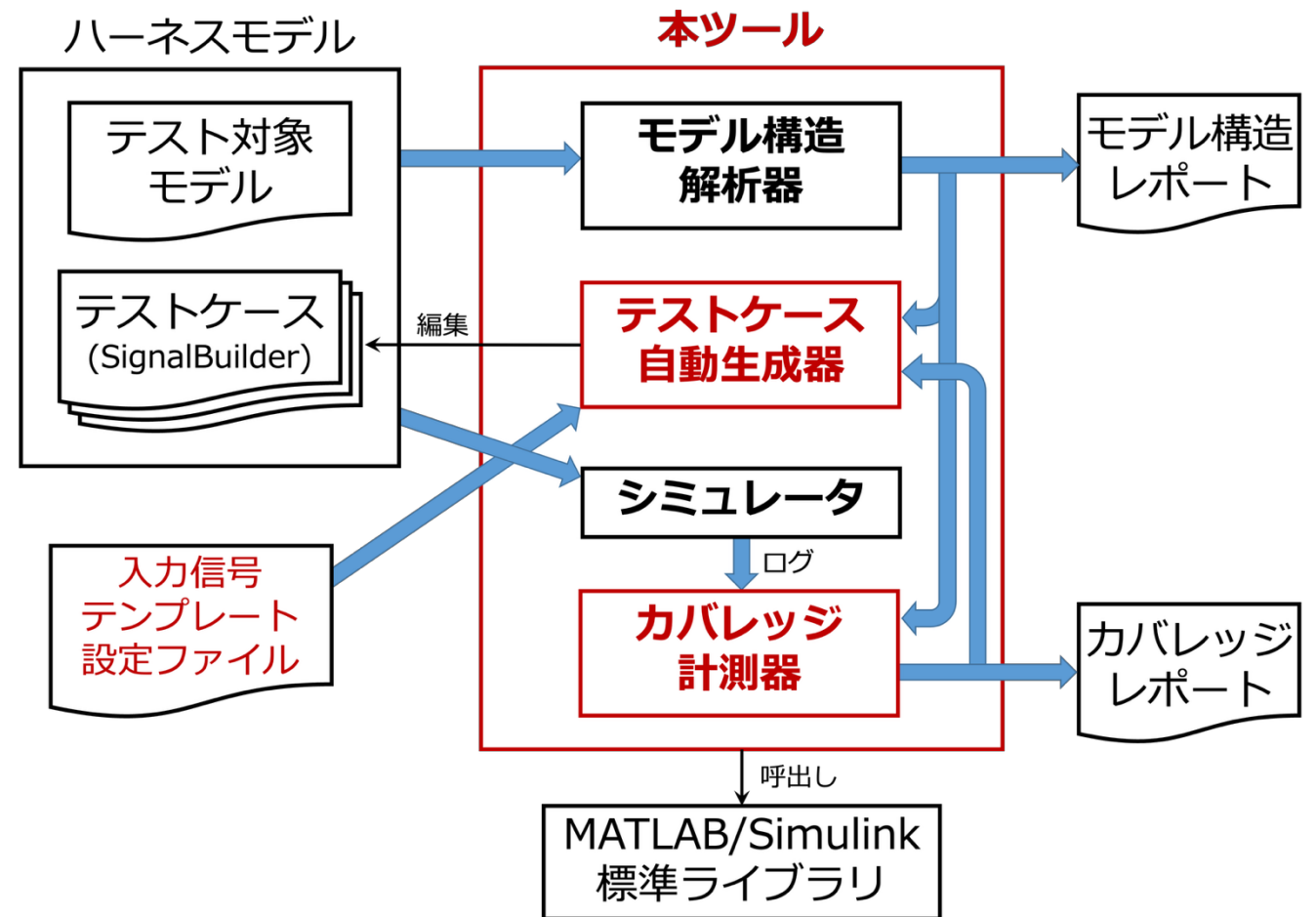
$\Rightarrow$ 共同研究先企業からカバレッジテストツールPROMPTをリリース

- モンテカルロ法 + 形式手法を採用



PROMPT-V1概要

- テストケース自動生成器
 - 対象はハーネスモデル
 - SignalBuilderの信号グループ = テストケース
 - テストケース生成は**テンプレートベースモンテカルロ法**
 - 各パラメタに具体値をランダムに割当てインスタンス化
- カバレッジ計測器
 - シミュレーションとカバレッジ計測を分離
 - **カバレッジを高速に計測**
 - 高速モードも利用可
 - 配列演算命令群を活用



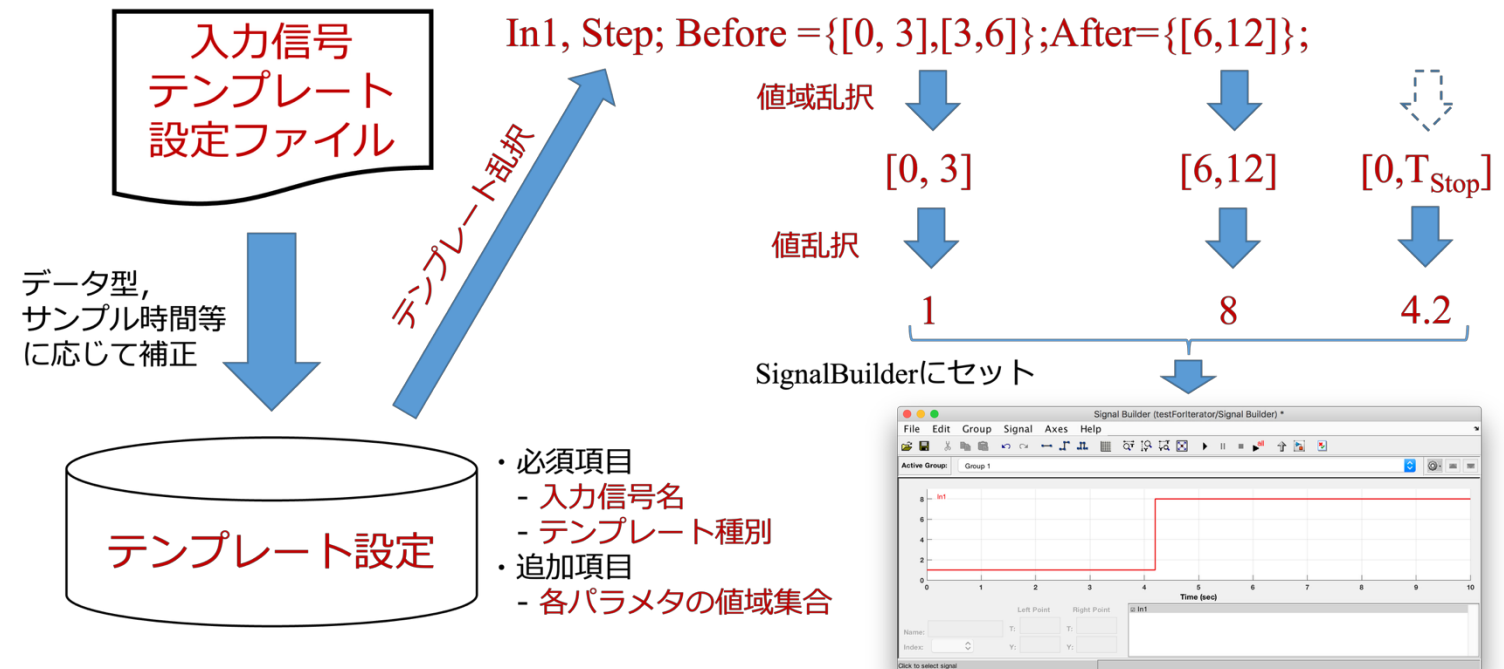
テンプレートベースモンテカルロ法によるテスト生成

• テンプレートベース

- 信号の概形をテンプレート（線形/ステップ/正弦波など）で指定する。
 - 信号の意図・意味を理解し易いし，現実的には任意波形を想定する必要はあまりない。
⇒ 探索空間を効果的に制限できる。

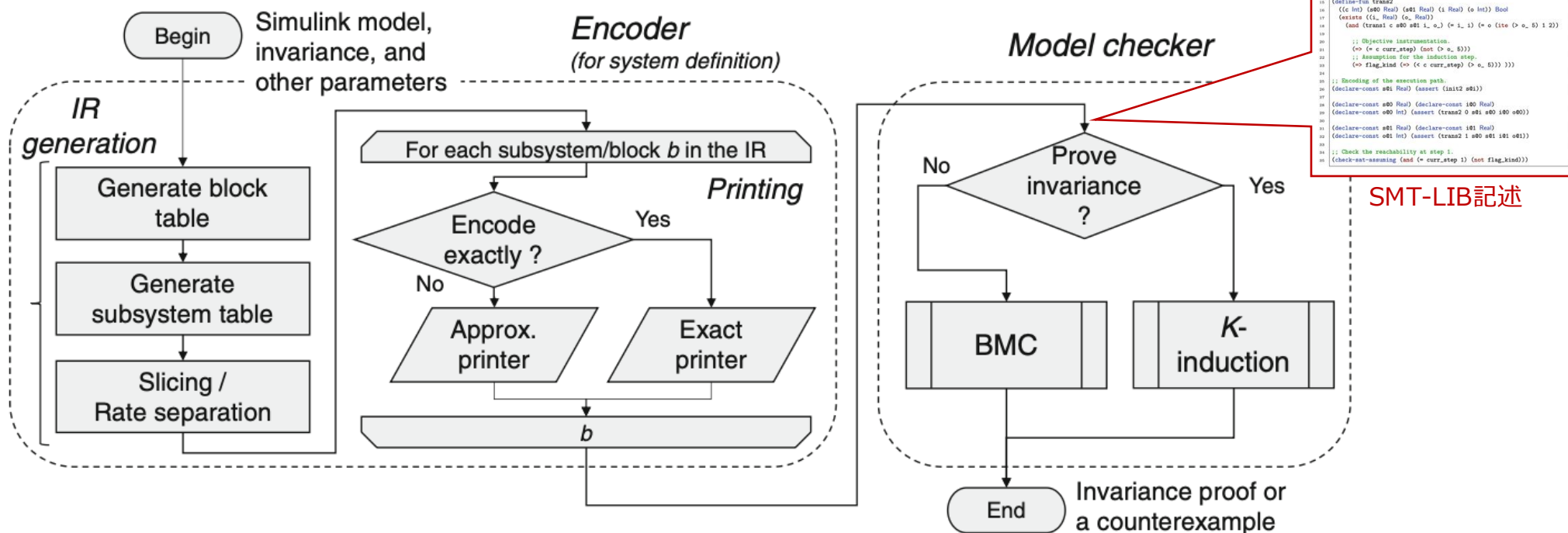
• モンテカルロ法

- ランダムに生成し，カバレッジに寄与しないものは破棄する。
 - シミュレーション可能などんなモデルでも扱える。
- デッドロジック検出はできない。



SMTベースモデル検査によるテスト生成 (PROPMT-V3)

- 「Simulinkモデルの振る舞い」と「目的状態に到達しないこと」をSMT-LIB記述にエンコードし、有界モデル検査とk-Inductionでテスト生成orデッドロジック証明をする。



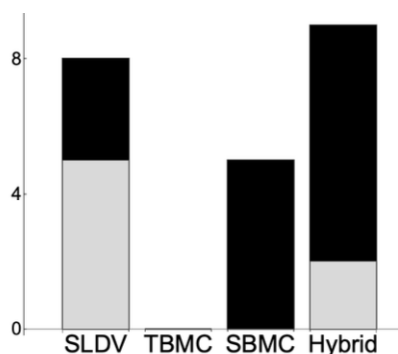
評価実験

• カバレッジ達成率を比較

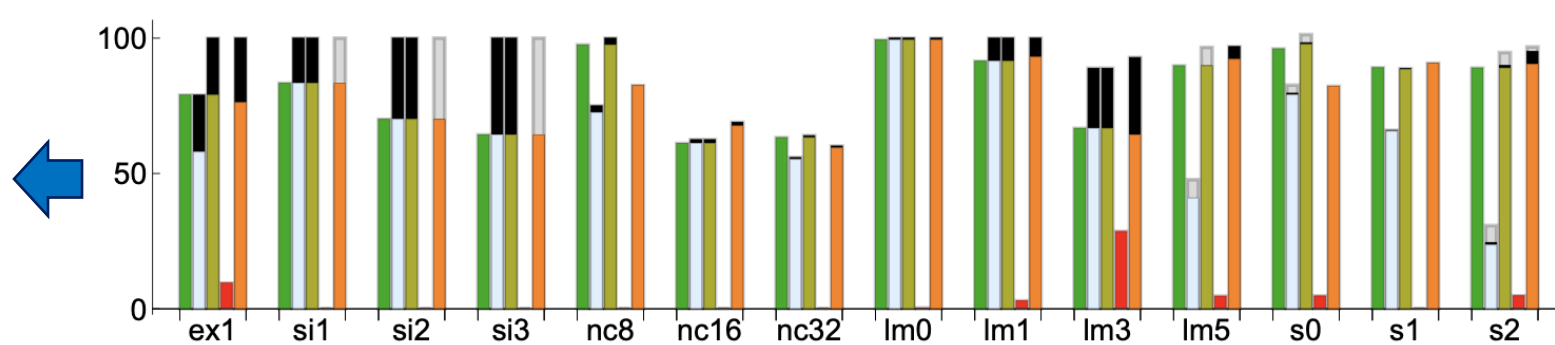
- ① テンプレートベースモンテカルロ法
- ② SMTベースモデル検査
- ③ テンプレートベースモンテカルロ法とSMTベースモデル検査のハイブリッド
- ④ SLDV

PROMPTでは未実装
(切替タイミング決定法に課題)

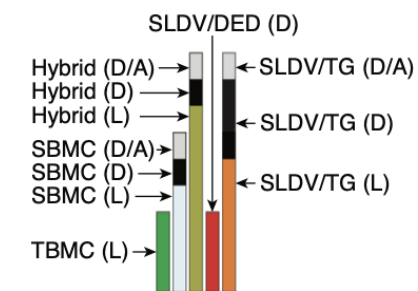
⇒ ①と②を組み合わせることで高カバレッジが達成できることを確認



フルカバレッジモデル数



カバレッジ



これまでの取り組み：まとめ

- 現代的車載システムに搭載されるECUの開発では、MATLAB/Simulinkを用いたモデルベース開発が行われている。
- MATLAB/Simulink向けカバレッジテストツールPROMPTをリリース
 - 大規模・複雑なモデルに対して効率的に高カバレッジテストスイートを生成するために、モンテカルロ法と形式手法を採用。

CRESTプロジェクト紹介：
次世代車載システムの形式的検証
の前に

次世代車載システム

- ここでは、基本的には**自動運転システム（ADS）**を指す。
 - 出典：SAE J3016:2021, JASO TP 18004:2022
- 用語
 - DDT（動的運転タスク）
 - 車両の操作に必要な、リアルタイムで実行するすべての操作上・戦術上の機能
 - OEDR（対象物・事象検知及び応答）
 - DDTのうち、運転環境の監視（物・事象の検知/認識/応答準備）と応答実行
 - ODD（運行設計領域）
 - ある運転自動化システム又はその機能が機能するように設計されている作動条件

表 1—運転自動化レベルの概要

レベル	名称	定義（口語表現）	DDT ⁽¹⁾		DDT のフォールバック	ODD	
			持続的な横・縦の車両運動制御	OEDR			
運転者が DDT の一部又は全てを実行							
運 転 支 援	0	運転自動化なし	運転者が全ての DDT を実行。 （予防安全システムによって支援されている場合も含む）	運転者	運転者	運転者	適用外
	1	運転支援	運転自動化システムが DDT の縦方向又は横方向のいずれか（両方同時ではない）の車両運動制御のサブタスクを特定の ODD において持続的に実行。 この際、運転者は残りの DDT を実行することが期待される。	運転者及びシステム	運転者	運転者	限定的
	2	部分運転自動化	運転自動化システムが DDT の縦方向及び横方向両方の車両運動制御のサブタスクを特定の ODD において持続的に実行。 この際、運転者は DDT のサブタスクである対象物・事象の検知及び応答を完了し、システムを監督することが期待される。	システム	運転者	運転者	限定的
ADS が（作動時は）全ての DDT を実行							
自 動 運 転	3	条件付運転自動化	ADS が全ての DDT を ODD において持続的に実行。 この際、フォールバックの準備ができている利用者は、他の車両システムにおける DDT 実行に関連するシステム故障だけでなく、 ADS が出した介入の要求を受け入れ、適切に応答することが期待される。	システム	システム	フォールバックの準備ができている利用者（フォールバック中運転者になる）	限定的
	4	高度運転自動化	ADS が全ての DDT 及び DDT のフォールバックを ODD において持続的に実行。 利用者が介入する必要があるということは期待されない。	システム	システム	システム	限定的
	5	完全運転自動化	ADS が全ての DDT 及び DDT のフォールバックを持続的かつ無制限に（すなわち、 ODD に依存しない）実行。 利用者が介入する必要があるということは期待されない。	システム	システム	システム	限定なし

⁽¹⁾ DDT には、目的地の決定及び移動すべきタイミングなどの、運転タスクの戦略的側面を含まない。

次世代車載基盤システム (1)

- Autoware

- ROS2の上に構築されるオープンソース自動運転ソフトウェア

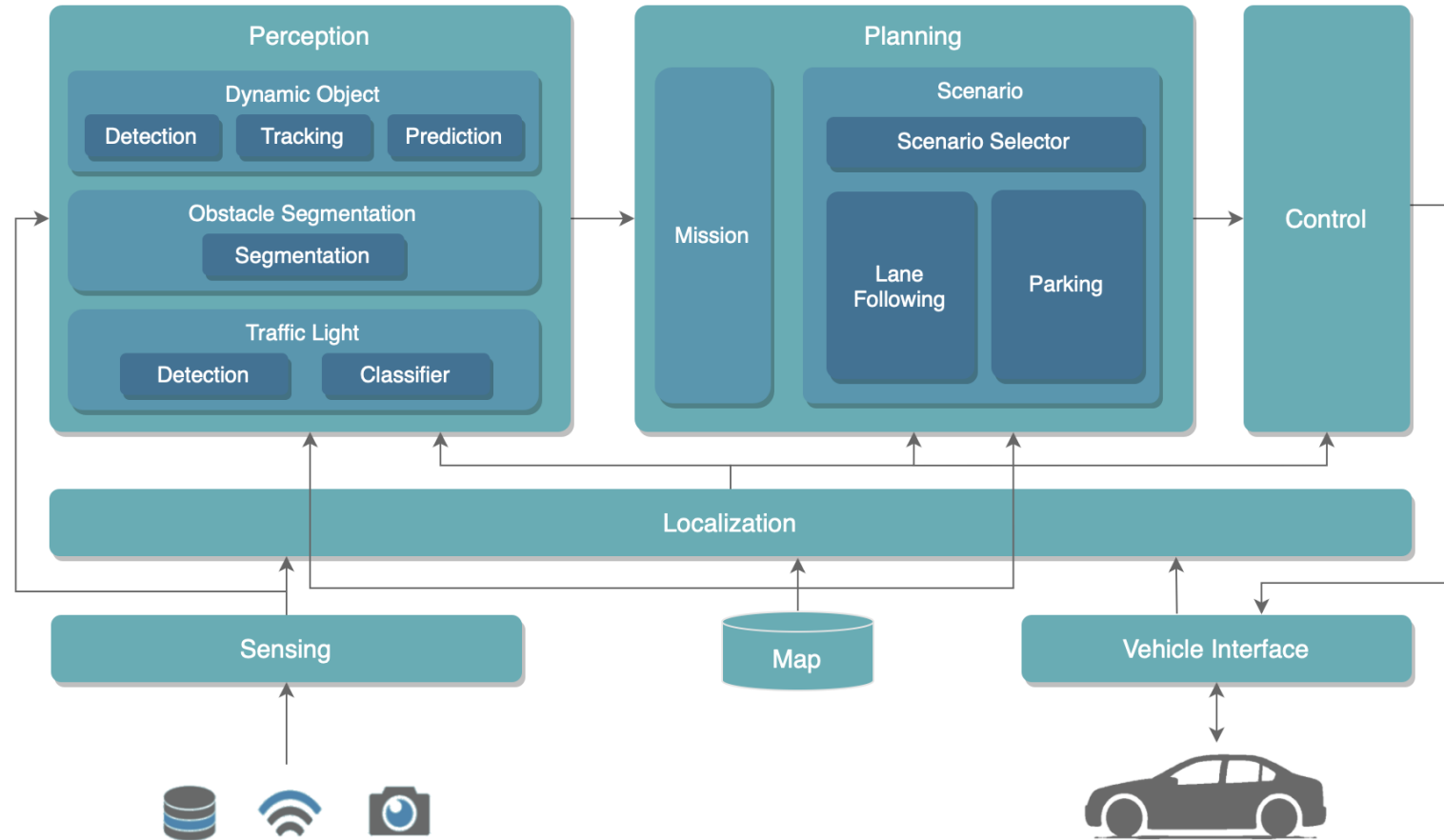
- ティアフォー社とAutoware Foundationが開発を主導

- Autoware Core

- 自動運転システムの基本機能 & IF
 - 厳格に品質管理

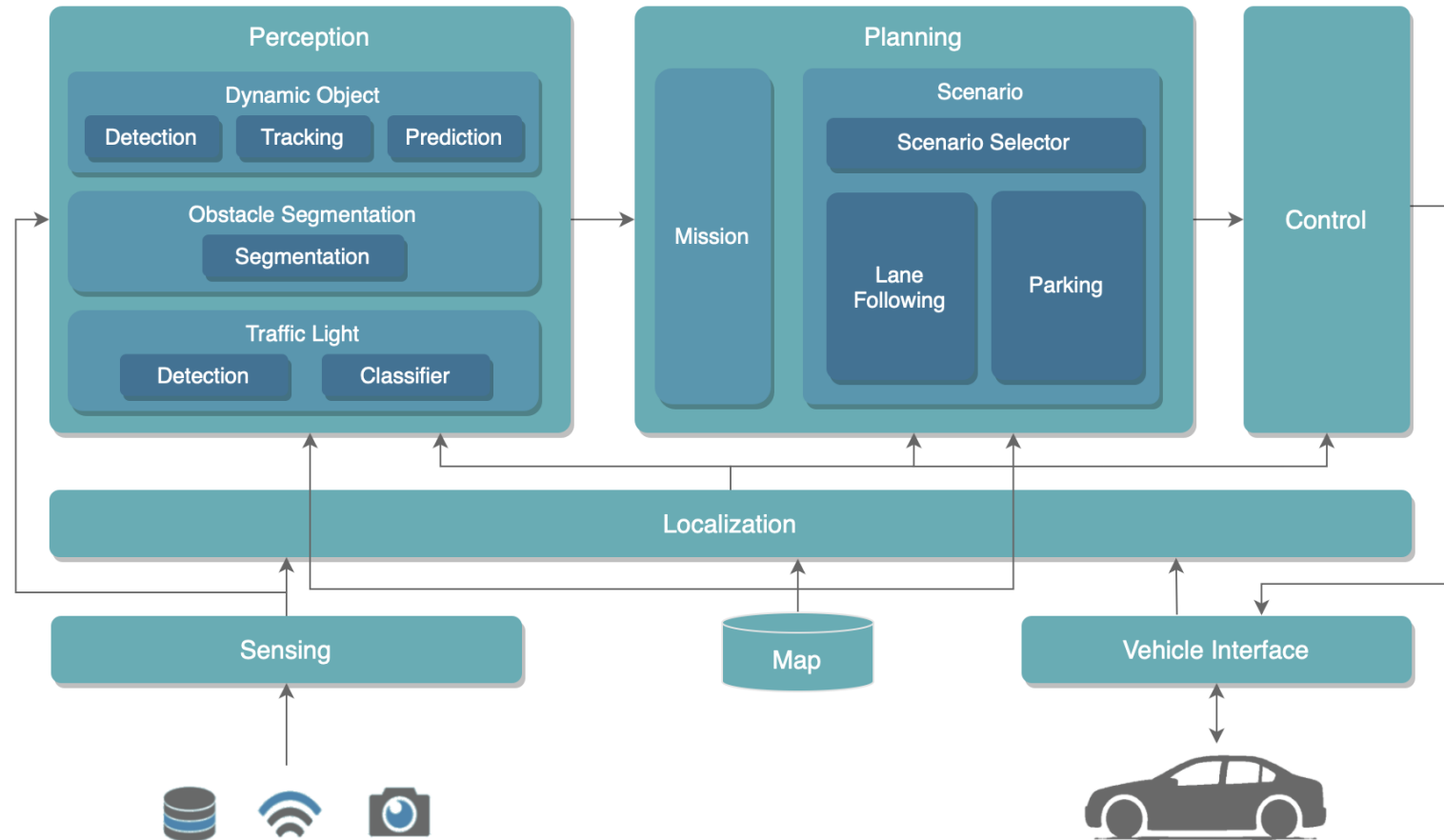
- Autoware Universe

- Coreの拡張
 - 実験的な機能を含む



次世代車載基盤システム (2)

- 認識/Perception
 - センサーデータから，物体の種類・位置等を判定する.
- 計画/Planning
 - 認識状況をもとに，車両が辿るべき軌道を定める.
 - 軌道 = 空間座標・方向・速度の点列
- 制御/Control
 - 軌道を辿るための方向・速度を算出する.
 - その方向・速度からスロットル/ブレーキ/ステアリング等の制御へのブレークダウンは車両ECU側で行う.



高レベル自動運転の安全性保証における技術的課題

- 安全の定義困難性
 - 基本的にはAI部分を含む.
 - なにをもって／どうやってAI部分の安全性を保証するのか？[認識・計画]
 - 常に徐行/停止していれば当然安全だが、実用的ではない.
 - どうやって実用性とのバランスを取るのか？[計画・制御]
- 走行環境の網羅性・充分性
 - ODDが限定的でも走行環境（歩行者・他者の存在/挙動，路面等の状態，外乱）には正に無数の可能性がある.
 - どうやって網羅性・充分性を担保するのか？[認識・計画・制御]
- 基盤レベルでの信頼性
 - システムが高度化するほど要求される計算能力が増加する.
 - リアルタイム性と信頼性を保証できるか？[基盤]

国際的取組みの状況

- シナリオベースの安全性検証

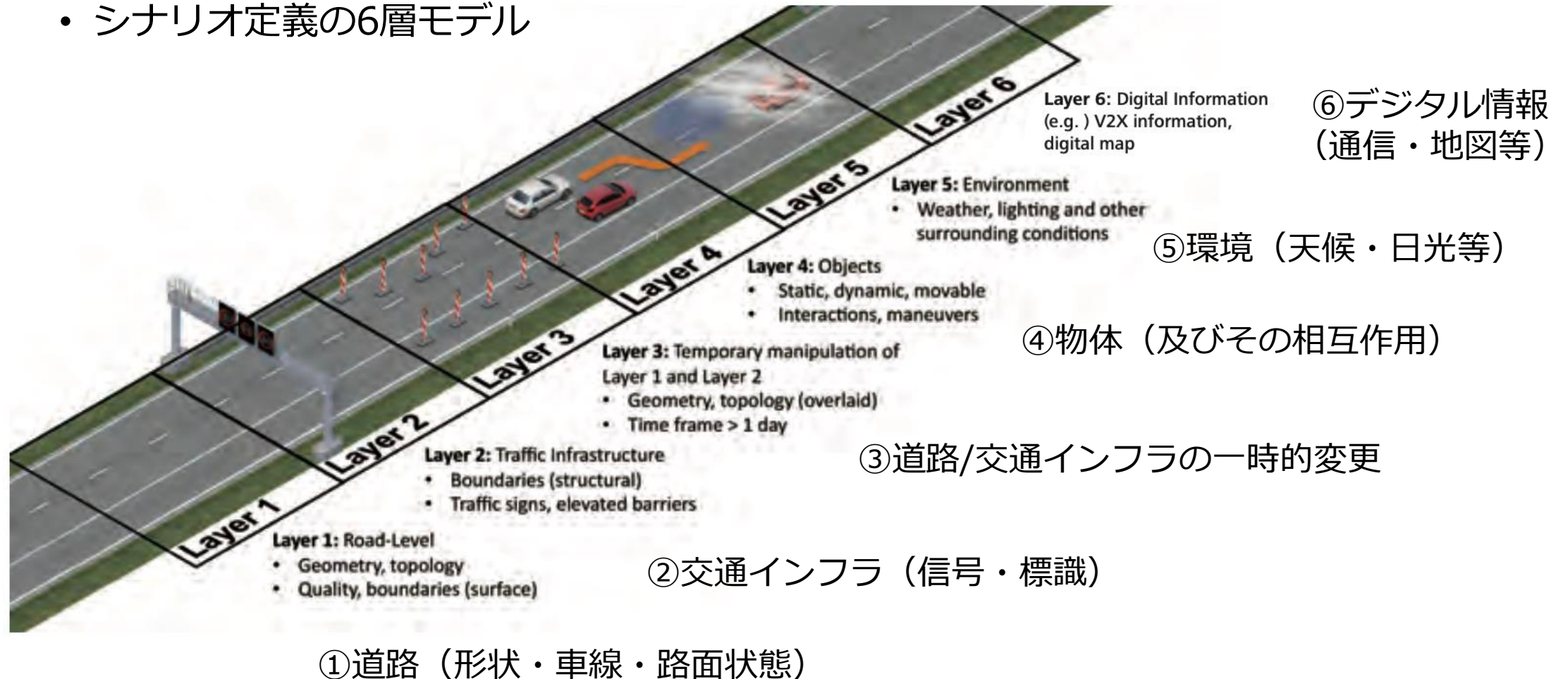
- A Framework for Automated Driving System Testable Cases and Scenarios (2018, NHTSA)
- PEGASUSメソッド (2019, PEGASUS Project)
- ISO34502:2022
 - 自動運転の安全性評価フレームワーク Ver3.0 (2022, JAMA)
 - SAKURAプロジェクト

- Safety Caseに基づく安全性論証

- ANSI/UL 4600:2023

シナリオベースの安全性検証（1）

- PEGASUSメソッド
 - シナリオ定義の6層モデル



シナリオベースの安全性検証（2）

- PEGASUSメソッド
 - 3抽象レベルの段階的詳細化
 - 実交通実態データを収集してパラメータ空間を導出

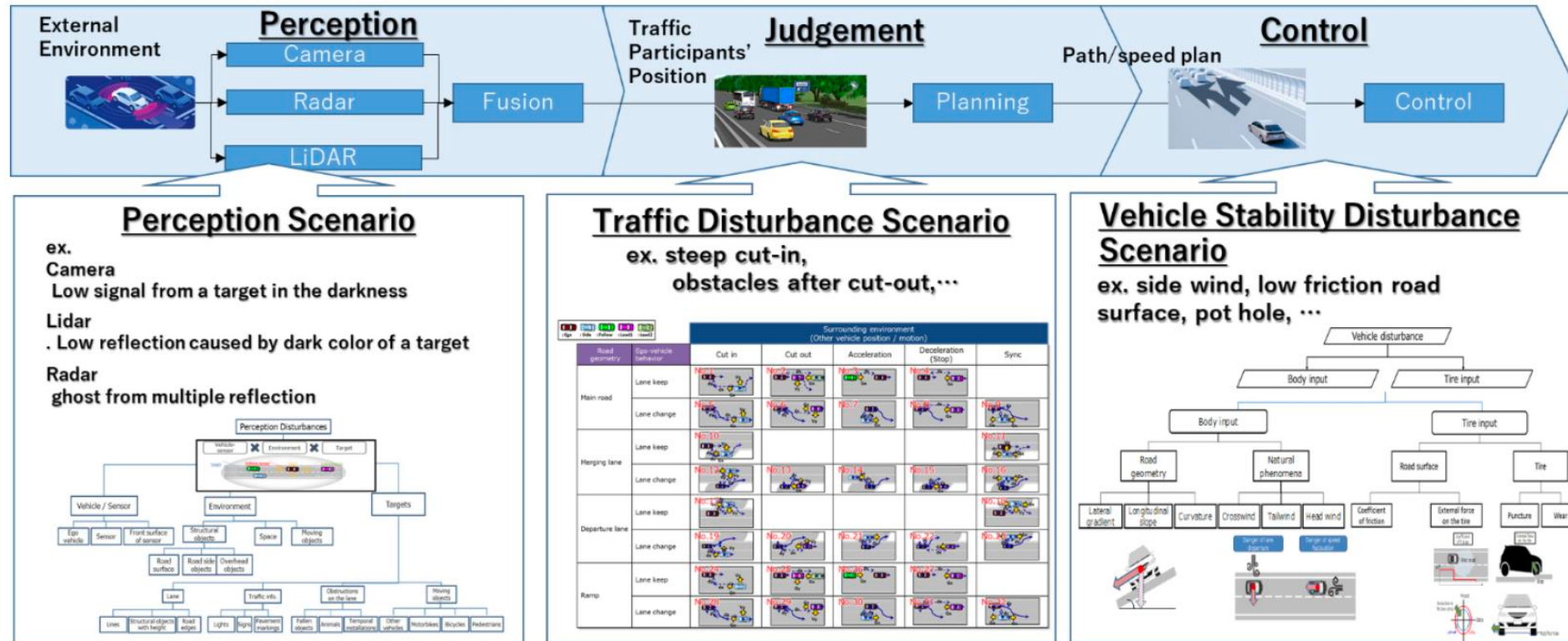
Functional scenarios	Logical scenarios	Concrete scenarios
<u>Base road network:</u> three-lane motorway in a curve, 100 km/h speed limit indicated by traffic signs	<u>Base road network:</u> Lane width [2.3..3.5] m Curve radius [0.6..0.9] km Position traffic sign [0..200] m	<u>Base road network:</u> Lane width [3.2] m Curve radius [0.7] km Position traffic sign [150] m
<u>Stationary objects:</u> -	<u>Stationary objects:</u> -	<u>Stationary objects:</u> -
<u>Moveable objects:</u> Ego vehicle, traffic jam; Interaction: Ego in maneuver „approaching“ on the middle lane, traffic jam moves slowly	<u>Moveable objects:</u> End of traffic jam [10..200] m Traffic jam speed [0..30] km/h Ego distance [50..300] m Ego speed [80..130] km/h	<u>Moveable objects:</u> End of traffic jam 40 m Traffic jam speed 30 km/h Ego distance 200 m Ego speed 100 km/h
<u>Environment:</u> Summer, rain	<u>Environment:</u> Temperature [10..40] °C Droplet size [20..100] μm	<u>Environment:</u> Temperature 20 °C Droplet size 30 μm

Level of abstraction

Number of scenarios

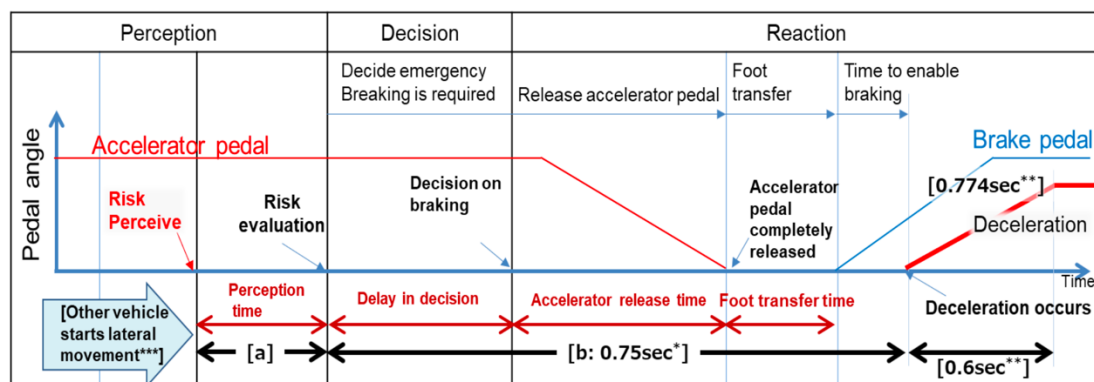
シナリオベースの安全性検証（3）

- JAMA自動運転の安全性評価フレームワーク Ver3.0
 - 3要素（認識・判断・操作）とその外乱に基づくシナリオ体系化
 - ODD体系-シナリオ体系間のマッピング



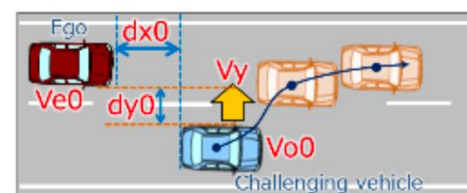
シナリオベースの安全性検証 (4)

- JAMA自動運転の安全性評価フレームワーク Ver3.0
 - 有能で注意深い運転者 (C&Cドライバ) モデル
 - C&Cドライバと同程度以上の安全性を持つことを検証する.

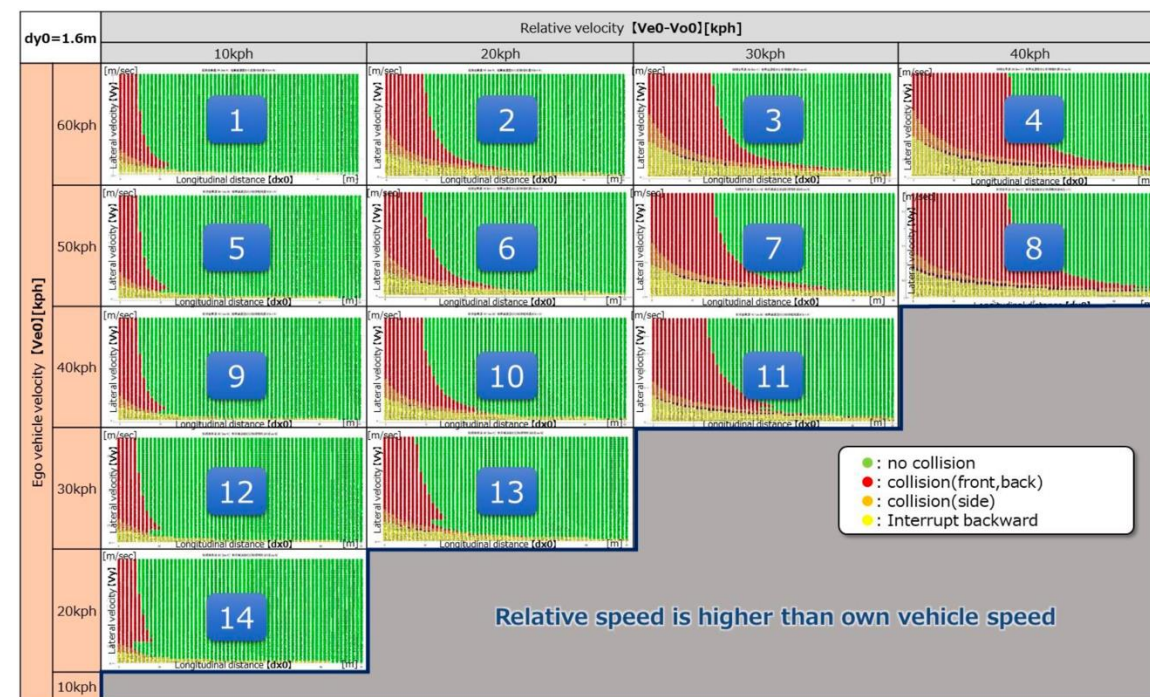


C&Cドライバモデル

算出



Initial condition	Initial velocity	[Ve0]	Ego vehicle velocity
		[Ve0-Vo0]	Relative velocity
Initial distance		[dy0]	Lateral distance [※]
		[dx0]	Longitudinal distance
Vehicle motion	Lateral motion	[Vy]	Lateral velocity



回避可能境界 (Preventable boundary)

CRESTプロジェクト紹介： 次世代車載システムの形式的検証

プロジェクト概要（当初）

- JST戦略的創造研究推進事業
CRESTプロジェクト「次世代車
載基盤システムのための形式手法
と検証ツールの創出」（2023年
10月-2029年3月）

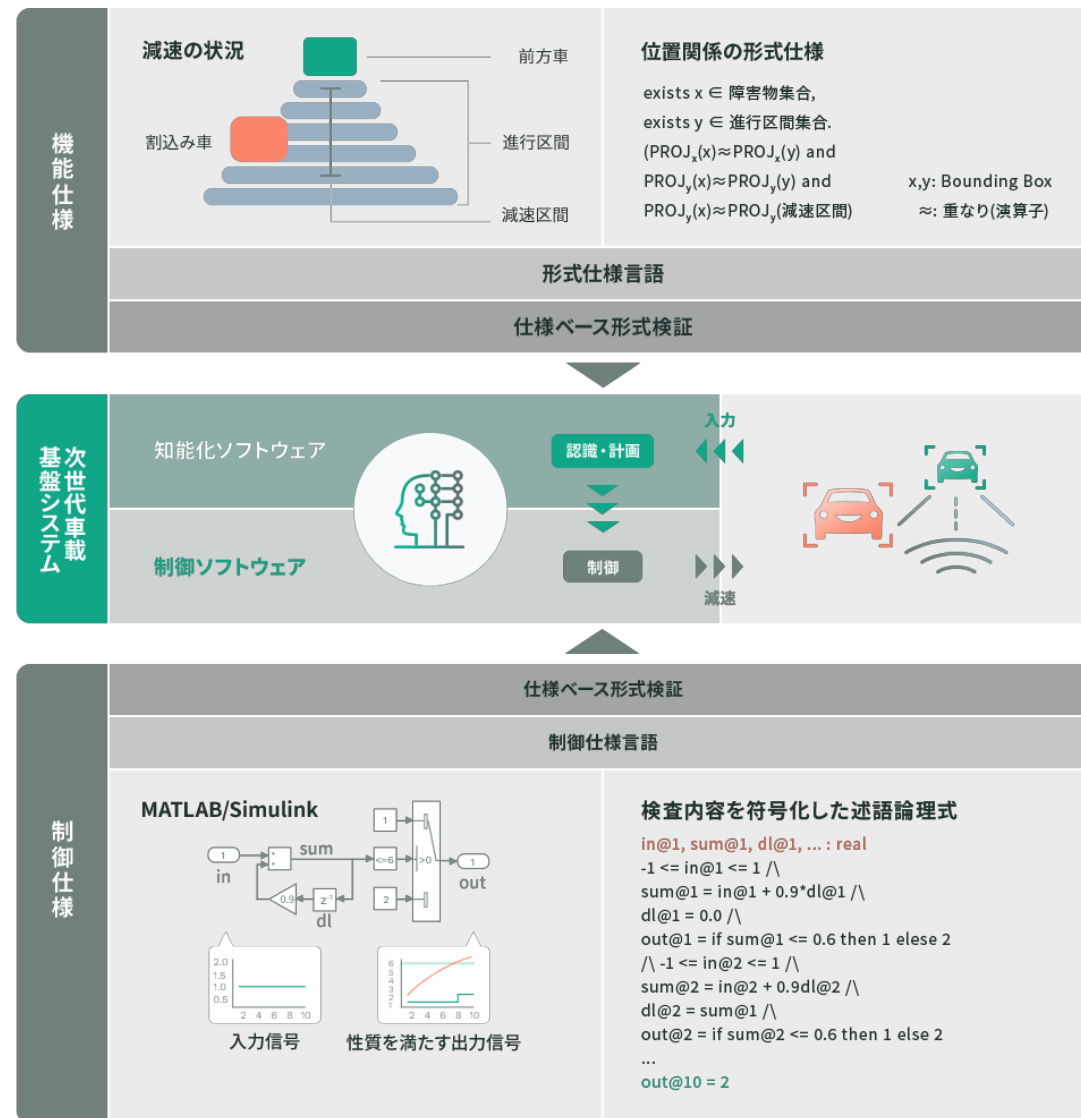
- 次世代車載基盤システムの安全性・
信頼性を保証する形式手法および検
証ツールを提案
 - 認識から制御までを一気通貫してカ
バーし，形式仕様言語，制御仕様言
語，それらを用いた形式検証手法で
構成
 - 自動化可能な部分を検証ツールとし
て実現

⇒自動運転を支える基盤の安全性と信
頼性の飛躍的向上，安全安心なモビ
リティ社会の実現を目指す

上流

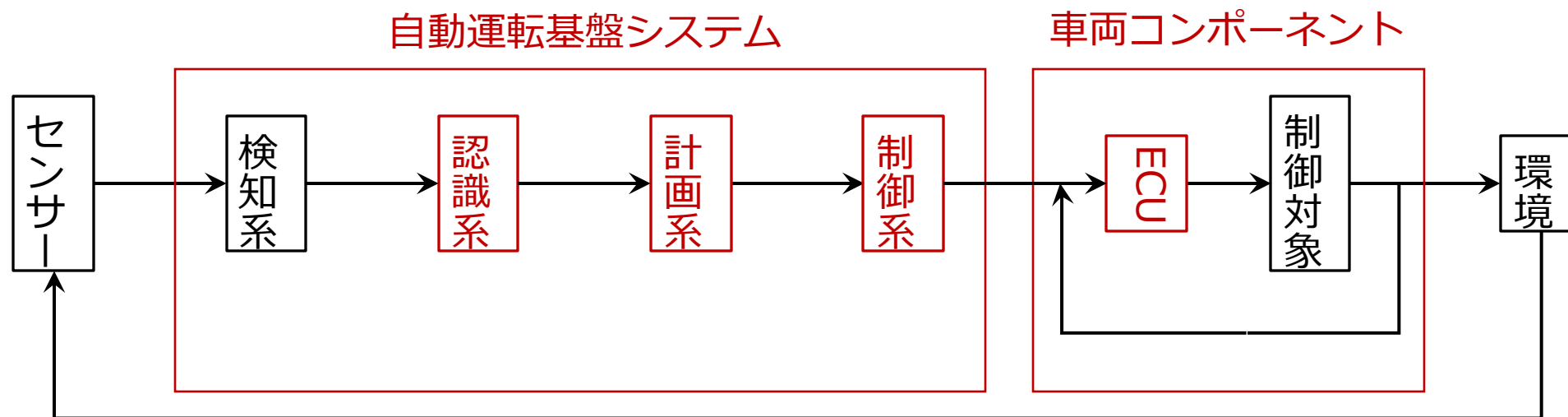


下流



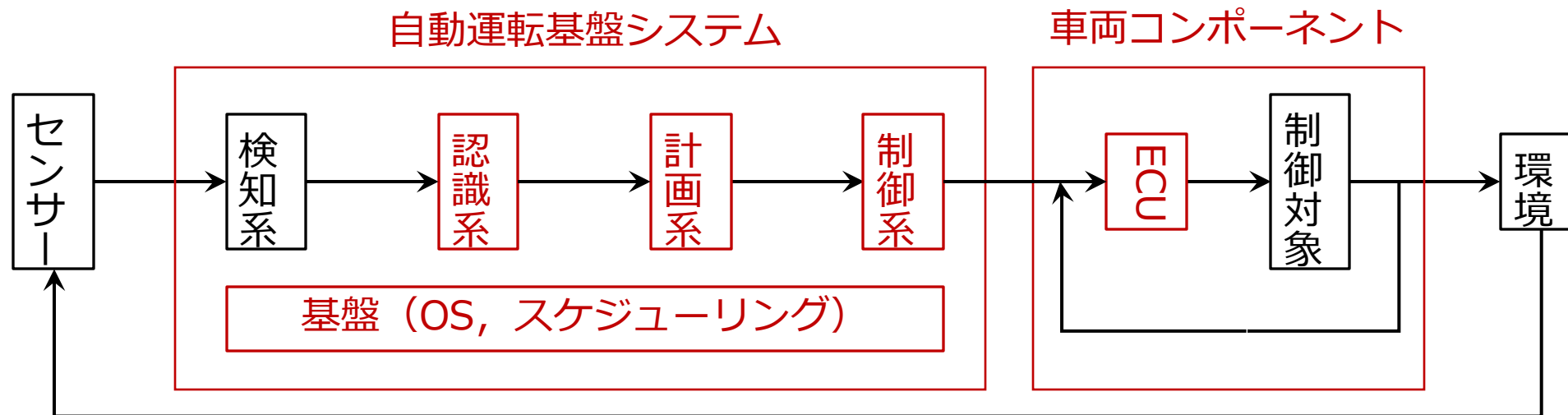
プロジェクト体制（2023年10月当初）

- 青木グループ
 - 研究代表者：青木利晃 教授@JAIST
 - 担当：認識・計画部分の検証
- 石井グループ
 - 主たる共同研究者：石井大輔 准教授@JAIST
 - 担当：制御（+計画）部分の形式検証
 - 車両コンポーネントの制御も含む.
- 富田グループ
 - 主たる共同研究者：富田堯 准教授@JAIST
 - 担当：制御（+計画）部分の統計的検証
 - 車両コンポーネントの制御も含む.



プロジェクト体制（2025年4月～）

- 青木グループ
 - 研究代表者：青木利晃 教授@JAIST
 - 担当：認識・計画・基盤部分の検証
- 石井グループ
 - 主たる共同研究者：石井大輔 准教授@JAIST
 - 担当：制御（+計画）部分の形式検証
 - 車両コンポーネントの制御も含む。
- 富田グループ
 - 主たる共同研究者：富田堯 准教授@JAIST
 - 担当：制御（+計画）部分の統計的検証
 - 車両コンポーネントの制御も含む。
- 安積グループ
 - 主たる共同研究者：安積卓也 教授@埼玉大学
 - 担当：基盤部分の検証
 - ティアフォーも参加



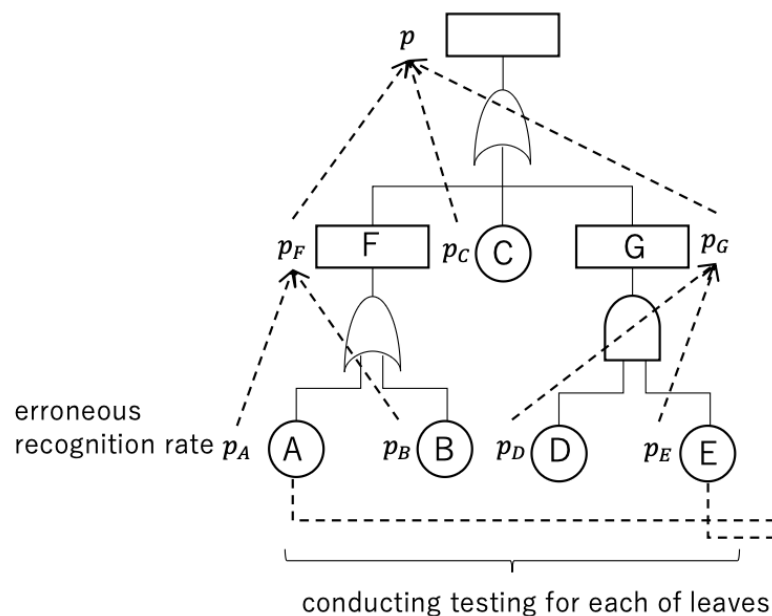
FT4D (Fault Tree for Datasets)

- 希少事象の誤り率より，頻出事象の誤り率の方が安全性に対するインパクトが大きい。

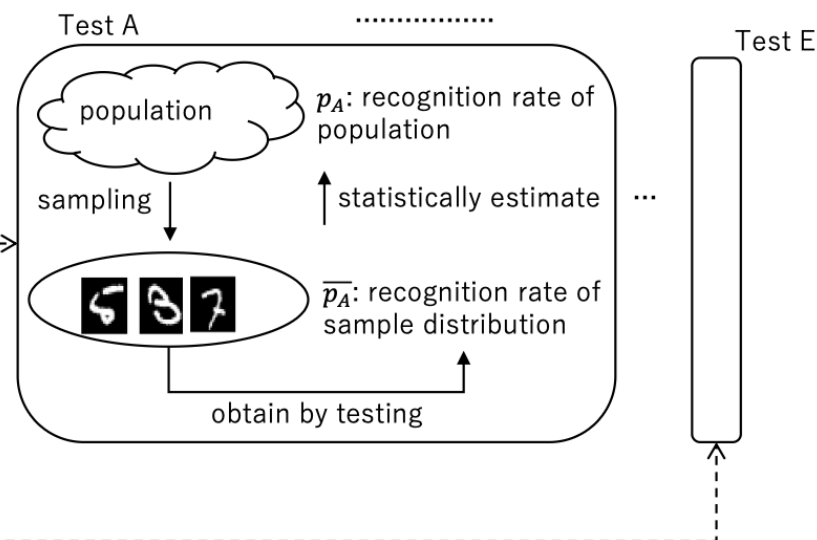
⇒故障木（Fault tree）解析を拡張して，データセットを扱えるように。

- テストデータセットは真の母集団の極一部なので，統計的推定に基づく補正を行う。

systematic fault analysis with fault tree for datasets



statistically evaluate test results

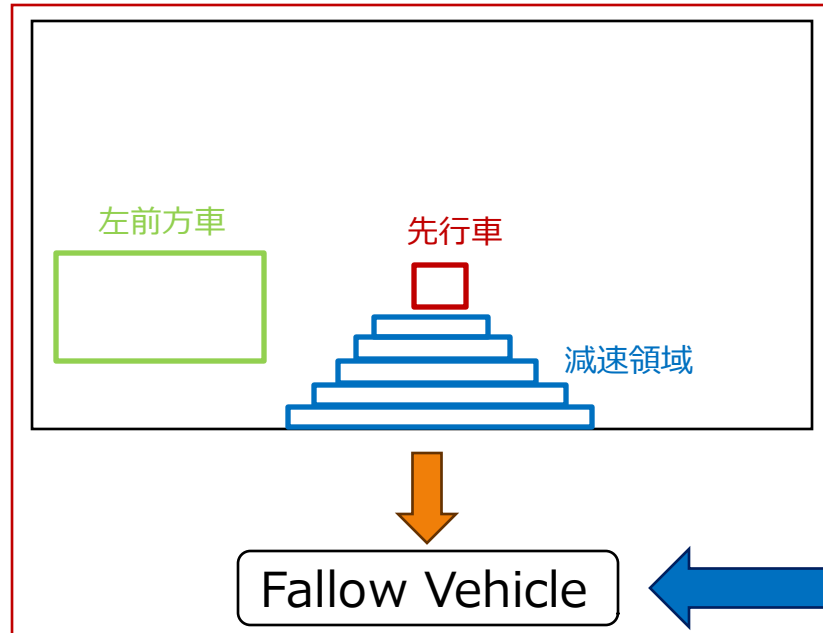


BBSL (Bounding Box Specification Language)

- あるシナリオにおける「BB (Bounding Box) の位置関係等」と「そのシーンで期待される応答」の関係を記述
 - 認識系の、自動運転システムの一部として期待する認識機能テストが可能になる
⇒ 点群データや3次元BB, 角度/速度情報等に対応するための拡張を検討中

BBSL仕様

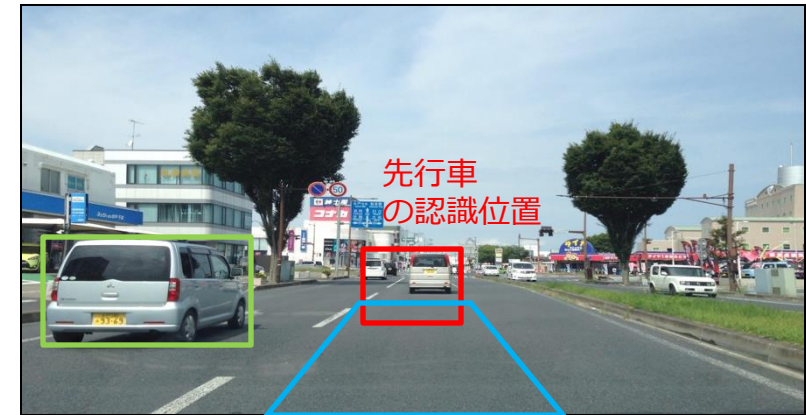
```
1 exfunction
2   vehicleExists():bool
3   vehicle():bb
4   stoppingDistance():interval
5 endexfunction
6
7 precondition
8   [vehicleExists() = true]
9 endprecondition
10
11 case stop
12   let vehicle : bb = vehicle(),
13   stoppingDistance : interval = stoppingDistance()
14   in
15     PROJy(vehicle) ≈ stoppingDistance
16 endcase
17
18 case NOT stop
19   let vehicle : bb = vehicle(),
20   stoppingDistance : interval = stoppingDistance()
21   in
22     not(PROJy(vehicle) ≈ stoppingDistance)
23 endcase
```



認識差異

応答差異

カメラ画像



減速領域

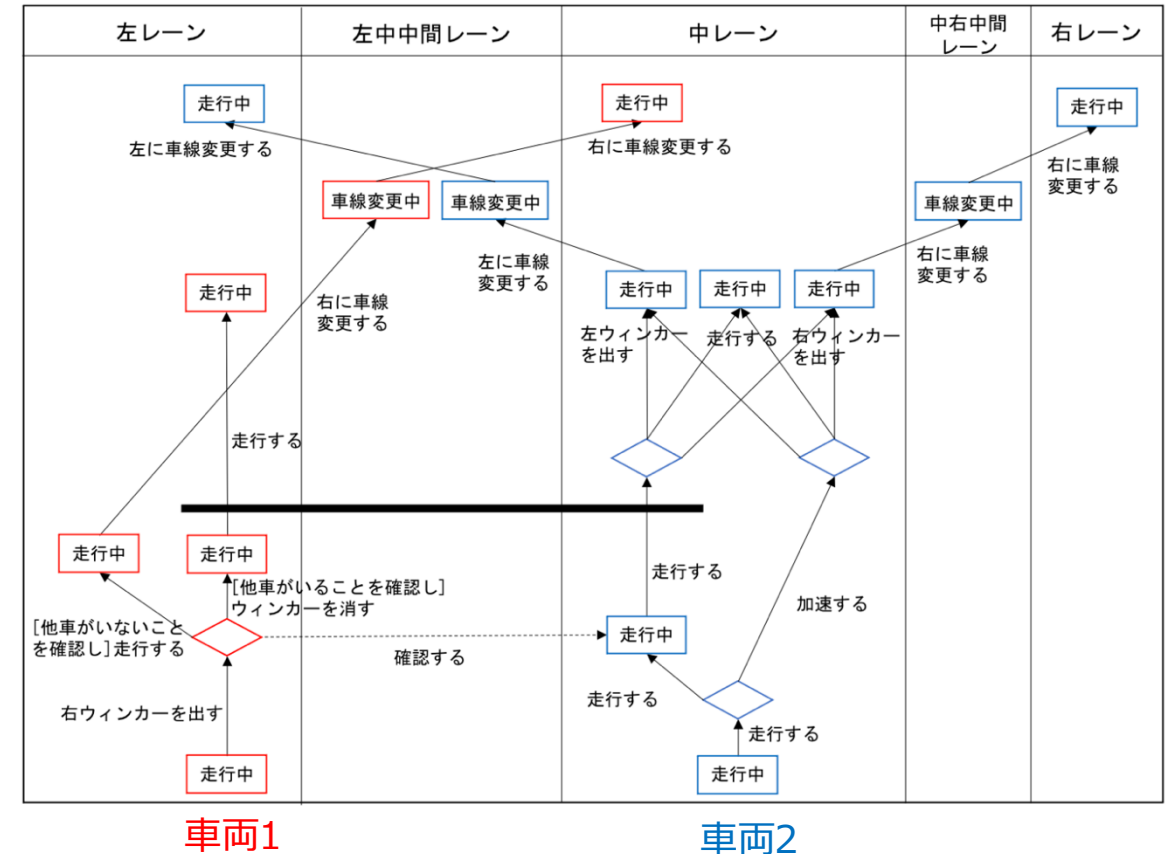
Deceleration



認識系
+
計画系

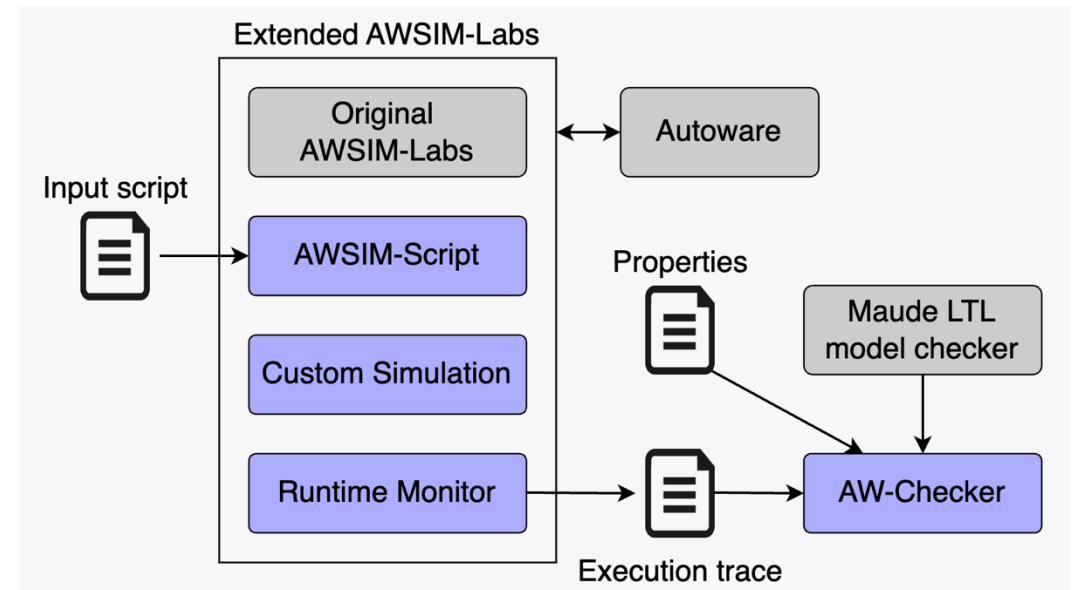
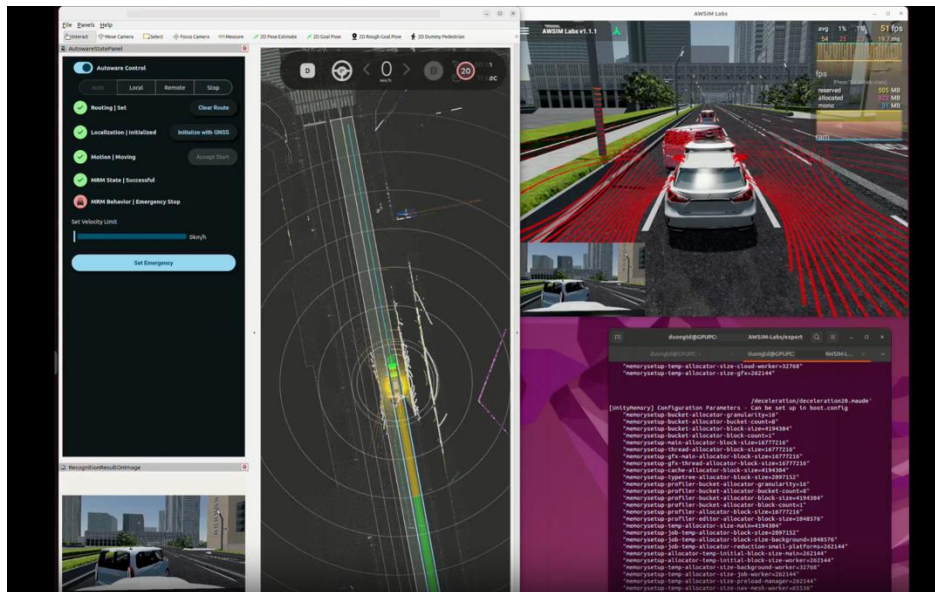
CPD (Car Position Diagram)

- スイムレーンをもつUMLアクティビティ図から着想を得たCPDによって、機能シナリオ群をコンパクトに表現
 - 道路・物体等の空間的配置とその変化を記述
 - 認識による条件付き遷移, 同期動作等も含む
 - SAT/SMTソルバ等を用いてシナリオを列挙し, 衝突シーンの検出等が可能
 - シーン = 移動物の位置の組み合わせ
 - シナリオ = 移動物の始点から終点までの遷移の組み合わせ
- ⇒ 具体的シナリオ (や実走行データ) とのマッピング法を検討中



AW-Checker

- シミュレーション環境AWSIM-Labsを拡張し，様々な具体的シナリオの振る舞いを検証できるように.
 - Autoware Universeについて，JAMAフレームワークに基づく高速時のシナリオの一部で，回避されるべき衝突を検出.
- ⇒より柔軟にシナリオ再現できるように，更なる拡張を検討
 - 現状，作成したシナリオをシミュレータ上で再現することが難しいケースがままある.

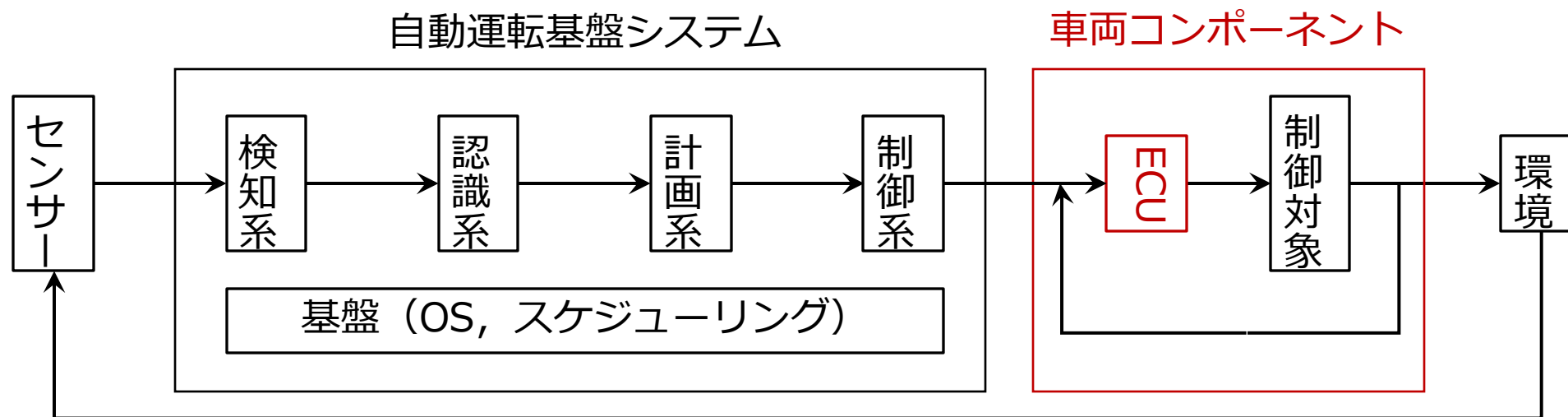


PROMPTの改良・拡張

- これまで、車両コンポーネントのECUモデルのカバレッジテスト生成の効率を向上させるため、継続的な改良・拡張を行ってきた。

⇒今後も継続的な改良と拡張を継続する。

- モンテカルロ法と形式手法の高度な連携
- テストケース生成/デッドロジック判定（= 到達可能性検証/不変性検証）からより高度な機能テスト/検証への拡張？
- 各種制御指標の統計的検証への対応？
- etc.



Awkernel

- 自動運転システム向けに Rust で実装された、非協調的な async/await 機構を備えたマイクロカーネルスタイルの単一メモリアドレス空間 OS

- 高いリアルタイム性と安全性を両立
- まずは安全機構 (Minimum risk maneuver; MRM) に適用予定

⇒ TLA+ / SPIN 等で形式検証を進めている

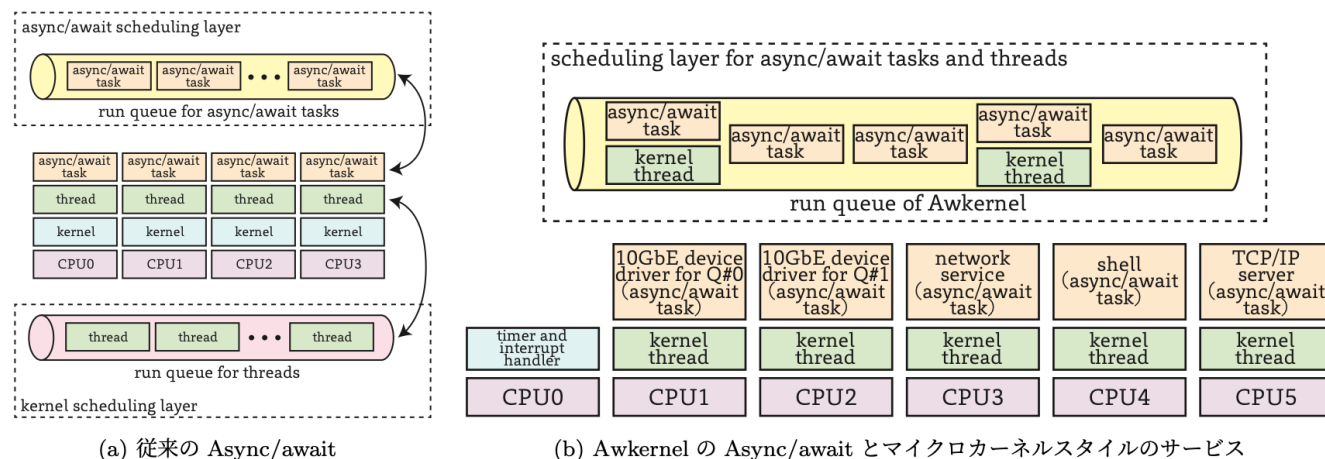


図 1: Awkernel と従来の Async/await の比較

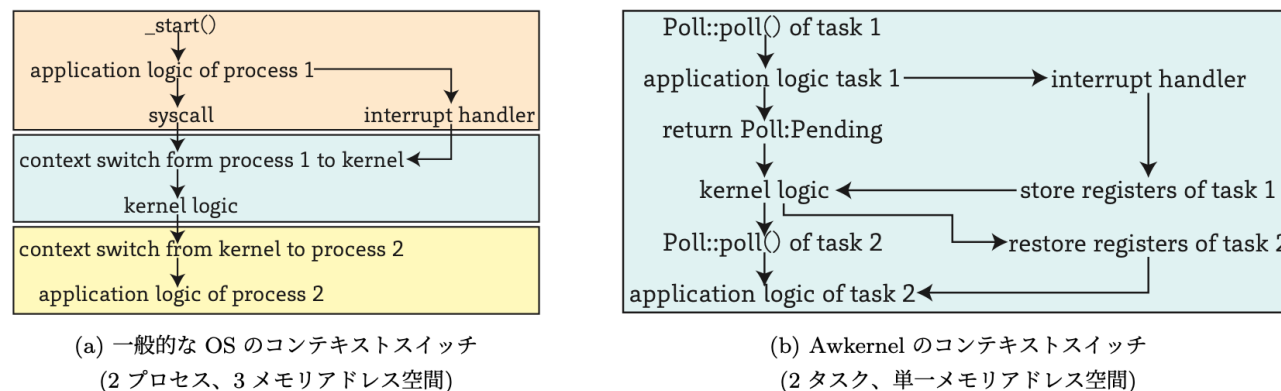


図 2: Awkernel と従来の OS のコンテキストスイッチの比較
(それぞれの四角い箱は仮想アドレス空間を表す)

高野&今井: Cyber Physical Systemsに適したリアルタイムOSの提案, IPSJ:2025-OS-166, 2025.

Hasegawa et al.: A Case Study on Model Checking and Runtime Verification for Awkernel, arXiv:2503.09282, 2025

確率的リアルタイムスケジューリング

- 自動運転システムには高いリアルタイム性が求められる.
 - 一方で, 大量のデータを処理し続ける必要がある.
 - 通常のリアルタイム性解析で用いられるWCET/WCRT (最悪実行時間/最悪応答時間) は実際の平均実行時間/平均応答時間より非常に大きい.
- しかし, 安全機構 (MRM) の存在を前提とすると, タスク完了がデッドラインに間に合わない確率が充分低いのであれば, 安全性と実用性の両立が保証できる.
 - ⇒ Probabilistic WCET/WCRT (確率的-) 等の概念に基づいた確率的リアルタイム性解析手法の提案・拡張を進めている.

まとめ

まとめ

- JST戦略的創造研究推進事業 CRESTプロジェクト「次世代車載基盤システムのための形式手法と検証ツールの創出」では,
 - 次世代車載基盤システムの安全性・信頼性を保証する形式手法および検証ツールを提案し,
 - 自動運転を支える基盤の安全性と信頼性の飛躍的向上, 安全安心なモビリティ社会の実現を目指している.
- ウェブサイト
 - プロジェクト : https://www.jaist.ac.jp/project/crest_fmap/
 - Github
 - FoMaAD: <https://github.com/fomaad>
 - Awkernel: <https://github.com/tier4/awkernel>
 - PROMPT: <https://www.gaio.co.jp/products/prompt-2/>

Q&A