

しなやかな問題解決を実現するための対話型モデリング手法「システムング」の提案 Systeming: Interactive Modeling Framework to handle the Complexity and the Uncertainty

三浦 政司[†]
Masashi Miura

南部 陽介[‡]
Yohsuke Nambu

山舗 智也[‡]
Tomoya Yamashiki

1. はじめに

VUCA などの言葉で表現されるように、社会の様々な場面において複雑で変化の速い状況が生じている[1]。そのため、様々な立場の人が頻繁に、複雑で分野横断的な問題解決に取り組む必要性に迫られている。例えば、そのような場面の一例として DX (デジタル・トランスフォーメーション) が挙げられる。競争力強化や豊かな暮らしの実現のために、企業や行政機関に DX が必要だと言われているエラー! 参照元が見つかりません。が、その実現は複雑で分野横断的な問題である。ビジネス環境や業務プロセス、行政へのニーズは多数の要素が絡み合った複雑系であり、それをデジタル化する ICT システムもしばしば複雑なものになる。経営・業務現場・顧客など様々な視点からの意見やニーズを調整する必要もある。

複雑な問題を解決するためには、問題領域の知識やスキルだけでなく、複雑さに起因する様々な難しさに対処するための知恵も必要となる。そのような知恵を整理・体系化したものが「システム工学」である。従来、複雑で分野横断的な問題を上手く解決するには、システム工学の専門家が必要であった。しかし、上記のように社会の様々な場面で頻繁に複雑な問題に出くわすようになった今、システム工学の専門家の手を借りずとも、様々な分野・階層の人々が自ら複雑さに挑むことができるようになることが望ましい。

そこで本研究では「システム工学の大衆化」を目指し、それを実現するためのフレームワークの構築に取り組んでいる。ここで「システム工学の大衆化」とは、誰もがシステム工学に基づいたアプローチによって複雑な問題に対処できるようになることを示す。それを実現するためには、第 3 節で示すように目的探索と解探索を同時に進めていく「しなやかさ」と、それを可能とする「良い上流設計」を誰もが実行できるようにしなければならない。本研究ではそのための具体的な方法として、対話型モデリングを中心とする独自のフレームワーク「システムング」を提案する。

本稿の第 2 節では、システム工学とそれが扱う問題について整理し、大衆化の必要性についてあらためて述べる。

その後の第 3 節では、システム工学の大衆化に向けた課題と本研究のアプローチについて説明する。第 4 節ではそのアプローチに基づく具体的な方法論としてシステムングを提案し、第 5 節ではシステムングを効果的に実践するためのツール開発や実践例、今後の展望などについて簡単に述べる。

2. システム工学と複雑な問題

2.1 システム工学

人類は、宇宙・航空・防衛などの分野の中で、他の領域に先駆けて複雑な対象を扱ってきた。そのような分野の中で培われてきた「複雑さに対処するためのベストプラクティス」を体系的に整理したものがシステム工学であると言える。他にもシステムズエンジニアリング、システムズアプローチ、システム思考などと呼ばれることもあり、それぞれ定義や位置づけを異にしている場合もあるが、本稿ではそれらを総称してシステム工学とする。世界的なシステム工学の専門家団体である INCOSE はシステム工学を「システムを成功させるための複数の専門分野にまたがるアプローチおよび手段」と定義している[2]。「システム」の定義については第 4 節で触れる。

2.2 複雑な問題

システム工学が対象とする「複雑な問題」とは、次のような対象や状況を伴う問題のことを指す。

- 多数の構成要素の間に多数の相互作用があり、ある要素を変更した場合の影響が、どこにどのように現れるか分かりづらい。
- 個々の構成要素の性質や振る舞いが分かっているにもかかわらず、それらが相互作用を持つと全体の振る舞いがどうなるか予測が難しい。
- 分野横断的で、幅広い専門分野の知識や技術を必要とする。様々な専門家が協力する必要がある。
- ステークホルダが多く、要求や制約が満たされているかどうかの確認が難しい。

複雑化した現代社会では、宇宙・航空・防衛に限らず、ビジネス、行政、生活の中の様々な場面でこのような複雑な問題が起きる。第 1 節で挙げた DX はその一例に過ぎない。

[†] 宇宙航空研究開発機構 Japan Aerospace Exploration Agency

[‡] 株式会社レヴィ Levii, Inc.

2.3 従来のシステム工学

2.2 節で見たような複雑な問題に対処するために、システム工学には様々な実践知が含まれている。本節ではそのうちの代表的なものをいくつか取り上げる。

2.3.1 ライフサイクルプロセス

システムのライフサイクルと、各ライフサイクル内で実行・管理すべきプロセスを規定することで、複数の専門家が力を合わせて同じ質でシステムを進歩させることができる。国際標準では合意プロセス群、技術プロセス群、技術管理プロセス群などが定められている[4]。

2.3.2 ベースラインとマイルストーン

プロセスを進めていく中では、組織やチームの中で合意したシステムに対する共通認識が何かを明確にしておくことが重要である。共通認識を明確に記述したものをベースラインと呼び、プロジェクトのマイルストーンや審査会(ゲート)ごとに作成・更新されていく。図 1 に、NASA Systems Engineering Handbook[5]に基づいて作成したベースラインとマイルストーンの概要図を示す。

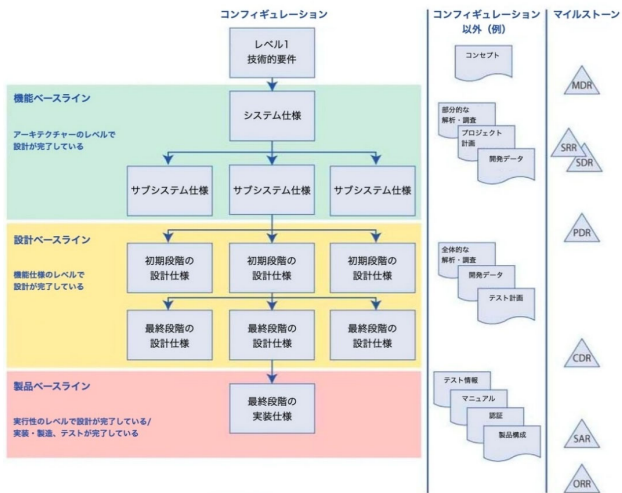


図 1. 設計情報の管理とマイルストーン (NASA SE Handbook FIGURE 6.5-3 に基づいて作図)

2.3.3 上流設計とアーキテクチャ

システムライフサイクルプロセスの国際標準[4]では、技術プロセス群の初期に「要求定義プロセス」と「アーキテクチャ定義プロセス」を設置している。これらのプロセスは一般的に「上流設計プロセス」と呼ばれる。要求定義プロセスではビジネス上の要求やステークホルダからの要求を分析・定義する。アーキテクチャ定義プロセスは、定義された要求に基づいて、システムの基本的な構成・機能・性質を定めるプロセスである。アーキテクチャの記述に関する国際標準エラー! 参照元が見つかりません。ではアーキテクチャは「ある環境におけるシステムの基本的な概念や性質のことであり、そのシステムの要素と関係、そして、設計と進化の原則が具現されている」と定義されている。

上流設計において要求とアーキテクチャについて関係者の中で合意を形成しておくことは、その後の関係者や各専門家による協調を円滑にして手戻りを防ぐ上で非常に重要である。図 2 は NASA が 30 以上の主要なプロジェクトを分析した結果であり、開発の各段階でどの程度のライフサイクルコストが定まるのかを示している[5]。この図から、システム開発の上流段階で 75%程度のライフサイクルコストが決まってしまうことが分かる。

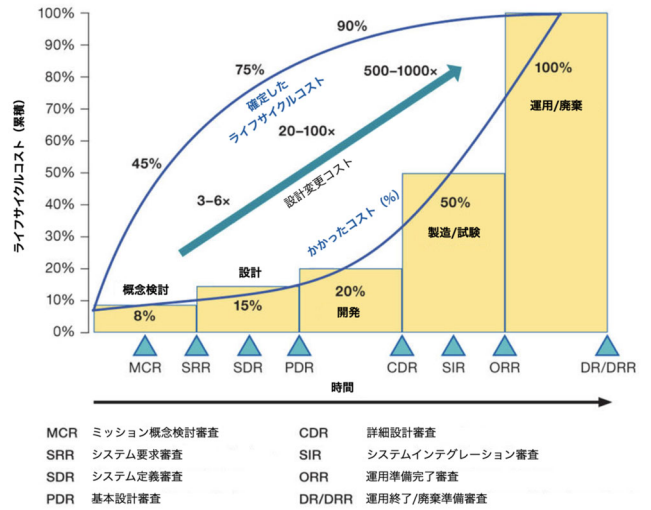


図 2. 開発フェーズとライフサイクルコストの確定 (NASA SE Handbook FIGURE 2.5-1 に基づいて作図)

2.4 問題の変化と新しいシステム工学

2.3 節で取り上げたシステム工学の代表的な考え方を見ると、従来のシステム工学は「プロセス」に重きを置いていることがわかる。複雑さへの対処方法をプロセスとして規定し、標準化することで、先人が築いたベストプラクティスを利用できるようにしている。

このようなプロセスの規定に基づく従来のシステム工学は、解決したい問題や課題が明確であり変動しない場合には極めて有効であるが、それらが不明瞭で変化してしまう場合には適用が難しくなってしまう。VUCA と表現される複雑な現代社会における問題では、「対象の複雑さ」だけでなく「環境や目的の複雑さ」も増しており、システムを取り巻く環境や、システムの目的そのものが不明瞭で予測できず、変化してしまう。井上ら[7]は目的に関する情報が既知である問題をクラス I・II、目的に関する情報が不完全で変化していくような問題をクラス III と分類しており、以下ではこの表現を借用する。

クラス III の問題が増えたことに対応して、システム工学の分野では MBSE (Model-Based Systems Engineering) が注目されている。MBSE では対象を一貫性のあるモデリング言語で記述することでトレーサビリティを確保し、階層的・段階的にベースライン化を進めることで変化に対応する[8]。他にもアジャイル開発との融合を目指した議論な

ども行われており、複雑な環境や目的への適応が進められている。

2.5 問題の普遍化

2.4 節で見たように、クラスⅢ問題への対応として MBSE やアジャイルとの融合などの新しいシステム工学が議論されている。しかし、社会の複雑化と変化はその勢いを増すばかりであり、クラスⅢに相当する問題がビジネス、組織、政治、社会生活、日常生活など様々な場面で生じるようになってきている。

クラスⅢの問題が普遍化している状況では、これまでのようにその都度システム工学の専門家を必要としていては、問題解決や価値創造が間に合わない。本研究はこの課題に着目し、システム工学の専門家の力に頼らず、誰もがシステム工学的なアプローチによってクラスⅢ問題に挑めるようになるためにはどうしたらよいか、すなわちシステム工学の大衆化について考え、フレームワークの構築やツールの開発に取り組んでいる。

3. 課題とアプローチ

3.1 目的と解の同時探索

目的に関する情報が不完全なクラスⅢの問題に取り組む際は、目的探索と解探索を同時に行う必要がある[7]。つまり、目的に対してある程度の合意を得たら、一度解探索に進み、解の探索や試行・試作などに取り組む。その中で目的側にもフィードバックが生じ、目的についてさらにシャープな議論に進むことができる。このような形で目的探索と解探索を行ったり来たりしながら反復的に取り組むことで徐々に目的のはっきりとしたり、目的の変化に追従し、それに対応した解決策を設計できるようになる。

このような反復的な探索を行うためには、ある種の「しなやかさ」が必要となる。目的が過度にぼやけたまま解探索に進んでも有効な解を得ることができない。一方で、目的に対する合意や解の構築を早い段階でシャープにし過ぎると、後々の変化に対応できず、大きな手戻りを生じることになる。クラスⅢの問題に取り組む際には、ちょうどよい硬さで目的側と解側を行ったり来たりする「しなやかさ」を持たなければならない。

3.2 しなやかさを実現するための上流設計

「しなやかさ」を持って探索的な問題解決に取り組むためには、上流設計が極めて重要となる。2.3.3 節で述べたように上流設計とは、要求とアーキテクチャについて関係者の間で合意を形成しておくことを指す。これはちょうど、目的（要求）と解（アーキテクチャ）の基本的な構造について認識を合わせておくということであり、同時探索の拠り所を形成する作業に相当する。

ここで、しなやかな同時探索を実現するためには、上流

設計における合意形成が、ちょうど良い抽象度で行われる必要がある。抽象度が高すぎるところで合意を得ても、その後に認識の齟齬が生じてしまう一方で、抽象度が低すぎる（具体・詳細過ぎる）ところでは専門性が異なる関係者の間で合意を形成するのが難しい。これはちょうど前節で述べた「ぼやけたまま」と「シャープ過ぎる」に相当する。また、要求とアーキテクチャに対してよく理解しないまま合意を形成したとしてしまうことも避けなければならない。関係者が皆、合意の対象である要求とアーキテクチャについて良く理解した上で合意形成を行う必要がある。そのためには、抽象度だけでなく、どのような視点と表現方法で要求やアーキテクチャを表現すればよいかも重要である。

このように、上流設計には「ちょうど良い抽象度で合意を形成する」「要求とアーキテクチャを良く理解するための視点と表現方法が必要」という難しさがある。図 3 はこれらの難しさを概念的に示したものである。

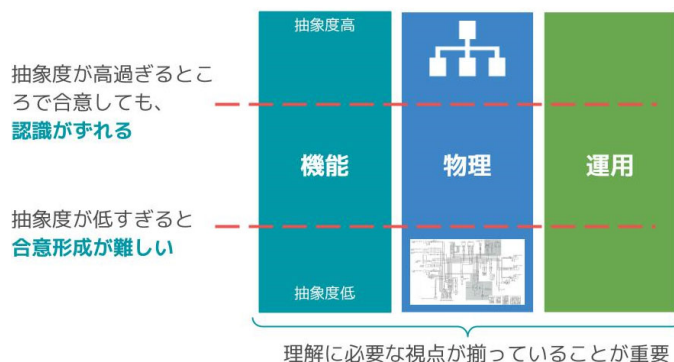


図 3. 上流設計の難しさ

3.3 対話型モデリングの提案

ここまで議論したように、クラスⅢに対しては目的探索と解探索を同時に進めるしなやかな問題解決が必要であり、それを実現するためには関係者が要求とアーキテクチャを理解するのに必要十分な視点とちょうどよい抽象度でシステムを表現し、合意を形成することが求められる。その問題領域に経験の深い専門家であれば「必要な視点」と「ちょうど良い抽象度」を暗黙的に知っているだろう。また、システム工学の専門家であれば、要求定義プロセスやアーキテクチャ表現の標準を適切にテラリングし、適用することで良い上流設計を実現できるかもしれない。しかし、本研究が目指す「システム工学の大衆化」のためには、そのような専門性がなくても「必要な視点」と「ちょうどよい抽象度」を発見できるようなすべを用意しなければならない。

そこで本研究ではシステムの姿をシステムモデルで表現し、システムモデルを通して対話しながらよい抽象度と必要な視点を探索していく「対話型モデリング」を提案する。

システムをノードとリンクから構成されるシステムモデルを用いて描くことで、要素間の関係性を直接的に表現し、

理解しやすくなる。また、システムモデルを描きながら対話することで、合意するのにちょうど良い抽象度や視点を探ることができる。どのような視点から、どのような抽象度でシステムを表現すればよいのか最初は分からないので、システムモデルを描きながら対話し、対話しながら探っていくというアプローチである。

4. システムング

著者らは、前節で提案した対話型モデリングを効果的に実践し、誰でもシステム工学的なアプローチによる問題解決に取り組めるフレームワークとして「システムング」を提案している[10]。本節ではシステムングの全体像と基本的な考え方を示す。

4.1 システムングの全体像

システムングは「対話型モデリングによる上流設計」を効果的に実践するためのフレームワークである。システム工学の知見に基づきつつ、一般的なシステム工学の方法論や解説と比較してより平易な言葉と実践に重きを置いた記述となっている。

システムングは次の3つから構成されている。

- マインドセット

対話型モデリングを実践する際に持つべき考え方。「システム思考」「視点をわける」「視点をつなげる」に分けられる。

- ディスクリプション

対話型モデリングを効果的に実践するための記述方法。ビュー、モデル、ビューモデルが定義されており、それらを使いこなすことで対話型モデリングを実践することができる。

- プロセス

対話型モデリングを実践するための手順。ディスクリプションで定義された概念を、具体的にどのような流れで使えばよいかの指針を与える。

以降ではこれらの3つの概念とそのサブ項目について、それぞれ簡単に紹介する。

4.2 マインドセット

システムングでは、対話型モデリングに取り組む際に持つべき基本となる考え方として「システム思考」「視点をわける」「視点をつなげる」の3つを定めている。4.3節で示すディスクリプション（記述方法）を使う際には、常にこれらのマインドセットを意識するとよい。

4.2.1 システム思考

システム思考は、モノゴトをシステムとして捉え、システムとして理解したり記述したりする考え方のことで、システムングのみならずシステム工学の基盤的な考え方である。システムングでは「システム」を図4のように定義する。

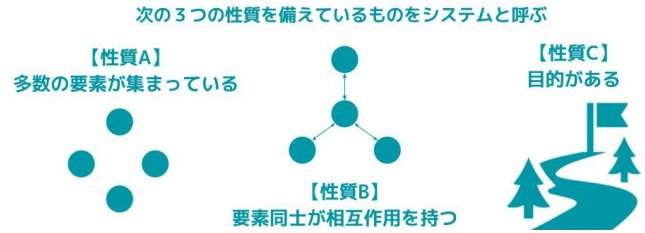


図4. システムの定義

この定義を採用すると、世の中のほぼ全てのものはシステムであると考えられることができる。一般的にシステムとして認識されている情報機器や Web サービス、家電、自動車、宇宙機などだけでなく、社会や組織、事業やイベントなどもシステムである。よってシステムングの適用範囲も、世の中のほぼ全てのこととなり、どのような問題も扱うことができる。ただし、対象をシステムとして捉えてシステムングを適用することにメリットがある場合とそうではない場合があることには注意が必要である。対象や状況がそれほど複雑でない場合には、システムとして捉えてシステムングを適用することで、解決に対して遠回りになってしまう場合もある。

4.2.2 視点をわける

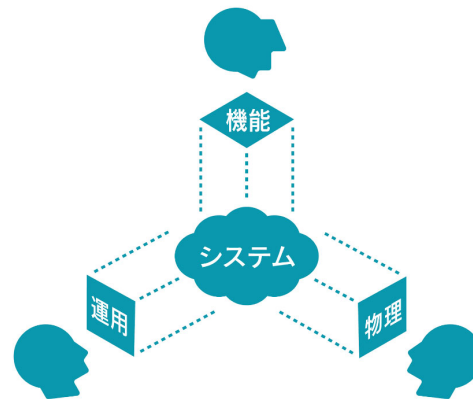


図5. 視点をわける

4.2.3 視点をつなげる

4.3 ディスクリプション

記述中

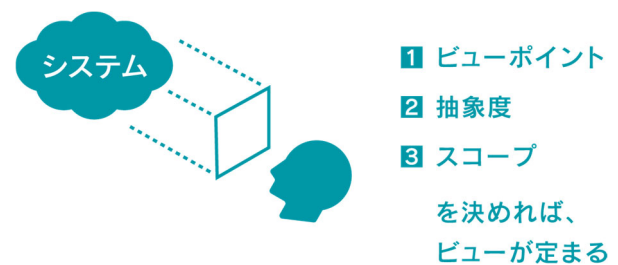


図6. ビューを規定する

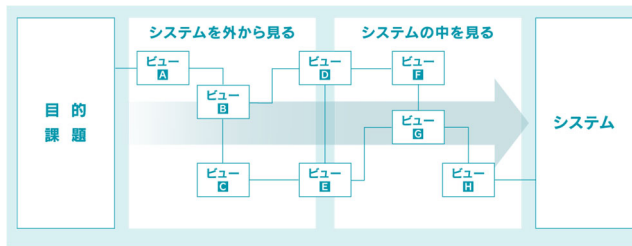


図 7. ビューモデル

4.4 プロセス

記述中

5. おわりに

記述中

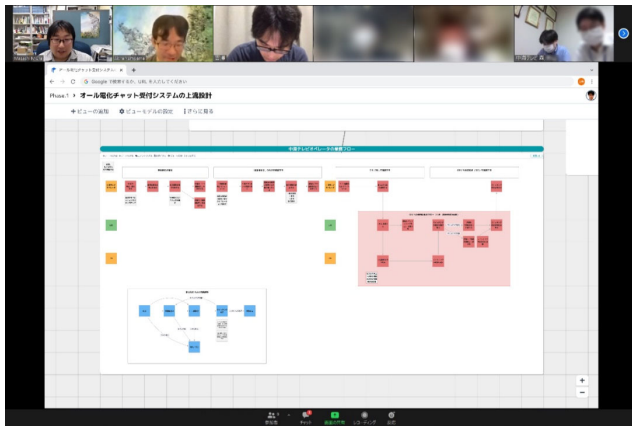


図 8. Balus で対話型モデリングを実践する様子[14]

ームをつくる,” 応用物理, 第 91 巻, 第 4 号, pp.235-239, 2022

- [12] Y. Nambu, M. Miura, R. Yoshizawa, T. Hagihara, S. Kimura, A. Yumiyama and S. Igarashi, “Development of Open Model-based Collaboration Tool and Application on Nano-satellite Project,” Transactions of the Japan Society for Aeronautical and Space Sciences, Aerospace Technology, Vol.17, No.4, pp.412-420, 2018
- [13] 三浦政司, 吉澤良典, 弓山彬, 南部陽介, 山舗智也, “システムモデルを用いた対話型上流設計によるサービス開発—モデルで納品・モデルで開発・モデルで検証—,” 情報処理学会第 84 回全国大会, 1B-01, 2022.
- [14] “中海テレビ放送様事例：Balus を用いた対話型モデリングによって、スムーズで手戻りのないサービス開発を実現” . <https://levii.co.jp/cases/21/>, (参照 2022-06-01).

参考文献

- [1] M. Oliver, K. Anshuman and K. Andreas, “Managing in a VUCA World,” Springer, 2016
- [2] “DX レポート～IT システム「2025 年の崖」の克服と DX の本格的な展開～” . https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/pdf/20180907_03.pdf, (参照 2022-06-01).
- [3] INCOSE, “Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities,” Wiley, 2015
- [4] ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes, 2015
- [5] NASA, “Systems Engineering Handbook,” rev.2, 2020
- [6] ISO/IEC/IEEE 42010:2011 Systems and software engineering — Architecture description, 2011
- [7] 井上 雅裕, 陳 新開, 長谷川 浩志 「システム工学-問題発見・解決の方法」, オーム社, 2021
- [8] David Long and Zane Scott, “A primer for Model-Based Systems Engineering 2nd Edition,” Vitech Corporation, 2011.
- [9] Bill Schindel and Rick Dove, “Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern”, 26th annual INCOSE international symposium, 2016
- [10] 三浦政司, “SYSTEMIN —システムデザインのためのフレームワーク—,” 株式会社レヴィ, 2020
- [11] 南部陽介, 三浦政司, 山舗智也, “協働を生み出すプラットフォームをつくる,” 応用物理, 第 91 巻, 第 4 号, pp.235-239, 2022