2J-09

STAMP/STPA 安全分析のためのツールの開発

佐藤　葉介†　　岡本　圭史†
仙台高等専門学校　情報システム工学科†

1. はじめに

IoT に代表されるように，情報システムは他の情報システムとの相互作用を要求されるようになりつつある．また，対象システム自体が複数のサブシステムを内包し，内部にサブシステム間の相互作用が存在するようなシステム・オブ・システムズとして構成されるようになってきている．このような状況下では，各コンポーネントの理解だけからシステム全体の動作を理解することは困難である．特にシステムの安全性において，システム全体を理解することができなければ事故を防止できない場合がある．事故を防ぐためにはシステム全体のメカニズム,技術,プロジェクト間の連携など複雑化したコンポーネント間の相互作用に着目してシステムを分析する必要がある．

2. STAMP/STPA

マサチューセッツ工科大学の Leveson 教授は，システム内で安全制御を行うコントローラと制御される被コントロールプロセスの相互作用に着目したモデルである STAMP (System-Theoretic Accident Model and Process) を提唱した．[1] さらに Leveson 教授は STAMP に基づきシステム内で事故を引き起こす潜在的な要因を見つける STPA (STAMP based Process Analysis) を提唱した．

STPA は 4 つのステップを通して実施される．初めに準備 1 としてアクシデント・ハザード・安全制約の識別を行う．次に準備 2 としてシステムの関係をモデル化し,コントロールストラクチャを構築する．作成したコントロールストラクチャをもとに 3 番目のステップとして安全でないコントロールアクションの抽出を行う．最後のステップとして安全でないコントロールアクションが起こる潜在要因を識別する．これによってシステムが事故につながる要因を特定し，システムの安全性を高めることができる．[1]

3. STPA 支援ツール

既存の STPA 支援ツールである XSTAMPP を紹介し，合わせて XSTAMPP の課題を示す．XSTAMPP[2]はドイツのシュトゥットガルト大学の Abdulkhaleq 氏が開発し公開しているツールであり，XSTMAPP は STPA 分析の流れに沿って結果を保持・連携する機能を持つ．STPA では多くの分析結果を生成することになるため,分析結果の保持・連携は大変有用な機能である．

XSTAMPP にはいくつかの課題点がある．1 つ目にネイティブアプリケーションであるため実行環境によって動作しないことがある． 2 つ目に明確なプラグインの開発手順のドキュメントがなく,プラグインのプログラムの設計が困難であるため，プラグイン開発が容易でない．XSTMAPP は Eclipse の RCP[3]により開発されたツールであり,プラグインを受け付ける構造になっている．しかしプラグインを開発するためには RCP のプラグイン開発に関する知識が必要である．3 つ目にコントロールストラクチャからその一部の切り取りができないことである．しかし STPA の最後のステップでは，コントロールストラクチャからその一部であるコントールループを切り出して利用する必要がある．

4. STPA 支援ツールの改善

本研究で開発したツールでは既存ツールのいくつかの課題点を解決し，新機能を追加した．

提案ツールにおける課題点の解決方法を示す．1 つ目の課題点に対しては，特定環境への依存度が低いプログラムを構成することである．本研究では，Web ブラウザ上で動作するアプリケーションとして支援ツールを実現することで対応した．2 つ目の課題点に対しては,XSTAMPP でプラグインを作成する必要がある拡張機能のうち，主な拡張機能をツールの機能として実装することで解決した．XSTAMPP に期待される拡張機能としては,STPA ベースの新分析手法への対応機能である．そこで，ツール上で STPA-sec などの新し

Development of a Tool for STAMP/STPA Safety Analysis
Yosuke Sato†
†Dept. of Information System Engineering, National Institute of Technology, Sendai
989-3128, Sendai, Japan

い分析手法の分析結果を残せる機能を作成し,分析者が分析に注力できるようにした．3 つ目の課題点に対して STPA の最後のステップにおいて,前のステップで作成したコントロールストラクチャから図の切り取り,保存ができるようにした．

XSTAMPP に無い以下の 4 機能を実装した．

1 つ目は順序判定機能である．コントロールストラクチャにおいて基本的なコンポーネント間の順序関係の正誤を判定できる．

2 つ目は STPA Step2 で使用する入力項目である,安全でないコントロールの入力欄の構成を変更した．XSTAMPP ではコンポーネントごとにデータの入力ができるが本ツールでは安全でないコントロールアクションベースごとにハザード要因を入力できる．（図 1）

Causal Factors Table

add guide word

unsafe control actions	guide word1	guide word2	guide word3
uca1		hcf1	
uca2			hcf2

図 1　ハザード要因の入力例

3 つ目はコントロールストラクチャを入力する際に,利用者定義のコンポーネント要素を利用できる機能である．（図 2）STPA で定義されているコントローラや被コントロールプロセスといったコンポーネント要素に加え，本ツールでは利用者がコンポーネント要素を定義し,利用できる．

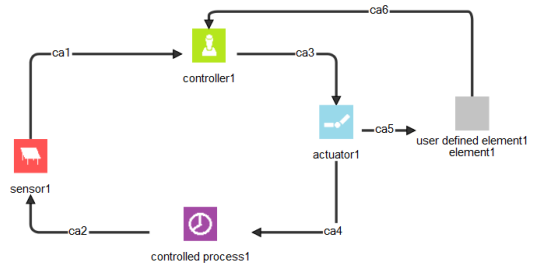

図 2　利用者定義のコンポーネント要素の使用例

4 つ目は利用者定義のガイドワードを利用できることである．STPA Step2 ではガイドワードを参考に安全でないコントロールを分析する．ガイドワードは STPA で明確に決められておらず,システムによって変わるためツール利用者がガイドワードを定義できると自由度が高くなる．

STPA 支援機能以外の特徴として，本ツールでは WebStorage をデータベースとして利用している．WebStorage はブラウザに付属していて,データはローカルの記憶装置に保存される．SQL と比較した場合,WebStorage はツールをブラウザ上で利用することから新たにデータベースのための環境を実装する必要がなく,ツール利用者への負担が少ない．

6．まとめ

本研究では STPA を支援するツールを開発した．既存ツールである XSTAMPP のいくつかの課題点を改善し，利便性向上のための新機能を実装した．本ツールはブラウザで動作するツールであるため,ネイティブアプリケーションである XSTAMPP よりも環境に依存しない．また利用者定義のコンポーネントを利用できることでコントロールストラクチャ構築の自由度が高くなった．なお，現段階では STPA ベースの分析手法の支援のみに対応しており,CAST ベースなどの他の分析手法には対応していない．

最後に将来課題を述べる．STPA は人手による作業が主であるが，モデル検査法等との連携による自動化が研究されている．[4] モデル検査器はモデルが時相論理式を満たすかの判定を行う．コントロールストラクチャはプロセスモデルを含む．プロセスモデルに含まれるプロセス変数はコンポーネントの状態を反映しており,コントロールストラクチャはシステムの状態を包含する図であるため,コントロールストラクチャは状態遷移システムと見ることができる．また安全制約やシステム要件などはモデル検査の検査式とみなせる．したがって，今後は STPA の分析用データ及び分析結果から検査用のモデルと検査式への変換法を提案し，提案変換法をツールとして実現する．

参考文献

[1] N. G. Leveson, “Engineering a Safer World”, The MIT Press, 2011.
[2] B. Asim, “XSTAMPP: An eXtensible STAMP platform as tool support for safety engneering”, 2015 STAMP Workshop, 2015.
[3] developersWorks IBM, “Eclipse RCP アプリケーションを独自ブランド化する”, https://www.ibm.com/developerworks/jp/opensource/library/os-eclipse-brand. [アクセス日: 31 12 2016].
[4] B. Asim, “XSTAMPP 2.0: New Improvements to XSTAMPP Including CAST Accident Analysis and an Extended Approach to STPA”, 2016 STAMP Workshop, 2016.