

機能共鳴分析手法（FRAM）を用いた  
不特定多数の人工知能エージェントによる  
自由行動の安全化に関する研究

平成 30 年 4 月

野 本 秀 樹

# FRAM によるマルチエージェントシステムの

## 安全性評価の研究

### 目次

第1章	序論.....	5
1. 1	本研究の背景.....	5
1. 2	本研究の目的.....	6
1. 3	参考文献.....	7
第2章	従来の安全解析手法の問題点と新しい安全解析手法.....	9
2. 1	従来の安全解析手法.....	9
2. 1. 1	ドミノ・モデル.....	9
2. 2	新しい安全解析手法.....	11
2. 2. 1	STAMP モデル.....	11
2. 2. 1. 1	STAMP モデルを使った安全解析.....	12
2. 2. 2	FRAM モデル.....	15
2. 2. 2. 1	FRAM モデルを使った安全解析.....	16
2. 2. 2. 1. 1	成功要因分析：.....	17
2. 2. 2. 1. 2	リスク要因分析：.....	17
2. 3	事故モデルの変遷のまとめ.....	18
2. 4	参考文献.....	19
第3章	FRAM 分析.....	21
3. 1	FRAM 分析の特徴.....	21
3. 2	FRAM モデルの作成.....	22
3. 2. 1	出力 (OUTPUT).....	24
3. 2. 2	前提条件 (PRECONDITION).....	25
3. 2. 3	入力 (INPUT).....	25
3. 2. 4	資源 (RESOURCE).....	25
3. 2. 5	時間 (TIME).....	26
3. 2. 6	制御 (CONTROL).....	27
3. 2. 7	モデル図の完成.....	27
3. 3	FRAM モデルの分析.....	28
3. 3. 1	接続ライン長・交差数の最小化.....	29
3. 3. 2	東京駅の FRAM モデルの構造的特徴 (STAMP モデルとの	

ハイブリッド分析)	31
3. 3. 3 成功要因の分析	33
3. 3. 4 ボトムアッププロセスによる同様の行動	36
3. 3. 5 リスク要因の分析	38
3. 4 分析結果のまとめ	39
3. 5 参考文献	40
<b>第4章 シミュレーション</b>	<b>42</b>
4. 1 FRAM モデルからのシミュレータの作成	42
4. 2 東京駅モデルのシミュレーション	44
4. 3 シミュレーションロジック	46
4. 4 シミュレーション実行結果の傾向分析	47
4. 4. 1 Minimum Distance = "1"の傾向分析	50
4. 4. 2 Minimum Distance = "2"の傾向分析	52
4. 4. 3 Minimum Distance = "3"の傾向分析	54
4. 4. 4 傾向分析のまとめ	55
4. 5 シミュレーション実行結果の統計分析	56
4. 5. 1 Minimum Distance = "1"の統計分析	56
4. 5. 2 Minimum Distance = "2"の統計分析	61
4. 5. 3 Minimum Distance = "3"の統計分析	66
4. 5. 4 東京駅コンコースの状態遷移と相転移現象の類似性	70
4. 5. 5 統計分析のまとめ	71
4. 6 参考文献	72
<b>第5章 対象システムの安全化に関する考察</b>	<b>73</b>
5. 1 Safety 2 (レジリエンス・エンジニアリング)	73
5. 2 Safety 2.0 (本質安全・協調安全)	76
5. 3 将来展望	77
5. 4 今後の課題	78
5. 5 参考文献	79
<b>第6章 結論</b>	<b>81</b>
<b>謝辞</b>	<b>85</b>
<b>著者発表論文等</b>	<b>86</b>
[査読付論文]	86
[口頭発表]	87
[著書 (共著)]	87
[その他の発表論文等]	88
[授賞]	88



## 第1章 序論

### 1.1 本研究の背景

IoT [1]や人工知能という、新しい技術が自動システムに導入されるようになりつつあるが、一方でシステムの安全性評価に対する大きな課題が顕在化しつつある。今日、システムの安全性評価に対しては、IEC61508 やISO26262などの国際規格にしたがって行うこととされている。これらの規格は産業いずれもソフトウェアを用いて構成される製品を対象としているが、ソフトウェアの安全性評価にはFTAやFMEAと呼ばれる解析手法の利用が前提とされている。

しかし、「故障モード」に相当するソフトウェアの「仕様誤り」の出方は千差万別であり、FMEA(Failure Mode and Effect Analysis)においても、どのような故障(バグ)からスタートすべきかを考えると実態との乖離が大きい。同様にFTA(Fault Tree Analysis)においても、障害に対する根源事象をソフトウェアの故障(バグ)に結びつけることの困難さが指摘されている。これに対し、レブソンは、プログラムが正しくてもシステムの障害は起こりえることを実証し、むしろ構成要素間のインタフェースに着目した解析のほうが有効であるとの主張を展開し、その手法をSTAMP(Systems Theoretic Accident Model and Process)として取りまとめ有効性が評価されている。

一方、本論文が対象としている「多くの自動化装置が混在するシステム(マルチエージェントシステム)」に対しては、システムのモデル化も困難なほか安全性や安定性を評価する方法もないといった困難な課題がある。特に自動車の自動運転が混在走行する分野や多くのロボットが人間とともに共存することを展望した場合、マルチエージェントシステムのモデル化や安全性の評価に対する研究が待たれている。

筆者は、そのソリューションとして自然界における自然知能(人の知能)が自由に行動する現場では、各個人の挙動は必ずしも予測可能ではないカオス的な状況であるにもかかわらず、大きな危険もなく、長年にわたって高い安全性を達成し続けている事例が数多く存在していることに着目した。その典型的な例として、本論文では、東京駅のコンコースにおける自由行動を取り上げた。東京駅コンコースは膨大な乗客が利用する通路であるだけでなく、全国に広がる広域鉄道網の巨大ハブとして、多様な利用目的を満足し、多くの店舗、待ち合わせ場所、休憩所、広告塔など、単にそこを通過するだけではなく、通行人の行き先や歩行速度などを様々な形態にす

る要素に満ちている。この性質は、他の駅の通路に比較した場合際立った特徴で、たとえば、品川駅コンコースなど、主な利用目的が通行に限られる場所と比べると、利用者の目的の多様性は高い。ロボットにそのような場を歩行させることは、他者の行動予測が難しいために難易度が非常に高いが、人間は老若男女を問わず、大きな困難も無く安全に利用している。ところが、東京駅のコンコースにおいては、人の行き来を制約するルールや、危険を検知して人の流れを制御するような機構は設けられておらず、一見するところ、単なる通路でしかない。つまり、東京駅の安全性は、既存の安全制御のスキームとは異なるものによって達成されている可能性が高い。そのような場所における安全化を考察するためには、従来のハザードから影響を解析する視点ではなく、まず、そもそもシステムがなぜ成功できているのかを分析し、その中からリスク要因をあぶりだすことで新たな方策が生み出されると考えた。

このためには、レジリエンス・エンジニアリングの一手法である、機能共鳴分析手法 (Functional Resonance Analysis Method: FRAM) [2] が適していると考えた。

## 1. 2 本研究の目的

本研究では、不特定多数の人工知能エージェントによる自由行動の安全化を目的とする。研究では、既存の安全解析手法の歴史的な変遷を概観したうえで、研究の位置づけを明確にする。東京駅のコンコースエージェントの安全性の振舞いについては FRAM を用いて、システムの成功要因 (システムが安全に目的を達成できる要因)・リスク要因を識別するとともに、同手法によって生成されたモデルの妥当性を検証するためのシミュレーションを行い、次世代のシステムの安全につながる知見を得る。

東京駅には、一日平均約 44 万人の人間が流出入する[3]。仮にメインコンコースに、全流入人口の半分の 20 万人が行き来すると考えると、朝 6 時から夜 12 時までの 18 時間、平均で毎時 1 万人強の流出入が繰り返されている、巨大かつ、極めて流動的なシステムと捉えることができる。

このコンコースを観察すると、以下の特徴がある：

- (1) 固定のプレイヤーが固定の位置にすることが無い。すなわち、最も重要なプレイヤーである「人」は固定構造の中の固定コンポーネントとなっていない。
- (2) コンコース全体の安全を管理している管理者、所謂 Safety

Controller がおらず、かつ、安全を確保するための信号や交通整理の指示と言った安全制御機構が存在していない。

(3) 大規模公共施設のハザードとして、何を考えるべきか確定することが難しい（「衝突」のような即物的なハザードを分析対象としてしまうと、ミクロな衝突防止のメカニズムに解析が限定される恐れがあり、自宅の廊下のような少人数の場所での安全性と本質的に変わらない解析になる可能性がある）。

(4) 外乱に対し局所的な振る舞いはあるものの、いずれ系全体としては安定状態に遷移する。

本研究の概要は以下のとおりである。

第1章「序論」では研究の背景と目的を明らかにし、研究の概要を述べた。

第2章「従来の安全解析手法の問題点と新しい安全性分析評価手法」では、現在主流となっている安全解析手法を本研究の対象であるインテリジェントな不特定多数のエージェントから成るシステム（マルチエージェントシステム）に適用する際の問題点について述べる。

第3章「FRAM 分析」では、本研究において使用する FRAM（機能共鳴分析手法）の概要、及び、これを発展的に適用したモデリング方法、研究対象のモデリング結果、モデルの評価方法、研究対象のモデル評価結果について述べる。

第4章「シミュレーション」では、FRAM モデルから知見を取得するためのシミュレータの作成、及びシミュレーションによる分析結果を述べる。

第5章「対象システムの安全化に関する考察」では、本研究に強く関連する次世代安全理論について述べ、今後の発展の可能性について言及する。

第6章「結論」では、本研究の結果明らかになったマルチエージェントシステムの安全性分析・評価について述べ、今後の課題を明らかにする。

本研究で作成するモデルは、今後、一般道における自動運転など、インテリジェントなマルチエージェントが自由行動を行うシステムの安全化に応用することを意図している。

### 1. 3 参考文献

[1] TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU (2012)  
"Overview of the Internet of things", pp.2, 2012

[2] 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”, pp. 25-38, 海文堂出版, 2013

[3] JR 東日本, “各駅の乗車人員”  
<http://www.jreast.co.jp/passenger/>, 2018



## 第2章 従来の安全解析手法の問題点と新しい安全解析手法

### 2. 1 従来の安全解析手法

本論では、これまで提唱されてきた各種モデルを概観し、モデル化手法の変遷と安全工学の拡張が密接に関係してきたことを述べたい。

#### 2. 1. 1. ドミノ・モデル

ドミノ・モデルを提唱したのは、ハインリッヒである。彼によれば、事故は、以下のような構造で記述できる[1]：

- (1) 社会的または家庭的欠陥
- (2) 個人的欠陥
- (3) 不安全な状態または行動
- (4) 事故
- (5) 災害

災害・事故が発生するためには、不安全な状態・行動が必要。さらにそれを引き起こすのは、個人的欠陥、さらにその遠因にあるのは社会的・家庭的欠陥であるとする。図2-1は、それら5つのピースの関係である。

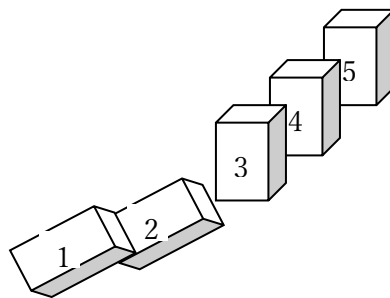


図2-1 ドミノ・モデル

3番のピースが倒れなければ（不安全な行動が無ければ）、災害（5番ピース）は発生しない

ドミノ・モデルは、事故の発生メカニズムが階層構造であるとするものである。具体的なモデルを作成する場合は、災害・事故をトップ事象とし、そこから事故原因に至る因果関係を定義する。

この考え方を解析手法化したものがフォルト・ツリー分析（FTA: Fault Tree Analysis）である。図2-2は、FTAで使われるダイアグラムである。

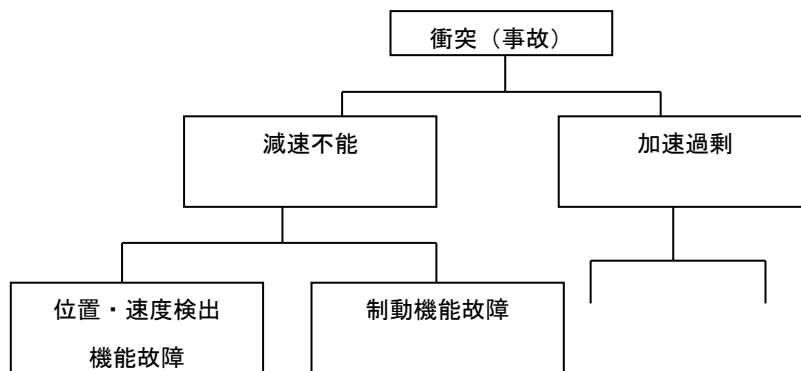


図 2-2 フォルト・ツリー

フォルト・ツリーは「故障(Fault)」のツリーである。ドミノ・モデルの基本思想は、小さな故障（ツリーの葉）が倒れることを発端に、徐々に大きな故障（ツリーの枝）に発展し、事故・災害（ツリーの幹）が発生するというものである。FTA は宇宙開発などで盛んに利用されており、その有用性が認められている。フォルト・ツリーの末端の葉全てを対象とし、それらの故障が発生した時にハザードに至らないようにする方法（ハザード制御）を、網羅的に実装することによって、ハザードの発生確率を最小化するという方法がとられている。

とはいえ、根源事象である末端の葉をソフトウェアのバグと結び付けることは困難であり、ソフトウェアの安全性解析には適していない。なぜなら、後にレブソンが主張したように、ソフトウェアは「故障」しないという、ハードウェアとは明らかに異なる性質を有しているからである。これについて次節で説明する。

## 2. 2. 新しい安全解析手法

### 2. 2. 1. STAMP モデル

ドミノ・モデルが事故の原因を故障によるとしたのに対して、レブソンは、事故は故障以外からでも起こるとして安全工学を拡張した。STAMP (Systems Theoretic Accident Model and Process) は、フィードバックループで記述される事故モデルと安全解析手法の総称である。STAMP モデルにおいては、事故は故障の連鎖ではなく、コンポーネント間のインタラクションの問題に起因して発生するとされる。[2]

事故が故障に起因しない最大の理由は、ソフトウェアが故障しないことにある。ソフトウェアは常に書かれた通りに動くため、ハードウェアのように劣化・故障しない。したがって、ソフトウェア起因の事故を FTA によって分析することは原理的に不可能となる。一方、従来の事故モデルは故障の分析を前提にしているため、新たな事故モデルが必要となったのである。

STAMP では、コンポーネントの故障を事故原因とせず、コンポーネント間のインタラクション不全を事故原因と捉える。コンポーネント間には必ずハザードを防止するためのアクションが有り、事故が発生するのは、そのアクションが想定通りに動作しなかったためであるとする。STAMP の事故モデルは、図 2-3 に示すように、Controller がハザードを制御し、Controlled Process が制御される形をとる。STAMP による安全解析は、まずハザードを定義し、この図を書くことから始めるのである。

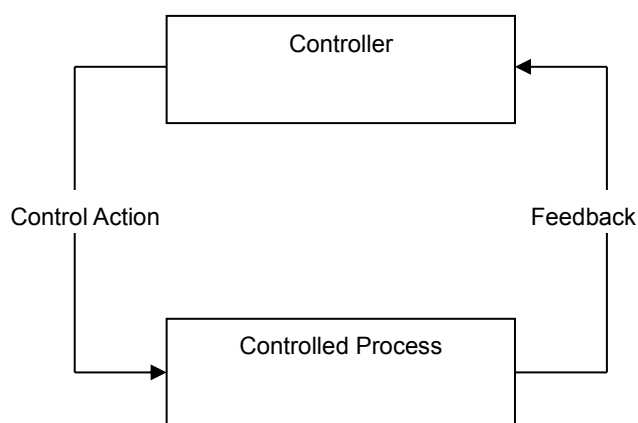


図 2-3 STAMP モデル

## 2. 2. 1. 1. STAMP モデルを使った安全解析

例えば、GPS が故障したことを見過ごすことにより、人工衛星を喪失するというハザード原因を FTA から導き、それを食い止めるハザード制御方法を考えるのが FTA による安全解析である (図 2-4)。

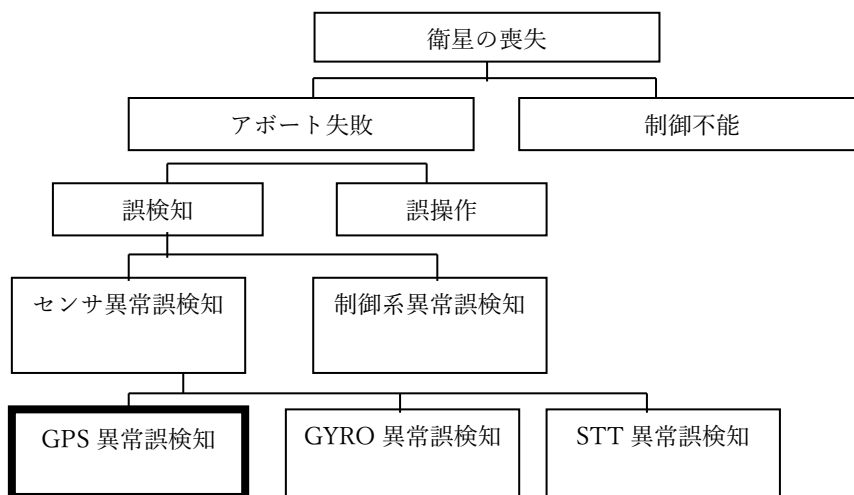


図 2-4 FTA によるミッション喪失原因の分析

一方、STAMP は、GPS が故障したとしても安全を確保できるようにするためのハザード制御が、どのコンポーネント間のインタラクションとして実装されているのかをモデル化する。(図 2-5)

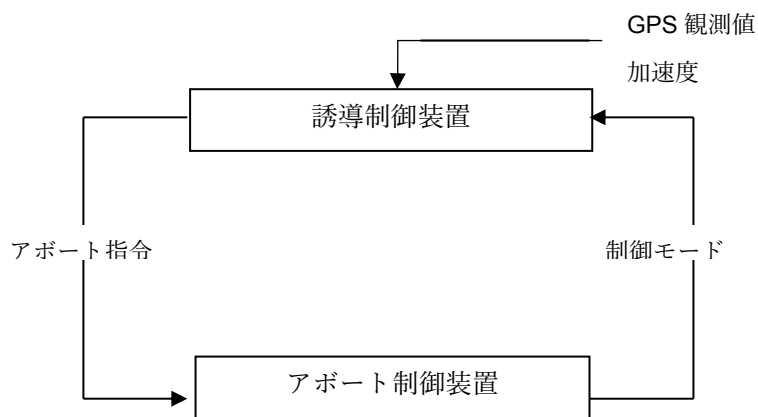


図 2-5 STAMP モデルの実例

ここでは、アボート指令という、安全を保証するために必要となるハザード制御 (Control Action) をピックアップし、それが実行される条件に使われているデータ群を識別している。具体的には、加速度、アボート制御装置の制御モードが、その条件として使用されている。

STAMP におけるモデリングとは、FTA のように、事故原因を直接詳細化して展開するのではなく、「事故を防ぐ仕組み」を定義する行為と言い換えることができる。

事故を防止するための仕組みがモデル化できたならば、次に、Controller から Control Action が想定通り提供されないという理由を分析するステップに入る。具体的には、Control Action が以下の 4 つのケースに陥るとハザードが発生すると考え、その原因を識別してゆく：

- (1) Control Action が提供されない
- (2) 間違った Control Action が提供される
- (3) Control Action の提供開始タイミングが悪い
- (4) Control Action の提供終了タイミングが悪い

この 4 つのカテゴリ分けにより、STAMP はソフトウェア起因のハザード解析を可能にした。ソフトウェア自体は故障しないため、機能不全を起こす原因は設計ミスである。しかし設計ミスの原因をすべて洗い出すことは、そのままでは極めて難しい作業となる。ミスの原因が無尽蔵にあるからである。

だが、「Control Action が提供されない要因」「誤ったものが (誤った条件で) 提供される要因」等の観点をとると、やみくもに設計ミスの原因を考えるのとは違い、カテゴリ毎に分析してゆくことができるようになる。そして、この 4 つのカテゴリ自体は、図 2-6 に示すように網羅的な構成となっているため、このカテゴリに従った分析に網羅性が与えられる。

提供されない	提供される	
	間違ったものが提供される	正しいものが提供される
	不適切な開始 タイミング	不適切な終了 タイミング

図 2-6 4 つのカテゴリの網羅性

たとえば、Controller である誘導制御装置の設計が、「宇宙機が加速している間は GPS 故障を検知しない」となっていた場合、Controlled Process

であるアボート制御装置側でも同様の条件が想定されているかどうかを分析する。想定されていない場合、意図せず Control Action が提供されない、あるいは誤った条件で提供されるというハザード発生原因が成立する。

ここで特筆すべきは、ハザード発生原因が機器の故障から導かれたのではないということである。ここでは、Control Action の実行条件についての Controller と Controlled Process 間の前提の相違をハザード発生原因と識別した。つまり、故障は無関係である。

STAMP では、まず、解析対象をハザード制御に係わる Control Action に絞ったうえで、この条件に使われるデータを図 2-5 のモデルに記述することによって、ハザード発生原因の発見をサポートする。「制御モード」データや、外部の加速度計から入力している「加速度データ」入力など、アボート指示の実行を左右するインタラクションをモデルに書き加えてゆくことで、モデル化を進めてゆく。しかしながら、本論で分析を試みる東京駅の安全に関しては、以下のような特徴があり、FTA/FMEA のみならず、STAMP でも解決できない課題がある：

- (1) 固定のプレイヤーが固定の位置にすることが無い。すなわち、最も重要なプレイヤーである「人」は固定構造の中の固定コンポーネントとなっていない。
- (2) コンコース全体の安全を管理している管理者、所謂 Safety Controller がおらず、かつ、安全を確保するための信号や交通整理の指示と言った安全制御アクションが存在していない。すなわち、STAMP における安全制御構造が存在していない。
- (3) 大規模公共施設のハザードとして、何を考えるべきか確定することが難しい。ハザードが定義できなければ、ハザード解析は始めることができない。すなわち、ハザード解析の実施が難しい。

これらの問題を解決するため、さらに新しい安全工学が提唱されはじめている。すなわち、次節に紹介する、レジリエンス・エンジニアリングのひとつ FRAM である。

## 2. 2. 2. FRAM モデル

レブソンは、事故原因はコンポーネント故障だけによって発生するのではなく、事故を防止する仕組みが失敗することによっても発生するとし、安全工学を拡張した。一方、ホルナゲルは、事故は失敗からだけでなく、成功からも発生するとして、さらなる拡張を行った。[3]

環境に応じて挙動を非決定論的に変える人工知能など、安全を確保するための固定の仕組みが想定しにくいシステムでは、どんな機能が安全を制御しているのか事前に絞り込むことは難しい。このようなシステムをも安全解析の対象にしてモデル化を行うためには、そもそも安全をもたらしている仕組み自体を発見する手法に工夫が必要となる。安全をもたらしている仕組み自体は固定の構造そのものであるとは限らず、構造は常に変動している中に創発的に現れては消える機能的な関係として捉えられるべきである。この考え方に基づいて開発されたモデル化手法に、機能共鳴分析手法 (Functional Resonance Analysis Method: FRAM) がある。図 2-7 は、図 2-5 と同等の内容の FRAM モデルの例である。

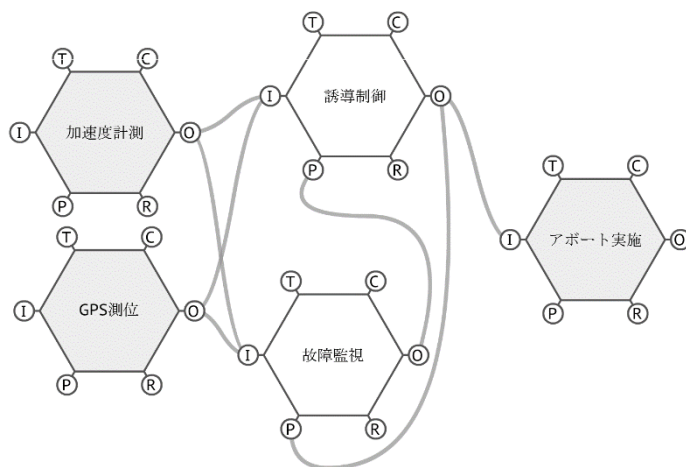


図 2-7 FRAM モデル

各六角形は、それぞれ「機能」を示している。FRAM 分析においては、事故が発生する原因を、FTA や STAMP のように特定の構造を持つモデルとして表現することは目指さない。FRAM 分析の目的は、環境変動に対して柔軟に対応し、安全をもたらす仕組み (成功要因) を明らかにすることである。さらに、その柔軟さゆえに、意図せぬ変動・共鳴となって逆に事故原因になるリスク要因を分析する。事故は失敗によって発生するとは限らず、システムの持つ柔軟な対応力が遠因となっても発生するという考え方である。

このようなリスクの典型が、ヒューマン・エラーであろう。人間は未知

の状況に対してもフレキシブルに対応できる柔軟性を持つがゆえに、単純なミスも犯すことがある。また、過度な反応や、エスカレーションという共鳴も起こしうる。柔軟さは成功要因であるとともにリスク要因ともなりうる。

同様に、人工知能システム等、柔軟性の高いシステムにおけるリスクを識別するためには、柔軟性が高く、成功に係わっている仕組みを知ることが、まず必要となる。

柔軟な環境変動への対応の仕組みをモデル化するために、どのような環境変動が入力されるのかを定義する。各入力は、図 2-8 のような 5 つの種類の入力に分類される。

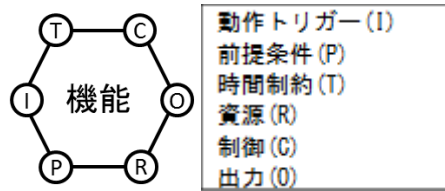


図 2-8 FRAM モデルの入出力記号の意味

これによって、単に「入力」とするよりも、変動への対応（出力）に与える影響をイメージしやすくなる。入力種別ごとの出力への影響を表 2-1 に示す。

表 2-1 各入力が出力に与える影響

I	処理のトリガー。即時処理が開始され、出力される
P	前提条件として常に I の前に入力されている必要有
R	暫時減少するため、枯渇により出力は停止する
T	時間制約により出力を許可・禁止する（非線形的効果）
C	出力値を直接変動させる

### 2. 2. 2. 1. FRAM モデルを使った安全解析

以下では、STAMP の事例と同内容の FRAM モデルを使って、安全解析を行う。安全解析は、上述したように、システムの成功要因から、リスク要因を割り出すという手順をとる。



### 2. 2. 2. 1. 1. 成功要因分析：

誘導制御と故障監視機能は、互いに相手方の機能実行の前提条件(P)を提供しあうという特徴を持っている(図2-9太線部分)。誘導制御がアボート実行を指示するためには、故障監視機能のチェック結果がなければならない。一方、故障監視機能も、監視を行うのは、誘導制御機能の制御モードが特定の状態にいるときのみとせねばならない。このように、互いに制約しあう関係があることにより、外部環境や内部状態の変化に順応してフレキシブルにアボート指示を実施することができる。

### 2. 2. 2. 1. 2 リスク要因分析：

誘導制御と故障監視の間の相互制約データは、両方とも「前提条件(P)」であるため、どちらの入力も自分の機能の開始よりも前に入力されなければならないが、どちらが先に入力されるのかは、モデル上不定である。どちらが先に処理を実施するかによって、以下のように2通りのリスクがある：

- (1) 誘導制御が先に動くとき、故障しているセンサの値を使って誘導制御してしまう可能性がある
- (2) 故障監視が先に動くとき、制御モードは受信したセンサデータよりも1周期古いデータなので、故障検知結果が意図せぬタイミングで誘導制御側に通知される可能性がある

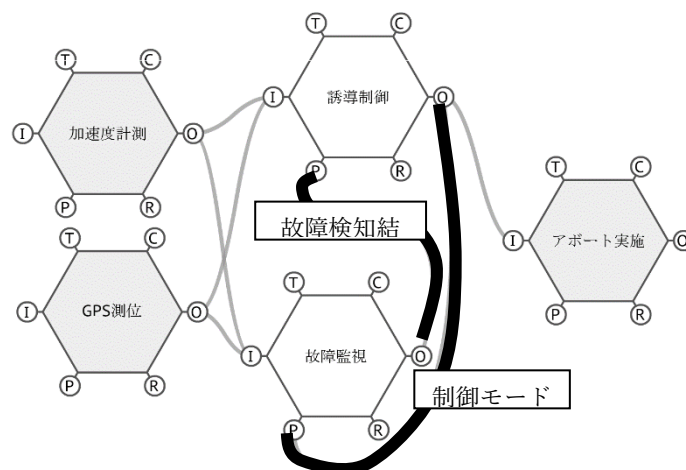


図2-9 誘導制御と故障監視の相互制約ループ(太線)

上記2つのリスクを比較すると、明らかに(1)のリスクは致命的である。したがって、故障監視が先に動くよう設計を進めるべきという分析結果が導かれる。かつ、1周期古い制御モードデータを扱っていることから、制御モードの変更直後の故障検知結果は、そのまま使わず、誘導制御側で1回棄却するなどの工夫が必要となることもわかる。この原因で事故が発生したのが、日本の人工衛星「ひとみ」である。ひとみは、姿勢制御に使用する高精度センサの観測値を故障検知より先に誘導制御側で使用したことにより、致命的な姿勢制御喪失を発生させるに至り、過回転により破壊した。[4]

また、図2-10のように誘導制御に2本入ってくる処理のトリガーは、どれか一つが予定通りに届かない時の処理等、柔軟に対応がなされる箇所であり、それゆえリスク源となる要素を含んでいる。たとえば、2つのデータの入力順序はモデル上不定であるが、もし順序に関して安全上の制約があるならば、それを守る仕組みこそが安全を保證している仕組みである。

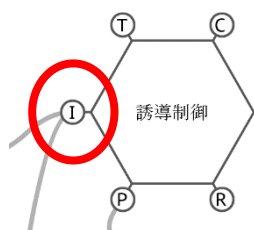


図2-10 順序不定の箇所の例

FRAM は柔軟性の高いシステムの分析に特に有用性が高い。この例では、システムの中で比較的柔軟性の高い箇所をモデルの特徴から特定し、その箇所の成功要因からリスクを導き出した。安全を守る仕組みを発見して初めてそれを脅かすリスクを導き出せるというのが、FRAMが「成功要因の分析による安全解析」を行う理由である。

### 2. 3. 事故モデルの変遷のまとめ

安全工学が、事故モデルの定義の変遷を通じて拡張されてきた歴史を概説した。ドミノ・モデルのような、コンポーネントの故障に着目していた安全工学が、ソフトウェアの複雑化に伴い、STAMPモデルのように、安全を保證する仕組みの不全に注目するようになってきた。さらに、人工知能等、安全を保證するための仕組みが見えにくいシステムの台頭に伴い、FRAM

モデルのように、機能間の共鳴という、よりダイナミックなインタラクションに着目する手法も登場しつつある。

加えて、安全解析の着眼点は、失敗原因から、成功原因へと拡張されてきた。これらの発展をもたらしたのは、「何をモデル化すべきか」という考え方の発展に他ならない。何が安全をつくっているのか、何が安全を脅かしているのかという問いへの答えは、モデル化手法の発展にともない、今後も拡張され続けると考えられる。

次章からは、本論文の目的である東京駅のコンコースの安全解析を新しい手法を使って実施する。東京駅の特徴として、安全を制御している主体が明確ではない（もしくは不在である）こと、システムへの人の流入が激しく、固定のシステム構造を定義できないことの2つの理由から、ここでは、FRAMを解析手法として採用した。ただ、FRAM分析は、これまでシステムの機能や事故の状況を表現することまでで留まっており、安全性の分析や解析にまで発展させた事例はなく、今後の研究が待たれていた。これに対し筆者は、FRAM分析によって明らかになったシステムの機能関係性を、一旦 STAMP の制御構造的な Controller/Controlled-Process 間のヒエラルキーに当てはめる STAMP と FRAM のハイブリッド的な手法を発想して適用した。その結果、機能関係が単なる Controller/ Controlled-Process 間の上下関係とは異なっているという分析結果が明らかとなり、東京駅の安全性をもたらしている要因を明らかにすることができた。

## 2. 4. 参考文献

[1] Heinrich, H., “Industrial accident prevention: A scientific approach”, pp.13-16, McGraw-Hill Book Company, inc, 1931

[2] Leveson, N., “Engineering a Safer World: Systems Thinking Applied to Safety”, pp.61-63, The MIT Press, 2012

[3] 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”, p.25-38, 海文堂出版, 2013

[4] 国立研究開発法人宇宙航空研究開発機構, “X線天文衛星 ASTRO-H 「ひとみ」 異常事象調査報告書 A改訂”, [http://www.jaxa.jp/press/2016/05/files/20160531\\_hitomi\\_01\\_j.pdf](http://www.jaxa.jp/press/2016/05/files/20160531_hitomi_01_j.pdf),

2016

## 第3章 FRAM 分析

本研究では、東京駅のコンコースにおいて自由な意思を持ったエージェント（人間）が整然と安全に移動できる状況を分析するため、レジリエンス・エンジニアリングの安全解析手法である FRAM (Functional Resonance Analysis Method: 機能共鳴分析手法) を適用する。FRAM 分析を用いることで、カオス的な状況にありつつも整然と推移できる隠れた構造が可視化できると期待している。東京駅の場合「プレイヤー」である人間は時々刻々入れ替わっても、そこに存在する目的の場所まで到達するという「機能」は変わらない[1]。

なお、東京駅コンコースにおける安全状態とは、パニックや渋滞、頻発する衝突がなく、エージェントがそれぞれ目的とする場所に到達できることとする。

### 3. 1 FRAM 分析の特徴

FRAM を使ったモデリングは、予測不可能な振舞いが支配的な状況において、シンプルなアプローチを可能にしている。以下にそのアプローチを示す。

- (1) まず、分析対象のシステムのうち、最も代表的、あるいはもっとも重要と思える機能を一つピックアップする。
- (2) 次にその機能が出力するものは何かを定義する。
- (3) 次に、その機能の実行条件を決める様々なパラメータ（動作トリガー、前提条件、資源、時間制約、制御パラメータ）が外部の機能から入力されているか否かを分析する。
- (4) 次に、その機能への入力を提供している他の機能についても、上記の(2)と(3)を行い、機能の相関関係を明らかにする。
- (5) 以上のように数珠つなぎに機能を追加してゆき、最後に、追加した機能が特に大きな変動要素もなく安定的に出力を繰り返しているものになるまで行う。このような機能をバックグラウンド機能と呼ぶ。バックグラウンド機能は、FRAM 分析の分析範囲の周縁である。

本論では、FRAM の公式ツールである FRAM Model Visualizer[2]を使用する。

### 3. 2 FRAM モデルの作成

東京駅コンコースの安全を分析するうえで一番注目したい機能を一つピックアップする。ここでは、コンコースの最大の目的である、「歩行者の移動」、すなわち、「歩く」という機能を取り上げてみる。

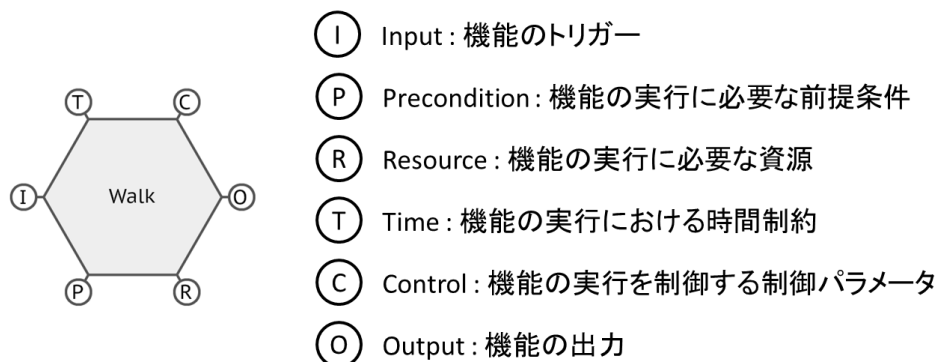


図 3-1 最初に注目する機能” Walk”

まず、「東京駅コンコースを歩く」という機能の出力は何かを考える。

次に、「東京駅コンコースを歩く」という機能を分析するときに5つの入力要素を考える：

- (1) 東京駅コンコースを歩くという動作のきっかけとなるトリガーは何か？
- (2) 東京駅コンコースを歩くという動作を行うための前提条件は何か？
- (3) 東京駅コンコースを歩くという動作を行うために必要となる資源は何か？
- (4) 東京駅コンコースを歩くという動作を行うための時間制約は何か？
- (5) 東京駅コンコースを歩くという動作を行うために使われる制御パラメータは何か？

歩くという機能の出力は当然、出される「一歩」である。

次に入力を考える。ここで注目すべき事実は、「歩く」や、「呼吸する」などのような、無意識で行われている機能においては、あらゆる要素が総動員されて動いていることが多いということである。歩行のためには、膨大な数の骨格筋が動作しているが、その動作指令には、大脳皮質、大脳基底核、小脳、脳幹、脊髄等、あらゆる制御系が協調しており、それらが視

覚、体性感覚、前庭感覚等、様々な感覚情報を統合している。すなわち、文字通り五感を総動員した機能という特徴を有している[3]。したがって、このような機能については、上図のように(1)から(5)までの全ての機能側面に対するに何らかの入力がある複雑なものとなる。

こうした機能の特徴をモデル化するパラメータは、表 3-1 のような、網羅的な構成となっている。

表 3-1 FRAM のモデル要素の全体構成

カテゴリ		データ名	説明		
入力	トリガー	I: 入力	機能を開始する直接のトリガー		
	事前条件	P: 前提条件	機能開始前に揃っている必要がある条件		
	事後条件	C: 制御	機能開始後に処理を変える	出力の中身を変える	制御パラメータ
		R: 資源		出力の有無を変える	資源枯渇時出力停止 (空間的側面)
T: 時間		締め切り時間オーバー時出力停止 (時間的側面)			
出力		O: 出力			

まず、機能には入力と出力が紐づいている。機能に対する入力には、大きく2種類がある。すなわち、機能が発生させるきっかけとなる入力「トリガー」と、機能の動作のあり方を変動させる「条件」の2種類である。このうち、「トリガー」が INPUT と呼ばれるパラメータに相当する。条件は、「事前条件」と「事後条件」の2種類に分割可能である。このうち「事前条件」は、それが存在しないとそもそも機能が開始しない。つまり、事前条件は、機能の動作・非動作をデジタル的に決定するパラメータである。FRAM では、これを PRECONDITION パラメータと定義する。

一方、「事後条件」には、PRECONDITION のように、動作・非動作をデジタル的に決定するパラメータと、動作の方法を細かくアナログ的に決定するパラメータの2種類に分けて考えることができる。動作途中で機能を継続するか、停止するか、というデジタル的影響を及ぼすパラメータとしては、動作の継続に不可欠となる「資源」と、動作の締め切りを決める「時間」パラメータがある。どちらのパラメータも、閾値を超過した場合は、それ以上機能が継続することができないという点で共通である。事後条件を構成する状態空間を時空間として捉えるならば、資源は空間的、時間は時間

的な尺度を持つパラメータとすることができる。一方、動作開始後に動作の出力値をアナログ的に変動させるパラメータも当然存在する。FRAM では、それを CONTROL パラメータと定義する。ここまで述べてきたように、FRAM の 5 種類の入力側面は、表 3-1 に見られるように、全体で機能の持つべき全ての側面を網羅できる体系を持っている。

以上のような網羅的な 5 種類の入力側面を定義することにより、ある機能に対する入力を過不足なくモデル化しようとするのが FRAM の手法の特徴の一つである。なお、それらの入力データは別の機能から出力されると考える。つまり、FRAM は、分析対象システムに存在している機能間の関係を網羅的にモデリングする手法とすることができる。

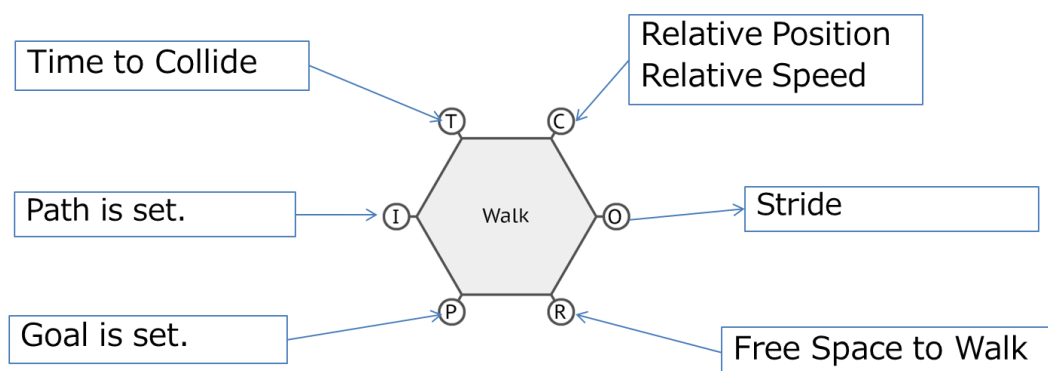


図 3-2 Walk 機能への入出力

東京駅における「歩行」機能に対する 6 つの側面は、上図に示したように、多種多様である。以下、それぞれの側面について記述する。

### 3. 2. 1 出力 (OUTPUT)

東京駅のコンコースを歩くとき、踏み出す一歩 (Stride) が出力となる。歩行は、この「一歩」をリズムカルに発生させるパターン発生器 (Central Pattern Generator: CPG) と呼ばれる脊髄の中の機構を中心に、指令系である脳、アクチュエータ系である筋肉、センサ系である各種感覚受容器の間の複雑なネットワークによって制御されている [3]。



### 3. 2. 2 前提条件 (PRECONDITION)

出力である「歩」を踏み出す大前提は目的地があるということである。目的地があって初めてそこへ向けた歩行が行われる。目的地が無ければ歩行は停止する。このようなデジタル的な意志決定要因となり得るのは、PRECONDITION と INPUT である。目的地が無ければ歩を踏み出すことは無いが、目的地さえあれば、必ず歩を踏み出せるかということ、東京駅の特殊事情により、そうはならない。すなわち、目の前が混雑しており、歩を踏み出すエリアが無い場合には、たとえ目的地があっても歩を踏み出すことはできない。つまり、目的地があれば、即時歩行できるわけではない。目的地があることは、歩行を開始するための前提条件と考えるのが妥当である。最終的に歩を踏み出す「トリガー」となる情報は、事項に示す INPUT パラメータである。

### 3. 2. 3 入力 (INPUT)

「歩行」は前述したように、脊髄に存在するパターン発生器からのリズムカルな指令が断続的に生成されることによって実行される。つまり、このリズムカルな指令が毎回の Stride を踏み出すトリガー (INPUT) として筋肉に出力される。FRAM 分析では、その「指令」は何かを分析する。目的地が存在するという前提条件が成立している場合、それが提供されると即座に歩行が開始される指令とは、どちらの方向に向かって歩を踏み出すべきかを指示する脳からの指令である。これを Path 情報と定義した。つまり、自分を中心とする座標系において、目標に近づくための次の歩のベクトルである。このベクトル情報が入力されない限り、目的地があろうとも、決して歩を踏み出すことはできない。Path は INPUT となる。歩行は、パターン発生器が生み出す信号によって生み出される連続的な運動であるため、目的地の有無という静的な情報ではなく、歩一歩更新され続ける最適な速度・方向を指示する指令をトリガーとして機能し続ける。したがって、INPUT は、最新状況を加味して計算された「最適 Path」が該当する。

### 3. 2. 4 資源 (RESOURCE)

次に、歩を踏み出す際の事後条件を考える。東京駅のような雑踏では、

Path が設定され、歩行が開始された後で、歩行を中断せざるを得ない状況が検知されることがある。すなわち、人との相対距離、相対速度が危険状態に達した場合、これを検知すると我々は、歩行を停止して安全化を図る。我々は目的地へ向かう最適 Path 周辺の安全エリアを常に監視している。安全エリアに進入してくる可能性のある相対速度や相対加速度をもった人間の動作と位置関係から、安全エリアが極端に小さくなった場合、近い将来にそこが人でふさがれることを予測して歩行停止するというインテリジェントな反応をする。つまり、Path 上の安全エリアの存在は、歩行を継続するか否かを決定するパラメータとなっている。さらに、安全エリアは、面積が状況にしたがって増減し続ける性質を持つ。安全エリアが潤沢に存在する時間帯と枯渇に近づくラッシュアワーというような増減である。局所的に安全エリアが減少している箇所等も、随時発生する。したがって、これを、同様に増減する「資源」と定義可能である。安全エリアという資源があるうちは、その中で最適パスにそって、自由に行動するが、ひとたび安全エリアという資源が閾値を下回ると、安全化のために歩行を停止する。この安全エリアを Free Space と定義する。実は、第 4 章のシミュレーションで述べるように、この Free Space の判定閾値が、システムの経済性・安全性を決定づける重要なパラメータとなっている。

### 3. 2. 5 時間 (TIME)

資源である安全エリアという「空間的」パラメータと同様、歩行を停止する可能性のあるパラメータとして、もう一つの監視項目がある。それが、「時間的」パラメータである。雑踏の中を歩く上での時間的制約として最大のもは、周囲の人間との時間的余裕、すなわち「衝突までの時間」である。Time To Collision (TTC) と呼ばれるが、我々がほぼ無意識に監視しているものには、空間的な余裕と同様、時間的な余裕度がある。視界に入っているあらゆる歩行者のベクトルを観察し、自分の設定している Path 上の、特定時刻における自分の予測位置と他者の予測位置が一致するとき、その状態に至るまでの現在時刻からの時間が計算され、その余裕がある一定値を下回る場合は、緊急的に歩行が停止される。これら空間的余裕度の計算と、時間的余裕度の計算は同時に行われ、歩行機能に対して「時間制約」と「資源」を並列的に提供している。

### 3. 2. 6 制御 (CONTROL)

さらにそれらを計算する際に使用される他者との相対位置・速度・加速度等のデータは、歩行の継続・停止を決定するだけでなく、歩行スピード、方向の微調整にも利用される。この微調整によって時間的余裕度、空間的余裕度が適切な範囲に保たれ、歩行を停止することなく継続し続けることができるように働く。相対位置・速度・加速度は、時間的・空間的余裕度計算への入力となると同時に、歩行機能の CONTROL 入力として動作ベクトルのアナログ的調整に使われている。

### 3. 2. 7 モデル図の完成

上記のような関係をモデルに現すと、下図のようなネットワークとなる。

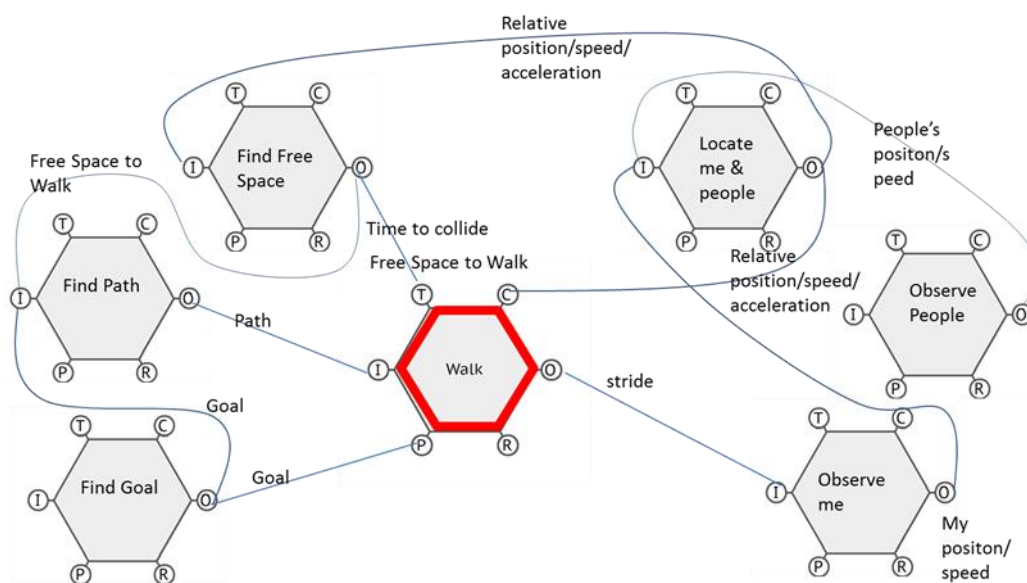


図 3-3 完成した FRAM モデル

FRAM のモデリングは、「歩く」機能のネットワークがバックグラウンド機能に到達し、分析対象範囲をカバーし終えた時、完成する。バックグラウンド機能とは、その入力条件について特段の分析を要さないような安定した動作を行う機能のことである。たとえば、「Find Goal」という機能は、歩く人が自分のゴール地点を探すという機能である。東京駅においては、この機能はきわめて単純なものである。すなわち、駅構内の随所に設置されている行き先案内板を探して、そこに指示された場所をゴールと見定め

るという機能となる。行き先案内板の見え方がある状況下で大きく変わる可能性はほぼ無く、また見つけるという動作に特別の能力やスピードを要することも無い。行き先案内板は、駅構内のあらゆる場所に設置されているからである。このように、機能の実行条件による変動を考慮するまでもない単純な機能の場合は、その機能への入力定義を省略し、そこで機能間の連鎖の追及をやめることができる。たとえば、単純なセンサ機能等がその典型となる。温度を入力するセンサの動作に特別な条件分岐が無い場合は、センサ機能はバックグラウンド機能となる。

バックグラウンド機能に達した時点で、その経路上のモデリングは終わりである。つまり、バックグラウンド機能で囲まれた範囲が、FRAMの解析範囲として後から決まる。ここが通常モデルベースの解析と異なる点でもある。すなわち、通常は、解析範囲をあらかじめ定めた上で、その範囲をモデル化する。FTA、FMEA、STAMPもそのような方法で解析を行う。

しかし、その方法には、一つ弱点がある。解析範囲を事前に限定するため、システムの動作に大きな影響を与えないと想定して最初から分析対象外とした部分が原因で発生するリスクや成功要因を見逃してしまうという弱点である。システムが想定外事象によって惨事に至る例は、東日本大震災をはじめ枚挙にいとまがない。むしろ、事故というものは、想定していない事象を起点に発生するので、想定していたハザード制御が働かないのだという側面が強いとも言える。その点、FRAMは、解析範囲を事前に想定せず、入力はバックグラウンド機能として特別に単純なものであると宣言したもの以外については、どこまでもモデリングを続けることが要求される。ツールの機能として、入力データを定義しているが、そのデータを出力している機能が存在しない場合は、モデルの完全性の欠陥として警告が表示され、モデリングを完了することができない。このような仕組みによって想定外を排除するため、モデル化し損じた想定外の入力が原理的に発生し難い。FRAMモデルが何の入力も定義しない機能は、上記の”Find Goal”のように実行条件が存在しない単純な機能であり、そうした機能には、想定外の条件入力というものがそもそも存在しえない。

上図のモデルでは、Find Goal機能がバックグラウンド機能、他の機能は、他の機能との間でループ構造に組み入れられており、この範囲で歩くことについてのネットワークが完結している。

### 3. 3 FRAMモデルの分析

作成した FRAM モデルは、FRAM Model Visualizer の機能を使って縦横無尽に構成を変えながらその特徴をとらえてゆくことができる。一見スパゲッティ状の複雑なネットワークであっても、機能間ネットワークである以上、純粹にランダムなものではなく、人工的なネットワークであれば、設計者の設計思想を反映した「設計アーキテクチャ」、自然物のネットワークであれば、長年の進化の過程で練り上げられてきた「創発的アーキテクチャ」を持っているはずである。そうした構造を探り出す作業は、FRAM 分析の中でもっとも重要なものであり、過去に同様のシステムを設計、または分析した経験を持つアナリストにとっては、経験を生かした分析能力の発揮のしどころであり、そのような経験を持たないアナリストにとっては、自らシステムアーキテクチャを創造、発見する建設的作業である。アーキテクチャが見えにくい複雑なネットワークは、接続ラインの長さ、交差数等、ネットワーク構造を複雑にするパラメータを最小にする機能ブロックの配置を最適解とみなし、ツールを使ってそれを見つけ出してゆく。

### 3. 3. 1 接続ライン長・交差数の最小化

図 3-4 のようなモデルがある場合、モデルの構造が見えにくい。この構造は、実は、階層構造なのであるが、機能の配置が不適切で、不必要に接続ラインが長い、または、不必要に接続ラインが交差しているため、真の構造（階層構造）が隠ぺいされている。

しかし、このモデルの配置を調整することにより、図 3-5 のように、真の構造を発見することができる。

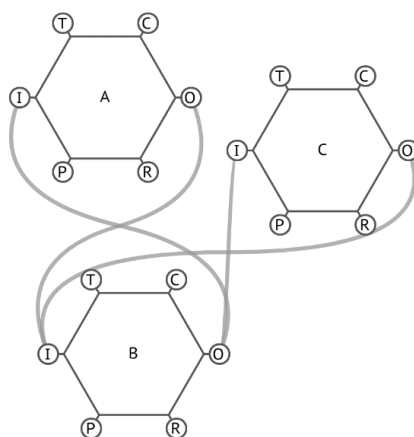


図 3-4 複雑で真の構造が隠ぺいされている配置

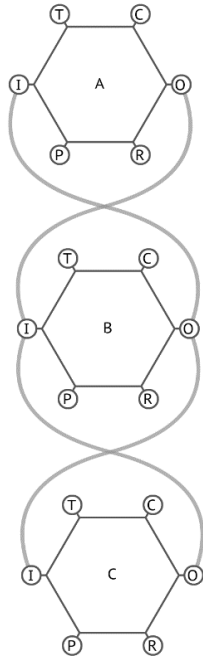


図 3-5 単純で真の構造（階層構造）を表現している配置

図 3-3 で作成したモデルの機能ブロックの配置を調節し、不必要に交差している線や不必要に長い線を排除すると、図 3-6 のような配置となる。

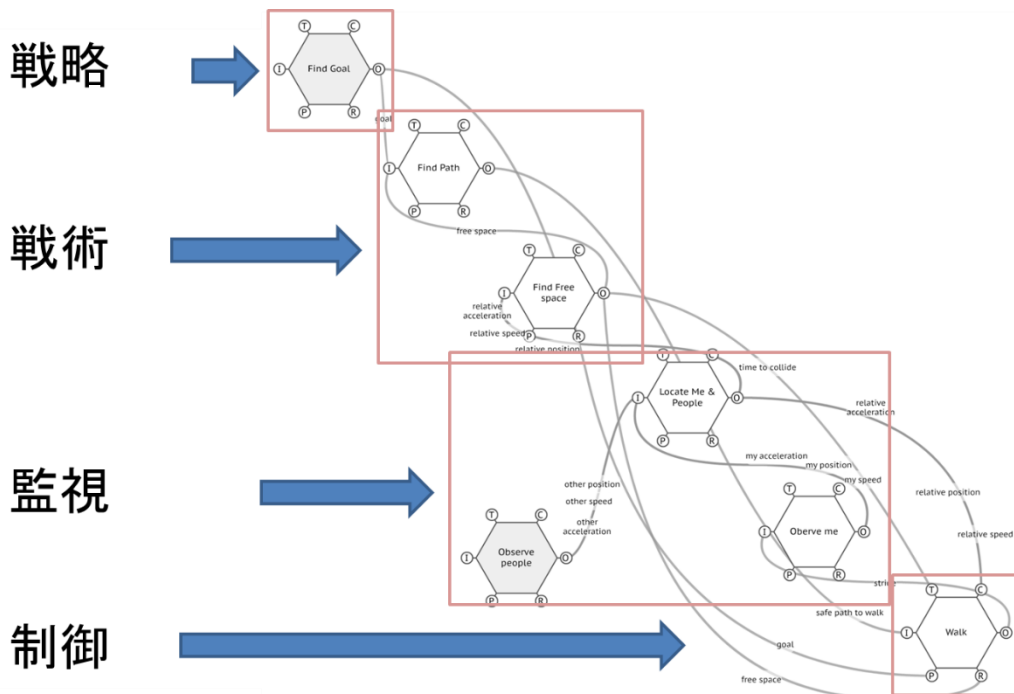


図 3-6 東京駅の FRAM モデルの構造

コンコースにおける歩行の機能ネットワークは、並べ替えを工夫することにより、上図のような階層構造と捉えることができる。ここから作成したモデルの評価を行ってゆくにあたり、FRAMを使った安全評価は、FRAMの工学的なバックボーンとなっているレジリエンス・エンジニアリングを抜きにしては語ることはできない。レジリエンス・エンジニアリングは、第2章で述べたように、従来の安全工学と異なり、システムのリスクやハザードが、失敗要因からではなく、成功要因からも発生すると考える。したがって、FRAMの安全評価作業においては、まず、対象システムの成功要因、あるいは、優れた特質を分析することから始める。東京駅コンコースのように、長年の運用を経て少しずつ練り上げられた人工物であれば、明治時代に設計された設計思想そのものではなく、運用実績をフィードバックして少しずつ改良を加えられた結果として現在の形となっており、かつ、時代の移り変わりや利用者の嗜好等を反映して、常に変化し続けているはずである。また、その結果作り上げられたコンコースのシステムは、現在、極めて安全で効率的なものになっており、成功要因の分析対象として優れた例題となっている。

### 3.3.2 東京駅のFRAMモデルの構造的特徴（STAMPモデルとのハイブリッド分析）

上図に示したように、4つのレイヤーに分割されていると捉えると、各レイヤーは、以下のような内容を持っている：

- (1) 戦略層：歩くという行動の最終目的を提示する
- (2) 戦術層：どこへ向かって、どのようなコースで歩くべきかを提示する
- (3) 監視層：戦術を達成するために自分の動きと群衆の動きの相対関係を監視
- (4) 制御層：歩くという物理的な行動を行う

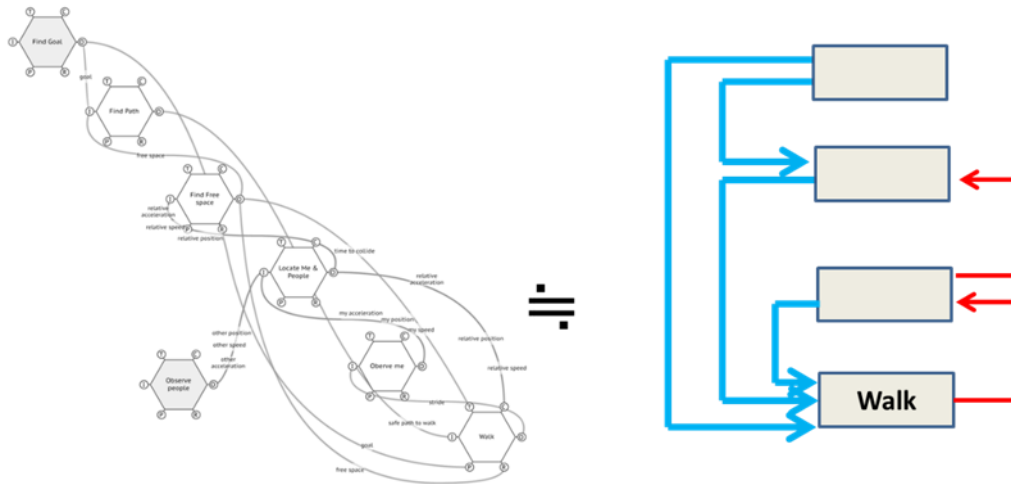
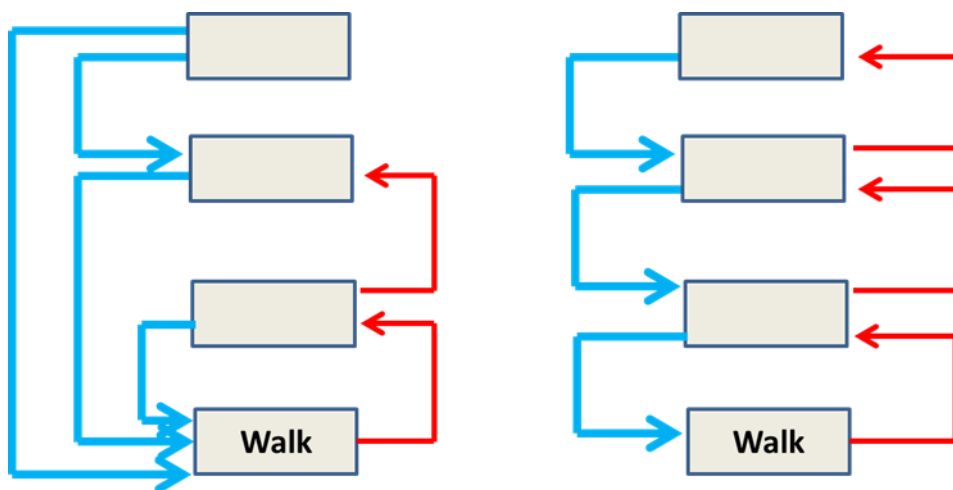


図 3-7 FRAM モデルを STAMP モデルに簡略化したもの

次に各層の間を関係性を分析する。各機能からの出力は、一つ上の上位層に伝達されているものとそうでないものが混在している。特に特徴的なのが、最下層の制御層に多くの入力集中しているという点である。この分析により、東京駅における歩行は、2章図2-5で紹介した STAMP の解析に頻繁にみられる階層的フィードバックループ構造[4]とは異なることがわかる。そこで、どれほど階層的フィードバックループ構造と異なっているのかを下図に示した。



東京駅の歩行

通常の階層制御

図 3-8 東京駅モデルと STAMP 的階層構造の違い



上図の左側が FRAM モデルの構造。右側が STAMP 的階層的フィードバック構造である。右側の STAMP 的なフィードバックの階層構造はトップダウン型の制御とすることができる。上位層は常に下位層をコントロールするために Control Action を出している。下位層から上位層へは、その Control Action からのフィードバックが行われ、フィードバック制御ループ構造を作っている。

一方左側の構造は、トップダウン構造ではない。レイヤー構造らしく上位層へデータがリレーされてゆく構造を持つてはいるが、上位層各層からは、出力が最下層の制御層に集中している。すなわち、下から吸い上げてきた情報が、随時さまざまな形で制御に使われているという構造である。かつ、上位層から降りてくる情報は「制御」指令ではなく、制御に使用するパラメータや、安全エリアや時間余裕の値などのような、参照情報がほとんどである。すなわち、制御の主体は最下層にあり、上位層はサーバーとしてサービスを提供しているだけであるといえることができる。これは、むしろボトムアップ構造となる。

このようなボトムアップ構造においては、戦術層は戦略層からトップダウンに決定されるのではない。むしろ、歩き方が戦術を決めている。戦術がボトムアップに決定され、環境変動に適応して自動的に修正され続けるという構造になっていると言い換えることも可能であり、まさに、これこそが、東京駅の歩行モデルの最大の特徴となっている。東京駅のコンコースが、何故現在のような単純な自由通路という形態に収斂してきたのか、そして、それによって何故膨大な人数が流出入する公共施設が大きな事故もなく毎日の様々な異常事態にも適応して極めて安全な場所として機能し続けることができるのかは、次節に示す成功要因分析の中で明らかにしたい。

また、興味深いのは、最上位層からは一方的に情報が出されるだけであるということである。ここだけオープンループになっており、従来型の制御構造とはさらに異なっている。何故、最上位層の「戦略」は、下位からのフィードバックを受けない一方通行で戦略情報を提供し続けているのかに関しても、次章の成功要因を分析する材料となっている。

### 3. 3. 3 成功要因の分析

前節までで、モデルの評価のための情報整理（機能の並び方の調整や、大まかな構造の理解）が終わると、いよいよ、レジリエンス・エンジニア

リングの主眼となる「成功要因」の分析に移る。成功要因を識別するためには、まずシステムの持つユニークな特徴を識別する必要がある。すべてのシステムには、それが達成しようとしている目的があり、その目的の達成（成功）のために、各システムは独自の性質を有している。この性質がシステムの特徴である。FRAM 分析によって明らかにするのは、その性質、すなわち、成功要因である。本研究においては、安全性の分析を対象としているため、東京駅が安全な状態にあることがどのように達成されているのかを分析する。システムの性質を明らかにする方法には、さまざまな方法が考えられるが、FRAM が提供するものは、たとえ東京駅のようにシステムの構造が流動的であっても、普遍的な性質を持つ「機能」を抽出しようとするものである。前章における制御構造の特徴の分析から、東京駅における歩行の特徴を以下の2項目として識別した：

- (1) 一旦ゴールが設定されると、ボトムアップに練り上げられる戦術にしたがって皆が歩き続けることができる。戦術は、常に自動的に参加者全員によって環境変化に適応して更新し続けられる。戦術を決めるのは、ルールや交通整理によるトップダウンプロセスではなく、参加者によるボトムアッププロセスである。(ボトムアップな戦術の自動的更新)
- (2) ゴールは環境に影響されず、常に安定的に提供されている。(安定的な戦略)

東京駅コンコースを行き交う人の流れを定点撮影すると、図 3-9 に見られるように、通行人は、1 秒毎に撮影された各フレームで、それぞれ別人に入れ替わっているほどの高い流動性を示している。つまり、東京駅のコンコース全体としては、各人の戦略、戦術、制御は全て異なり、数秒先の状況を正確に予測することは困難である。



図 3-9 東京駅コンコースの定点撮影（4 秒間毎秒撮影）

一方、FRAM モデルを見ると、各人のとるべき戦術はボトムアップに更新され続ける仕組みが出来上がっているため、集団全体は戦術を失うことなく、安全に歩き続けることができる。各人の作り上げる戦術はそれぞれ異なるが、集団にとって最適な戦術に更新され続けているため、統一行動ではないが集団を安全にする戦術は維持されている。図 3-9 のような目まぐるしく変化する状況の中で、常に安全な経路を計算し、膨大な数の他者との相対運動を並列計算しつつ、最短時間で目的地に向かうという、極めて高度な行動を、我々は苦も無く安全に実施している。ここで実現されている安全化は、前もってトップダウンプロセスに基づき計画された安全化ではなく、集団自体のボトムアップな自己組織化プロセスによって作り出されていると考えることができる。東京駅の特徴として、そのような計画に基づく安全化行動を行う交通整理者やルールが不在だからである。

また、安定的な戦略情報の提供という、このシステムの特徴の一つも明らかになった。FRAM モデルから明らかのように、最上位の戦略層へは下位からの入力がない。すなわち、戦略は常に固定のものが、日中における太陽光のように安定して提供され続けている構造となっている。FRAM の用語で言うと戦略を提供する機能は「バックグラウンド機能」という、安定的な機能である。構内の混雑状況など、歩行者が監視しているパラメータを入力して戦略が書き換えられるということが無く、戦略は常に不変である。東京駅で安定的に常に示され続けている戦略情報とは、行き先案内版によるゴールの表示である。東京駅に関しては、この表示が変わることが無いことはもちろん、どのような場所からでも、全ての場所への案内板が必ず視野に入るように、夥しい数の案内板が設置されているという特徴も備えている。すなわち、東京駅における戦略層の安定とは、ゴール自体が不変であるという論理的な安定性と、ゴールが、あらゆる場所から参照できるという空間的な安定性の両面を備えている。加えて多国語表示や理解しやすい色やアイコンを使用したユニバーサル・デザインの採用等によって、様々な側面から隙の無い安定性が追及されている。

### 3. 3. 4 ボトムアッププロセスによる同様の行動



図3-10 滞留している乗客と周囲の乗客

ボトムアップに練り上げられる戦術というものは、極めて柔軟であり、環境変動に即座に対応する。たとえば、団体客がコンコース内に滞留しているときには、歩行者全員が無意識のうちに戦術を変更してコースを変えてゆく。誰も交通整理をしていないからこそ、このような安全化が可能となる。もし、交通整理を行う者が群衆全体の流れを統制しようとするれば、単一の交通整理者が、数百人単位の歩行者全員の相対距離・速度・加速度を全て把握する必要がある。しかし、仮に100人程度の人数の交通整理であっても、100人の保有している相対距離データは $100 \times 100 = 10000$ 件に上り、それらが数秒後にどのような状態に変わるのかを現在の各人のベクトルから瞬時に予測しつつ、流れ全体を最適に導くことは一人の人間には事実上不可能であり、結果的に交差点の信号制御のような単純な方法を選択せざるを得なくなる。もし交差点の信号制御に依存して東京駅の交通整理を行うならば、常に全体の半数は赤信号待ちとなるため、一度に動ける人数は約50%まで縮小し、現在の利用者数を維持することは不可能となる。ここに待ち行列特有の連鎖的な渋滞[5]が発生すれば、東京駅の交通は破たんすることとなる。つまり、集団パニックのような危険な状態に容易に遷移し得る。

このように、ボトムアップな戦術の更新は、全体を統制することが不可能な複雑なシステムにおいて有効であることは、東京駅という、長い歴史の中で改良され続けてきたシステムの実績と、現在の姿によって、証明されているといっても過言ではない。

同様な研究事例として、レイノルズの「群れのルール」の研究がある。  
[6]

レイノルズはシミュレーションによって、鳥の群れは、以下の3つの要

因だけで統一のとれた団体行動をとることができることを示した：

- 分離 (Separation)  
鳥オブジェクトが隣の鳥オブジェクトと衝突しないように距離をとる。
- 整列 (Alignment)  
鳥オブジェクトが隣の鳥オブジェクトと速度を合わせる。
- 結合 (Cohesion)  
鳥オブジェクトが近隣グループ中心へ位置を近づける。



図 3-10 レイノルズによる鳥の群れのシミュレーション画像

Craig W. Reynolds, (1987), Flocks, Herds, and Schools: A Distributed Behavioral Model 1, Computer Graphics, 21(4), p.25-34.  
ACM SIGGRAPH '87 Conference Proceedings, Anaheim, California, July 1987.

それぞれの鳥は、各々、隣の鳥との距離・速度を保ち、近隣小グループの中心方向を指向する。これらは、特に上位コントローラが存在しなくても可能な制御である。つまり、東京駅と同様にボトムアッププロセスによって創発的に実現されている。鳥の群れには、明確なリーダーが存在しなくとも、頻繁にみられる統一のとれた編隊飛行が可能となっている。東京駅では各々が異なる目的地を持っているが、鳥の群れは、餌の捕獲や外敵からの防御等、群れとしての共通ゴールを本能的に共有している。一方、東京駅の群衆の行動は、鳥の群れが同じ方向へ向かう行動とは異なり、それぞれ

のオブジェクトは、それぞれの目的を持っている。目的が異なり、統一的な行動をとる必要は無くとも、それぞれがボトムアップに戦術（歩き方・飛び方の決定）を練り上げてゆくという点は同様である。東京駅の群衆の行動は、それぞれが異なる利己的な目的に向かってそれぞれの好みの戦術のもとに行動しているが、「結果的にそれが最も安全な行動」になっている。

東京駅や鳥の群れのような、明示的に制御・防御されることを要さず、危険な状態に陥っても創発的な自己組織化によって自ずと安全に向かうような安全性を、安全工学上の用語で「先天的安全性」と呼ぶ[7]。先天的安全性は、制御による Active な安全、防御による Passive な安全を上回る、安全性における最高の状態を示すものである。制御や防御のための資源やルールを必要とせず、集団は、放置されていれば自然に安全に向かう。

### 3. 3. 5 リスク要因の分析

前項で見てきた成功要因は同時にリスク要因にもなりうる。ボトムアップに戦術が決定されるということは、環境変動に適応して柔軟に戦術を変えられる反面、列車乱れによる番線変更や運休などにより一旦ゴールが提供されなくなる、あるいは、ゴールが変化すると、歩行の前提条件が失われ（下図）、無目的に徘徊する大量の歩行者を生んでしまう。結果としてコンコースは即座に渋滞し始める。

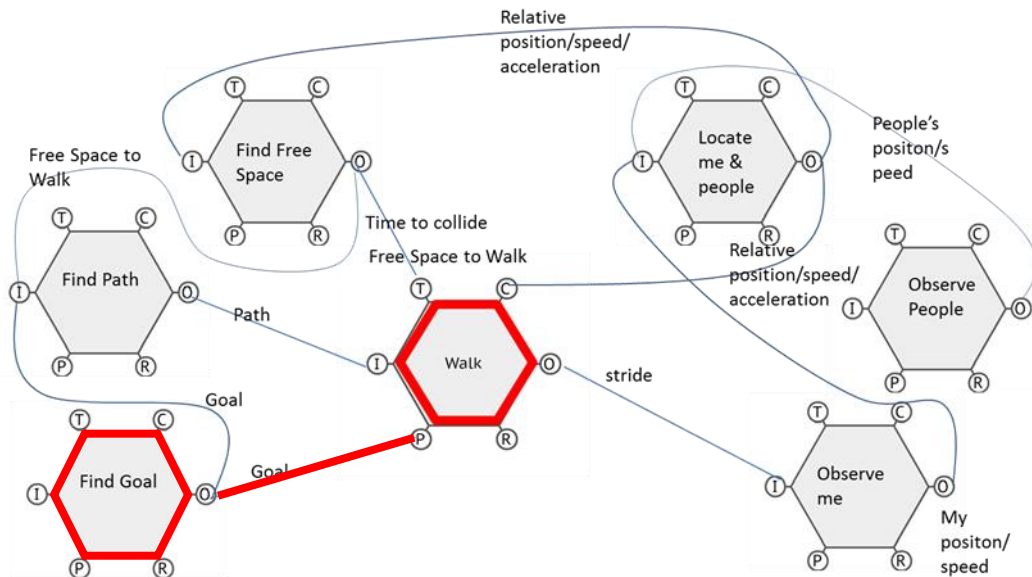


図 3-11 ゴールの提供機能と歩行機能の関係（赤線）

もし東京駅でこのような状態が発生すると、局所的な渋滞が発生することになる。そして、それは、新規流入者によって、待ち行列に特有の伝搬を経てより広範囲な渋滞につながる。これこそが東京駅のハザードである。FRAMはハザード解析手法ではないため、解析前にハザードを定義することをしない。FRAMにおける安全解析は、事前に解析対象となるハザードを定義するのではなく、解析の結果としてハザードが明らかになる。したがって、本研究における「安全」とは、人と人とが衝突せずに移動できるというだけでなく、衝突を回避するために立ち止まる回数の増加による渋滞が引き起こされにくいことである。

ゴールが提供されるか否かは個々人からのフィードバックによらずに一方向的に決まることであるため、この機能は、ひとえに駅の設定によって実現されている。コンコースにおけるゴールを提供する機能とは、すなわち行き先案内板の表示機能である。東京駅を観察すると明らかにわかるのは、コンコースのどこにいても、必ず案内板が見えるように、夥しい数が設置されていることである。「あらゆる場所から案内板が見えるようにする」ことこそが、東京駅のハザード制御となっている。このハザード制御は、長い東京駅の歴史の中で徐々に改良を受けて完成されてきた仕組みであり、歩行者が自己組織化によって安全になっているのと同様、創発的に練り上げられてきた工夫である。

東京駅のコンコースは、行き先案内板以外には、特に特徴的な安全フィーチャーを持っていない、単なる通路である。しかし、それゆえに、通行人の自由度が高く、最小面積に最大乗客数が出入りできるため、容易にパニックには至らず、結果的に極めて安全なシステムになっている。前述したように、信号機制御等を使った交通整理を東京駅に導入すると許容流出入人口の激減・パニックを招く可能性があるが、東京駅は、そのような非効率な安全制御を行わず、創発的な自己組織化による安全化に任せているため、ハザードになりやすく、高いレジリエンスを有している。

### 3. 4 分析結果のまとめ

東京駅の安全解析を、FRAMによって行った。FRAM分析によって明らかになった制御の階層構造の特殊性から以下の成功要因とリスク要因を導いた。

#### (1) 成功要因

- ・ 一旦ゴールが設定されると、ボトムアップに練り上げられる戦術にしたがって皆が歩き続けることができる。戦術は、常に自動的に参加者全員によって環境変化に適応して更新し続けられる。戦術を決めるのは、ルールや交通整理によるトップダウンプロセスではなく、参加者によるボトムアッププロセスである。
- ・ ゴールは環境に影響されず、常に安定的に提供されている。

## (2) リスク要因

- ・ ゴールの提供が滞ると、歩行が停止するため、新規流入者を受け入れることが困難となり、パニックに至る

上記の成功要因、リスク要因の分析により、東京駅を安全化する方策としては、以下が考察される。

## (3) 東京駅安全化策

- ・ 歩行者が自由・利己的に行動することによって先天的に安全化されているため、あえて交通整理等によって歩行の自由度を減殺しないこと。また、歩行区分の設置によって歩行可能領域を削減しないこと。これは現在すでに達成されている。
- ・ 行き先案内版を見やすく、どのような場所からも見える位置に十分な数量を設置すること。これも現在すでに達成されている。

以上、東京駅の安全性を FRAM 分析から述べた。

FRAM モデルは、現実の東京駅の状況を、機能を切り口にして解釈したものである。このモデルの正確性や妥当性は、機能を6つの側面に分けて定義する手法の妥当性と網羅性に依存している。表 3-1 で示したように、この6つの側面には、十分な網羅性と妥当性を含むと考えられるが、いかに網羅的なモデリング手法を使ったとしても、実際のモデルに、必要な側面を全て定義できているか否かは別問題となる。次章以降では、モデルの妥当性を検証し、成功要因として識別された考察を裏付けるため、モデルから作成したシミュレータを使い、FRAM モデルが示すボトムアップ構造のロジックで動作する多数のオブジェクトの動きを模擬した。

## 3. 5 参考文献



[1] Deleuze, G.; Guattari, F., "What Is Philosophy?" pp.118, Verso Books ISBN-10: 0860916863, 1984

[2] Hollnagel, E., "The FUNCTIONAL RESONANCE ANALYSIS METHOD - FRAM Model Visualizer (FMV) - "  
<http://functionalresonance.com/FMV/index.html>

[3] 青井, "歩行と走行の脳・神経科学", 市村出版, ISBN-10: 490210931X, pp.47, pp.100, 2013

[4] Leveson, N., "A STPA Primer", <http://sunnyday.mit.edu/STPA-Primer-v0.pdf> pp.13, 2013

[5] Sundarapandian, V., "7. Queueing Theory", Probability, Statistics and Queueing Theory PHI Learning, ISBN 8120338448, 2009

[6] Reynolds, C., "Flocks, herds and schools: A distributed behavioral model", SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques, Association for Computing Machinery, pp.25-34, doi:10.1145/37401.37406, ISBN 0-89791-227-6, 1987

[7] Heikkila, A., "Inherent safety in process plant design. An index-based approach", Espoo, Technical Research Center of Finland, VTT Publication, pp.384, ISBN 951-38-5371-3, 1999

## 第4章 シミュレーション

FRAM モデルは、他のモデル化手法と同様、アナリストの才能や経験によってフィデリティに差が出る可能性がある。したがって、モデルの妥当性を検証する手段を持つことが望ましい。FRAM は機能ベースのモデリング手法であるため、これを容易に実現可能となっている。本章においては、モデルの妥当性を検証することを目的に、モデルから生成したシミュレータを使い、実際の東京駅における群衆の行動と大きな乖離が無いことを確認した。シミュレーションの実行結果に対しては、定性的な傾向分析、及び、定量的な統計分析を行った。その結果は、前章に示した自然界における群れの行動との類似性を傍証するものとなった。

### 4. 1 FRAM モデルからのシミュレータの作成

FRAM モデルは、機能間のインタラクションを定義したモデルであるため、プログラミング言語における「関数」や「クラス」間の構造図に相当する情報を有している。つまり、FRAM モデルは、それ自体が実行可能な構成を表現しているため、モデルからシミュレータを作成することが可能である。

以下に、モデルからシミュレータを作成する方法を示す。FRAM Model Visualizer の機能を使用し、FRAM モデルを表形式で出力すると、以下のような6つの側面が機能毎に出力される。

----- FRAM Model -----

Function Walk

Input	safe_path_to_walk
Output	stride
Precondition	goal
Resource	free_space
Control	relative_position relative_speed relative_acceleration
Time	time_to_collide

---

上記の設計情報から、以下のようなロジックを実装することができる。

```
01 // Precondition がそろっているときのみ実行
02 If (goal != "") {
03     // Input があったら、処理開始
04     If (safe_path_to_walk != "") {
05         // Free Space が閾値以上であれば処理続行
06         If (free_space > threshold_fs) {
07             // TTC が閾値以上であれば処理続行
08             If (time_to_collide > threshold_ttc) {
09                 // Safe_path_to_walk で推奨された方向に進む
10                 Do_walk (safe_path,
11                         relative_position,
12                         relative_speed,
13                         relative_acceleration);
14             }
15         }
16     }
17 }
```

各関数間の入出力は、全て FRAM モデルに記述されているため、シミュレータの実装は、入力パラメータをどのように組み合わせるのかというロジック部についてのみ行うことになる。しかも、そのロジック部自体は、各入力パラメータの種類が決定されていれば、上記のように既に骨格は決定されている。たとえば、PRECONDITION は、処理を実行するか否かを最初に決定するための条件であるため、ロジック部の先頭で一度確認するとき使用する（1 行目）。続いて、処理を継続するか否かを決定する事後条件（RESOURCE 及び TIME）については、処理開始後のタイミングで確認するとき使用する（5 行目、7 行目）。最後に処理の制御パラメータは、出力値を計算するサブ関数のパラメータとして使用する（10 行目）。以上のルールに従って全ての主要な関数（FRAM モデルで「機能」名がつけられているもの）は、上記のような構成で統一的に作成可能である。これが、シミュレータのひな形となる。

## 4. 2 東京駅モデルのシミュレーション

前章に示した方法により作成されたひな形にロジックの追加を施し、シミュレータを作成した。これを実行し、FRAM モデルが実際にボトムアップに戦術を決定・修正しながら全ての歩行者を効率的に歩行させているか否かを検証した。シミュレーションでは、歩行者の数を任意に設定し、全ての歩行者が、それぞれに行き先を持ち、シミュレーションの開始と同時に一斉にスタートし、ゴールを目指す。行き先がない場合、その場で立ち止まる。途中、他の歩行者の動きが自分のゴールまでの Path と交差することを検知した場合は、衝突を回避しつつ、なるべく Path から外れない範囲で軌道の修正を行う。その際、歩行者は Free Space と呼ばれる他の歩行者と交錯しない仮想の安全エリアを維持しつづける。具体的には、制御パラメータである Relative Position/Speed/Acceleration を利用して歩く方向と速度を微調整し、Free Space が閾値を下回るほど小さくならないようにする。Free Space の面積の計算には、Relative Position/Speed/Acceleration から計算した衝突までの時間 (Time to Collide) を使用する。

シミュレータの実行に際しては、以下の条件を設定可能とした：

- (1) 歩行者の人数
- (2) 各歩行者の初期位置
- (3) 各歩行者の目的地
- (4) Free Space の評価閾値

なお、Free Space は、Relative Position/Speed/Acceleration 及び Time to Collide から決定される従属変数である (下図)。

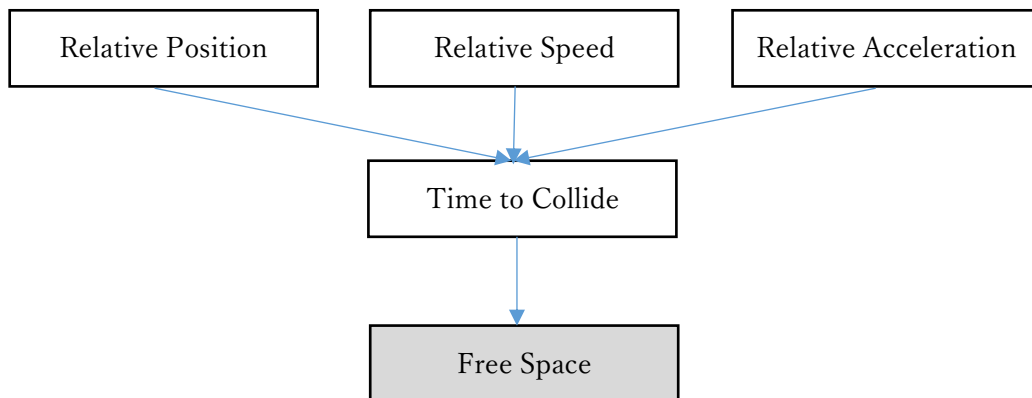


図 4-1 従属変数 Free Space を決定する独立変数群

Relative Position 等の独立変数、及びそこから自動更新される Free Space という従属変数がシミュレータの出力となる。歩行者モデルは自他の動きの観測から得られた Relative Position/Speed/Acceleration から Free Space を計算し、それが、ユーザが任意に設定可能な Free Space 評価閾値を下回るほど小さくなっていないことをチェックしながら歩行を継続する。以降、Free space 評価閾値を Minimum Distance と呼ぶ。

上述の4つの設定を可能とするシミュレータを作成し、各歩行者の行動を模擬した。行動の模擬を実施するに当たり、目の前の Free Space 面積が閾値 (Minimum Distance) を下回り、次の一步が踏み出せない状態に陥った回数を記録するようにプログラムした。また、全ての歩行者が目的地に到着するまでの実時間を記録できるようにした。これにより、どのような歩き方をすれば、最も安全に歩行できるのかを、各種設定の組み合わせで観察することが可能となる。たとえば、20人の歩行者を配置し、Free Space 閾値を人間3人分のスペースに設定した時と、50人の歩行者を配置し、Free Space 閾値を1人分に設定した時で、駅全体の安全性(立ち止まってしまった回数)と、経済性(全員が目的地にたどり着けるまでの時間)にどのような違いが出てくるのかを観察できる仕様とした。立ち止まってしまった回数の総計が多ければ多いほど、その周辺の歩行者との衝突の危険性が高まり、同時に衝突を避けるために立ち止まる行動によって連鎖的に渋滞が発生し、結果パニックに至るようなエスカレーション(機能の共鳴)が生まれることになる。これは安全性の指標である。また、全員が目的地にたどり着くまでの時間が長ければ長いほど、同一時間内にコンコース内にいる人数は増加してゆくため、やがて飽和状態に達する。飽和状態に早く達してしまう駅は、可能利用者数も少なくなるため、経済的に不利である。つまり、これは経済性の指標となっている。東京駅は混雑しているが、渋滞によって頻繁に停止させられるような事態に遭遇することは稀であることから、本シミュレータの実行時においても、停止回数は現実の東京駅のように少なく維持されなければならない。また、一人で歩く速度とも大きく乖離せず、実際の東京駅のように、素早く目的地に移動できる戦術が維持される必要もある。これらの要件が満たされる実行を行えるならば、シミュレータ(FRAMモデル)の特徴であるボトムアップな戦術維持アーキテクチャが、現実と大きく乖離しないものであることを示すことができ、トップダウンに戦術が決まらないからこそ安全になっているとする本研究の仮説を裏付ける根拠の一つとすることができる。

#### 4. 3 シミュレーションロジック

FRAM モデルからひな形として作成したソースコードに対して、以下の修正を実施した。

(1) Find\_Free\_Space 機能は、前後左右のエリアに存在している空白を識別し、複数の空白箇所を Precondition として Walk 機能に伝える。空白箇所が無ければ Walk は出力を停止する。また、Find\_Path 機能に対しては、最適な Path の選択を行うための複数の Free Space 情報を Path 候補として提供する。Free Space を識別する際には、以下の情報を利用する：

- ・ 他者との相対位置
- ・ 他者との相対速度
- ・ 他者との相対加速度

最新の相対位置と相対速度・加速度により、識別されている Free Space に相手が進入してくる可能性が有ればポイントを低くし、相手が遠ざかる動きをしていればポイントを高くする。全ての他者との間の相対運動を、周辺の Free Space 全点に対して評価し、全 Free Space の中で最も危険の少ない地点から危険の多い地点までにそれぞれポイントを付ける。

(2) Find\_Path 機能は、Find\_Free\_Space 機能が識別した複数の Free Space の候補から、最適な Path を見つけるため、以下の情報を利用する：

- ・ ゴールの座標
- ・ Free Space 情報
- ・ Free Space を起点として将来展開予定のエリアの人口密度

Free Space 情報に列挙された複数の空白地点に対して、ゴールに向かう最短コースに重なるベクトルを最有力候補としつつ、複数の有力候補の中から、将来 Path を設定しやすいベクトルを選ぶ。具体的には空白地点を起点にしたその先のエリアの人口密度を計算し、複数の Path 候補がある場合は、より人口密度の低いエリアに高い優先度をつける。これにより、人口密集エリアの一つだけ空いた空白地点に進入する代わりに、よりその後の展開が有利になるエリアを選択し、将来の Path 取りのしやすい

戦術をとる。なお、ゴールに向かう方向の Path が存在せず、代わりにゴールと逆方向に向かう Path が存在している場合の戦術については、実際の歩行者の動きを観察しつつルールを決定した。実際の歩行者には、前方が完全にふさがれている場合、そこで立ち止まって渋滞が解消するのを待つよりも、あえてゴールから遠ざかる人口密度の低い方向への軌道修正を入れながら、回り込む Path をとる挙動が見られた。そこで、プログラムロジックとしても、ゴールと逆行する Path の選択を禁止しない方針としたところ、予測通り停止する回数の低減、及びゴール到達時間の短縮に寄与することが分かった。停止は衝突回避のための最終手段であるため、衝突と同等の非安全な状態と考えることができるが、これを低減することによって、ゴール到達時間をも低減できる。ゴール到達時間の低減は、安全性の向上というよりも、むしろ、効率の向上であり、歩行者にとっては、極めて利己的な目標である。システム全体の安全性の向上と、歩行者個人の利己的な利益追求が合致するという特徴は、東京駅コンコースの特徴の中でも最も重要なものである。これについては、次節のシミュレーション実行結果の分析で述べる。

#### 4. 4 シミュレーション実行結果の傾向分析

以下に、シミュレーションの実行結果と、その考察について述べる。シミュレーション空間として、100 人で立錐の余地の無い飽和状態になる空間を想定し、そこに、1 人～50 人までの範囲で任意の人数を設定できるようにした。その人数が一斉に各自の目的地に向かって出発するシミュレーションを実施した。

各自の初期位置は、空間内でランダムに設定され、全員がゴールに到達するとシミュレーションを終了する。

各自の目的地は、自分の位置から遠い側の縦方向の終端上のいずれかの場所を、シミュレーション開始前にランダムに設定することとした。

シミュレーション時間は、障害物が無い直線移動にかかる平均時間（10 秒）の 10 倍の時間（100 秒）を最長時間とし、これ以上かかる場合は、渋滞によって新規流入者を受け入れることのできないパニック状態と判定し、シミュレーションを打ち切る。

シミュレーション実行時には、Free Space 閾値（Minimum Distance）条件を変えて実行を行った。すなわち、Free Space として有効と判定す

る余裕度の閾値を、一人分の空きスペース、二人分の空きスペース、三人分の空きスペースの3種類の値を使って実行した。

「一人分の空きスペース」とは、接触することなくギリギリですり抜けられるスペースがあれば、そのスペースに向かって一步を踏み出せると判断する、積極的な戦術を持つ歩行者の判断閾値である。実は、実際の東京駅コンコースでは、このようにギリギリのスペースを使ってすり抜けるような積極的な歩行戦術をとる歩行者が支配的である。歩行中に他の歩行者が使用するキャスター付きバッグに躓く事故が発生する主な理由は、この「ギリギリのすり抜け」を我々が無意識に実行していることから発生していると考えられる。前章で、東京駅構内においては、いわゆる安全上のルールは無く、各人は自由に戦術を維持・修正し続けていると述べたが、唯一の例外的ルールとして、「キャスターバッグを持つときは、周りの歩行者に配慮」することを呼びかけるポスター等を見出すことができる。これは、歩行者がキャスターバッグに衝突する多くの事例が存在し、その事象については、駅の他の仕組み等では安全化が困難であることを示唆している。ぎりぎりのすり抜けが一般的に行われている東京駅コンコースを模擬する本シミュレータの実行結果としては、一人分の空きスペース

(Minimum Distance = “1”) の条件で実行されたものがこの条件に近く、実行結果を見ても、その設定でのゴール到達時間や停止回数が、現実の状況に近いことが分かった。

次章に、歩行者数、Free Space 閾値毎の実行結果を示す。グラフの横軸が実行時間、縦軸が歩行者の停止回数である。歩行の停止とは、Free Space が閾値を下回り、安全に次の一步を踏み出す空きスペースが消滅したことにより、衝突までの時間 (Time to Collide) が危険領域に達した時に行われる安全化である。この時、歩行者は、他者との衝突を避けるため、一時的に歩行を停止する。この歩行停止は、衝突を避けるという意味で安全な行動であるが、歩行を突然停止した歩行者の近傍にいる他の歩行者にとっては、停止という不連続な行動が、予測できない動作となるため、逆に非安全となりうる。しかも、近傍の歩行者が停止したことにより、自らもそれに対応して停止、あるいは、歩行経路・速度を急変させるという行動を行うと、それが連鎖的に周辺に波及し、結果的にショック・ウェイブのように、多くの歩行者に停止行動をとらせることとなる。この瞬時の戦術変更の連鎖は、一方では、鳥の群れに見られるような、見事な編隊飛行のマヌーバを実現するが、他方では、安全に保たれているコンコースで突然大人数が歩行停止するという状態に陥り、それが連鎖して交通麻痺状態に至る可能性をも示唆している。一方、コンコースの歩行者は、



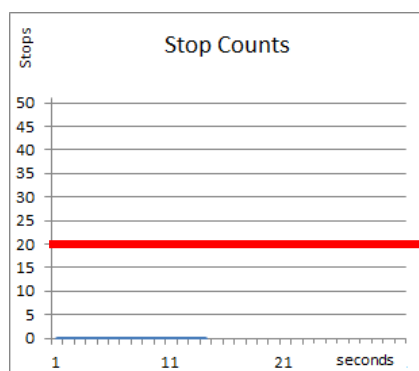
ルールに制約を受けることが無いため、一時的に停止に追い込まれても、Free Space が見出されれば何ら躊躇することなく歩行を再開する。シミュレータのロジックにおいても、それと同様、歩行者は瞬時に歩行の停止・再開を行う。

なお、今回行ったシミュレーションは、各エージェントの持つ Minimum Distance を統一的な値として使用しているが、各エージェント固有の閾値を様々に分散させ、郡としての行動様式をより詳細に分析することも可能である。そのためには、各エージェントの歩行目的を途中で変える、あるいは、歩行速度や障害に際しての反応などにも多様性を持たせるなど、多くのパラメータの自由度を高める必要があり、今後の課題と考えている。

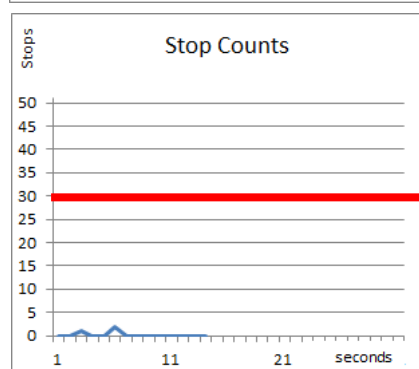
各実行結果には、以下の凡例に基づき、情報を付記した：

- (1) Minimum Distance: Free Space の判定閾値 (1~3 のいずれかの値)
- (2) Total Stops: シミュレーション時間内に発生した衝突回避のための停止の総合計回数
- (3) Max Stops: 同時刻に発生した最大停止数。全員が一斉に一步を踏み出そうとしたときに、危険を検知して踏み出すことをやめた人数。たとえば、50 Walkers で Max 25 stops となれば、50 人の歩行者のうち、25 人が同時刻に歩行を停止したことになる。
- (4) Duration: シミュレーション開始から全員がゴールに到達するまでにかかった時間。100 秒を超えてもシミュレーションが収束しなかった場合は、” 100 (時間切れ) ” と記した。
- (5) 歩行者の数はグラフ中で赤線によって示した。

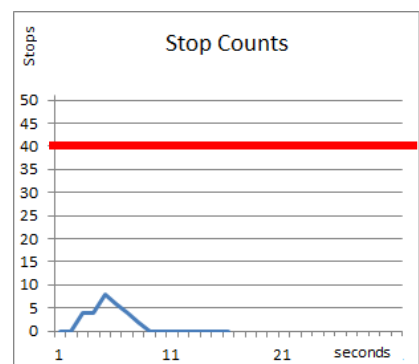
#### 4. 4. 1 Minimum Distance = "1"の傾向分析



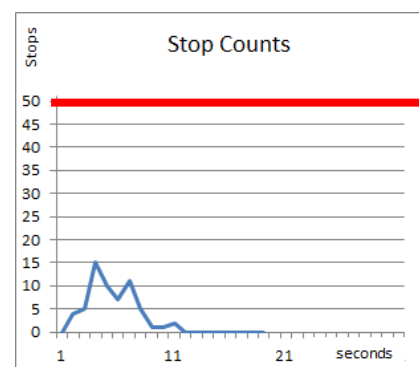
Min distance : 1  
Walkers : 20  
Total stops : 0  
Max stops : 0 (0%)  
Duration : 14



Min distance : 1  
Walkers : 30  
Total stops : 3  
Max stops : 2 (6%)  
Duration : 14



Min distance : 1  
Walkers : 40  
Total stops : 203  
Max stops : 8 (20%)  
Duration : 16



Min distance : 1  
Walkers : 50  
Total stops : 61  
Max stops : 15 (30%)  
Duration : 19

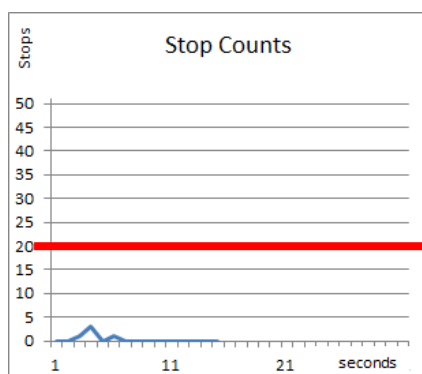
図 4-2 Minimum Distance = 1

Minimum Distance が” 1” の場合は、歩行を停止する人の数は歩行者数 50 人のケースでも最大 30%にとどまり、停止した多くの歩行者も、即座に歩行を再開している。つまり、50 人でも運用可能ということができる。実際のコンコースにおいては、100 人で立錐の余地がなくなる空間に 50 人がいるという極端な混雑は稀であるが、そのような状況にも対応可能となっている。

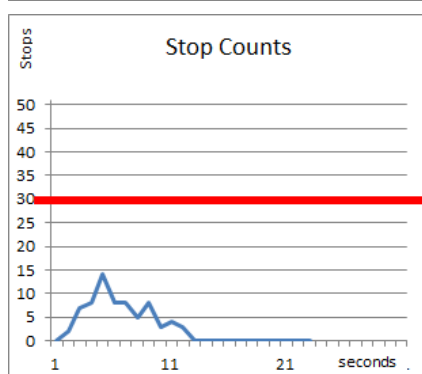
停止した人全員が歩行再開するまでの時間は、最長で 11 秒となるが、これはコンコースに一人の歩行者のみが存在し、最短コースで目的地に向かい到達する平均時間とほぼ同等である（平均時間は約 10 秒）。すなわち、歩行停止は長時間継続せず、短時間に回復する。新規に流入する歩行者を想定すると停止は断続的に続くと考えられるが、コンコースの 50%が歩行者で埋め尽くされており、歩行者間の平均空きスペースが一人分という極端な混雑状況において、システムが安全を維持できていることは、特筆すべき安全性ということができる。

本実行結果は、自分の周囲に一人分のスペース（一步踏み出せるスペース）があれば、そこに踏み出すという、歩行者のいわば積極的・利己的な戦術が幸いし、システムが安全に保たれているということを示している。次項には、Minimum Distance = “2” の実行結果を示す。

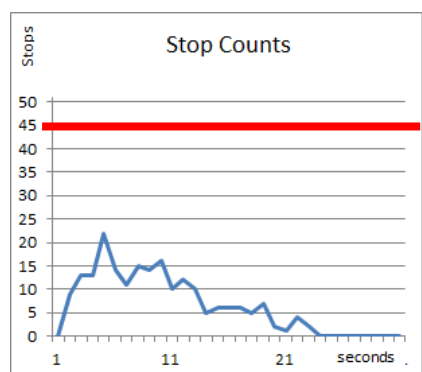
#### 4. 4. 2 Minimum Distance = "2"の傾向分析



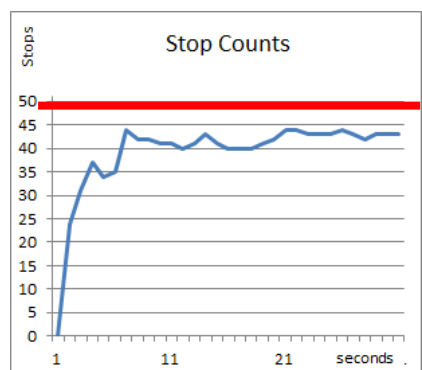
Min distance : 2  
Walkers : 20  
Total stops : 5  
Max stops : 3 (15%)  
Duration : 15



Min distance : 2  
Walkers : 30  
Total stops : 70  
Max stops : 14 (47%)  
Duration : 23



Min distance : 2  
Walkers : 40  
Total stops : 203  
Max stops : 22 (55%)  
Duration : 32



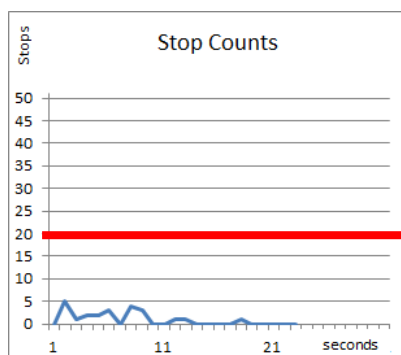
Min distance : 2  
Walkers : 50  
Total stops : 4181  
Max stops : 44 (88%)  
Duration : 100 (時間切れ)

図 4-3 Minimum Distance = 2

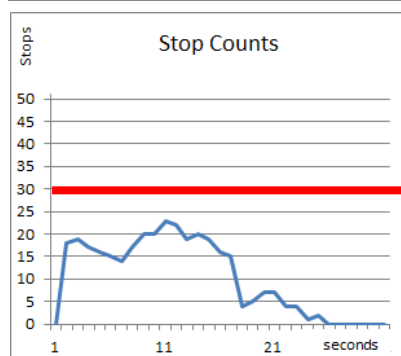
Minimum Distance が” 2” の場合は、自分の周囲に最低二人分のスペースがある状態が必要となる。つまり、Minimum Distance = “1” のケースに比べ、より、防衛的な戦術で歩行者が行動しているということが出来る。しかしながら、より防衛的な戦術をとっているため、より安全になるということにはならない。40 人までは、極度に渋滞しつつも、最終的には全員がゴールに到達したが（同時停止は 55%：二人に一人が停止）、50 人を超えると歩行停止している人の割合は 88%に急上昇し、コンコースのほとんどの利用者が衝突防止のために停止したまま、その場に居続けなければならない状態にとどまっている。すなわち、時間をかけてもゴールに到達できる人数がゼロのまま手詰まり状態に陥り、新規流入者を受け入れることが不可能となる。このように、ほとんどの歩行者が立ち止まっているような状況では、新規流入者によって状況は常に悪化し続け、完全にデッドロック状態になるまでエスカレーションは止まることができない。公共施設の安全という観点では、これは極めて危険な状態ということになる。すなわち、防衛的な戦術によって、非安全な状態になるのが、東京駅コンコースの特徴である。

次項では、その防衛的な戦術をさらに強め、Minimum Distance = “3” によるシミュレーションを実施した結果を示す。

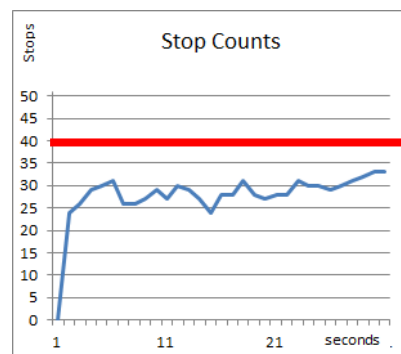
#### 4. 4. 3 Minimum Distance = "3"の傾向分析



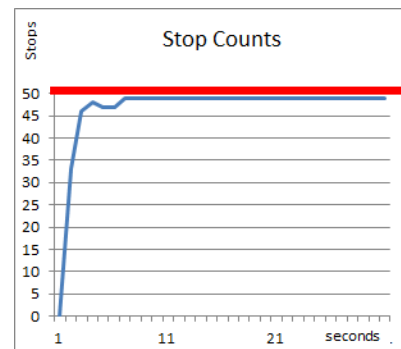
Min distance : 3  
 Walkers : 20  
 Total stops : 23  
 Max stops : 5 (25%)  
 Duration : 21



Min distance : 3  
 Walkers : 30  
 Total stops : 323  
Max stops : 23 (77%)  
 Duration : 29



Min distance : 3  
 Walkers : 40  
 Total stops : 3139  
Max stops : 33 (83%)  
Duration : 100 (時間切れ)



Min distance : 3  
 Walkers : 50  
 Total stops : 4827  
Max stops : 49 (98%)  
Duration : 100 (時間切れ)

図 4-3 Minimum Distance = 3

Minimum Distance が” 3” の場合は、自分の向かう方向に三人以上の空きスペースが必要となる。同時に歩行停止する人の割合は、30 人で 77%、40 人で 83%、50 人で 98%となるため、飽和率 30%程度（30 人）の混雑状況でも運用は不可能なレベルである。特に 40 人以上の場合は、シミュレーション時間以内に全員が目的地に到達できなかつたため、新規流入者が流入する余地が生まれず、コンコースは大混雑となる。

#### 4. 4. 4 傾向分析のまとめ

上記の分析から、Minimum Distance をいかに小さな値にできるのかが、東京駅の安全性にも、経済性にも同時に重要なことであるということが出来る。これは、東京駅においては、安全性と経済性は競合しないことを意味する。Minimum Distance を小さな値とし、経済性を最優先すると、結果的にシステム全体の安全性向上が達成されるという好循環により、3 章に示した先天的安全性が達成されているという仮説を立てることが可能である。

仮に人工知能を搭載した自由歩行型ロボットを東京駅コンコースに投入する場合、人間並みに、他歩行者とギリギリですれ違うことのできる認識能力、行動予測能力、行動計画能力、歩行制御能力が必要であることがわかる。そのような高度な能力を有しているならば、Minimum Distance = “1” での動作が可能となり、人間にとって障害とならない行動が可能となる。

しかし、このようなロボットが人間と同等の認識・行動予測・制御能力を有するレベルに達するためには、多くの課題を解決せねばならず、現時点において現実的な解とは考え難い。現時点では、ロボットの性能が人間に大きく劣ると考えられるため、Minimum Distance は、大きな値を設定して、ロボットの制御誤差による衝突を防止する安全化が必要となる。つまり、現時点におけるロボットにとっての現実的な Minimum Distance は、東京駅で人間が行っている高度な安全化に用いられる閾値ではなく、群衆の安全化の観点からは好ましくない「隔離」による防衛的な閾値とならざるを得ない。この閾値を採用せざるを得ないならば、人間の群衆の中にロボットを投入することは現時点では安全上問題がある。逆に、Minimum Distance = “1” という非防衛的な閾値を設定できるほどに洗練された認識能力・行動予測能力・行動計画能力・歩行制御能力を持てるかどうか、安全か否かの評価基準となるであろう。

#### 4. 5 シミュレーション実行結果の統計分析

シミュレーション結果に対して、上述のような渋滞のたまかな収束傾向を分析することはシステムの特徴（積極的・利己的な歩行が安全性と経済性を向上する）を把握する上で有効であるが、統計的な分析によって、さらに興味深い考察を得た。以下では各実行結果の統計的分析により、東京駅の歩行に関わる特徴を詳しく分析する。

##### 4. 5. 1 Minimum Distance = "1"の統計分析

Minimum Distance = “1” のケースで、30人、40人、50人の各ケースにおける複数実行結果の統計的な分析を行った。

以下は、9回の実行によって得られた実行結果の一覧である。

表 4-1 30人のシミュレーション (Minimum Distance = “1”)

Walkers	30		
Run#	Duration	Total stop	Max stop
1	15	7	5
2	16	3	1
3	15	2	1
4	15	1	1
5	14	1	1
6	16	3	1
7	14	5	2
8	14	1	1
9	17	15	9
Average	15.11	4.22	2.44
Max	17	15	9
% (Max stop / Walkers *100)			30%



表 4-2 40 人のシミュレーション (Minimum Distance = “1”)

Walkers	40		
Run#	Duration	Total stop	Max stop
1	18	38	9
2	17	29	8
3	15	14	4
4	16	17	5
5	18	17	6
6	17	9	4
7	18	16	5
8	17	13	4
9	16	18	7
Average	16.89	19.00	5.78
Max	18	38	9
% (Max stop / Walkers *100)			23%

表 4-3 50 人のシミュレーション (Minimum Distance = “1”)

Walkers	50		
Run#	Duration	Total stop	Max stop
1	20	70	12
2	21	22	7
3	17	44	15
4	21	62	10
5	18	47	12
6	18	47	16
7	18	26	8
8	19	43	10
9	19	46	11
Average	19.00	45.22	11.22
Max	21	70	16
% (Max stop / Walkers *100)			32%

これら3種類の実行結果を比較すると、大きな特徴を見出すことができる。すなわち、同時に歩行停止する最悪ケースの歩行者の割合は、いずれ

も約 30%であり、有意な差が出ないということである。また、同時に歩行停止する平均的な人数も、10%~20%で大差が生じていない。すなわち、Minimum Distance = “1” の場合、同時停止者数は、どの初期人数であっても、最悪で 30%近辺となっている。言い換えると、Minimum Distance = “1” であれば、最悪の事態であっても、30 人~50 人の範囲内では、瞬間的に全体の 30%程度の人間が一時的に歩行を停止するに留まるということを示している。人数が増えても危険度が比例して上昇しないという特徴は、大規模公共施設としては、好ましい特性であろう。人と人の距離が人間一人分になる混雑状況（50 人のケース）においても、最悪の渋滞発生は、全体の 3 割を滞留させるに留まるため、パニックに陥る可能性は低く、かつ、一時的に人数が増減しても、停止している人の割合に変化がないため、戦術を大きく変更する必要が無く、安心して歩行し続けることが可能になる。歩行者の感覚でいうと、東京駅は、どの時間帯に利用しても、歩行速度を極端に遅くすることなく、また、頻繁に行く手を阻まれることなく歩ける場所ということになる。この傾向を検証するために、東京駅において、ラッシュアワー、閑散時、それぞれの時間帯の定点観測を実施した。東京駅におけるビジネスマンの歩行者の平均的なピッチは、毎秒 2 歩で、どの時間帯においても一定である。一般的な傾向として、ビジネスマンは最も高速で移動しているが、その速度は、ラッシュアワーにおいてもほとんど低下しない。つまり、上記のシミュレーションにおける Minimum Distance = “1” の状況と最も近く、混雑状況に左右されにくいという特性を有していると考えられる。

以下に、Minimum Distance = “1” における 30 人ケース、40 人ケース、50 人ケースそれぞれの実行結果（ゴール到達時間、総停止回数、最大同時停止回数）の比較グラフを示す。各グラフから見て取れるように、システムの性質は人数が 50 人にまで増加しても変わらず安定している。

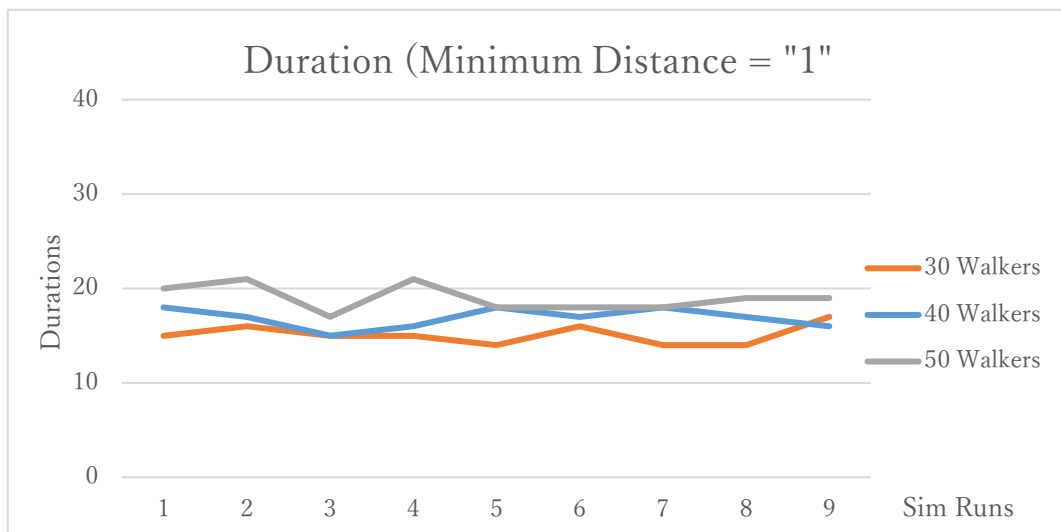


図 4-4 Minimum Distance = “1” でのゴール到達時間

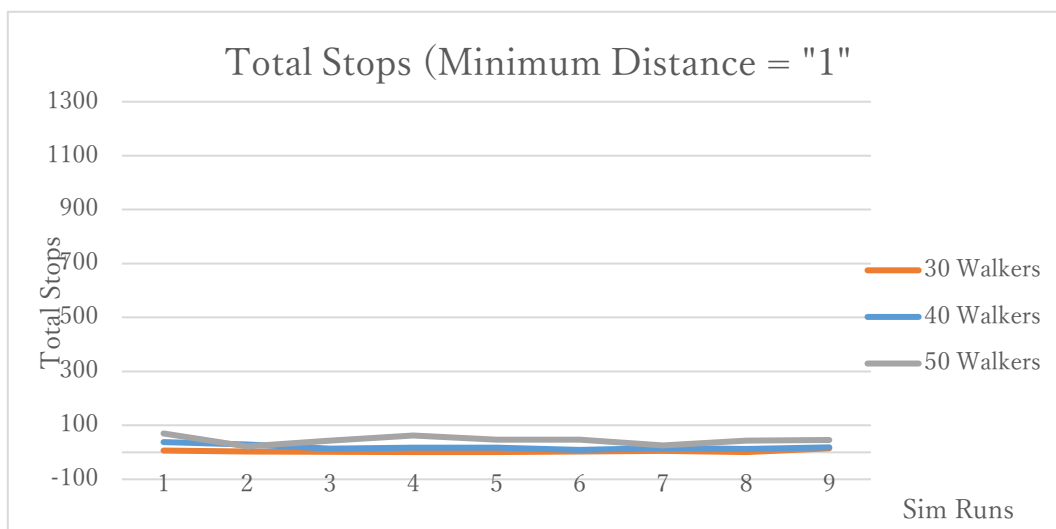


図 4-5 Minimum Distance = “1” での総停止回数

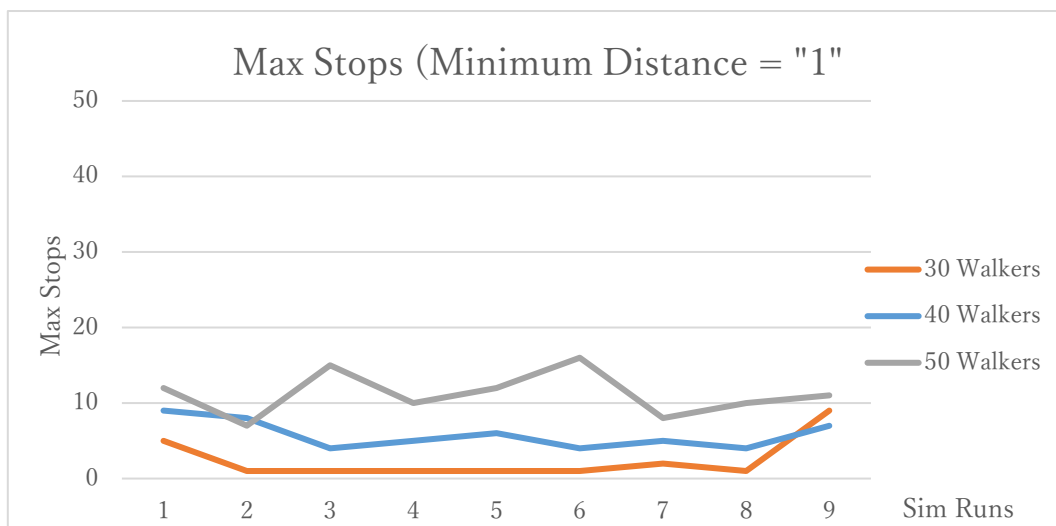


図 4-6 Minimum Distance = “1” での同時停止回数

一方、次ページ以降には、Minimum Distance = “2” のケースで同様の統計を取った結果を示す。こちらのケースでは、人数が増えるに従い、歩行停止する人の割合も増加するため、人数が 40 人となると、渋滞のエスカレーションが始まり、50 人となると、もはやほぼ全員が停止したまま麻痺状態に陥っている。

#### 4. 5. 2 Minimum Distance = "2"の統計分析

表 4-4 30 人のシミュレーション (Minimum Distance = “2” )

Walkers	30		
Run#	Duration	Total stop	Max stop
1	28	55	6
2	22	54	10
3	22	51	9
4	23	70	11
5	21	32	7
6	23	52	7
7	20	21	4
8	23	70	14
9	20	37	6
Average	20	38	8
Max	28	70	14
% (Max stop / Walkers *100)			47%

表 4-5 40 人のシミュレーション (Minimum Distance = “2” )

Walkers	40		
Run#	Duration	Total stop	Max stop
1	34	294	24
2	28	216	19
3	30	235	19
4	36	330	26
5	28	225	22
6	28	257	27
7	29	239	20
8	43	333	19
9	28	233	20
Average	34	363	24
Max	43	333	27
% (Max stop / Walkers *100)			68%

表 4-6 50 人のシミュレーション (Minimum Distance = “2”)

Walkers	50		
Run#	Duration	Total stop	Max stop
1	時間切れ		44
2	時間切れ		45
3	時間切れ		42
4	時間切れ		44
5	65	1266	38
6	47	836	38
7	53	930	42
8	69	1229	43
9	時間切れ		46
Average	58.5	1065.25	42.44
Max	69	1266	46
% (Max stop / Walkers *100)			92%

上記のシミュレーションは、全て Minimum Distance = “2” で実行したものである。Minimum Distance = “1” のケースとは異なり、人数が 40 人、50 人と増加するにつれ、%(Max stop / Walkers \*100)の数値も上昇する。その傾向は人数が 30 人を超えてから顕著になる。Minimum Distance = “2” を実現するためには、常に、自他の間の距離が二人分以上離れていることが必要であるが、100 人で満員となる空間においては、この条件を完全に満たし、一度も停止することなく全員が歩行し続けるためには、全員が各自の周囲に、自分専用の空間を占有する必要がある（下図）。

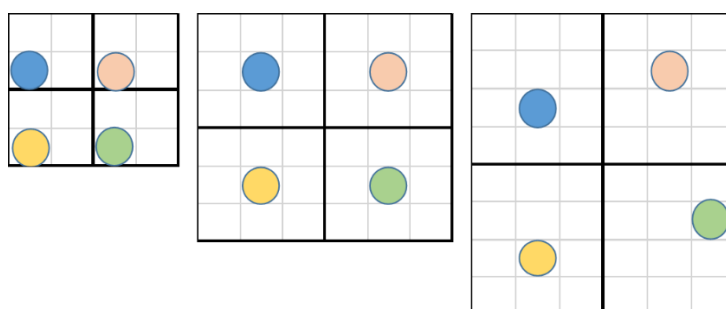


図 4-7 Minimum Distance = 1, 2, 3 の制約を守った各人の占有空間例

たとえば、上図のような配置では、Minimum Distance = “1” の場合、各自が 4 セル分、Minimum Distance = “2” の場合は、各自が 9 セル分、Minimum Distance = “3” の場合は、各自が 12 セル分の空間を必ず占有できていないと、無停止で歩行を続けることを保証することはできない。すなわち、Minimum Distance = “2” で運用するためには、コンコース全体のうち、10%程度のみ人で埋まっている状態を維持することが、無停止で歩行を続けられる十分条件となる。これを満たせる最大人数はたかだか 10 人であり、上記に示したような 30 人以上の状況においては、無停止で歩行が続くことはない。しかし、そのような厳しい状況にあっても、戦術をフレキシブルに切替える歩行方法は、有効に働いており、30 人でも運用が可能なレベルを維持している。十分条件である許容人数制限の 3 倍の人数にも対応可能なレジリエンスを有していることは、むしろ非常に高い安全性ということも可能である。

Minimum Distance = “2” のケースでは、人数が増加するに伴い、収束能力が急激に減衰する。この傾向は、特に 50 人を越えたケースにおいて顕著である。以下に、各人数におけるシミュレーション収束結果を示す。

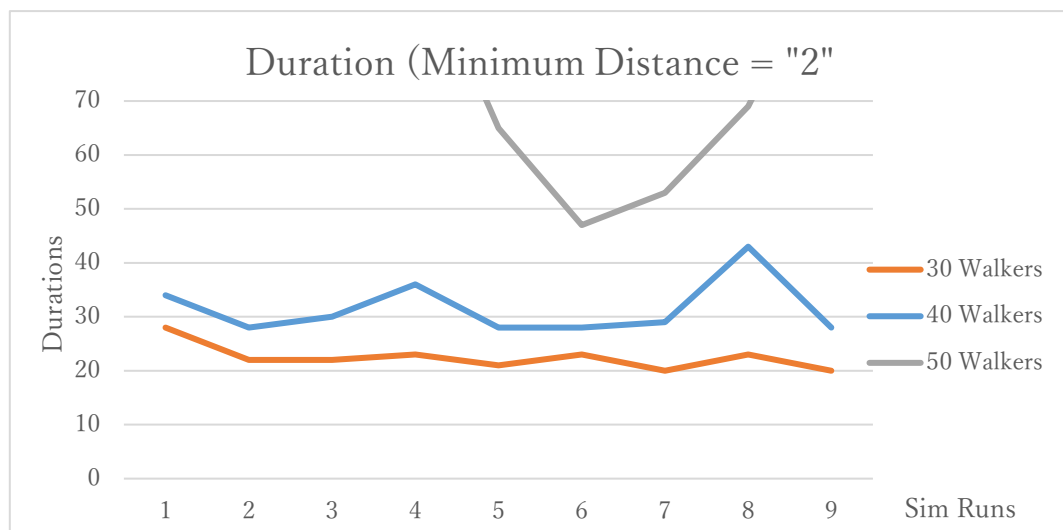


図 4-8 Minimum Distance = “2” でのゴール到達時間

Minimum Distance = “1” におけるゴール到達時間（図 4-4）では、人数による差異がほとんど無く、どれも 15 秒から 20 秒で収束しているが、Minimum Distance = “2” における上図から明らかなように、3 ケースの差異は明らかであり、かつ、50 人のケースにおいては、収束したのは 3 ケースのみであり、残りのケースではシミュレーションがデッドロックに陥り、発散した。人数の増加に伴い、収束能力が指数関数的に悪化してい

ることがわかる。この傾向は、下図に示すように、総停止回数の推移にも全く同様に表れている。

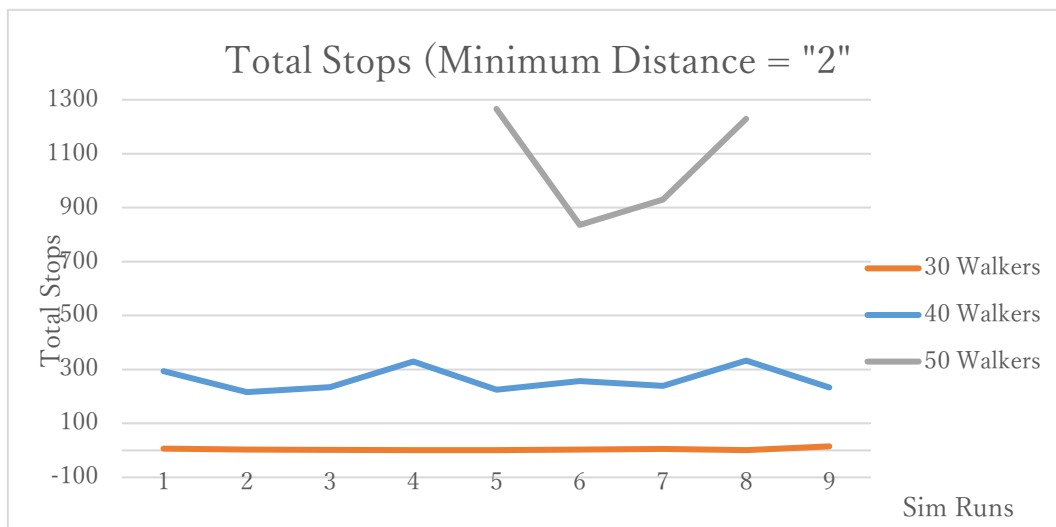


図 4-9 Minimum Distance = “2” における総停止回数

図 4-9 と前述の図 4-8 を比較すると、その傾向が全く同一であることがわかる。すなわち、ゴール到達時間は総停止回数に比例している。図 4-7 は、ゴール到達時間、すなわち、駅コンコースの経済的な効率性を示しており、図 4-8 は、停止回数、すなわち、駅コンコースの安全性を示している。言い換えると、東京駅では経済性と安全性には強い正の相関が存在していることがわかる。この傾向は、下図の傾向からも明らかである。

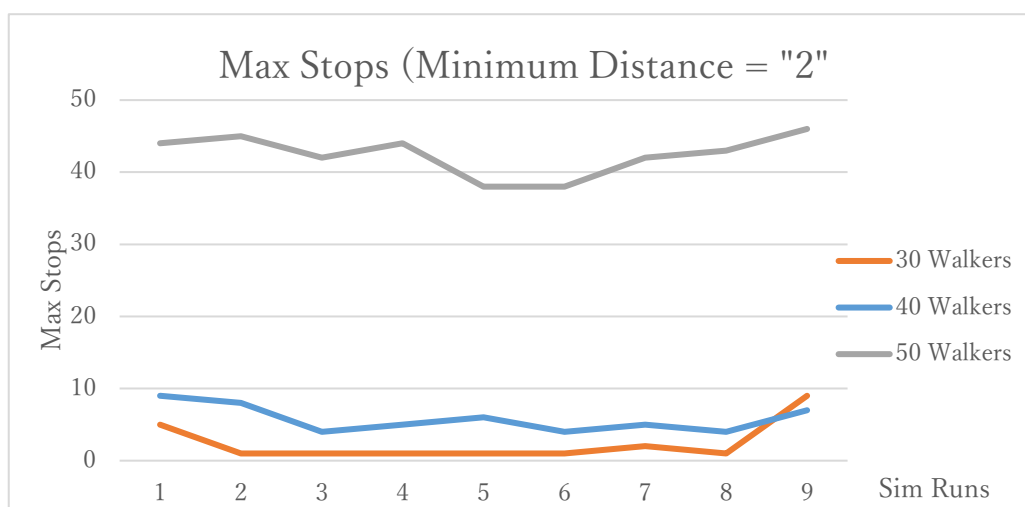


図 4-10 Minimum Distance = “2” における同時停止数



30人、40人のケースでは、同時停止数の最悪値は10人未満であるが、50人では急激に悪化し、常時90%程度の人数が停止している。コンコースにいるほぼ全員が停止している状態が、すり抜けられると判定する閾値が一人分多くなるだけで、これほど簡単に発生することは、我々が無意識にぎりぎりのすり抜けを行っている理由を説明していると言えるのではないだろうか。すなわち、ぎりぎりをすり抜けることにより、安全上の問題に遭遇することなく目的を効果的に達成したという成功体験が、我々の無意識の行動様式を決定している可能性を示唆していると考えられる。また、他人を避けるための方向転換や、急停止等の「安全行動」を行うと、群衆の中では逆に衝突につながり、逆に非安全になり得ることも学習していると考えられる。この学習の結果、ゴールへの最短直線コースをなるべく外さず、ぎりぎりのすり抜を繰り返すことで、他人との衝突を防止でき、安全性を逆に高められるという行動規範が身につけていると考えることができる。

これは、4.4.4項での傾向分析において立てた仮説を裏付ける結果となる。4.4.4項では、安全性（衝突・停止回数の低減）と、効率性（ゴール到達時間の低減）が両立することが東京駅コンコースの特徴であり、安全になればなるほど効率が良くなり経済的になるという好循環が存在しているという仮説を立てたが、シミュレーションの実行結果の統計的な分析を通じ、特に図4-9（効率性）と図4-10（安全性）が、ほぼ相似形となるほど強い相関関係を持つことが、この仮説を裏付ける根拠となると考えられる。

一般的には安全性を高めることは、経済性を犠牲にすることにつながることが多い。異常時には安全のために動作を停止する設計は鉄道、プラント等の分野で一般的である。また、異常時に備えて経済的に不利なバックアップを搭載するのは航空宇宙等の分野に共通である。鉄道では、信号機などの安全装置が故障した場合は、列車を停止して安全化する。プラントも同様に、制御系が異常になった場合は、動作を停止して安全化する。航空・宇宙ではコンポーネントの故障に備えて、経済的には不利な高価なバックアップを搭載している。こうした人工的な技術システムにおいては、経済性と安全性は競合することが一般的であり、安全を確保するために、やむを得ず経済性を犠牲にするという判断を強いられることになる。

ところが、東京駅コンコースにおいては、歩行者が経済性を最優先してぎりぎりのすり抜けを行うと、逆にシステム全体としては安全性が向上するという正の相関を持つ。我々が人工的に作り出しているシステムでは実現が難しい本質的な安全が東京駅コンコースには存在している。システム

がその効率を最大化することは、そのシステムの本質的な目標であり、その本質的な目標を達成することによって安全性が得られるならば、それを本質安全と呼ぶことが可能である。このような「本質安全」を目指す、新しい安全の概念として、Safety 2.0 という概念が提唱されている。Safety 2.0 と東京駅コンコースの安全性の関係については、5章、「関連技術」の章で述べる。

#### 4. 5. 3 Minimum Distance = "3"の統計分析

ここまでの章で述べてきたように、Minimum Distance を” 1” から” 2” へと増加させ、個人の安全のためのバリアを強固にすると、逆にシステム全体の安全性が低下する傾向が東京駅コンコースには存在している。Minimum Distance を” 3” に増加させると、その影響はさらに顕著となる。

表 4-7 30 人のシミュレーション (Minimum Distance = “3” )

Walkers	30		
Run#	Duration	Total stop	Max stop
1	29	139	16
2	35	324	23
3	31	152	15
4	38	240	19
5	29	140	16
6	31	197	16
7	30	163	14
8	34	210	16
9	27	127	14
Average	33	153	12
Max	38	324	23
% (Max stop / Walkers *100)			77%

表 4-8 40 人のシミュレーション (Minimum Distance = “3”)

Walkers	40		
Run#	Duration	Total stop	Max stop
1	34	294	24
2	28	216	19
3	30	235	19
4	36	330	26
5	28	225	22
6	28	257	27
7	29	239	20
8	43	333	19
9	28	233	20
Average	34	363	24
Max	43	333	27
% (Max stop / Walkers *100)			68%

表 4-9 50 人のシミュレーション (Minimum Distance = “3”)

Walkers	50		
Run#	Duration	Total stop	Max stop
1	時間切れ		49
2	時間切れ		50
3	時間切れ		50
4	時間切れ		50
5	時間切れ		47
6	時間切れ		47
7	時間切れ		49
8	時間切れ		50
9	時間切れ		44
Average	時間切れ		46
Max	0	0	50
% (Max stop / Walkers *100)			100%

Minimum Distance = “2” のケースで顕著となった、歩行者数増加に対する鋭敏な停止回数の増加傾向は、ここでも同様にみられ、表 4-9 にみら

れるように、50人ケースにおいては、全ケースで歩行者全員が歩行できないまま時間切れに至っている。これは、東京駅コンコースが通路としての基本機能を全く提供できていない状態と言い換えることができる。Minimum Distance = “1”であれば、全員が直線コースに近い11秒程度でゴールに到達できていたにも関わらず、Minimum Distanceの閾値を変えるだけで、完全な機能不全に陥った。

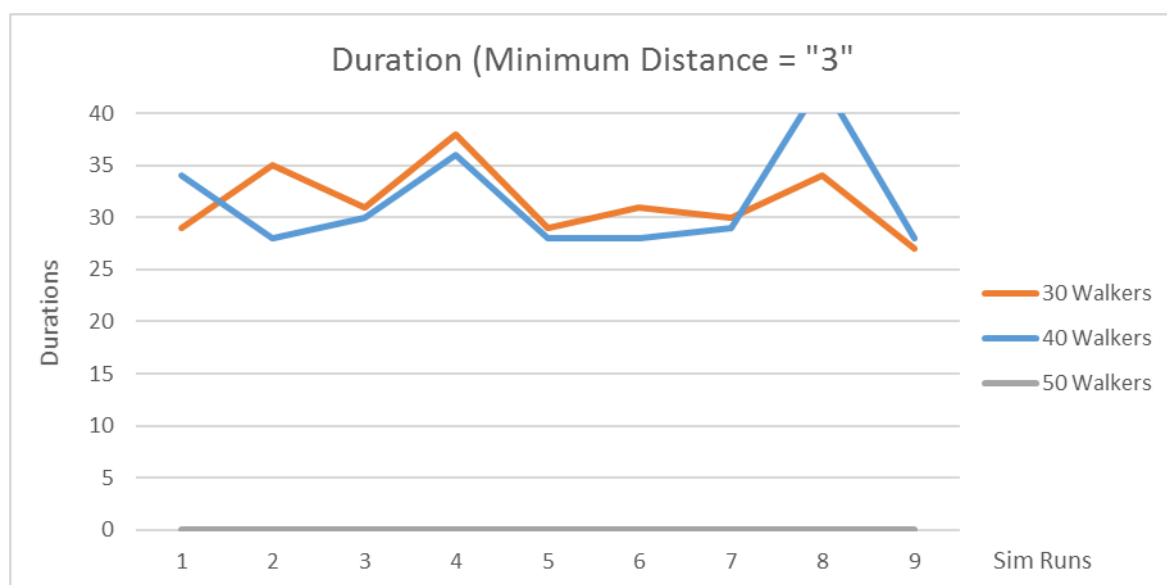


図 4-11 Minimum Distance = “3” におけるゴール到達時間

Minimum Distance = “3” においては、50人のケースでは、デッドロック状態に陥っているため収束不可能である。30人及び40人ケースは、ほぼ同様の傾向を示しているが、概ね30～40秒を要しており、直線距離を最短時間で歩行する時間の3～4倍の時間が必要である。つまり、進出する利用者の3～4倍の利用者が常時新規流入してくることを意味する。30人を超える利用客がいる場合、4倍の流入者があると100人を超えるため、この収束性能で渋滞を発生させないことは不可能と考えられる。

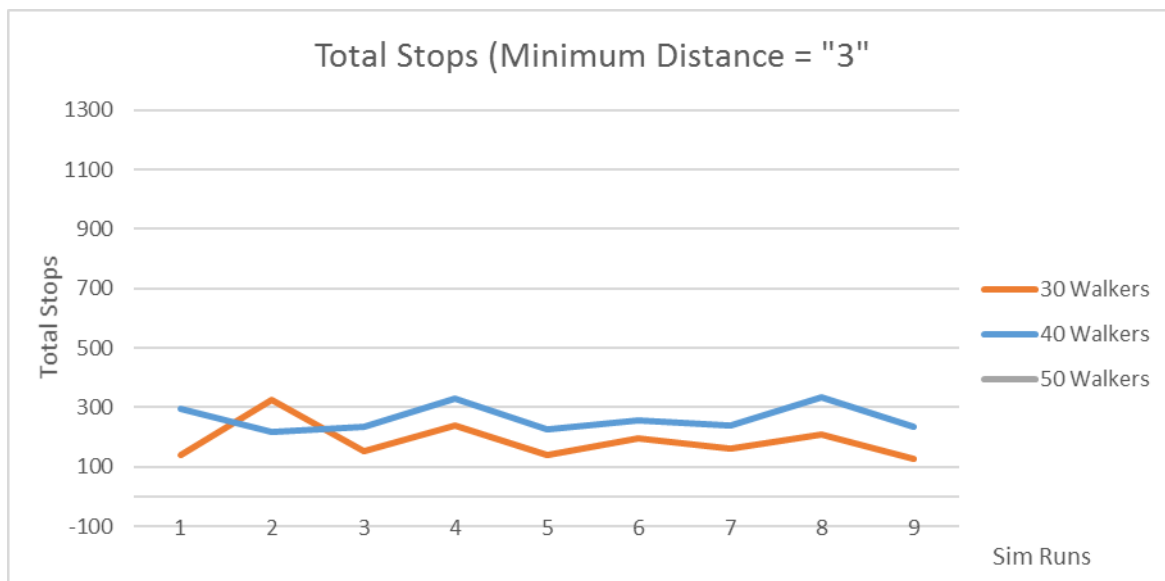


図 4-12 Minimum Distance = “3” における総停止回数

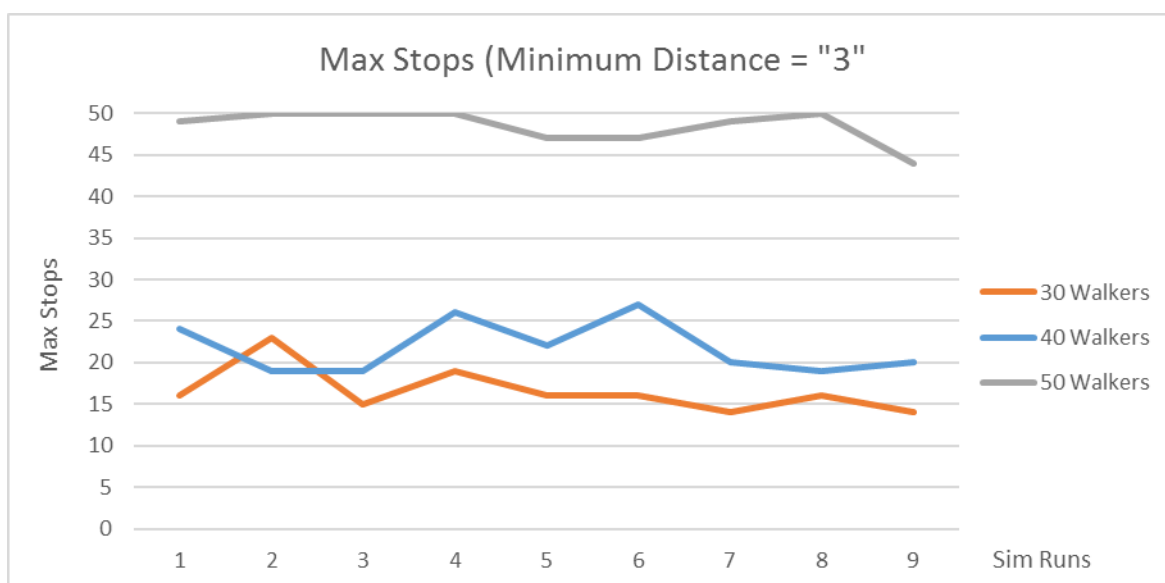


図 4-13 Minimum Distance = “3” における同時停止回数

Minimum Distance = “2” のケースと同様、図 4-11（ゴール到達時間：経済性）と図 4-12 及び 4-13（停止回数：安全性）に強い傾向の一致をみることができる。50 人ケースにおいてはシステムが破たんしているため、計測不能となっているが、30 人ケース、及び 40 人ケースにおける傾向は、3つのグラフで共通となっている。すなわち、安全になればなるほど経済性が向上する傾向にある。Minimum Distance = “3” の設定によ

ってシステムが飽和状態に近い極端な混雑を発生させていても、結果の現れ方は、経済効率の高い Minimum Distance = “1” のケースと同じ特徴を保存しており、デッドロックによってシステムが破たんする直前まで同じ傾向が継続する。

#### 4. 5. 4 東京駅コンコースの状態遷移と相転移現象の類似性

Minimum Distance = “3” における 30 人と 40 人の両ケースの同質性と、50 人ケースの極端な異質性は、流体と個体との相転移現象に現れる不連続な性質の転移に類似している[1]。流体における相転移現象では、流体が個体に相転移するまでは、流体は流体の性質を維持する。同様に、東京駅の群衆も、流動性を残す 40 人以下のケースでは、安全性が向上すると経済性も向上するという群衆の性質を保存している。50 人を超えると群衆がデッドロックによって流動性を無くし、個体への相転移と同様の性質の不連続な転移が発生する。群衆が個体への相転移を起こす点（相転移点：クリティカル・ポイント）は、Minimum Distance と歩行者数の組み合わせで表すことができる。流体の相転移における温度と圧力の組み合わせに相当するものである。

相転移現象には、流体・個体間の「構造相転移」以外に、磁気相転移、金属-絶縁体転移、常伝導-超伝導転移、常誘電体-強誘電体転移、真空の相転移と、多種多様なものが存在するが、その中で、鳥の群れの相転移に関する研究は、本論の結論と符号するものである[2]。バイアレク等は、鳥の群れにみられる瞬時の情報伝達（方向・速度変更）が、どのように行われているのかについて群れの観測データによる分析を実施し、パラメータ  $g$ （個々の鳥が群れ全体の平均速度に合わせる制御の強さ）がパラメータ  $J$ （個々の鳥が隣の鳥との間の速度差を最小にする制御の強さ）に比べ極めて小さい ( $g/J = 10^{-3}$ ) ことを見出した。パラメータ  $g$  が極めて小さいということは、系の中の速度分布の乱雑さが大きい、すなわち、系のエントロピーが大きいことを意味しており、相転移点に近い状態を維持していることを意味する。つまり、鳥の群れは、100degC 近傍の水と同様の最大エントロピーを持つ状態近傍で維持されており、小さな外乱（外敵の接近等）の入力をきっかけに相転移することにより、瞬時に状態が群れ全体に伝搬するとしている。群れ全体の統合制御を行っていないがゆえに、逆に群れ全体に情報が瞬時に伝達すると主張しているのである。

東京駅のコンコースにおける創発的な制御もこれと同様の特徴を有しており、個々の歩行者は、東京駅全体の平均的な歩行速度に合わせて行動するようなことは稀で、単に近傍にいる他者との相対位置・速度・加速度によって計算される Free Space の大きさを閾値以上の大きさに保とうとしているに過ぎない。極めて利己的に、自身の周囲の中で有利な Path を見つけ出すために、周辺の他者との相対位置・速度・加速度を最小限度に保ち、なるべくぎりぎりをすり抜けようとする行為が、システム全体を見た場合には、全体の安全化のために最適となっている。さらに、コンコース内に不具合が発生（団体客が一斉に停止等）したとしても、瞬時に全員が次になすべき軌道修正を自ら実施する様子は、外敵の接近によって群れ全体が相転移を起こす仕組みにも通じうる自己組織化の一例である。

前述のレイノルズによる研究に端を発し、バイアレク等に引き継がれて、現在も盛んに研究されている鳥の群れの統一のとれた行動の原理は、相転移現象の「アナロジー」として説明する段階から、集団の自己組織化（最大エントロピー状態からの相転移）というメカニズムが実際に働いているという論点で説明される段階に至っている。一方、東京駅の利用者は群れではなく、各々別の目的を持つ集団であり、外部との流出入も激しいシステムであるが、鳥の群れと共通項を持つ自己組織化による安全化というメカニズムを内包していると考えられる。今後、相転移理論との融合により、新しい安全理論を創出することも可能であると考えられる。

#### 4. 5. 5 統計分析のまとめ

シミュレーション結果の統計的分析により、以下の事項を見出した。

- (1) Minimum Distance = “1” のケースにおけるような安定した系の性質が見られる領域では、系の安全性（衝突・停止回数）と経済性（ゴール到達時間）は、強い相関を有しており、安全であればあるほど経済的になるという好循環が、東京駅コンコースの特徴となっている。
- (2) 安全性、経済性が共に良好な状態を維持できている安定状態から、構内の多くの歩行者が停止する麻痺状態に陥る状態へ遷移する条件は、Minimum Distance（最小 Free Space 閾値）と歩行者数の組み合わせで決定される。

- (3)安全であればあるほど経済性が高まるという東京駅コンコースの持つ性質は、従来の制御系システムの安全設計では実現が困難な長所であり、「本質安全」と呼べる優れた性質を備えている。
- (4)東京駅コンコースには自己組織化による自律的な安全化メカニズムが働いており、それは、鳥の群れに見られる自己組織化と通じるものである。特に、鳥の群れが群れ全体としてエントロピーを最大の状態に保った状態（相転移点近傍状態）を維持しており、外乱をきっかけにした相転移によって瞬時に情報を伝達できるとする仮説は、東京駅の歩行安全化がボトムアッププロセスにより個々人で異なる Free Space を持つ乱雑な状態を保ちながら、全体として最適化に向かっていくとする本研究の主張と符号する。

上記のような統計的に見出された特徴は、特に(4)に見られるように、相転移理論との接点を示唆し、今後の学術的発展を期待し得るものである。一方、これらの特徴は、21世紀に生まれた安全理論とも軌を一にする。次章には、この「新しい安全理論」と東京駅コンコースの安全との関係について述べたい。

#### 4. 6 参考文献

[1] Landau, D.,; Lifshitz, M., “Statistical Physics”, Course of Theoretical Physics Vol.5, Pergamon Press, SBN: 978-0-7506-3372-7, chapter 14, 1969

[2] Bialek, W.,; Cavagna, A.,; Giardina, I., “Social interactions dominate speed control in poising natural flocks near criticality”, PNAS. 111 (20): 7212-7217. arXiv:1307.5563, Bibcode:2014PNAS..111.7212B. doi:10.1073/pnas.1324045111. PMC 4034227, PMID 24785504, 2014



## 第5章 対象システムの安全化に関する考察

本章では、これまでの分析に強く関連している2つの安全理論との関連性を通じ、この分析が将来どのような発展の可能性を持っているのかについて考察を行った。2つの安全理論は、それが世に出た時期、その目指すものに高い類似性があると同時に、共に「次の世代の安全」を意味する「Safety 2」、「Safety 2.0」という名前を冠している。

### 5.1 Safety 2 (レジリエンス・エンジニアリング)

東京駅コンコースのモデルを作成する際に利用した機能共鳴分析手法 (FRAM) は、レジリエンス・エンジニアリングの安全理論を手法化したものである[1]。レジリエンス・エンジニアリングの目指すのは、従来の安全理論がFTAやFMEA等、Failure (失敗要因) に着目するのは逆に、成功要因に着目する理論である。ホルナゲルはこれを、従来安全理論のSafety-1に対する新しい安全Safety-2と呼んでいる。

東京駅の安全解析においても、失敗する要因を直接分析していない。その代わりに、安全に歩くために、東京駅にはどのような仕組みが存在しているのかを分析している。その結果、見出された成功要因は、ボトムアップに戦術が練り上げられ、安全化に向かって、群衆が自己組織化しているという性質であった。この自己組織化によって、多様な方向・速度・加速度・目的・歩行方法を持つ集団の中で、常に最適な安全化が実現されている。

物事が失敗する理由はほぼ無尽蔵に存在しており、常に新しい失敗理由を原因として事故は発生し続ける。しかし、物事が成功する理由は無尽蔵ではなく、少数の「定石」に収斂されて行く傾向を持つ。過去に成功したシステムを手本にすることにより、あるいは、生物的な進化の過程でDNA情報に記憶されることにより、この「定石」が引き継がれ、勝利の方程式として、磨き上げられてゆく。結果、弱い定石は淘汰され、最も強い成功要因が残ることになるため、成功要因は無尽蔵にはなりにくい。無尽蔵の失敗要因が「存在しない」ことを証明することは、原理上不可能であるが、少数の成功要因が「存在する」ことを証明することは可能である。レジリエンス・エンジニアリングは、この少数の成功要因がうまく機能できる仕組みを確実にすることによって安全化を行う。

東京駅の分析では、この成功要因を支えている戦略（目的地）の安定的な提供が停止すると、多数の歩行者が停止し、パニック状態に遷移する可能性が見出された。つまり、東京駅のハザードとは、目的地を見失った歩行者の増加による多数の停止の発生、そして、それに起因する渋滞、パニックの発生であることが見出された。また、ハザードの発生を防止する方法は、戦略、すなわち行き先案内版を多数設置すること、及び、自由な歩行を優先すること（ルールや区分の設定等の通常的安全化を行わないこと）である。

自由な歩行を優先することによって安全性を向上できるという発見は、東京駅的安全化を実現するうえで極めて重要な意味を持っている。すなわち、歩行者は、利己的な目的のために最短コースを進むという経済性を追求することにより、自然に安全性をも向上していることになる。経済性を向上すればするほど安全性を向上できるため、この安全性は簡単なことでは破たんしない。安全工学上、このような安全性を「先天的安全性」と呼ぶ。先天的安全性を有するシステムは、パッシブに「止まることによる安全化」、及び、アクティブに「制御することによる安全化」と比較して、極めて強靱である。様々なシステムの安全解析によって、このような優れた安全化の仕組みが明らかになることにより、それを手本にして定石がさらに洗練され、多くのシステムが強靱になる。東京駅的安全化は、ロボットや自動運転システムが、人工知能を使って行う自由行動の安全化という、これからの時代の最重要課題に対する解決策の手本の一つとなるであろう。

また、成功要因の分析を行う Safety-2 により、東京駅的安全化のように、従来の安全化（止める安全、制約による安全、制御による安全）とは全く異なる解決策を提示することができるようになると、イノベーションを加速することが可能となる。安全であることが経済効率をも向上させるならば、従来システム設計の主要制約事項であった「安全と性能のトレードオフ」が不要になり、大幅なコストの削減が可能となる。たとえば、東京駅のコンコースの場合、交通整理を行うことや歩行区分制約を課すことは不要であるため、コンコース内全域を自由通行エリアにすることができ、駅の収容人数は、理論上最大となる。コンコースの両サイドの商店は、どの方向に向かう利用客も顧客とすることができるため、歩行区分のあるコンコースと比較して、施設が収容できる理論上最大の人数全てを潜在顧客とすることができる。東京駅の場合、一日 44 万人が利用するため、その全てを潜在顧客とできる経済的価値は大であり、このような恵まれた環境が創出されることで発生する新たな商業形態により、駅そのもの

の機能が変容するというイノベーションを生むことにつながる。つまり、安全によって社会経済が活性化することになる。

これと同様の Safety-2 によるイノベーションに関して、植田、野本等による、次世代国際宇宙ステーション補給機（こうのとり）の設計検討の例を述べる[2][3]。

この設計では、従来異常時のバックアップとして搭載されていた A 系、B 系、C 系の 3 冗長システムを見直し、動植物の持つ性質を使ったアーキテクチャに変更することによって、コストの半減と、安全性及びミッション達成能力の改善を同時に実現した。

人間が二足歩行する際に、二本の脚のうち、片方をバックアップとして温存することは無い。しかしながら、従来の宇宙機は、基本的にバックアップ系の温存を設計指針として来た。現行の宇宙ステーション補給機の安全性を高めつつ、ミッション達成能力とコストを大幅に改善するために、動植物が何故想定外の外乱に対してもしなやかに対応でき、コストパフォーマンスの高い安全性と信頼性を実現できているのか、そして、宇宙機の設計との違いは何なのかを分析した結果、その主要因は、冗長系をバックアップとして温存せず、常時複数系を稼働させ、故障時にはデグレードモードでサバイバルすることであることを見出した。宇宙機においても、この動植物の持つ高いレジリエンスを成功要因として基本設計に組み込むことにより、複数系のスラスタを常時使用し、異常時には、スラスタの切り替えではなく、噴射量の微調整によって対応することにより、スラスタ基数を削減可能であることが見出された。さらに、スラスタ基数を大幅に削減しながらも、従来の「1 故障運用継続・2 故障アボート（退避）」の能力を、「2 故障運用継続」に改善可能であることも判明した。これは、まさに安全性と経済性の両立が可能であることの発見である。また、スラスタの配置に関しては、遺伝的アルゴリズムによる最適化を行い、人間の体幹周辺に集中している推進系（脚）と同様、推進系を機体の中心に配置することで、配管や通信ケーブルの簡素化、機体の完全独立モジュール化等を実現し、深宇宙探査にも対応できる拡張可能な機体構成となった。

以上のように、安全性と経済性の両立による先天的安全性は、東京駅や鳥の群れにおける群衆の安全のような自然物だけでなく、人工衛星の設計のような、典型的な従来型技術システムにもイノベーションをもたらす可能性を有している。したがって、21 世紀のシステムイノベーションを先導するために、安全設計を変革することは、極めて重要なテーマとなる。

## 5. 2 Safety 2.0 (本質安全・協調安全)

Safety-2 と同時期に注目を集め始めてきた安全理論に、中村等による” Safety 2.0” がある[4]。Safety 2.0 とは、安全の担い手の違いによる安全工学の歴史を解釈しなおす考え方である。レジリエンス・エンジニアリングと軌を一にしながらも独自の視点を持つ安全工学の再構築を目指している。

### (1) Safety-2

- ・ Safety-1: 失敗要因に注目
- ・ Safety-2: 成功要因に注目

### (2) Safety 2.0

- ・ Safety 0.0: 人の注意力に依存
- ・ Safety 1.0: 機械が人のミスのカバー
- ・ Safety 2.0: 人と機械の協調

上記のように、それぞれのアプローチは異なる。しかし、それらが目指す安全性は、下記に示す共通の目標を持っている。

- (1) 人と機械の協調 (Joint Cognitive System) の実現。
- (2) 人をエラーの温床と捉えない。むしろ人の柔軟性を活かそうとする。
- (3) 安全性と経済性を対立する概念と捉えない。安全は制約ではなく、安全によって性能が高まる姿を目指す。

Safety-2 と同様、Safety-2.0 も上記のような目標を持っているが、アプローチは異なり、より、「協調安全」に重きを置いている。協調安全とは、自律的に分散しているエージェントが、互いに連携しながら安全状態を達成することである。東京駅においては、ボトムアッププロセスにより、各自の戦術は各自が自律的に維持している。言い換えると、各自の持っている戦術は全て異なっており、システム全体としては乱雑である。つまり、群衆全体の中の統制が取れているというよりも、乱雑さを持つ全体が、その乱雑さゆえに安全な状態を保っている。これは、鳥の群れにおける乱雑さが、群れのエントロピーを最大化した状態を維持するがゆえに、外乱の発生に対して瞬時の相転移によって一斉に反応するという逆説的な

安全化に通じるものである。Safety 2.0における協調安全とは、このような自律分散型システムでこそ真価を発揮する。システムのエージェントは、それぞれ自律的に行動しているが、お互いに何らかの方法により協調し、安全化する。協調の方法は、データ通信のような従来型の制御システムが採用されているものや、東京駅のようにデータを送ることなく行われる高度な協調等、様々な形態が有り得る。いずれの協調方法をとるとしても、重要なのは、各エージェントが自律していることが許容されていることである。各エージェントは、システムの目的や安全を達成するために必要な情報交換を行っているが、その行動は厳密に制御されているとは限らず、常に変動を含んでいる。この変動を最小化し、エントロピーを最小化しようとするのが Safety 1.0 の安全であるが、Safety 2.0 では、システムのエントロピーを最小化することは必要条件ではない。むしろ、鳥の群れのように、エントロピーを最大化することで安全化するアーキテクチャをも許容するのが Safety 2.0 の特徴といえることができる。これは従来の防衛的な安全に対して積極的な安全、止める安全に対して止めない安全、制約する安全に対して協調する安全といえることができる。

Safety 2.0 を実現するためには、安全性、信頼性、保全性といった、個々のステークホルダーが牽制しあうのではなく、安全になればなるほど信頼性が上がり、保守性が上がるという相関を生むシステムアーキテクチャが必要となる。東京駅のコンコースにおいては、安全性の向上は、自由エリアの増大によってもたらされ、複雑な制御の仕組みは不要である。これは、高い保全性を意味する。また、システムが提供しなければならない唯一の情報である「行き先案内板」は、あらゆる場所であらゆる種類の人に理解できる高い信頼性が要求され、それがそのまま安全性に直結する。このような安全性・信頼性・保全性の融合の上に、自律分散しているエージェントが、互いに協調しながら最大利益を得られるシステムは、Safety 2.0 の目指すべき究極の姿といえるかもしれない。

### 5. 3 将来展望

本論では、東京駅のコンコースが先天的安全性を有しているがゆえに、人々は単に利己的な目標のために最小相対距離を保って歩けば、システムが自ずと安全化されるという主張を行った。もし、このような形で不特定多数の人工知能エージェントによる自由行動が安全化されうるならば、そ

の安全性は極めて強固なものになるであろう。一つの例として、ビットコインのシステムについて述べる。

ビットコインシステムでは、ブロックチェーンと呼ばれる参加者全員の間で行われる合意形成の仕組みにのっとり、トランザクションの正当性を全員で確認しあいながら取引が行われる[5]。ここで使用されるPoW(Proof of Work)という合意形成アルゴリズムにおいては、参加者はデータのマイニングを行い複雑な数値計算を高速で行い合意形成データを自ら作成することにより、ネットワークから報酬を受け取るという仕組みが働く。つまり、ビットコイン参加者は、そこで利益を得るために、システムの合意形成のための計算を競って行うことになる。より高速に計算を行う参加者が最も多くの報酬を得るという、東京駅と同様の利己的な競争環境である。もし、不正な利用者がトランザクションを改ざんし、存在しない取引データを偽造しようとするならば、他の全ての利己的な競争相手を凌駕する計算を行わなければならない。つまり、他の参加者が利己的になればなるほど、その計算能力を凌駕することが困難になるため、安全性も高まるという仕組みである。東京駅が安全になっている仕組みと全く同じ仕組みがここに働いていると言っても過言ではない。

通貨システムという、流動性の高く、多くの思惑が交錯するシステムは、現在、人工知能を含むソフトウェアによる自動取引によって大きく変容しつつあり、旧来の国家財政の信用度に依存したシステムがどこまで安全であり続けられるかは不透明である。偽造防止のための紙幣の印刷技術等を国家が独占的に保有できた過去の時代には鉄壁であった通貨の安全性も、今日のキャッシュレス取引全盛の時代においては限界を露呈し始めており、ネットバンキングシステムへの不正侵入や情報漏えいによって危機に直面する機会が増加している。つまり通貨システムは、もはや国家の印刷局が中心となっている紙幣の流通システムではなく、ネットワークで接続された参加者全員による仮想取引単位に変容しようとしている。まさにIoTに変容しつつある。このような今日的なIoTに対して、旧来の中央集権型の安全管理や国家によるトップダウン型の安全管理に依存する限界を認識し、分散・共有型、ボトムアップ型の安全を育ててゆくことは、本研究の成果の将来的な発展としてあるべき姿と言えよう。

#### 5. 4 今後の課題

不特定多数の人工知能エージェントによる自由行動の安全が達成されるためには、Minimum Distance = “1” の設定（ギリギリのすり抜けを許容する設定）においても、十分安全といえるだけの認識能力、予測能力、制御能力、予期せぬ相対関係に陥った場合の緊急回避能力、衝突時に衝撃を瞬時に軽減する反射能力等がシステムに要求される。しかし、現状における我々人類が到達した技術レベルは、これには程遠く、未だ、人間とロボットは物理的に隔離されている必要がある。人間が行っているギリギリのすり抜けが実現できなければ、群衆の中にロボットを投入することは、現状達成されている安全性を阻害する要因になってしまう。システムが Minimum Distance = “1” を実現できるレベルに達しているのか否かを検証するクライテリア、検証手法の開発が今後の課題である。これは、とりもなおさず、人工知能の安全性を検証する仕組みの構築であるが、人工知能に見られる非決定論的動作原理が存在する限り、安全性の検証は、特定条件において、必ず安全行動を実施できることという固定のクライテリアでは実現が難しい。システムが非決定論的に動作するため、検証を決定論的にしてしまうと、全てが不合格になってしまうためである。今後、我々が人工知能の安全性の検証に取り組む場合は、従来のルールベースのシステムの検証方法を超えて、一人の人間を評価するときのような柔軟で多角的な評価指標を持たなければならないであろう。その点で、本論文はひとつの方法論を提示したといえる。今後、この分野の研究が進み、安全を損なうことなく人工知能を搭載したロボットや自動車が人間と共存できる社会の実現に進むことを願っている。

## 5. 5 参考文献

[1] Hollnagel, E., “The ETT0 Principle - Efficiency-Thoroughness Trade-Off. Why Things That Go Right, Sometimes Go Wrong.” Sjsikkerhetskonferansen, Haugesund, Norway, pp.23-24, 2009

[2] Ueda, S.; Nomoto, H.; Kasai, T., “A study on new GN&C and propulsion system architecture by resilience engineering approach”, 9th International ESA Conference on Guidance, Navigation & Control Systems, 2014

- [3] Nomoto, H.; Ueda, S., Takata, S., Kasai, T., Fukatsu, T., Kobayashi, R., Nogami, M., Wakabayashi, Y., "RESILIENT REDUNDANCY - Design Study of the New HTV (H-II Transfer Vehicle)" International Symposium on Space Technology and Science, 2014
- [4] 中村英夫, "IoT時代の新しい安全「Safety 2.0」の全貌", ET2017 独立行政法人 情報処理推進機構 SEC 先端技術入門ゼミ, 2017
- [5] Satoshi N., "Bitcoin: A Peer-to-Peer Electronic Cash System", <https://bitcoin.org/bitcoin.pdf>, 2008



## 第6章 結論

不特定多数の人工知能エージェントによる自由行動の安全化という目的のために、そのような複雑系システムと共通点の多い東京駅のコンコースの自由歩行が、どのような仕組みで安全に行われているのかを分析した。

分析には、レジリエンス・エンジニアリングの安全解析手法であるFRAM(機能共鳴分析手法)を用いた。作成したFRAMモデルの分析から、このシステムの成功要因として、以下の2点が識別された：

- ・ 成功要因
  - ・ 一旦ゴールが設定されると、ボトムアップに練り上げられる戦術にしたがって皆が歩き続けることができる。戦術は、常に自動的に参加者全員によって環境変化に適応して更新し続けられる。戦術を決めるのは、ルールや交通整理によるトップダウンプロセスではなく、参加者によるボトムアッププロセスである。
  - ・ ゴールは環境に影響されず、常に安定的に提供されている。

つまり、東京駅の安全性は制御や制約、ルールなどにより実現されているのではなく、利用者それぞれが自律的に自分専用の戦術を作り上げているが、その異なる戦術がシステム全体としては、協調的に働いていることが見出された。同様の分析が鳥の群れの統一的な行動に見られる。個々の鳥は、自分の近傍の鳥との関係から自分専用の戦術を作り上げているが、その異なる戦術が、システム全体としては、統一的な群れの行動として発現していると考えられている。

一方、東京駅の安全性を実現している成功要因が識別されたことにより、その成功要因を阻むリスクも識別された：

- ・ リスク要因
  - ・ ゴールの提供が滞ると、歩行が停止するため、新規流入者を受け入れることが困難となり、パニックに至る

リスク要因が識別されれば、安全化のための方策を検討することが可能となる。成功要因を最大化し、リスク要因を最小化するための方策として、以下が必要であるが、これらはいずれも既に実現されている。つま

り、現在の東京駅は、長い歴史の中で、対策が練り上げられてきており、既に十分に安全なレベルが実現されているといえることができる。

- ・ 東京駅安全化策

- ・ 歩行者が自由・利己的に行動することによって先天的に安全化されているため、あえて交通整理等によって歩行の自由度を減殺しないこと。また、歩行区分の設置によって歩行可能領域を削減しないこと。
- ・ 行き先案内版を見やすく、どのような場所からも見える位置に十分な数量を設置すること。

次に、FRAMモデルの分析に続き、モデルの妥当性を検証する目的で、モデルから作成したシミュレータを使い、モデルが表現しているボトムアッププロセスが実際に効果的に作用しているのか否かを確認した。シミュレータでの検証の結果、以下の知見を得た：

- ・ Minimum Distance = “1” のケースにおけるような安定した系の性質が見られる領域では、系の安全性（衝突・停止回数）と経済性（ゴール到達時間）は、強い相関を有しており、安全であればあるほど経済的になるという好循環が、東京駅コンコースの特徴となっている。
- ・ 安全性、経済性が共に良好な状態を維持できている安定状態から、構内の多くの歩行者が停止する麻痺状態に陥る状態へ遷移する条件は、Minimum Distance（最小Free Space 閾値）と歩行者数の組み合わせで決定される。
- ・ 安全であればあるほど経済性が高まるという東京駅コンコースの持つ性質は、従来の制御系システムの安全設計では実現が困難な長所であり、「先天的安全性」、あるいは「本質安全性」と呼べる優れた性質を備えている。
- ・ 東京駅コンコースには自己組織化による自律的な安全化メカニズムが働いており、それは、鳥の群れに見られる自己組織化と通じるものである。特に、鳥の群れが群れ全体としてエントロピーを最大の状態に保った状態（相転移点近傍状態）を維持しており、外乱をきっかけにした相転移によって瞬時に情報を伝達できるとする仮説は、東京駅の歩行安全化がボトムアッププロセスにより個々人で異なるFree Spaceを持つ乱雑な状態を保ちながら、全体として最適化に向かっていると本論の主張と符合する。

次に、今回分析した東京駅コンコースの安全に関して、Safety-2 および Safety 2.0 といった、21 世紀になって登場した新しい安全の概念との親和性を分析した。Safety-2 は今回使用した解析手法である FRAM の理論的基礎であり、高い親和性を有しているが、Safety 2.0 に関しても、高い親和性を見出すことができた。東京駅において実現されている安全性と経済性の両立は、Safety 2.0 の目指す協調安全と符合する。協調安全とは、安全性・信頼性・保全性を実現するステークホルダーが互いに牽制しあうのではなく、互いに協調するものであり、東京駅において実現されている先天的安全性（利用者が利己的に動いても、それによって安全化が実現される）が、Safety 2.0 の目指す究極の「本質安全」である。

最後に、今後の課題として、人工知能の安全性を検証するための課題を識別した。シミュレーションの結果明らかになったように、群衆の安全性は、Minimum Distance = “1” という、ギリギリのすり抜けをも実現可能な高い認識能力、予測能力、制御能力、予期せぬ相対関係に陥った場合の緊急回避能力、衝突時に衝撃を瞬時に軽減する反射能力等がシステムに備わっていて初めて達成可能となる。このような高い能力を備えるためには、人間のようなインテリジェンスが必要となるが、この際、問題となるのは、人間や人工知能の持つ非決定論的動作原理の妥当性を検証するためには、固定のクライテリアは使用できず、検証する側にも揺らぎを評価できるインテリジェンスが必要となるということである。今後、我々が人工知能の安全性の検証に取り組む場合は、従来のルールベースのシステムの検証方法を超えて、一人の人間を評価するときのような柔軟で多角的な評価指標を持たなければならない。しかも、人工知能システムの実社会への導入は既に始まっており、この課題は緊急に解決される必要がある。

本研究は、今後急速な技術進歩が予測されるロボットや自動車の自動運転といった自律したエージェントが、安全性を損なうことなく共存できるための解析手法の確立を意図したものである。安全を積極的に制御するプレイヤーがいないマルチエージェントシステムがどのように安全化されているのかを FRAM により可視化するという試みは初めてのものであり、自己組織化による先天的安全性の創発が最大の特徴との分析結果が導き出された。

自由度の高い FRAM モデルを、トップダウン型安全制御モデルである STAMP モデルに写像することにより、逆に、トップダウン型ではない安全

制御が実在し、特にマルチエージェントシステムにおいては有効に働いていることを見出した。

ホルナゲルの FRAM はシステムの表現（成功要因）はできるが、安全性解析には限界があり、どのように利用するかは今後の課題とされていた。

これに対し、野本（筆者）は、STAMP の概念を融合させることにより、構造化することで分析が可能との結論を得た。

この結果、FRAM が単なる表現手段としてではなく、安全性分析のツールとしても利用できることになった。筆者の研究成果の一つでもある。

従来、安全は経済効率と背反するとされてきたが、先天的安全性（本質安全）への方策が示されることで、安全理論が単に安全な社会作りに貢献するだけでなく、経済発展にも貢献するという、新しいパラダイムを作るきっかけとしてゆきたい。

## 謝辞

本研究は、近年注目され始めた人工知能の安全性をいかに評価するかという問題への解決策を、自然知能の安全の成立要因を分析することにより得ようとする試みである。研究内容について、日本大学中村英夫特任教授、高橋聖教授、泉隆教授、細野裕行教授のご指導のもとにまとめたものである。ご指導を賜った中村英夫特任教授、高橋聖教授、泉隆教授、細野裕行教授に深く感謝を申し上げる。

また、研究の着想に関して、南デンマーク大学エリック・ホルナゲル教授、カーディフ大学デビッド・スレーター教授より、様々なご助言を賜った。心より感謝申し上げる。

## 著者発表論文等

### [査読付論文]

1. 野本; 道浦; 片平; 石濱, “FRAM (機能共鳴分析手法) による成功学に基づく安全工学”, SEC Journal 53号, ISSN: 1349-8622, pp.42-49, 2018
2. Nomoto, H.; Ueda, S.; Takata, S.; Kasai, T.; Fukatsu, T.; Kobayashi, R.; Nogami, M.; Wakabayashi, Y., “RESILIENT REDUNDANCY - Design Study of the New HTV (H-II Transfer Vehicle)”, The 7th International Association for Advancement of Space Safety Conference, Space Safety is No Accident, ISBN978-3-319-15981-2, pp.529-536, 2014
3. Nomoto, H.; Katahira, M.; Fukatsu, T.; Okabe, H.; Ymanaka, K., “Independent Test”, Proceedings of the First International Association for Advancement of Space Safety Conference, a New Beginning, ISBN: 92-9092-910-3, ESA-SP Vol. 599, pp.501-505, 2005
4. Nomoto, H.; Ueda, S.; Ujiie, R., “Risk Assessment based on Resilience Engineering for Adaptive GNC System”, International Symposium on Space Technology and Science 2015, 2015
5. Nomoto, H., “Preliminary Risk Analysis for Software Verification & Validation”, International Symposium on Space Technology and Science 2000, 2000
6. Kakimoto, K.; Yoshikawa, S.; Nomoto, H., “A study of applying past accident cases to software design across domains using Control Structuring Diagram”, International Symposium on Space Technology and Science 2017, 2017
7. Sakemi, K.; Matsumoto, M.; Hoshino, N.; Nomoto, H.; Aman, R.; Fukuta, Y.; Kitamura, S.; Kunifuji, T., “A new software IV&V approach based on CAST and Safety-Driven Design”, International Symposium on Space Technology and Science 2017, 2017
8. Nakao, H.; Nomoto, H.; Katahira, M.; Ishihama, N., “Applying Statistical Test to Spacecraft Redundant Navigation Sensor System”, International Symposium on Space Technology and Science 2002, 2002

9. Port, D.; Nakao, H.; Nomoto, H.; Mamiya, H.; Katahira, M.,  
“Resolving COTS System Assessment Clashes  
COTS-Based Software Systems”, ICCBSS’05 Proceedings of the 4th  
international conference on COTS-Based Software Systems, ISBN  
3-540-24548-0, pp.248-257, 2005

#### [口頭発表]

1. 野本, ” STAMP による航空管制システム分析”, 組込みシステム開発技術展 (ESEC) 専門セミナー, 2018 (招待講演)
2. 野本, “Walk in Tokyo Station”, 情報処理学会, 2018 (招待講演).
3. 野本, “宇宙ステーション運用におけるレジリエンス・エンジニアリング”, 山梨大学医学部, 2017 (招待講演).
4. 野本, “FRAM とは”, 第 14 回クリティカルソフトウェアワークショップ, 2017 (招待講演)
5. 野本, “Walk in Tokyo Station”, 慶応義塾大学大学院システムデザインマネジメント研究科 システムアシュアランス講義, 2017
6. 野本, “STAMP と FRAM”, 日本信頼性学会 2017 年度第 1 回フォーラム, 2017 (招待講演)
7. 野本, “宇宙ステーション運用におけるレジリエンス・エンジニアリング”, 大学病院医療安全セミナー, 2017 (招待講演).
8. 野本, “FRAM の安全解析”, SEC 特別セミナー 破綻を回避するためのレジリエンス・エンジニアリング, 2017 (招待講演).
9. 野本, “STAMP 理論と手法”, JR 東日本本社, 2016 (招待講演)
10. 野本, “STAMP 理論”, 品質保証研究会 第 43 回講演会, 2016 (招待講演)

#### [著書 (共著)]

1. 独立行政法人 情報処理推進機構 技術本部ソフトウェア高信頼化センター ソフトウェア高信頼化推進委員会 IoT システム安全性向上技術 WG, “はじめての STAMP/STPA (活用編) ～システム思考で考えるこれからの安全～”, 2018
2. 独立行政法人 情報処理推進機構 技術本部ソフトウェア高信頼化センター ソフトウェア高信頼化推進委員会 IoT システム安全性向上技術 WG, “はじめての STAMP/STPA (実践編) ～システム思考に基づく新しい安全性解析手法～”, 2017
3. 独立行政法人 情報処理推進機構 技術本部ソフトウェア高信頼化セン

ター ソフトウェア高信頼化推進委員会 システム安全性解析手法 WG,  
“はじめての STAMP/STPA ～システム思考に基づく新しい安全解析手  
法～”, 2016

#### [その他の発表論文等]

1. 野本, “IoT時代の安全性検証技術 Resilience Engineering の適用 ;  
日経 BP オンライン, 2016
2. 野本, “安全におけるモデル化手法—STAMP と FRAM;日本機械学会誌,  
Vol.119, No.1177, pp.675-678, 2016
3. 野本; 片平; 酒井, “JEM ソフトウェア IV&V によるリスクアセスメン  
ト”, 宇宙ステーション講演会講演集, Vol.14, pp.29-30, 1998
4. 野本; 片平; 上杉, “米国におけるソフトウェア IV&V の概要”, 宇宙  
ステーション講演会講演集, Vol.14, pp.27-28, 1998
5. Ueda, S.; Nomoto, H.; Kasai, T, “A study on new GN&C and  
propulsion system architecture by resilience engineering  
approach”, 9th International ESA Conference on Guidance,  
Navigation & Control Systems, 2014.

#### [授賞]

1. HTV5 Rendezvous Trajectory Team, “NASA JSC Group Achievement  
Award”, 2016
2. HTV /H-IIB ロケットの開発 ; 日本産業技術大賞文部科学大臣賞, 2010