研究ノート

System of Systemsの安全性論証の基本構造に関する提案

徳　田　昭　雄*
波多野　祥　二**
山　本　輝　俊***
浅　井　龍　男****

要　旨

インテリジェント交通システムやスマート都市構想など情報通信技術（Information Communication Technology: ICT）を活用する社会基盤システムへの期待感が高まりを見せている中で，その有力なアーキテクチャの候補として System of Systems（SoS）が議論されている。SoS は，自律的に振る舞うコンポーネントと，コンポーネント間の協調動作から構成されるシステムとして定義されるが，その導入と運用は社会が初めて経験するものである。加えて，SoS は，その「複雑さ」故に，従来の法体系や責任のとり方と整合的ではないと指摘する声が多い。私たちは，どのように SoS を構想し受容すれば良いのだろうか？　これが，SoS に関する包括的な問いである。

本稿では，先ず，SoS の複雑さの捉え方について検討し，次いで安全性に関する議論の先駆的事例とも言える ISO 21448（Road Vehicles-Safety of the intended functionality, SOTIF）と PEGASUS Research Project の考え方と，SoS への拡張に関する試論を通じてこの問いに答える準備を行う。

キーワード：System of Systems（SoS），複雑なシステム，リスク評価，安全性論証，アジャイルガバナンス

＊ 立命館大学経営学部　教授
＊＊ 株式会社 OTSL
＊＊＊ 株式会社 OTSL
＊＊＊＊ 立命館大学デザイン科学研究所　客員研究員

はじめに

インテリジェント交通システムやスマート都市構想など情報通信技術（Information Communication Technology: ICT）を活用する社会基盤システムへの期待感が高まりを見せている。このような基盤システムは，自律的に振る舞うコンポーネントと，コンポーネント間の協調動作実現される System of Systems（SoS）と呼ばれるアーキテクチャを採用する方向が示されている。しかし，SoS の導入と運用は社会が初めて経験するものであるため，どのようにこれに接し，コントロールするのか？できるのか？が，根本的な問いとして発せられている。これから発表を予定している研究ノート群は，この問いに関して，SoS の「安全性の論証」という視点から答えることを最終的な目的としているが，本稿では，その準備的段階として，リスク分析方法および分析に必要なデータに関する現状調査結果と分析結果を示す[1)]。現状調査及び分析は，以下の視点により実施している。

1） SoS の安全性論証の構造と，事故など危険発生時に，軽量なプロセスで安全性論証と信頼性確保ができる方法はどのようなものか？
2） Unknown の危険事象の判定方法はどのようなものか？
3） 安全性論証におけるシミュレーションを活用すべき箇所とシミュレーションに必要なデータはどのようなものか？

1. 安全性論証に関する先行的研究と本稿の問題意識

1-1. 安全概念に関する先行研究及び国際標準

安全概念は，本質安全，機能安全に区分して議論されることが一般的である。本質安全はリスク要因そのものの排除により安全を確保するというアプローチであり，「ロウソクは火災原因となる危険性があるから，ロウソクの使用を禁止する」などといった例が考えられる。機能安全は，コンピュータなどの電子機器を含んだ機器の追加により安全を確保するというアプローチであり，「ロウソクに監視装置をつけて，転倒あるいは類焼の可能性を探知した時に自動的に消火する」といった例を考えることができる。自動車の自動ブ

レーキなどが機能安全の具体的事例である。そして，近年，人と機械の協調動作環境の安全を考えるために協調安全という概念が提唱されている。協調安全[2]は，情報通信技術（ICT）の活用により，人・モノ・環境が情報を共有して安全を構築するアプローチであり，「ロウソクと人が近傍の可燃物情報を共有し，ロウソクの位置の決定や着火／消火の自律的制御を行う」などが事例として考えられる。

図表 -1 は，安全概念と標準規格の例を示示している。

図表 −1　安全概念と標準規格など

安全概念	主な標準規格など
本質安全	「ISO 12100（JIS B 9700）：機械類の安全性ー設計のための一般原則ーリスクアセスメント及びリスク低減」
機能安全	IEC61508” Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems (E/E/PE, or E/E/PES)” ISO26262 “Road vehicles – Functional safety”
協調安全(Safety2.0)	IEC MSB,『IEC White Paper Safety in the future：2020』(参考文献)

出所）徳田，浅井（2024）

本稿における安全性に関する議論は，機能安全と協調安全の両方を対象としている。

1-2. System of Systems における「安全」の課題

複数／多数の自律的システムが，インタフェースを介して相互作用して，特定の目的を実現するタイプのシステムは，社会がこれまで経験したことのないシステムだと言える。SoS を構成する個々のシステム（サブシステム）そのものは従来の安全概念を基にその安全性を議論できるが[3]，サブシステムが相互に連携してシステム系としての秩序を形成する SoS の安全性に関してはサブシステムレベルの議論とは別物と認識するべきであると著者等は考える。つまり，SoS の安全性に関する議論は二重構造となっている点にまず注目するべきである。本稿では，サブシステム・レベルの安全性に関しては従来の法規やルールの問題として議論の対象とせず，サブシステム群の相互連携レベルを対象に議論を進めることとする。

2.　System of Systems（SoS）の定義と複雑さの捉え方

第 2 章では，SoS の定義例を示し，それぞれの「複雑さ」の考え方と，安全性論証の基礎となるリスク評価のフレームワークを検討する。

2-1. SoS の定義

SoS について，「自律的な振る舞いをするサブシステムがお互いに影響を及ぼしあいながら協調動作し，システム全体としてより高位の振る舞いを行う」というイメージは共有さ

れている一方で，その定義や説明は微妙に異なっているのが現状である。そこで，本稿は，ISO21839:2019[4] の定義を軸に，INCOSE（International council on System Engineering）[5] の文献に見られる説明を参照しながら議論を進めることとする。

2-1-1. ISO21839:2019 における SoS の定義

ISO21839:2019 は，システム及びソフトウェア・エンジニアリングに関する国際標準である。そこでは，SoS を，サブモジュール（原語は Constituent system）から構成され，サブモジュール間の相互作用を通じて固有の能力を提供するものとして説明している。SoS の構成要素であるサブモジュール（Constituent System）自体も，独自の開発過程，独自のマネージメントゴールや資源を持つ完結したシステムであるとされている。つまり SoS におけるサブモジュールは，SoS の部分としてのコンポーネントではなく，相互作用を通じて SoS に固有の能力を与えるものと説明されている（徳田，浅井　2023）。

また，ISO 21839:2019 は，SoS の類型として，以下の四つを挙げている[6]。以下に，筆者らによる試訳を記す。

- Directed: SoS は，特定の目的のために生成され管理される。サブモジュールは SoS の要素として構成され独立して運用されるが，通常の運用時には，SoS の目的を実現する構成要素として運用される。
- Acknowledged: SoS は，目標の理解，熟練した管理者，十分なリソースを持つ。個々のサブモジュールは，独立した所有権，目標，資源，開発／保守アプローチを維持する。システム群の変更は，SoS と特定のシステムの合意に基づく。
- Collaborative: 個々のシステムの相互作用は，中心的目的の実現のために，多かれ少なかれ自発的に行われる。中心的プレーヤーが，サービスがどのように提供されるかあるいは拒否されるかを正しく判断する。従って，強制あるいは標準の維持努力が中心的プレーヤーにより行われる。
- Virtual: SoS は，中心的な管理機能あるいは，中心的な目的を持たない。広域的な振る舞いは創発的（emerge）であり，時として魅力的である。しかし，このタイプの SoS は，自己を維持するための不可視な（未知な）メカニズムに依存している。

2-1-2. INCOSE 文書における SoS の説明

INCOSE の関連文書に見える SoS の説明の代表的なものは以下の通りである。

- a collection of independent systems, integrated into a larger system that delivers unique capabilities（INCOSE Systems of Systems Primer）

  独立したシステムの集合体であり，独自の機能を提供する大きなシステムに統

合される

- set of systems and system elements that interact to provide a unique capability that none of the constituent systems can accomplish on its own（New Standards for System of Systems Engineering）
  どの構成システムも単独では達成できない独自の能力を提供するために相互作用するシステムとシステム要素の集合
- an SOI（System Of Interest）whose elements are managerially and/or operationally independent systems.（Systems Engineering Handbook）
  SOI（System Of Interest）と呼ばれる，マネージメント面と運用面の一方あるいは両方で独立したシステムを構成要素とするもの

2-1-3. SoS の定義と複雑さの考え方に関して注目するべき点：本質的予測不能性と完全性の維持の困難さ

ISO 21839:2019 の定義と INCOSE の各種文書の説明は，ほぼ共通しているが，それぞれに特徴的な定義あるいは説明が含まれている。

ISO 21839:2019 のそれは，Virtual と表現される創発的なメカニズムを定義に含んでいる点であり，具体的なシステムの振る舞いが原理的に予測不可能であることを含意している。同様に，Systems Engineering Handbook は，SoS の構成要素が，マネージメント面と運用面の一つあるいは両面で独立したものであると説明する。このことは，SoS の一貫性や完全性の維持が時間経過と共に困難になる可能性を表現していると理解できる。SoS が事故あるいは不具合を起こした時，このような差は上記表面的には些細なものと見えるかもしれないが，事故や不具合の原因分析とそれに基づく刑事／民事上の責任と賠償／補償について考える時，その差が大きな意味を持つ可能性があると我々は考えている。

2-2. 事象の予測可能性に関する議論

2-1-3 節で，SoS は予期せぬ挙動や時間経過と共に完全性が失われる傾向があることを指摘した。社会的基盤システムとしての SoS では，それらは事故や障害と呼ばれる現象として現れる。それらは，損害賠償や補償の対象となり，それらへの対応は，便益と損失のバランスをどのように考えるかという社会的な問いをもたらす。便益と損失のバランスを考える際に考慮すべきことの一つは「リスク」である。本節では，SoS におけるリスクの考え方を検討する上で有益と思われるフレームワークについて，ジョハリ（Johari）・ウインドウを紹介し，リスク評価への応用の経緯を確認する。

ジョハリ・ウインドウ[7]は，Joseph Luft と Harry Ingham が，人の相互作用の理解のために 1969 年に提唱した四象限モデルである。このフレームワークを応用して様々な領域で

リスクの説明のために応用されてきた経緯がある。特に 2002 年のラムズフェルト国防長官（当時）が行った記者会見での使用が契機となって広く知られるようになった[8)]。様々なものが例示されるが，本稿の趣旨に沿った表現として図表 -2 を例示する。

図表 −2　SoS のリスク分類のためのジョハリ・ウィンドウの応用例

出所）Johari の原図を基に著者らが作成（徳田，浅井　2023）

2-3. 小括：安全性論証に関して

リスクを評価するためには，既知（known）と未知（Unknown）の区分，そして，未知（Unknown）な事象の捉え方とリスク評価方法が大きな意味を持っていることを，Johari window モデルは示唆している。

SoS の安全面でのガバナンスにおいても，事象を known-unknown という形で整理し，ステークホルダー間で共有することは，迅速な対処と安全性の漸進的向上を図るための前提となると考える。

リスク評価という観点から見る場合，SoS の特徴は，1）創発性に由来する予測不可能性，2）運用と管理が分離した多数のサブシステムの時間経過に伴う完全性の喪失　の二点から来る予測困難性にある。前者は，例えば天気予報のように，観測パラメータ・セットのヒューリスティックな発見と計算モデルの改良により精度を高めることはできるが根源的に予測不可能なものであり，後者は，コストをかければ解消可能かもしれないが得られる利益とのバランスが問題となる。

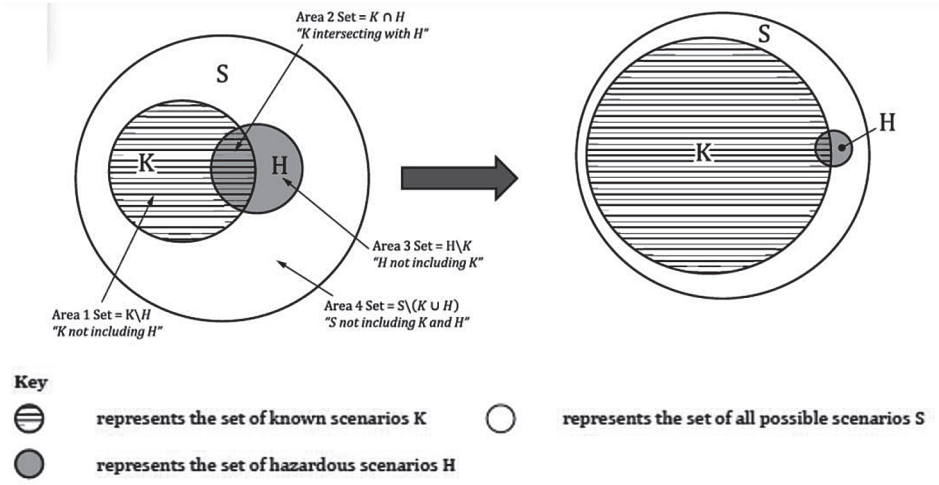
3. 事例研究

3-1. ISO 21448 の概要と先駆的事例としての性格

ISO21448 Safety of the intended functionality（以下，SOTIF）は，センサー，複雑なアルゴリズム，電子制御（E/E system）によるアクチュエータからなる自動運転車の安全性の担保を目標としている[9)]。SOTIF では，安全性を E/E system の動作不良を原因とするハザード（hazard）に起因する予測不可能なリスクの不在と定義し，安全性確保のためのハザード分析とリスク・アセスメント（HARA）の評価を定義している[10)]。

SOTIF における表現の基本は，風景，動的物体に加え全てのアクターや観察者とその関係のスナップショットであるシーンおよびシーンの連続体であるシナリオである[11)]。図表 -3，図表 -4 に，SOTIF におけるシナリオの概念と施策の概念を示す[12)]。

図表 −3　SOTIF におけるシナリオの集合

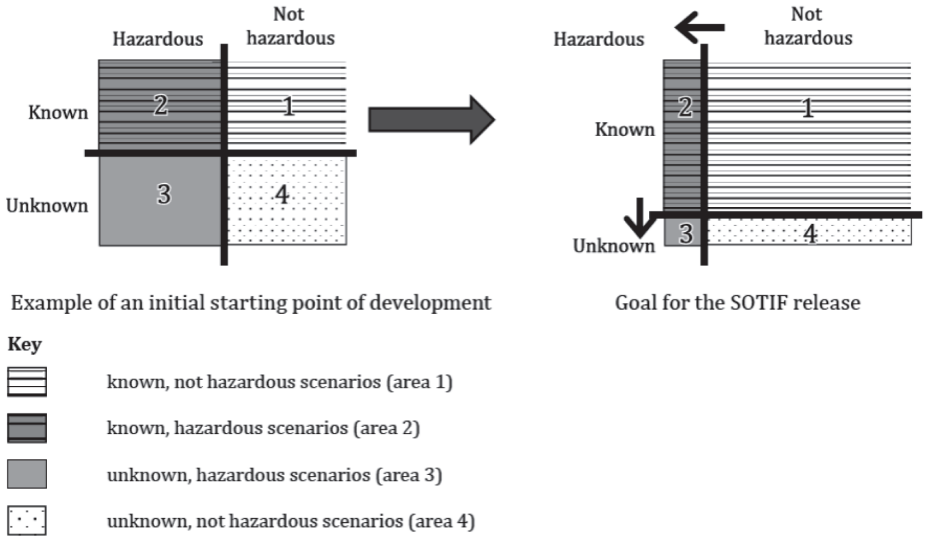


出所）ISO21448 SOTIF p14

図表 -3 において，集合 S は可能性を持つシナリオ全ての集合，集合 K は既知（known）なシナリオの集合，集合 H は危険シナリオの集合とその包含関係を示している。SOTIF が提示するゴールは，危険シナリオ H と集合 K∩H で示される潜在的に危険な振る舞いを評価し，これらにより引き起こされる余剰リスクを十分なレベルにまで低下させるための議論を提供することである。

図表 -4 は，シナリオを known-Unknown 軸と Hazardous-Not hazardous 軸による 4 象限により分類している。そこでの基本戦略は，適切な理解と措置により Unknown 領域と Hazardous 領域を圧縮することである。筆者らは，事故／不具合の実況検分と事故シミュレータにより，シナリオの解析とシーンを表現するパラメータの最適化や拡張により所期の目的を達成することができると考えている。

図表－4 Known/Unknown とハザードに基づくカテゴリ



出所）ISO21448 SOTIF p15

SOTIF は，シナリオが含むべき要素の例示[13]，安全性分析方法の例示[14]へと論を進めることで体系的に完成する。

本稿の議論の対象である SoS においても，シナリオによる事象表現，known-unknown と Hazard による事象の分類，安全性分析という SOTIF のアプローチは参考になる。しかし，複数の自律的サブシステムを含む複雑なシステムへの適用可能性については別途検討が必要であると判断している。

3-2. PEGASUS Research Project の概要と事故の表現とデータ利活用のアプローチ

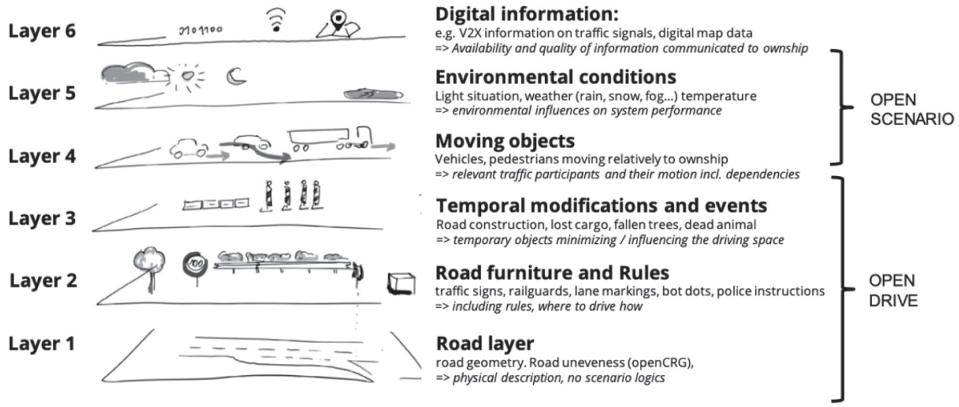
PEGASUS Research Project（以下，PEGASUS）は，ドイツ経済エネルギー省（BMWi）による高度な自動運転における安全確保に関する標準仕様を普及させるためのプロジェクトであり，自動運転機能のリリースのためのシナリオとシチュエーション，品質基準の一般的受容，ツールと方法論の確立と導入を目的としたものである[15), 16)]。プロジェクトは，6 層からなるデータモデルを軸に，「シナリオ & 品質評価」，「実現プロセス」，「テスト」，「結果反映と組み込み」という 4 つのサブプロジェクトから構成されている（図表 -5 参照）[17)]。

図表 -5 は，大きくオープンデータのレイヤとオープンシナリオのレイヤに分かれるが，PEGASUS における主要な議論は，シナリオ & 品質評価とテストの領域であり，本稿における主な関心と重なっている。

PEGASUS では，自動車の走行シナリオをファンクショナル・シナリオ，ロジカル・シナリオおよびコンクリート（Concrete）シナリオの三種類で表現する。ファンクショナル・シナリオは車両／走行／道路状況などに関する定性的な記述であり，ロジカル・シナリオ

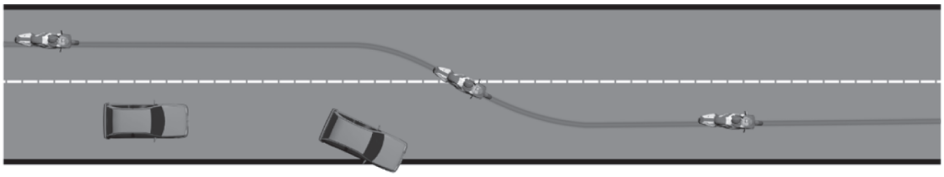
は，定性的な記述事項に対して想定する範囲の数値を追加したもの，コンクリート・シナリオは記述事項の値を確定させたものと説明できる[18]。図表 -5 で，バイクの割り込みで

図表 −5　PEGASUS のデータモデル



出所）PEGASUS Symposium 2019

図表 −6　PEGASUS のシナリオ例



Abstract

```
MotCyc: Motorcycle
PCar: PassengerCar

do serial():
    parallel:
        PCar.drive with:
            direction(straight)

        MotCyc.drive with:
            speed(faster_than: PCar)

       get_ahead: serial:
            left: MotCyc.drive with:
                direction(straight)
                lane(left_of: PCar, at:
start)
                position(behind: PCar,
                         at: start)
                position(front: PCar, at:
end)

             right: MotCyc.drive with:
                 direction(right, at: start)
```

Concrete

```
MotCyc: Motorcycle
PCar: PassengerCar

do serial():
   parallel:
      PCar.drive with:
         speed(50kph)
         direction(straight)
         lane(0)

      MotCyc.drive with:
         speed(80kph)

      get_ahead: serial:
         left: MotCyc.drive with:
            direction(straight)
            lane(1)
            position(20m, behind: PCar,
                     at: start)
            position(2m, front: PCar, at:
end)

         right: MotCyc.drive with:
```

ASAM OpenSCENARIO: Version 2.0.0 Concepts

出所）ASAM OpenSCENARIO: Version 2.0.0 Concept

車が車線逸脱をする事故シナリオの例を示す。図表 -5 の左側は抽象化されたパラメータを組合せたシナリオで，右側がパラメータ空間に具体的な値を与えたコンクリート・シナリオである。

PEGASUS における，シナリオの構築，品質評価とテストの実践的な検証活動の成果は，本稿の議論の対象である SoS においても参考になる。しかし，複数の自律的サブシステムを含む複雑なシステムへ適用するために必要十分な論点が含まれているか否か，そして含まれていない場合，何を拡張するべきなのかについて，本学における実証プロジェクトの中で検討し解を見出すための研究を継続的に行う必要があると考える。

3-3. 小括：インプリケーション

本稿の目的である SoS の安全性論証に関する議論に関して，以下の知見が得られた。

a） リスク評価に関しては ISO21448 SOTIF のフレームワークを基礎に，Unknown 事象の由来に配慮した拡張を施すことにより対処可能

b） SoS における安全性論証は，PEGASUS で定義されたシナリオとその拡張で記述可能。

c） SoS におけるシナリオ構造，パラメータ・セットの定義，シミュレーション・モデルの定義とメンテナンスは，専門知識を基にさらに研究が必要

ただし，以下の制限などを考慮する必要があることも認識することができた。

d） PEGASUS のシナリオは，アクターの相対的な位置関係や動作関係がパラメータ化されており，アクター同士の相互関係で事故に至る経緯を表すことができるようになっている。しかし，車両に搭載されたセンサーデータなど制御システムに関する具体情報はシナリオに登場しない。従って，事故の因果関係を明文化できないので，もう一段階掘り下げた機器レベルのシナリオが必要

e） PEGASUS が導入するシナリオ・データベースは，あるインシデントの Known/Unknown の判断の基礎的情報を提供する。シナリオ・データベースに存在するインシデントは Known として扱うことができる。ただし，インシデントの同一性については，パラメータ・セットとその値の類似性のレベルを超えて，複雑系を記述するレベルでの比較が要求される可能性がある

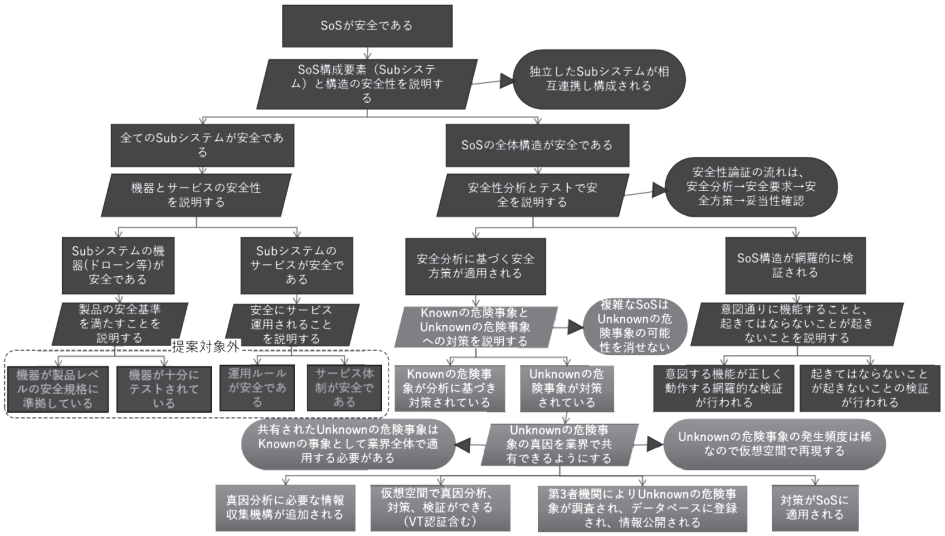
上記に基づいて，事故の検証や SoS のアセスメントを行うためのシミュレータが備えるべき要件の研究を継続して行う予定である。

4. SoS の安全性論証と運用に関する論点整理と課題提示

4-1. 安全性論証の基本構造

SoS の安全性論証をするには，「SoS が安全であること」をゴールにおいたとき，ゴール達成を説明するための論証構造が必要となる。この論証構造に基づき SoS の安全性論証を行う。図表 -7 に SoS 安全性論証の構造を示す[19)]。

図表 −7　SoS の安全性論証の基本構造



出所）OTSL 2023

SoS は，構成要素であるサブシステムと，その協調動作により成立する SoS そのものからからなっている。このため，「SoS が安全である」という安全性論証のゴールは，二つのサブゴール「全てのサブシステムが安全である」と「SoS の構造全体が安全である」に分けることができる。その際，「全てのサブシステムが安全である」は，「SoS の構造全体が安全である」の前提条件となる。

「全てのサブシステムが安全である」ことは，ロボットのような機器と，機器や情報を用いてシステム利用者に価値を提供するサービスの両者について安全性を説明することである。機器は機器製造事業者により開発された工業製品であることが専らであるため，SoS に含まれる全ての機器が製品レベルの安全規格の要件を満たしており，かつ，十分なテストシナリオを用いて検証されていることを製造者が示すことを以て安全性論証とすることができる。サービスに関しては，運用ルールおよび運用体制の安全性を示すことで安全性論証とすることができる。これらは，国際標準規格や JIS で既に示されているものを援用することを想定しているので，本稿でこれ以上の議論は行わない。

SoS 構造の安全性論証は，複数の Sub システムで構成される構造（SoS 構造）に対して，

安全性分析を行うこと，および，検証を行うことで実施する。通常，安全性論証は，「安全性分析」→「安全要求の導出」→「安全方策の適用」→「安全性の妥当性確認」の流れで行われるので，安全性分析に基づき安全方策が適用されること，および，安全方策を含めた SoS 構造を網羅的に検証することで安全性の妥当性確認を行う。

第 2 章で議論したように，SoS は Unknown な事象の発現を前提とする必要がある。そのため，SoS の安全性に関する議論は，常に事象の Known/Unknown を念頭におきつつ行われる必要がある。

4-2. Known な事象に関する安全性分析

Known な事象に関する安全性分析は，静的な安全分析手法を適用可能である。静的な安全性分析手法の代表的存在は STAMP/STPA である[20]。STAMP/STPA は，人間や環境を含むシステム構成要素間の相互作用の分析を前提とするので，SoS に於けるサブシステムの相互作用に関する分析への適応性を期待できる。

4-3. Unknown な事象に対する安全性分析

Unknown な危険事象に対する安全性分析については，以下に示すような動的なアプローチの採用を提案する。

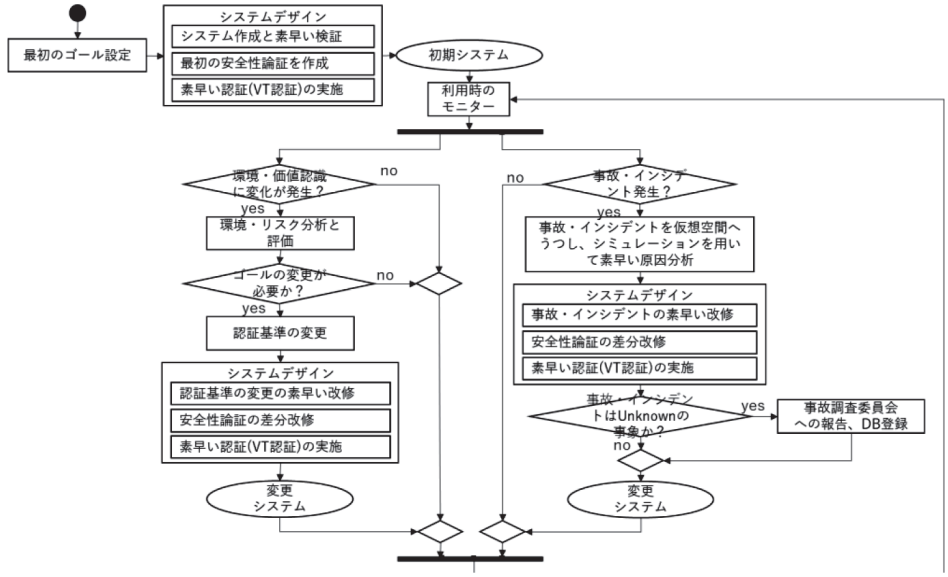
前例がないと思われるような危険事象が起きた時，起きた危険事象が Unknown のものであることが確認する必要がある。そのために，Unknown の危険事象の真因を明文化するための情報収集機構を SoS あるいは機器などの動作環境に設置する。取集された情報から危険事象を仮想空間で再現して真因分析を行う。真因分析結果に基づく対策を立案し，仮想空間で検証およびバーチャルテスト認証（VT 認証）を行う。仮想空間を用いる理由は，Unknown の危険事象の発生頻度が稀であり，現実世界で再現が困難だからである。仮想空間での作業には適切なシミュレータを用いる。仮想空間で検証された対策は SoS に速やかに適用されなければならない。また，Unknown の危険事象は個別の事故情報と対策案を含め，事故調査委員会など第 3 者機関に報告される。第 3 者機関は事故情報と対策をデータベースに登録して一般向けに情報開示を行う。情報開示された危険事象は Known の危険事象として扱われ，業界全体に対策の適用が求められる。

SoS 構造の網羅的な検証は，次の 2 つ，「設計意図通りに機能することを正常系の網羅性のあるテストで確認」，および，「起きてはならないことを意地悪テストで確認する」ことで行う。前者は網羅性のあるテストシナリオを用意する必要がある。後者は利用環境を考慮して，より厳しい条件でテストを行う必要がある。考慮すべき対象を以下に示す，関係者（利用者や通りがかりの人など）の行動，運用環境，運用規則（ルール），あるいは，Sub システムの不具合など。また，現実世界で行うテストのコストなどを考慮し Virtual Testing も認めることとする。

4-4. 安全性論証のためのシステム・デザインと運用の流れに関する提案

事故およびインシデントの発生時に，軽量なプロセスで安全性論証と信頼性確保ができる方法として，システムデザインとシステム運用の処理の流れを具体化する。図表 -8 に SoS 安全性論証のためのシステムデザインと運用の流れを示す。

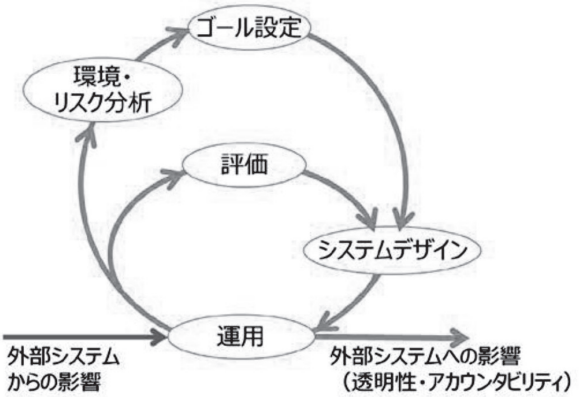
図表 −8　SoS 安全性論証のためのシステムデザインと運用の流れ



出所）OTSL　2023

まず，最初のゴール設定に従い設計された初期システムが運用される。このとき，システムに監視機構を設け，システム利用時に起きた事象を監視できるようにする。監視対象は環境・使い方の変化と事故・インシデントなどの障害（危険事象の発生）である。前者は 2 重ループ（図表 -9）の外側のループ，後者は内側のループの起点となる。ただし，ループに対応する活動は常に複数存在しており，同時並行的に動いている。

図表 −9　アジャイルガバナンスの 2 重ループ



出所）経済産業省 [21)]

処理の流れの左側は環境･価値認識の変化に対応したフローであり，変化が発生した場合，環境・リスク分析を評価しゴールの変更が必要かどうか判断する。ゴールの変更が必要なとき，認定基準（制限速度のテストシナリオなど）を変更し，システムを速やかに改修して運用に戻す。処理の流れの右側は事故など危険事象の発生に対応したフローであり，危険事象が発生した場合，危険事象を仮想空間にうつし，シミュレーションによる素早い原因分析を行う。原因分析を速やかに行うため，システムにはあらかじめ情報収集機器を搭載しておく必要がある。分析結果に応じシステムを速やかに改修する。また，危険事象がUnknown の事象だったとき，事故情報と対策を事故調査委員会へ報告し DB 登録することで情報開示できるようにする。改修されたシステムは速やかに運用へ戻す。

5. まとめと今後の課題

本稿では，SoS の安全性論証の構造と，事故など危険発生時に，軽量なプロセスで安全性論証と信頼性確保ができる方法，そして，Unknown の危険事象の判定方法として，以下の各項目を検討し対策の方向性を示した。

1. 複雑なシステムとしての SoS の二つの予測不能性，すなわち創発性に由来する本質的予測不可能性と，管理と運用の分離に由来する完全性の喪失による予測困難さを確認した。
2. SoS のリスクを考えるための基本的なフレームワークとして Johari ウィンドウを発展させたものを用いて Unknows（知られていない事象）への対処が中心的課題であること
3. 先行事例としての SOTIF と PEGASUS Project の概要と意味

そして，第 4 章で，安全性分析の方法について基礎となるモデルを提示した。そこでの主張は，以下の通りである。

a） 事故／インシデントデータベースの内容と実際に起きた事故／インシデントを比較照合することにより Knowns/Unknows の判定ができる。

b） Unknowns と認識された事象の原因分析が重要である。それにより事故データベースと Virtual Test（シミュレータ含む）の更新によりリスクの正しい評価と低減を期待できるからである。

c） SoS が運用される環境や SoS の使い方の変化といった大きな変化と，事故・インシデントなどへの迅速な対処は区別するべきであり，それは二重ループを持つアジャイルガバナンスとして実現される必要がある。

また，上記の議論により，安全性論証におけるシミュレーションを活用すべき箇所とシミュレーションに必要なデータに関して基礎的な要件を導き出す準備を行うことができた

と考えている。

しかし，SoS の安全性論証とガバナンスに関する課題の全てに解を提示できた訳ではない。とりわけ Knowns/Unknowns の判定に関してはさらに検討が必要であると考えている。この判定は，過去の事故 A と検討対象となっている事故 B の同一性を判断することになるが，何が一致すれば同一と言えるのかが問われなければならない。例えばシミュレーション・モデルが持つ全変数の一致が条件となるならば，ほぼ全ての事故／インシデントは同じとは言えないであろうことは直感的に明らかである。SoS の創発的性格と完全性の喪失による予測困難さ，それぞれの原因追求アプローチの差と法的責任など，なお考慮すべき点は多く，学際的な検討が要求される領域であり，これらに関する論点抽出と研究を継続したいと考えている。

謝辞

この成果は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP22006）の結果得られたものです。

This working paper is based on results obtained from a project, JPNP22006, commissioned by the New Energy and Industrial Technology Development Organization（NEDO）.

【注】

1）本稿の一部は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP22006）の一環として行われた株式会社 OTSL（以下，OTSL）の報告を参照している。
2）一般社団法人セーフティグローバル推進機構　https://institute-gsafety.com/safety2/
3）例えばサービス・ロボットやドローンの安全基準や認証手続きで前提とされる安全性が相当する
4）ISO21839:2019 Systems and Software engineering, System of Systems（SoS）consideration in life cycle stage of a system
5）INCOSE ホームページ　https://www.incose.org/
6）INCOSE ホームページ　https://www.incose.org/
7）Johari は，Joseph Luft と Harry Ingham の名からとった呼称
8）https://archive.md/8k6bU
9）ISO 21448 Road Vehicles -Safety of the intended functionality p-vi
10）ISO 21488-3
11）ISO 21448 Road Vehicles -Safety of the intended functionality p9
12）ISO 21448 Road Vehicles -Safety of the intended functionality p13
13）ISO 21448:2022 pp99-106
14）ISO 21448:2022 pp106-120
15）https://www.pegasusprojekt.de/en/about-PEGASUS
16）経済産業省が実施する SAKURA プロジェクトは，シナリオに基づく安全性評価フレームワークを構築し，ISO34502 として国際標準化された。https://www.meti.go.jp/press/2022/11/20221116006/20221111005.html

17）https://www.pegasusprojekt.de/files/tmpl/Symposium2019/1_2_MUSICC_Saigol.pdf

18）ロジカル・シナリオは，抽象化されたパラメータを組み合わせたファンクショナル・シナリオ（例：速度型の最低速度を持つ）に，より具体的なパラメータ空間を与えたシナリオ（例：10km/h < 最低速度 < 30km/h）を指す。コンクリート・シナリオはパラメータ空間を具体的な値に置換えたシナリオ（例：最低速度 = 28km/h）を指す。シナリオのロジックとパラメータ空間の定義を分けることで，パラメータのサンプリングとシナリオの実行を分けて考えることができる。

19）記法は，GSN（Goal Structure Notation）に準拠している

20）MIT Nancy Leveson が提唱したシステム理論で定義された事故モデルにおけるハザード要因を分析する手法である。

21）https://www.meti.go.jp/press/2021/07/20210730005/20210730005-1.pdf p ix

【参考文献】

経済産業省「アジャイル・ガバナンスのデザインと実装に向けて」p ix, https://www.meti.go.jp/press/2021/07/20210730005/20210730005-1.pdf（2024 年 6 月 14 日閲覧）

経済産業省「日本発の自動運転システムの「シナリオに基づく安全性評価フレームワーク」に関する国際標準が発行されました」https://www.meti.go.jp/press/2022/11/20221116006/20221111005.html（2024 年 6 月 20 日閲覧）IEC 2020「将来の安全」pp32-34

一般社団法人セーフティグローバル推進機構「セーフティ 2.0」トップページ，https://institute-gsafety.com/safety2/（2024 年 3 月 29 日閲覧）

独立行政法人製品技術基盤機構「製品安全におけるリスクアセスメントの概要」，https://www.meti.go.jp/product_safety/koureisya/risk_assessment_outline.pdf（2024 年 3 月 29 日閲覧）

日本認証株式会社「Safety2.0・協調安全について」，https://www.japan-certification.com/safety_registration/safety2/about/（2024 年 3 月 29 日閲覧）

徳田昭雄，稲葉光行，山田希，浅井龍男（2023）「System of Systems の特性とステークホルダーの分析，及び，ガバナンスに関する論点整理の試み」，『立命館デザイン科学研究』vol.3 pp 215-227

Hollnage, E. What is SafetySynthesis? Website https://safetysynthesis.com/（accessed March 29, 2024）

INCOSE Systems of Systems Primer Website https://www.incose.org/publications/technical-product-catalog/sos-primer（accessed March 29, 2024）

INCOSE（2023）Guide for Systems of Systems（SoS）Use of Requirements WG Products

ISO/IEC ISO/IEC Guide 51:2014, Safety aspects: Guidelines for their inclusion in standards

ISO/IEC ISO/IEC 21448 Safety of the intended functionality

Joseph Luft and Harry Ingham The Johari Window Website https://web.archive.org/web/20071222001124/http://www.noogenesis.com/game_theory/johari/johari_window.html（accessed March 29, 2024）

Marco Caccamo etc. Physical Deep Reinforcement Learning: Safety and Unknown Unknowns Website https://arxiv.org/pdf/2305.16614.pdf（accessed March 29, 2024）

Nicolas Becker（2021）The Safety of the Intended Functionality Report on ISO/TC22/SC32/WG8 activities

PEGASUS Research Project Website https://www.pegasusprojekt.de/en/about-PEGASUS（accessed March 29, 2024）

Presenter: Secretary of Defense Donald H. Rumsfeld（February 12, 2002 11:30 AM EDT）DoD Transcript Website https://archive.md/8k6bU（accessed March 29, 2024）

Proposal on the Basic Structure of Safety Argumentation for System of Systems

Akio Tokuda*
Shoji Hatano**
Terutoshi Yamamoto***
Tatsuo Asai****

Abstract:

Expectations for social infrastructure systems that rely on information communication technology (ICT), such as intelligent transportation systems and smart city concepts, are on the rise. The System of Systems (SoS) is being discussed as a promising architectural candidate for such infrastructure systems. At present, a SoS is defined as a system consisting of components that behave autonomously and cooperative behavior among components, the introduction and operation of which society will experience for the first time. In addition, many have pointed out that SoS, due to its "complexity," is not consistent with conventional legal systems and ways of taking responsibility. How should we conceptualize and accept the SoS? This is the overarching question regarding SoS.

In this paper, we first examine how the complexity of the SoS is viewed, followed by a discussion of ISO 21448 (Road Vehicles-Safety of the intended functionality, SOTIF), which can be considered a pioneering example of the safety discussion, and the PEGASUS Research Project and its extension to the SoS.

Keywords:

System of Systems (SoS), Complex System, Risk evaluation, Argumentation of secureness, Agile governance

* Professor, College of Business Administration, Ritsumeikan University
** OTSL Inc.
*** OTSL Inc.
**** Visiting Researcher, Institute of Design Science, Ritsumeikan University