2024 年（令和 6 年）4 月 10 日

ISO 34502 の自動運転車危険シナリオを数学的に定式化

～安全性保証タスクの自動化・効率化により自動運転の社会受容を促進～

情報・システム研究機構 国立情報学研究所（ＮＩＩ、所長：黒橋 禎夫、東京都千代田区）のアーキテクチャ科学研究系教授 蓮尾 一郎と、京都大学（総長：湊 長博、京都府京都市）大学院情報学研究科助教 和賀 正樹らの研究グループは、科学技術振興機構（ＪＳＴ、理事長：橋本 和仁、東京都千代田区）の戦略的創造研究推進事業 ERATO蓮尾メタ数理システムデザインプロジェクト(*1)（ERATO MMSD、研究総括：NII アーキテクチャ科学研究系教授 蓮尾 一郎）などのもと、自動運転車の安全性保証の枠組みである国際標準 ISO 34502 (*2) で示された危険シナリオ群について、その意味内容の数学的定式化を行いました。

本研究では、従来英語などの自然言語で記述された危険シナリオを「STL（シグナル時相論理、Signal Temporal Logic）(*3)」という形式言語で記述することで、解釈の違いが生じる可能性がある危険シナリオの意味内容を確定させ、また危険シナリオを用いた安全性評価タスクの自動化・効率化を可能にしました。本成果は自動運転車の安全性保証に貢献する成果であり、また同時に、情報システムと人間社会との間の「契約」たる要求仕様の活用において、数学が果たしうる重要な役割を指し示すものでもあります。

本研究成果を、情報学応用に関する主要国際会議 The 39th ACM/SIGAPP Symposium On Applied Computing（SAC）2024 で 2024 年 4 月 9 日（中央ヨーロッパ時間）、発表しました。

【ポイント】

- 自動運転車の本格普及に向けて、広範・詳細な安全性保証による社会的信頼の樹立が急務である。
- この目的に向けて、ISO 34502 は自動運転車の危険シナリオ群を網羅的に定めている。しかし自然言語で記述されているため、意味内容に解釈の違いが生じる可能性があり、ソフトウェアツールによる機械的処理が容易ではなかった。
- 本研究では、「STL」という形式言語を用いて、ISO 34502 の危険シナリオ群を数学的に定式化した。これにより危険シナリオ群の意味内容を確定させ、モニタリングなどの安全性評価タスクの自動化・効率化が可能になった。

情報・システム研究機構 国立情報学研究所
京都大学 情報学研究科
科学技術振興機構（JST）

- 自動運転車の安全性保証に貢献する成果であり、また一般に、自動運転をはじめとする新技術の社会受容に向けて数学が果たすべき重要な役割を指し示す成果でもある。

【背景】

今後の社会に期待されている自動車の自動運転技術の普及のためには、自動運転車の安全性を高めるだけでは十分ではありません。高い安全性を社会に対して保証しそれを説明して、自動運転車を公道に受け入れてもらう必要があります。国内外でさまざまな安全性保証の枠組みが提案されていますが、その中でも ISO 34502 は、一般財団法人日本自動車工業会（以下、日本自動車工業会）の取り組みを起源とする日本発の安全性保証の枠組みです。

ISO 34502 では、自動運転車の動作を「認知」「判断」「動作」の 3 段階に分けたのち、各段階における危険要素を同定してそれらの組み合わせを考えることにより、自動運転車の危険シナリオを網羅的に列挙しています。これらの危険シナリオにおいて適切な安全行動をとれるかどうかを評価することにより、自動運転車の安全性を保証しようというのが同規格の考え方です。

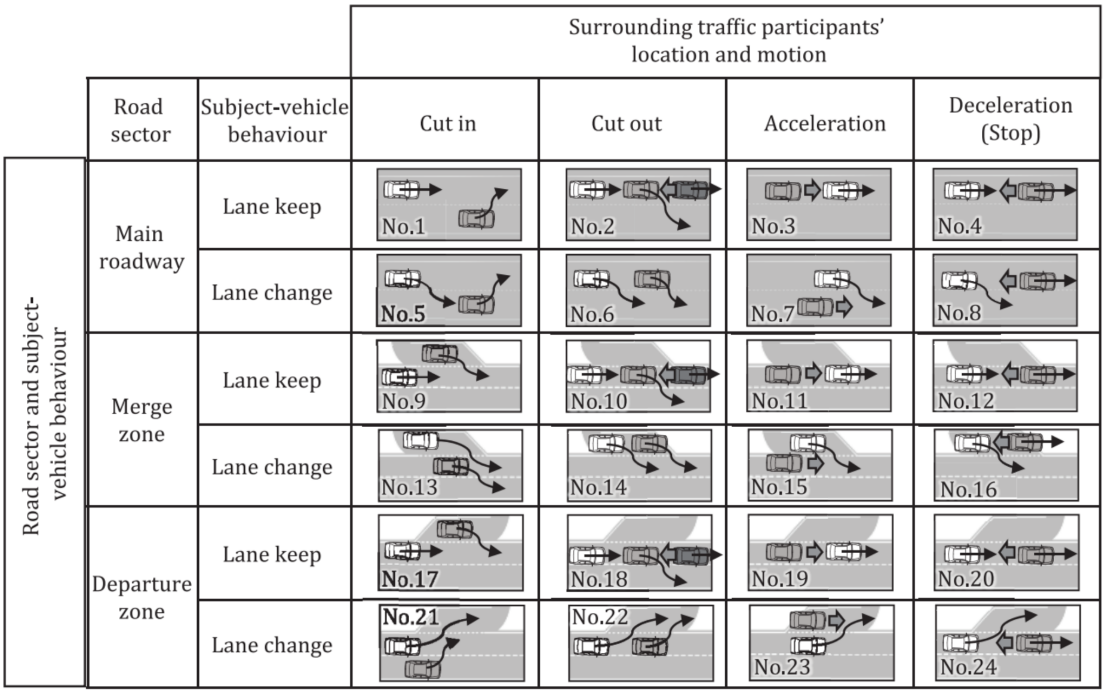
しかし ISO 34502 では、これらの危険シナリオが自然言語（英語）で記述されており、その大規模応用の妨げになっていました。その 1 つ目の問題は自然言語の曖昧さであり、たとえば「強引な割り込み」が正確には何を指すのかについて、解釈の違いが生じてしまっていました。

2 つ目の問題はソフトウェア処理の困難さです。危険シナリオ群を用いた安全性評価においては、モニタリング（走行データにおける危険シナリオの発生を検知する）や、テストデータ生成（危険シナリオが生じるような走行状況をシミュレーション用に生成する）などの安全性評価タスクを大量に実行する必要があり、ソフトウェアによる自動化が必須です。しかし自然言語で記述された危険シナリオに対しては、これらタスクを実行するソフトウェアをシナリオごとに改めて一から作成しなければならず、多大な労力がかかってしまいます。

【研究手法・成果】

本研究では上記の問題を解決するため、ISO 34502 の危険シナリオ群のうち、特に「判断」段階の危険要素に起因するシナリオ群（図 1 ）に対して、これらの数学的定式化を行いました。この定式化により危険シナリオそれぞれに「数学的定義」を与え、その意味内容を確定させました。

情報・システム研究機構 国立情報学研究所
京都大学 情報学研究科
科学技術振興機構（JST）



<図 1> ISO 34502 の危険シナリオ群のうち、特に「判断」段階の危険要素に起因するものの一覧。表は ISO 34502:2022 からの引用。

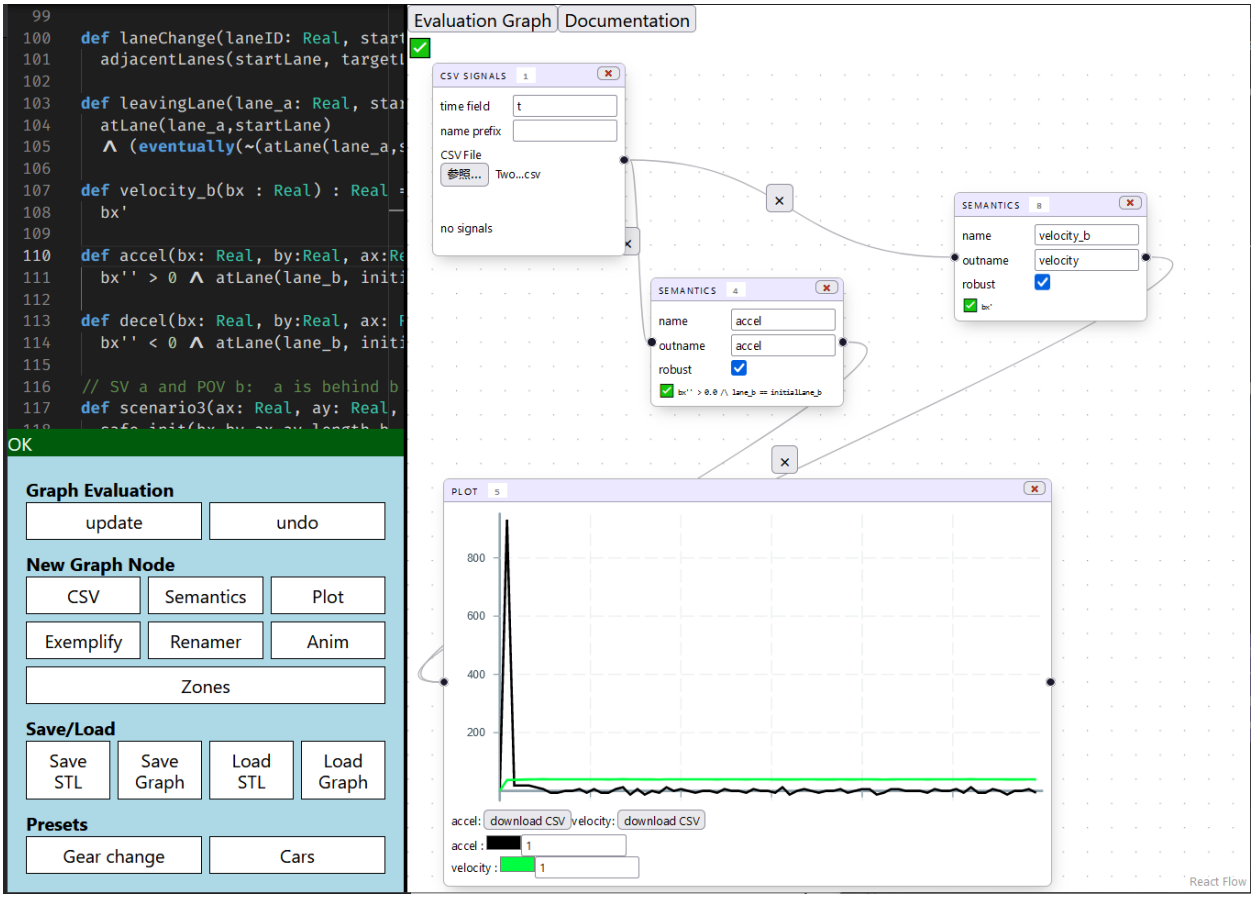
この数学的定式化（図２）のために STL を用いました。プログラムを書く際にプログラミング言語という形式言語を用いるのと同じように、STL という形式言語を用いて危険シナリオを記述していくことになります。そうすると、STL の語彙それぞれの意味がすでに数学的に定義されているため、危険シナリオの意味が数学的に定義できます。さらにこの定式化においては、本研究グループが開発中の対話型ツール「STL デバッガ」を用いて、記述した数学的内容が ISO 34502 のもともとの意図に合致しているかを確認しながら、定式化を進めていきました（図３）。

情報・システム研究機構 国立情報学研究所
京都大学 情報学研究科
科学技術振興機構（JST）

$scenario_i(SV, POV, L) \;:=\; initSafe(SV, POV) \wedge roadSector_i(SV, POV) \wedge disturb_i(SV, POV, L)$, $i = 1, \ldots, 24$ (cf. this is (1). *initSafe* is from §4.3)
$disturb_i(SV, POV, L) \;:=\; initialCondition_i(SV, POV, L) \wedge behaviourSV_i(SV, L) \wedge behaviourPOV_i(POV, SV, L)$, $i = 1, \ldots, 24$ (cf. (2) in §3)

i	$roadSector_i$ (cf. §4.1)	i	$initialCondition_i$ (cf.§4.2)	$behaviourSV_i$ (cf.§4.4)	$behaviourPOV_i$ (cf.§4.5)
$1-8$	$mainRoad(SV, POV)$	1	$\top$	$laneKeep(SV, L)$ $\mathcal{U}\, danger(SV, POV)$	$cutIn(POV, SV)$
		2	$sameLane_3(SV, POV_1, POV_2, L)$ $\wedge aheadOf(SV, POV_1)$ $\wedge aheadOf(POV_1, POV_2)$	$laneKeep(SV, L)$ $\mathcal{U}(\neg sameLane(SV, POV_1, L))$	$leavingLane(POV_1, L)$ $\wedge(laneKeep(POV_2, L)$ $\mathcal{U}(\neg sameLane(POV_2, POV_1, L)$ $\wedge danger(SV, POV_2)))$
		3	$aheadOf(POV, SV)$ $\wedge(sameLane(SV, POV, L)$ $\vee inAdjLanes(SV, POV, L))$	$laneKeep(SV, L)$ $\mathcal{U}\, danger(SV, POV)$	$accel(POV, SV, L)\ \mathcal{U}\ danger(SV, POV)$
		4	$aheadOf(SV, POV)$ $\wedge(sameLane(SV, POV, L)$ $\vee inAdjLanes(SV, POV, L))$	$laneKeep(SV, L)$ $\mathcal{U}\, danger(SV, POV)$	$decel(POV, SV, L)\ \mathcal{U}\ danger(SV, POV)$
		5	$\top$	$leavingLane(SV, L)$	$cutIn(POV, SV)$
		6	$\top$	$leavingLane(SV, L)$	$cutOut(POV, SV, L)$
		7	$aheadOf(POV, SV)$	$enteringLane(SV, L)$	$accel(POV, SV, L)\ \mathcal{U}\ danger(SV, POV)$
		8	$sameLane(SV, POV, L)$ $\wedge aheadOf(SV, POV)$	$leavingLane(SV, L)$	$decel(POV, SV, L)\ \mathcal{U}\ danger(SV, POV)$
$9-16$	$mergeZone(SV, POV)$	$9-16$	$initialCondition_{i-8}$	$behaviourSV_{i-8}$	$behaviourPOV_{i-8}$
$17-24$	$departZone(SV, POV)$	$17-24$	$initialCondition_{i-16}$	$behaviourSV_{i-16}$	$behaviourPOV_{i-16}$

<図 2 > 本研究の成果たる ISO 34502 危険シナリオ群の STL による数学的定式化の例。各危険シナリオ scenario$_i$（i=1,2,⋯,24）のテンプレートとその構成要素を表で示している。



<図 3 > 「STL デバッガ」のスクリーンショット。左上のテキスト部で入力した STL 論理式の意味内容を、右側の GUI 部で対話的に確認することができる。

情報・システム研究機構 国立情報学研究所
京都大学 情報学研究科
科学技術振興機構（JST）

STL を用いた数学的定式化は、上記の２つ目の問題も解決します。STL を入力フォーマットとしてモニタリングやテストデータを生成するアルゴリズムは（本研究グループのこれまでの研究成果も含め）多数存在しますが、今回の研究成果により、これらのアルゴリズムが ISO 34502 の安全性評価に適用できるようになります。

【今後の展望】

STL は製造業における広い応用が期待されている形式言語であり、STL に基づく品質保証ソフトウェアツールのエコシステムが形成されつつあります。今回の研究成果は、このソフトウェア・エコシステムと自動運転車の安全性保証のための枠組み（ISO 34502）の２つをつなぐものであり、両者の発展をさらに促進して、自動運転の社会受容のみならず、製造業の設計過程の自動化・効率化をも推し進めるものです。

一方、STL の産業界活用の１つの障壁として、STL に習熟した技術者でないと意図した内容を容易に記述できないという問題がありました。STL は決して難しい形式言語ではありませんが、それを学ぶためには新しいプログラミング言語を学ぶのと同様の学習過程が必要になります。今回用いた「STL デバッガ」は、一般的なプログラミング言語のデバッガと同様の役割を果たすものであり、STL の学習過程を助け、その産業界での活用を推し進めるものです。

本研究の定式化では、危険性を定義するために、RSS（責任感知型安全論、Responsibility-Sensitive Safety）[(*4)] 安全距離の概念を用いています。RSS は自動運転車の安全性を数学的に証明する方法論として注目を集めており、本成果を通じて RSS の活用の場面がさらに広がっていくことが期待されます。

より一般には、さまざまな情報システムの性質・要求仕様・動作シナリオなどを数学的に定式化することは、意味内容の明示化やデータ処理の自動化を可能とさせ、信頼性が高く、効率的な製品開発に貢献できるため、大きな産業的・社会的意義を持ちます。このような数学の社会応用の一つのあり方を広く発信し、またこれを支える技術とソフトウェアツールをさらに発展させることで、情報システムの信頼性樹立と社会受容を実現すべく、研究を続けていきます。

蓮尾　一郎　教授からのコメント：

「本研究の契機は三菱電機株式会社様との協働であり、STL による要求仕様の数学的定式化のケーススタディとして ISO 34502 を提案いただいたことにより、本成果が可能になりました。

自動運転システムや生成 AI など、新しい情報技術には『十分に安全か、社会に受け入れてよいほど安全か』という社会的信頼の問題が常につきまといます。その際、情報システムが満たすべき要求仕様は『社会との契約』であり、社会的信頼の基盤になるものですから、本研究のような数学

的定式化は非常に重要です。今後、情報技術と社会との関わりを整理し情報技術を安全に使役する人間中心の社会を実現するために、数学的技術の研究開発をさらに進めていきます。」

【研究プロジェクトについて】

本研究は、科学技術振興機構（JST） 戦略的創造研究推進事業 ERATO「蓮尾メタ数理システムデザインプロジェクト」（JPMJER1603）、JST 研究成果展開事業 大学発新産業創出プログラム START プロジェクト推進型 起業実証支援「ソフトウェア品質の論理的説明技術による、自動運転の本格普及の実現」（JPMJST2213）、JST 戦略的創造研究推進事業 CREST「AI 集約的サイバーフィジカルシステムの形式的解析設計手法」（JPMJCR2012）の一環で行われました。また本研究では、三菱電機株式会社情報技術総合研究所との協働も行いました。

【論文タイトルと著者】

タイトル：Temporal Logic Formalisation of ISO 34502 Critical Scenarios: Modular Construction with the RSS Safety Distance

著　　者：Jesse Reimann, Nico Mansion, James Haydon, Benjamin Bray, Agnishom Chattopadhyay, Sota Sato, Masaki Waga, Étienne André, Ichiro Hasuo, Naoki Ueda, Yosuke Yokoyama

発表会議：The 39th ACM/SIGAPP Symposium On Applied Computing (SAC) 2024

発 表 日：2024 年 4 月 9 日（中央ヨーロッパ時間）

〈メディアの皆様からのお問い合わせ先〉

大学共同利用機関法人 情報・システム研究機構 国立情報学研究所
総務部企画課 広報チーム
TEL：03-4212-2164　E-mail：media@nii.ac.jp

国立大学法人 京都大学
渉外部広報課国際広報室
TEL：075-753-5729
E-mail：comms@mail2.adm.kyoto-u.ac.jp

国立研究開発法人 科学技術振興機構（JST）
広報課
TEL：03-5214-8404　E-mail：jstkoho@jst.go.jp

〈JST の事業に関すること〉

国立研究開発法人 科学技術振興機構（JST）
研究プロジェクト推進部 ICT／ライフイノベーショングループ
今林 文枝
TEL：03-3512-3528　E-mail：eratowww@jst.go.jp

（*1） ERATO 蓮尾メタ数理システムデザインプロジェクト：国立研究開発法人 科学技術振興機構（JST）の「戦略的創造研究推進事業 ERATO」に採択されている研究プロジェクトで、Society 5.0 の大きな柱となる物理情報システム（CPS）の品質保証手法の学術的研究を推進している。特に、CPS の典型例の一つとして注目される自動運転システムを重点応用対象として、その信頼性保証を支えるモデリング手法・形式検証手法・テスト手法、さらにこれらを包括する実用的な 検証と妥当性確認（V&V）技術の研究開発に取り組んでいる。このような大きなチャレンジでは、ソフトウェア・制御・AI といった多様な学術分野の協働が必要となるため、学術分野融合の基礎となる数理的（メタ）理論も重視して研究を推進する。略称は ERATO MMSD。プロジェクト詳細は https://www.jst.go.jp/erato/hasuo/ja/参照。2022 年 3 月に本研究期間を終了し、現在は追加支援期間として研究を推進中（2025 年 3 月まで）

（*2） 国際標準 ISO 34502：日本自動車工業会の「自動運転の安全性評価フレームワーク」を起源とする日本発の自動運転車の安全性保証の枠組みの提案。詳細は経済産業省のニュースリリースを参照 https://www.meti.go.jp/press/2022/11/20221116006/20221111005.html

（*3） STL（シグナル時相論理、Signal Temporal Logic）：∧（かつ）、∨（または）などの演算子を持つ基本的な論理体系である命題論理に対し、$F_{[0,T]}$（これから T 秒以内に）、$G_{[0,T]}$（これから T 秒間ずっと）などの時相演算子を加えて拡張して得られた時変シグナルの性質を記述するために適した論理体系。O.Maler と D.Nickovic により 2004 年に導入された。

（*4） RSS（責任感知型安全論、Responsibility-Sensitive Safety）：Mobileye 社の研究者が最初に提唱した自動運転車の安全性に数学的証明を与えるための方法論。追突回避という特定の運転シナリオに適用すると、適切にブレーキをかけることで必ず追突を回避できるような車間距離（RSS 安全距離）の公式が得られる。2022 年の ERATO MMSD の研究の成果により、レーン変更や非常停止といった複雑な運転シナリオへも適用が可能となった。詳細は以下のニュースリリースを参照。 https://www.nii.ac.jp/news/release/2022/0707.html

情報・システム研究機構 国立情報学研究所
京都大学 情報学研究科
科学技術振興機構（JST）