従属故障分析を考える

今回は、ISO 26262 における従属故障分析（DFA: Dependent Failure Analysis）について考えてみます。一般的には、安全分析（演繹的・帰納的手法）を通じてリスクを特定し、適切な対策を講じる流れが取られますが、従属故障分析がどのようにこのプロセスに関与するのか、また、どのような課題があるのかについて考察していきます。

従属故障分析とは

従属故障分析は、カスケード故障（CF：Cascade Failure）、共通原因故障（CCF：Common Cause Failure）を特定するために実施される分析活動です。

安全分析（FMEA、FTA）との違い

安全分析（例：FMEA、FTA）と従属故障分析の違いは下表の通りです。

手法	目的	考え方
FMEA	安全機構の各故障モードが安全目標を侵害するかを分析する。	各エレメントや機能の故障モードを洗い出し、それが安全目標に与える影響を評価。 重大度、発生頻度、検出可能性を基にリスクを評価し、高リスクな故障モードを特定する。
FTA	安全目標の侵害を引き起こす故障経路を特定し、根本原因を分析する。	トップ事象（安全目標の侵害）を分析し、その事象を引き起こす可能性のある故障モードを分析しツリー状に展開する。
DFA	データフローやインタフェースを通じて、故障がシステム全体に与える影響を分析する。	システム内の依存関係や相互作用を分析し、各エレメント間のデータフローやインタフェースに関連する故障が他のエレメントにどのように影響を与えるかを分析する。

従属故障分析は本当に必要なのか

FMEA や FTA でインタフェースやデータフローを考慮したツリー構成を採用しているのであれば、従属故障分析を実施する必要はないのではないか、という疑問を抱く方もいらっしゃるかもしれません。本稿では、FTA を例にとり、その必要性を考えていきたいと思います。

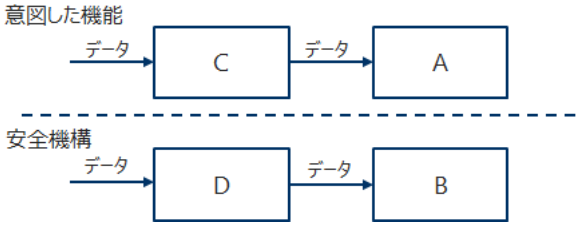

図 1 前提とするシステムアーキテクチャ

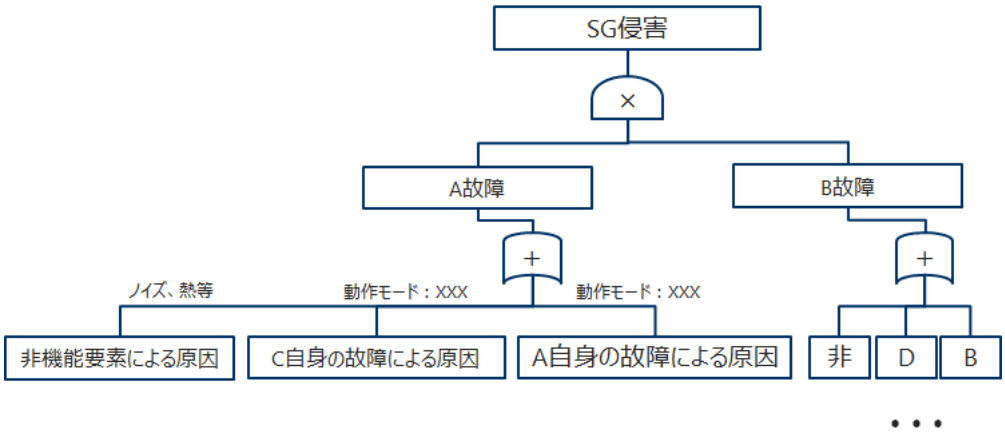

図 2 データフローが表現されていない FTA

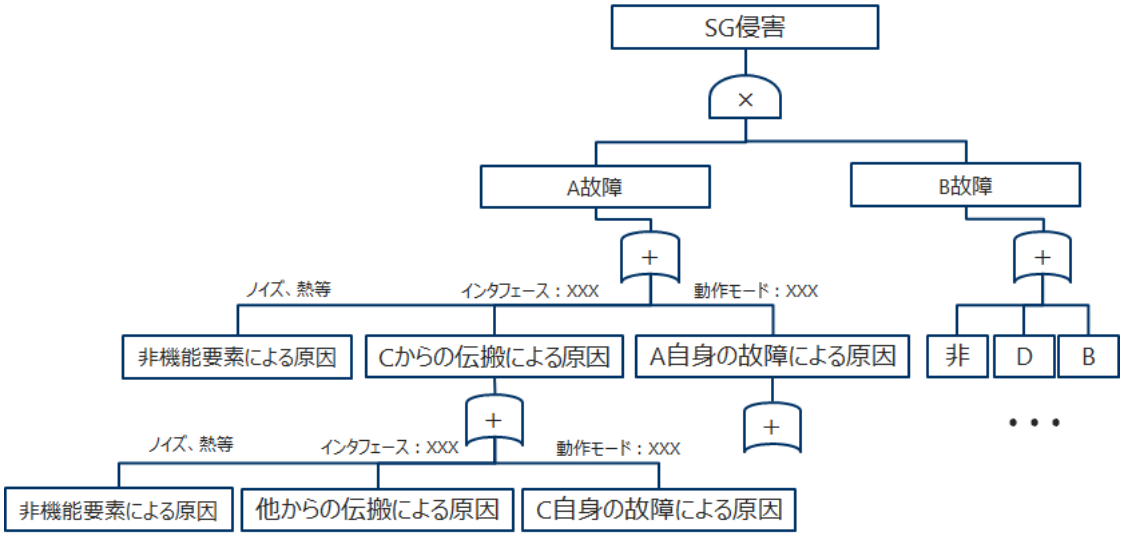

図 3 データフローが表現された FTA

図 1 は前提とするシステムアーキテクチャを示しており、A、B、C、D はシステムエレメントを表しています。A は C の出力を入力として動作し、B も同様の動作を行います。

図 2 はミニマルカットセット化されたツリー構造で、データフローの関係は表現されていません。OR ゲートの下にある要素はすべて並列であるため、エレメント間の依存関係を把握することができません。また、共通原因故障については AND 要素から外れ、単一故障として表現されてしまうため、その因果関係を本ツリーから

直接検証することは困難です。

一方、図 3 は各エレメント間のデータフローを表現したツリーであり、カスケード故障や共通原因故障の関係を可視化できる構造となっています。そのため、安全目標を侵害する要因の妥当性に限っては、本ツリーを用いて検証することが可能です。しかし、どちらの方法においても共通して言えることは、「安全目標を侵害しない根拠」を示すことができないという点です。

安全分析が先か、従属故障分析が先か

「なぜ安全目標を侵害するのか」という点よりも、「なぜ安全目標を侵害しないのか」の根拠の方が実は重要です。侵害する要因が判定されれば対策を講じることができますが、侵害しないと判断された要因に関しては、抜け漏れを引き起こす可能性があるからです。このように安全分析には、担当者の思い込みが入り込むリスクがあり、かつ外からは確認しづらいという特徴があります。そのため、事前に従属故障分析を行い、網羅的に伝播経路や共通原因故障を特定してから、その結果を基に安全分析を実施することが合理的であると考えます。

課題

とはいえ、各エレメントのインタフェースや振る舞いに対して従属関係を分析するには、莫大な工数がかかるだけでなく、分析結果の記録方法（ルールやテンプレート）の策定も大きな課題です。論証に耐えうる根拠を残しつつ、分析を簡略化する必要があるため、効率的な方法を模索することが求められます。

ワークショップ

現在、弊社では従属故障分析に関するワークショップを企画しています。開催時期は未定ですが、皆様にとって有意義な時間となるよう、鋭意開発中です。告知の際は、ぜひご参加いただければ幸いです。

2025/4/23 山田 毅
yamada@biz3.co.jp

