複雑システムの安全設計と事故モデルについて

Safety Design and Reliability Management of Complex Systems

会津大学　　兼本　茂　　Shigeru KANEMOTO　Member

Abstract
The present paper discusses a concept of safety design and accident models in complex systems. Modern artifacts or engineering products in our daily life are usually controlled by embedded computers and inevitably connected to Internet. Due to big progress of computer hardware and software performance, control algorithms become more and more sophisticated, intelligent and complex. This makes difficult to assure the safety of the system. Since the current safety assessment tools such as FTA/ETA, FMEA or HAZOP were developed more than 50 years ago, it is difficult to use them in modern complex system safety assessment. In the present paper, one of new hazard analysis techniques, called STAMP/STPA, will be discussed to meet a new safety requirement of 'Safety2.0'.

Keywords:
Safety2.0, Complex embedded system, FRAM, STAMP/STPA

1．はじめに

最近、産業界で「Safety2.0」なる考え方が議論されている[1]。これは、人工知能、ビッグデータ、IoT（Internet of Things）といった情報化社会の技術潮流に沿って、第４次産業革命（Industry4.0）や、自動運転のような高度な製品・制御システムが提供される時代になり、旧来の安全設計の考え方ではない新しい安全設計が必要とされてきたためといえる。また、3.11 福島事故の頃から、Safety-II（レジリエンス工学）という考え方が広まっている[2]。これは、システムの破局的な事故を防ぐために、人間の過誤を防ぐという旧来の考え方ではなく、人間の柔軟な問題解決能力を利用することも必要ということから注目される考え方でもある。

このような安全設計に関する時代の変化を明快に示したのがN.G. Levesonにより与えられたFig.1のような背景である[3,4]。近年の製品・制御システムは、そのほとんどが組込み計算機のソフトウェアにより制御されており、多くの場合、インターネットとも繋がった複雑なシステム構成となっている。ソフトウェアにより知的で複雑な制御が実装可能になる。運転員からの指示や運転員への干渉操作が複雑に絡み合っている点も近年のシステムの特徴である。このような複雑システムの進展に対して、それに対応する安全設計では、図にあるように５０年以上前に開発されたFTA・ETA、FMEA、HAZOPといった手法が依然として使われている。

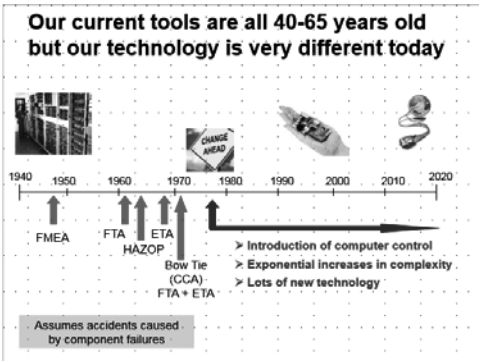

Fig.1 Background of current safety design tools

筆者は、組込みシステム・情報システム分野での安全設計や障害診断といった分野での技術動向調査に関わっているが[5,6]、計算機性能の指数的な増大に伴って、システムの複雑さや開発速度も指数的に早まっており、そこでの安全設計の欠陥や不具合が社会に与える影響の大きさは、日々のニュースでも実感しているところである。原子力プラントも巨大な組込みシステムということがで

連絡先: 兼本　茂、〒965-8580 福島県会津若松市一箕町鶴賀、会津大学・コンピュータ理工学部
E-mail: kanemoto@u-aizu.ac.jp

き、複雑システムの安全設計という共通の視点で安全設計を考察することは意味があると考える。

冒頭述べた「Safety2.0」の考え方を向殿[1]に沿って要約すると Fig.2 のようになる。機械技術で安全を確保する Safety1.0 の時代から、人、モノ、環境が協調しながら安全を確保する Safety2.0 に移行し、止めない安全を達成することを目指している。「止める安全から止めない安全」というのは言葉としてはわかり易いが、これを実現することはそう簡単なことではないし、一部の業界を除いて、具体的な方策が提案されているわけではない。生産要求と安全のトレードオフは永遠の課題でもある。しかしながら、人と機械が日常の場で共存する時代を迎えた今、このような安全の考え方を整理・考察してみることは意義がある。本稿ではそのきっかけとして、N.G.Leveson の提唱している STAMP/STPA というハザード分析法を紹介する。

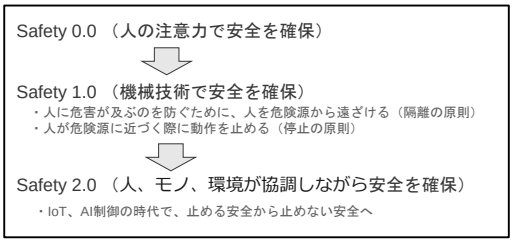

Fig. 2 Transition of Safety Design

2．複雑システムの安全設計とは

2.1 IoT 時代の環境変化

組込み計算機が小型化し、無線端末を通してインターネットとつながる IoT 時代を迎え、以下述べるような環境変化が起こっている。人、モノ、環境の相互作用がより緊密になること、インターネットを介して悪意のある攻撃が日常化している点で、セキュリティとセーフティの境界がなくなってくること、安全性に関わるソフトウェアの開発体制もオープン化が進んでくることなど、安全設計に関わる環境が大きく変わりつつあることを考慮する必要があるが、これに対応できるハザード分析手法や安全設計手法として適したものがないというのが現実であろう。また、マスコミやネットを介した批判に応えるという社会への説明責任がより大きくなっていることにも留意が必要であり、第三者による客観的な安全性の評価が重要にもなってくる。以下に、これらの環境変化をまとめる。

・人と機械の協調制御が日常化、さらに、IoT によりクラウドとの連携が深まる
・AI 技術、ビッグデータ解析技術の進展で、機械の知能化（ソフトウェアによる制御）がより進む。人間の指示に従わないことが出てくる。
・自動化の進展で、人間の能力が低下し、想定外の危機対応能力が低下する
・特別な訓練を受けてない人がシステムを使う（自動運転、介護ロボットなど）
・Industry4.0 で、生産現場で究極の効率性が求められる
・Security が IoT を通して Safety に影響する
・ウォータフォールからアジャイル開発へ移行し、安全設計・信頼性管理の不安がでてくる。
・オープン化・System of Systems。複数の独立企業による共同開発での意思疎通の問題や部品調達仕様の不完全性の問題がでてくる。
・PL 法による製造物責任だけでなく、マスコミなどを通した社会への説明責任が強く問われる

2.2 Safety2.0 の目標

向殿[1]によると、safety2.0 の達成目標として下記の４項目が挙げられている。

(1) 止めない安全／人とモノと環境を情報でつないで、それぞれが協調して高次元の安全を実現する取り組み。例えば、熟練度が高い人の不安全行動では機械の速度を落とす、また、熟練度が低い人では機械を止めるといった、柔軟な安全制御機構。人と機械の協業により、フレキシブルな生産、生産性の向上に寄与できる。
(2) IoT による安全（不安全）の見える化／人の体調や部品の劣化状況を常時監視し、人に優しい経営や安全への的確な投資を行う。
(3) 協調安全（コラボレーションフェールセーフ）/人や環境に問題が発生した時、その情報に基づいて機械が自律的に人を安全側に誘導する
(4) セキュリティの課題/IoT による情報空間と物理空間の連結でセキュリティがセーフティに影響する。セキュリティとセーフティは、「完璧であることはありえないという」共通点がある。「許容可能なリスク」という機械安全の考え方の適用が必要。

いずれも、今の時代にあった目標と言えるが、特段新しいものでもない。(1)止めない安全では、適応ユーザーインターフェイスのような概念はこれまでに検討されているし、また、緩和制御という考え方で、非常停止をするまえに、警報レベルに近づくと、出力や速度を下げる緩

和操作をする考え方も既に実用に供されている。今後、このようなソフトウェアによる安全制御は、ますます高度化することが予想されるが、思わぬバグでトラブルが起こり得ることは、近年のニュースに散見される。(2)のIoT による安全（不安全）の見える化も、米国では、Prognostic Health Monitoring(能動的状態監視）といったシステムが原子力発電所に導入されつつある[7]。日本でも、状態監視保全という考え方で回転機の挙動監視が行われている。これが、IoT の普及で、より大規模かつ低コストで実行されることが予想される。(3)協調安全では、かって、航空機の自動制御で、パイロットと機械の判断のどちらを優先させるかといった議論があった。原子力プラントでは、10 分ルールといった形で原子炉の緊急停止の直後は、機械の判断を優先するということになっている。人間と機械の判断の優先度の問題は、問題解決の状況に応じて変わるが、今後の製品・制御システムは、人間との共存がますます親密になるため、状況依存の優先度の問題は極めて重要になるが、同時に、難しい問題でもある。(4)のセキュリティの課題は、StuxNet によるイランの濃縮設備の破壊で注目された。遠心分離機の回転数情報が外部から改ざんされたわけであるが、これをアナログメータにして本質安全を図るといった考え方もSTAMP のなかで提案されている[4]。現実的に実施可能かどうかは別にして、セキュリティでも、機能安全規格で取り入れられている SIL(Safety Integrity Level)といった考え方で、重要な情報を特定し、「誰から守るか」から「何を守るか」「どの程度の完璧さで守るか」といった考え方に移行することも大事である。

このように、Safety2.0 では、新しいことを言っている訳ではなく、できることは、既に行われている。しかし、このような目標を明確にし、その標準的な達成方法をガイドラインのようなもので示しておかないと、安易に取り組むと大きな失敗にもなりうることに注意が必要である。

2.3 パフォーマンス変動の監視と制御

複雑システムの安全に関して、前節で述べたような目標を達成するのは、従来の標準規格に基づく安全設計や、綿密な事前テストを行ったりするだけでは困難と考えられる。しかしながら、いくつかの先駆的な研究や提案がある。ホルナゲルは、複雑システムの安全分析に、機能共鳴解析手法（FRAM）という考え方を提唱している[8]。失敗の本質に焦点を当てるのではなく、日々の活動の本質、すなわち、「うまくいっている」理由を考えるということである。この「うまくいっている」状態からのズレをパフォーマンス変動と称し、この変動がお互いに共鳴して大きな事故に至ることがあるという考え方である。従って、このパフォーマンスの変動を監視し制御できれば大きな事故は防げる。このための方策として、ホルナゲルは、排除、予防、防護、促進という考え方を挙げている。排除はハザードを発見して排除することを、予防は安全装置の設置によるハザードの防護という意味で用いている。防護は、事故を起こす原因より結果の緩和や回復手段と定義している。促進は、予防が有害なものを防ぐという視点に対して、有用なものに焦点を当て、過誤を防ぎより使いやすくするような改良設計などを行うもので、能動（プロアクティブ）保全といえよう。

筆者なりに、これを再整理したものを以下に示す。防護を、通常トラブルの緩和手段と、破局的事故の緩和手段に分け、さらに、選抜・訓練を追加した。

(1) 排除／設計の前段階の要求仕様まで含めたハザード分析、第三者 V&V、本質安全策の検討、ソフトウェア制御に関する形式手法による数学的証明など
(2) 予防／安全規格にのっとった多重化や冗長化
(3) 防護／深層防護と多様化（D^3）
(4) 促進／有用性の積極評価と能動（改良）保全（失敗学から成功学へ）、安全が効率向上にも役立つ改善活動、能動的状態監視（Prognostic Health Monitoring）、保全 PDCA サイクル
(5) 破局からの防護・レジリエンス／事故の緩和・回復手段の提供、未予見外乱の兆候を認識する注意力と発想力（小さな改善活動（促進）は、小さな失敗は防げるが、大きな失敗を防げるわけではないことにも留意が必要）
(6) 選抜・訓練／専門技術とメンタルスキル（注意力）の訓練[9]、安全文化の醸成

これらの方策のなかで、実施可能なものはすでに実際の現場に取り入れられている。一方で、(1)のハザード分析などは、多様な環境下（例えば、定期検査時の作業、新人と熟練者の混じった作業現場など）で、旧来の FTA や FMEA のような手法が使えるかというと、疑問が残る。また、(6)の選抜・訓練においても、予想できるシナリオでの対応訓練は十分に行われているが、想定をはずれたシナリオをどこまで想定できるかといった発想力のテストのような訓練がなされているかは不明である。こういった視点で役立つ可能性のある手法として、以下の節で、

新しいハザード分析法の紹介をする。

2.4 新しい事故モデル・STAMP

複雑システムに対応可能な事故分析法としてFRAMを挙げたが、もうひとつの注目すべき事故モデルとして、N.G.Levesonにより提唱されたSTAMP（System Theoretic Accident Model and Process）がある。いずれも、非線形複雑システムを想定し、創発（Emergency）や共鳴（Resonance）といったフィードバックを含む動的なシステムの中で、安全が保たれたり事故が顕在化したりするという考え方を基礎においている。FRAMが正常時の挙動からの偏差をどう調整して成功に導くかという視点に立ったモデルであるのに対して、STAMPはハザード（失敗）に至る非安全な制御行動をどのように網羅的に発想できるかという視点に立っており、対極的なモデルにも見えるが、システム内の要素の動的で非線形の相互作用によりハザードが発現するという本質的な考え方は同じであろう。ただし、成功に至るパスや失敗に至るパスを新たに見つけるという発想力を喚起する手段としては異なり、具体例による比較が待たれるところでもある。

ここでは、STAMPに付随するハザード分析手順であるSTPA（System Theoretic Process Analysis）を含めて、複雑系のハザード分析にどのように用いてゆくかを簡単な事例で説明する。

STAMP/STPAの詳細は、N.G.Levesonの教科書[4]ないしIPA（情報処理推進機構）で公開している解説書[7,8]を参照していただきたいが、ここでは、その概略手順を述べる。具体的手順は、次章の事例研究を参照されたい。

(1) アクシデント、ハザード、安全制約、ならびに、制御構造図を定義する。ここで、ハザードはアクシデントに至る前の状態と定義され、安全制約は、ハザード状態を防ぐための制約条件である。制御構造図は、対象プロセスの安全を保証するための制御アクションを、制御器、操作器、フィードバック信号、対象プロセスという抽象化した構造の中で定義する。

(2) 非安全なコントロールアクション（CA）の識別：STPAの特徴は、非安全CAを下記の4つのキーワードをもとに識別することである。
- 与えられないとハザード
- （不適切な状況で）与えられるとハザード
- 早すぎ、遅すぎ、誤順序で与えられるとハザード
- 早すぎる停止、長すぎる適用でハザード

(3) 非安全なコントロールの原因（ハザード誘発要因／ハザードに至るシナリオ）の特定、ならびに、その回避策（下位のコンポーネントへの安全制約）の導出。原因を考える参考として、Fig.3に示すような、ハザード誘発要因のガイドワードが与えられている。

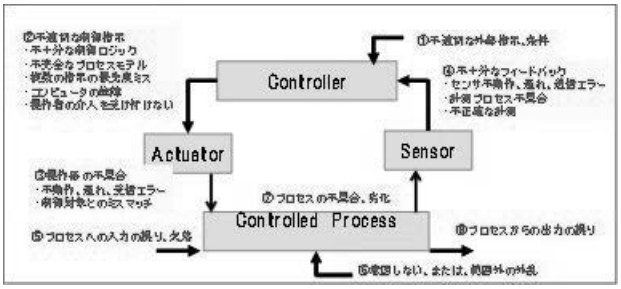

Fig.3 Typical hazard causal factor

3. STAMP事例研究

3.1 化学プラントシミュレータ[8]

Fig.4にSTAMP事例研究の対象にした簡単な化学プラントシミュレータのUI画面を示す。二つのタンクがあり、下のタンクの水をポンプで汲み上げ、さらに上のタンクの水位を一定に保つ制御器を持たせている。何らかの故障で水位がアラームレベルを超えると、緊急排水弁を開放してオーバーフローを防ぐ仕組みである。右下の赤丸で示した部分で、各種の故障を模擬できるようになっている。このシミュレータの特徴は、右上の赤丸で囲ったボタンで、システムの起動（注水操作）、緩和操作（水位がアラートレベルを越えた際に数秒間だけ緊急排水弁を自動ないし手動で開けてアラームでの緊急排水を回避する）、緊急停止、通常停止という4つの操作を運転員の指示で行う点である。各種の故障のもとで、機械による自動制御と運転員によるオーバーライド操作を模擬できるようになっている。

なお、このシミュレータの詳細は、IPA/SECのWebサイトで公開されている[8]。

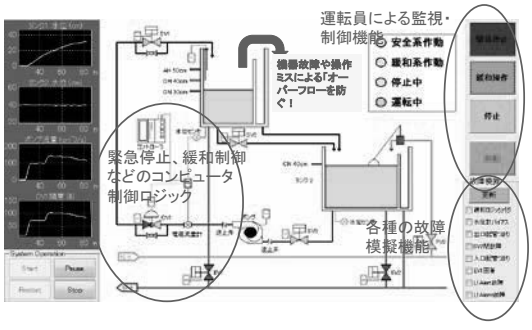

Fig.4 Simple chemical plant simulator

3.2 分析結果[8]

この化学プラントのアクシデントをタンク１からの溢水として分析した結果を以下で説明する。ハザードは、水位がアラームレベルを超えた状態であり、安全制約は、水位がアラームを超えた危険な状態にならないこと、となる。この制御構造図は、Fig.5 に示すように、運転員、コントローラ、緊急排水弁が階層的に並んだものとなる。ここで、コンピュータからプラントへの指示（コントロールアクション、CA）は、タンクへの注水と緊急排水になる。さらに、運転員からのオーバーライド指示として、緊急排水操作がある。この 3 通りの CA に対して、4 つのタイプの非安全 CA の影響を評価したのが、Fig.6 の表である。5 通りの非安全 CA(UCA)が抽出されていることがわかる。

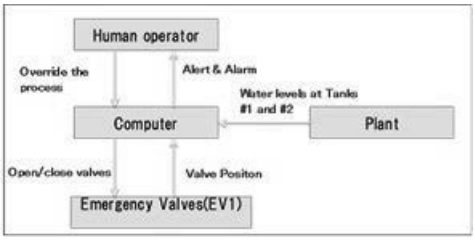

Fig.5 Control structure of chemical plant

	Not providing causes hazard	Providing causes hazard	Incorrect Timing / Order	Stopped Too Soon / Applied too long
Computer Action 給水弁(SV1)閉とドレン弁(EV1)閉		水位アラームレベル状態で、コンピュータが給水弁閉とドレン弁閉指示を出す(UCA1)	水位アラームレベル以下になる前に、コンピュータが給水弁閉とドレン弁閉指示を出す(UCA1)	
Computer Action 給水弁(SV1)閉とドレン弁(EV1)開	水位アラームレベルに達した際、コンピュータが給水弁を閉しない、または、ドレン弁を開しない(UCA1)		水位アラームレベルに達した後X秒以内に、コンピュータが給水弁を閉しない、または、ドレン弁を開しない(UCA1)	コンピュータが給水弁が完全に閉まる前に指示をやめる、または、ドレン弁が完全に開く前に指示をやめる(UCA2)
Human Action 運転員の手動操作の介入(コンピュータ指示に優先)	コンピュータ操作が溢水を引き起こしそうな時に、手動操作で介入しない(UCA3)	コンピュータが意図通りに動いている時に、運転員が介入して溢水を引き起こす(UCA4)	コンピュータ操作が溢水を引き起こしそうな時に、手動操作の介入がX秒以上遅れる(UCA3)	誤解により運転員が緊急排水を中断する、または、給水を継続する(UCA5)

Fig. 6 Unsafe control action table and hazard prediction

この５つの UCA のハザード誘発シナリオの分析結果の一部を以下に紹介する。詳細は文献[8]を参照されたい。

(1) UCA2：コンピュータが給水弁が完全に閉まる前に指示をやめる、または、ドレン弁が完全に開く前に指示をやめる

[Scenario 2-1]

・コンピュータはバルブ開閉信号を出したので、バルブは指示どうりの状態にあると、コンピュータは思っている

・バルブの位置情報のフィードバックがないか、間違っていたため、開き終わったと勘違いして、開閉指示をやめる

(2) UCA4：コンピュータが意図通りに動いている時に、運転員が介入して溢水を引き起こす

[Scenario 4-1]

・コンピュータが適切に制御しているのに、運転員がこれを不十分と誤解する

・運転員への、水位に関する間違った情報提示、または、不適切な操作ガイドの提示

・コンピュータの動作に関する設計を理解していない

[Scenario 4-2]

・不注意によるコンピュータへの間違った操作介入

・操作介入機能の設計ミス（間違えやすい設計）

[Scenario 4-3]

・運転員の意図的な操作介入（安全マージンを犠牲にした生産効率の向上など）

・過負荷やノルマなどのプレッシャー

・安全文化の不備

これらの STPA で抽出されたハザード誘発要因を、コンピュータとプラントの相互作用の不具合を主にして、FTA による結果と比較したものが Fig.7 である。ほぼ同様の要因が抽出されていることがわかるが、STPA では、ガイドワード図にもとづいて自由な発想で要因をリストアップできるため、より柔軟な結果が得られている。

また、Fig.8 は、運転員とコンピュータの相互作用、および、運転員によるプラントの直接操作でのハザード誘発要因をまとめて示したものである。リストアップされている要因は、経験のあるエンジニアからすると、おおよそ妥当なものであることが分かろう。

これらの結果を FTA と比べてみると、Fig.9 に示すように、起動時など複雑な状況（今回の例では起動時の注水操作）や、人間の多様な過誤行動の分析に STPA は役だっていることがわかる。STPA は四つの非安全 CA の制約のもと、自由な発想でハザード誘発シナリオの導出が可能になる。

原子力分野への応用で公開されているものは少ないが、Thomas による NRC レポート[10]がある。PWR 蒸気発生器のリーク事故の分析をしており、NRC として、規制等の第三者レビューに役立つのではないかという評価を受けている。

FTAの要因	STPAの要因
安全水位センサの故障	A-4:水位が高くなったことが検出されない B-4:水位が高くなったことが遅れて検出される
アラーム判定の故障	ABD-2:判定ロジックの間違い
コントローラの故障	A-2:排水命令が出されない A-3:水位を間違って認識する B-2:排水命令が出されるのが遅い C-2:排水命令が途手に中断される(制御ロジックエラー)
人間による開の指示なし	A-1:人が排水命令を出さない A-1:人が排水命令を中止させる A-5:人が排水弁を間違って閉める A-5:人が排水弁を手動で開けない
緊急排水弁の故障	A-5:排水命令を受け取ったが排水弁が動作しない B-5:排水弁の動作の遅延 B-5:排水弁が開いてから水位が下がるまで時間がかかる(排水弁の容量不足)
その他(通信系)	A-2:排水命令が出されるが排水弁に伝わらない B-2:排水命令が出されるが排水弁に伝わるまでが遅い A-3:水位の状態についてのフィードバックの欠失や間違い B-3:水位の状態についてのフィードバックの遅れ

Fig.7 Comparison of hazard causal factors in FTA and STPA

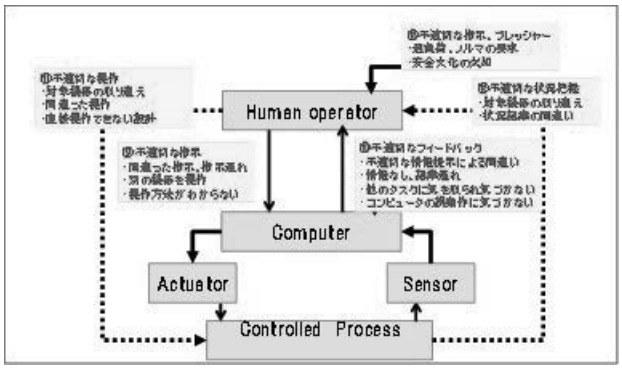

Fig.8 Summary of hazard causal factors in human and computer interaction

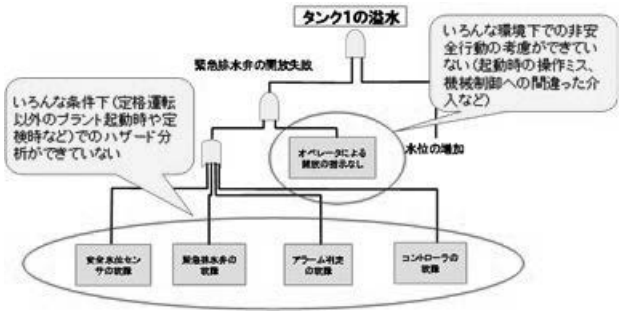

Fig.9 Merits of STPA evaluation in hazard analysis

4. まとめ

複雑システムの安全に関して最近の議論をまとめてみたが、多くの事例は原子力分野に携わっている人には思い当たることばかりであろう。一方で、3.11 の大地震・大津波のような破局的な災害に対する事前の発想力が足りなかったという反省もある。また、現在でも、トラブルが起こるたびに手順書を積み重ね、それが却って過誤を招く逆効果になることを感じることもあるのではなかろうか。N. G. Leveson は、「手順書やガイドワードに囚われすぎると自由な発想ができなくなるのでダメである」と、事あるごとに指摘している。また、リーズンのいう「専門知識とメンタルスキル（注意深さ）」を持った人材を選抜・訓練することが安全を守る最後の砦として必要というのは、当たり前ではあるが慧眼である[9]。メンタルスキルには、想定外を想定するという発想力、リーズンの表現を借りると、「生産要求とありそうでありえない潜在的な危険性の幅広い理解をする能力」が問われるが、その方法論のひとつとして STAMP があるかもしれない。

このような、原子力、航空、医療などの分野で考察されてきた安全に関する議論は、情報分野でのシステム開発にも大いに参考になる考え方である。もちろん、産業分野に応じて、安全や損失に関わる考え方は異なるが、人、モノ、環境の協調で安全を守る時代における共通の安全思想は必要であり、それを実現または支援するツールや手法が望まれる。

謝辞

本稿での考察は、IPA/SEC の製品・制御システム高信頼化部会の中の障害診断WGならびにシステム安全性解析手法WG の議論の一部をまとめたものである。参加した委員諸氏に感謝いたします。

参考文献

[1] http://kenplatz.nikkeibp.co.jp/cp/Safety2015/ http://techon.nikkeibp.co.jp/atcl/column/15/335160/021700004/?rt=nocnt

[2] エリック・ホルナゲル : Safety-I & Safety-II、海文堂、2015 年

[3] N. G. Leveson : Engineering a Safer and More Secure World、IPA/SEC-Seminar、Tokyo, 2015 年 6 月 18 日

[4] N. G. Leveson,“Engineering a Safer World: Systems Thinking Applied to Safety (Engineering Systems)”, The MIT Press, 2012.

[5] 日本技術者連盟、原子力発電所の異常診断・予防保全技術訪米調査報告、2015 年 7 月

[6] エリック・ホルナゲル : 社会技術システムの安全分析・FRAM ガイドブック

[7] https://www.ipa.go.jp/sec/reports/20160331_4.html

[8] https://www.ipa.go.jp/sec/reports/index.html

[9] ジェームス・リーズン : 組織事故とレジリエンス、日科技連、2010 年

[10] Thomas, J. and Leveson, N.: Evaluating the Safety of Digital Instrumentation and Control Systems in Nuclear Power Plants, NRC-HQ-11-6-04-0060, 2012.