衛星内の情報伝達経路モデルに基づく不具合分析支援に関する研究

2020 年 12 月 01 日　　03-183005 西本 慎吾

概要

近年，大学や高専などの教育機関や，民間企業による超小型衛星の開発，およびそれを利用した事業の展開が盛んになっている．一方で，超小型衛星の信頼性の低さが問題となっている．軌道上故障に関する調査の結果信頼性の低さの原因として，設計および製造過程における不良が多いことが分かっており，地上試験によって不具合の改修，対策を十分に行うことが重要である．しかし，衛星のような複雑なシステムでは，一つの不具合事象に対して多くの故障が考えられ，不具合事象から故障箇所の特定を行うことは非常に多くの知識と経験を必要とする．そこで，本研究ではコンポーネント間の接続関係モデル，情報伝達経路モデルを用いて衛星の故障候補の検証手順（打つべきコマンド，確認事項）を探索し，それらをコマンドの安全性及び，故障候補切り分け能力を示す指標と共に提示することで，不具合分析を支援する手法を提案する．本手法を用いて，簡易的な衛星モデルに対して不具合分析を実践することで，コマンドによる故障箇所の特定作業が体系化できること，設計不備の発見につながることを確認した．

C : コマンド
T : テレメトリ
l : リンク (コンポーネント間の接続関係)
R : 経路
$P(l_i = \text{normal})$: リンク l_iの正常確率
$\mathbb{F}$: 経路 R 内の故障候補リンクの集合
$\mathbf{N_F}$: コマンドが形成する経路内の故障候補リンクの数

1　序論

1.1　研究背景

超小型衛星開発に大学などの参加が増加している中，信頼性の低さが問題となっている [1]．軌道上故障の調査の結果，衛星の故障原因の多くは設計・製造過程にある [2] ことが分かっており，それらの多くは地上試験によって確認することができるものであるという調査結果が出ている [3]．

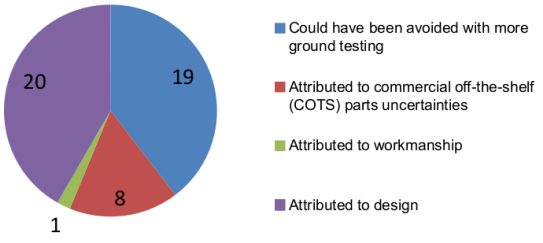

図 1　超小型衛星の故障原因に関する調査結果 [2]

1.2　問題提起

以上より，地上試験での不具合分析が不十分になっていることが，超小型衛星の信頼性の低さの原因の一つである．地上試験での不具合分析を十分に行うためには以下の 2 点の作業に高い知識と経験が必要とされる．

- 故障仮説の生成
- 故障候補の検証

まず，故障仮説の生成は FTA(Fault Tree Analysis) などを用いて不具合事象から考えられる故障モードを網羅的に洗い出す．衛星は内部機器の物理的相互作用が複雑に絡み合っているため，人による思い付きでは網羅的に行うことは困難である．
また，故障候補の検証は衛星から得られる情報を元に衛星の安全を確保しながら行う必要がある．実ミッションで使用するコマンドとテレメトリは膨大な数であるため，その中から切り分けを行うための情報を選択し，仮説の検証を行う作業は無駄やヒューマンエラーを生むきっかけとなる．
これらの課題に対して，表 1 に示すように，故障候補の洗い出しを網羅的に行う研究が盛んにおこなわれている．一方で，不具合分析の大きな課題の一つである検証過程に関して取り組んだものは少ない．

表 1　不具合分析手法の比較

手法	故障網羅性	手法の目的
GDE	低	故障仮説生成
GDE+[4]	中	故障仮説生成
網状故障解析 [5]	中	異常モード洗い出し
故障オントロジー [6]	高	故障仮説生成

1.3　本研究の目的

以上より，衛星の故障箇所を特定する作業を体系化し，不具合分析経験の少ない人が十分に不具合分析を行える様に支援することが必要である．よって，本研究では以下の機能を満たす不具合分析支援手法の提案を目的とする．

- 故障候補を確認するためのコマンドおよびテレメトリを提案する．
- コマンドを選ぶ際の判断の指標を定量的に提示する．

2　情報伝達経路モデルに基づく不具合分析支援手法

2.1　手法の概要

上述した機能を実現するために，本手法は以下の要素から構成されている．

- 衛星内部機器の接続関係モデル及び情報伝達経路モデル
- 故障仮説検証の流れ及び検証用コマンドの探索アルゴリズム
- コマンドの安全性及び，故障候補切り分け能力を示す指標

本手法を用いた不具合分析システムの構成を図 2 に示す．本手法は人間と対話的に故障箇所の特定を行う．これにより 実機の情報をシステムに反映しながら故障箇所を絞り込むことができる．

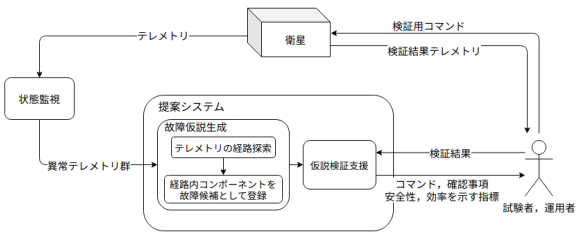

図 2　本手法による不具合分析の構成

2.2　モデル

本手法で用いる情報伝達経路モデルを作成するために必要な，各モデルに関して以下に示す．

2.2.1　コンポーネント間接続関係モデル

來村ら [7] は拡張デバイスオントロジーとして，機器間の接続関係を「ポート」と「導管」という概念を用いて表現している．これを元に，コンポーネントの接続関係を表す「リンク」(表 2) を定義した．各リンクには正常確率を属性として持ち，これを用いて後ほど述べるコマンドの故障候補切り分け能力を定量化している．また，表 3 のように各コンポーネントがリンクを属性として持ち，コマンドの情報伝達で使用するリンク (コマンドリンク) とテレメトリの情報伝達で使用するリンク (テレメトリリンク) を区別している．

表 2　リンク定義

ID	Link_name	Compo1	Compo2	Medium	Probability
19	PCU-XTx	PCU	XTx	Power	0.5
20	TOBC-HTR_PROP	TOBC	HTR_PROP	Power	0.5
21	TOBC-HTR_PANEL	TOBC	HTR_PANEL	Power	0.5
22	TOBC-HTR_BAT	TOBC	HTR_BAT	Power	0.5
23	TOBC-HTR_CAM	TOBC	HTR_CAM	Power	0.5
24	TOBC-TS_PROP	TOBC	TS_PROP	Signal	0.5
25	TOBC-TS_PANEL	TOBC	TS_PANEL	Signal	0.5
26	TOBC-TS_BAT	TOBC	TS_BAT	Signal	0.5
27	TOBC-TS_CAM	TOBC	TS_CAM	Signal	0.5
28	AOBC-AOCS	AOBC	AOCS	Signal	0.5
29	AOBC-AOCS	AOBC	AOCS	Power	0.5
30	MIF-XTx	MIF	XTx	Signal	0.5
31	MIF-CAM	MIF	CAM	Signal	0.5
32	HTR_PROP-PROP	HTR_PROP	PROP	Heat	0.5

表 3　コンポーネント定義

Component	Com_linkID	Tel_linkID
GS	1	
MOBC	7,8,9,10,11	6
PCU	11,12,13,14,15,16,17,18,19	11
TOBC	20,21,22,23,24,25,26,27	8
AOBC	28,29	
MIF	31	30
XTx		3,7
STx		2
SRx	4,5	

2.2.2　情報伝達経路モデル

以下の表 4,5 のようにコマンドおよびテレメトリを定義した．それぞれ情報伝達の経路を上述のリンクによって表現している．また，コマンドの属性として「コマンド送信によって変化するテレメトリ」及び「種別」を定義している．これらによって，コマンドが起こす状態変化を表現し，内部状態の更新を行っている．

表 4　コマンドモデル

ID	CommandName	impact_TEL_ID	type	path				
19	HTR_CAM_OFF	5,6,10,18,22	ACTION	1	4	8	23	35
20	HTR_BAT_OFF	5,6,10,19,23	ACTION	1	4	8	22	34
21	AOCS_ON	5,7,11,25	ACTION	1	4	9	29	
22	AOCS_OFF	5,7,11,25	ACTION	1	4	9	29	
23	RW_START	5,7,11,26	ACTION	1	4	9	28	
24	RW_STOP	5,7,11,26	ACTION	1	4	9	28	
25	M_DATA_DOWN	5,8	GET	1	4	10	31	
26	GET_PANEL_TEMP	5,6	GET	1	4	8	25	

表 5　テレメトリモデル

ID	TelemetryName	TransitionTrigger	path				
10	TOBC_Current	Command	13	11	6	2	
11	AOBC_Current	Command	14	11	6	2	
12	MIF_Current	Command	15	11	6	2	
13	SRx_Current	Command	17	11	6	2	
14	STx_Current	Command	18	11	6	2	
15	XTx_Current	Command	19	11	6	2	
16	PANEL_Temp	Command	41	25	8	6	2
17	PROP_Temp	Command	40	24	8	6	2

また，テレメトリのモデルでは，テレメトリに変化を及ぼすトリガの種類を指定しており，これによって故障箇所特定に必要な情報取得のために取る行動を決めることができる．ここでは簡単のため，軌道運動などによる状態変化は考慮せず，時間とコマンドによる状態遷移のみを考えている．

2.3　故障仮説検証の流れ

本手法によって故障仮説の検証を行う流れ及び検証用コマンド探索のアルゴリズムを図 3 に示す．故障仮説の検証を行う流れとして，まずは取得済みテレメトリによって確認を行い，その後コマンドによる確認に移る．この時，検証用コマンドの探索アルゴリズムは図に示す通りで，コマンドとテレメトリによる経路が故障候補のリンクを通り，そのコマンドによって発生する状態変化があれば，故障候補を確認可能なコマンドとしている．

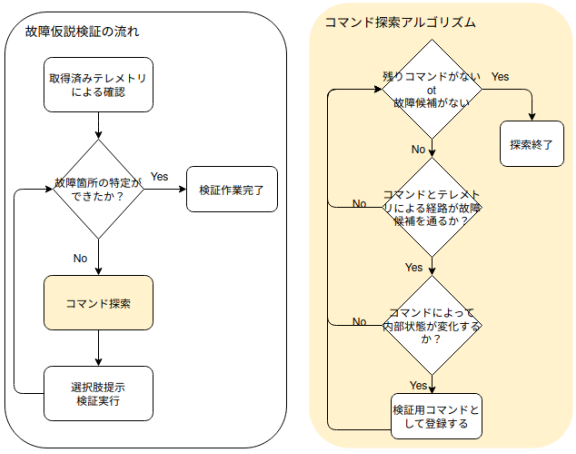

図 3　故障仮説検証の流れ及び検証用コマンド探索アルゴリズム

2.4　評価指標の提案

本手法の対象は地上試験における支援であるが，不具合分析に利用する情報の粒度がコマンドとテレメトリのみであるため，軌道上不具合発生時の故障箇所特定にも利用可能である．そこで，以下では地上試験及び軌道上での運用時の両方で重要となる指標を提案し，本手法が両状況で使い分け可能な枠組みであることを示す．

2.4.1　コマンドの衛星生存性への副作用

まず，生存への副作用を示す指標として，以下の 3 点を与える．

- コマンドを打つ前の電力状態と，コマンドを打つことによって発生する電力消費量
- 姿勢変化を起こすか否か
- コマンド送信によって変化するテレメトリの数

前者 2 点の電力と姿勢の制約による指標は運用時に特有のものであり，コマンドを打つことで衛星の安全を脅かすことがないように危険な動作を明示的に示すことで，未熟な運用者による誤ったコマンド送信を防ぐ目的がある．また，不具合発生時は衛星の状態に対する把握が不十分であるため，衛星の状態を大きく変化させるコマンドは危険であるといえる．そのため，コマンドによって発生する状態変化の大きさを定量的に示す指標として 3 点目の指標を与えている．

2.4.2　コマンドの故障候補切り分け能力

運用時は可視時間が限られており，その時間内に不具合の改修を行わなければミッション失敗につながるような，時間制約を考慮した不具合分析を行う場面が考えられる．その際には，少ないコマンド数で効率的に故障箇所の特定を行えることが望ましい．以下ではコマンドの切り分け能力として，一つのコマンドによる切り分け能力を示す指標と全体の効率を考えた指標に関して示す．

まず，一つのコマンドで切り分けられる故障候補の数を表す指標に関して述べる．以下の図 4 の例に示すように，あるコマンド C_k によって形成される経路が複数存在する場合を考える．あるリンク l_i の状態を確認するためにはその経路 R_j 内にある他のリンクが正常である必要があるため，l_i を確認できる確率は式 (1) となる．

$$P(l_i|\mathrm{R}_j) = \prod_{m\in\mathbb{F}_j, i\neq m} P(l_m = \mathrm{normal}) \qquad (1)$$

また，$P(l_i|\mathrm{R}_j)$ はコマンドが形成する各径路すべてに対

して求まるのでそれらの最大値を取りコマンド C_k による l_i の確認可能性は式 (2) となる．

$$P(l_i|\mathrm{C}_k) = \max\{P(l_i|\mathrm{R}_{j_k})\} \tag{2}$$

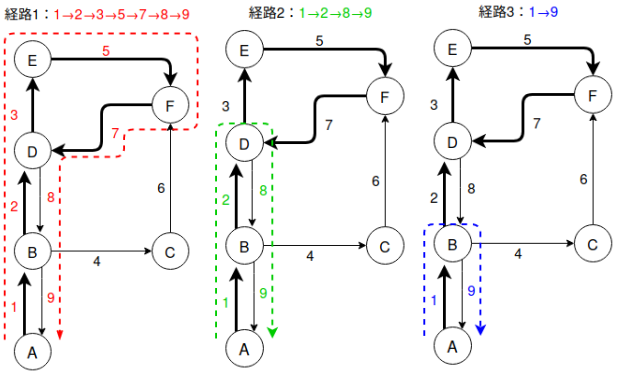

図 4　故障候補 (太矢印) と検証用コマンド C_1 による情報伝達経路の例

また，式 (2) が各リンクに対して求められるのでそれらの平均である「平均確認可能性 $P_m(\mathrm{C}_k)$」，及び C_k によって確認可能なリンクの数を表す「確認可能リンク数 $E(\mathrm{C}_k)$」が以下のように求まる．

$$P_m(\mathrm{C}_k) = \frac{1}{\mathbf{N_{F_k}}} \sum_{i=1}^{\mathbf{N_{F_k}}} P(l_i|\mathrm{C}_k) \tag{3}$$

$$E(\mathrm{C}_k) = \mathbf{N_{F_k}} P_m(\mathrm{C}_k) = \sum_{i=1}^{\mathbf{N_{F_k}}} P(l_i|\mathrm{C}_k) \tag{4}$$

式 (3),(4) がどちらも高いコマンドを選択することで，一つのコマンドでより多くの故障候補を絞り込むことが可能である．

次に，あるコマンドから検証を開始した時に，最終的に故障箇所の特定を行うまでにかかるコマンドの総数に関して述べる．図 5 に示すように，あるコマンドによる検証を考えると各テレメトリの結果によって検証結果が異なる．この時，故障候補が残っている場合にはそれに応じたコマンドの探索を行う必要がある．そのため，図 5 に示す各 Case それぞれで，最終的に故障箇所を特定するまでのコマンドの総数が異なる．

各テレメトリが正常値を示すか否かは各リンクごとの正常確率を用いて式 (5), (6) のように算出でき，これを元に図 5 の各 Case になる確率が求まる．

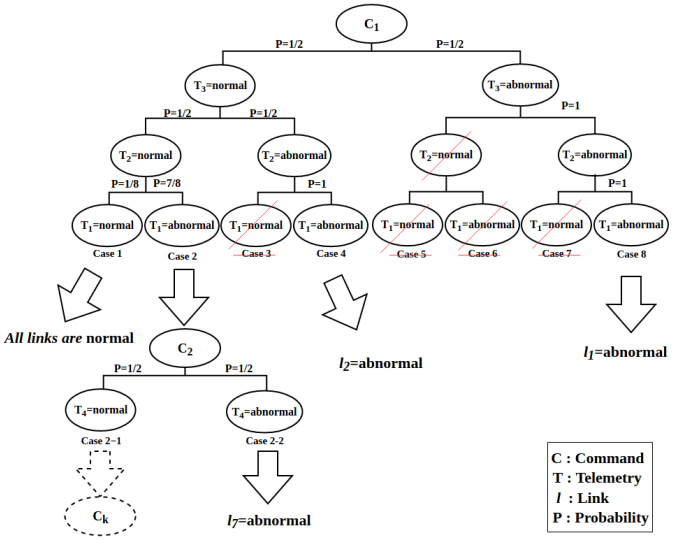

図 5　検証プロセスの全体像（テレメトリの ID は図 4 の経路と対応）

$$P(\mathrm{T}_j = \text{normal}) = \prod_{i \in \mathbb{F}_j} P(l_i = \text{normal}) \tag{5}$$

$$P(\mathrm{T}_j = \text{abnormal}) = 1 - P(\mathrm{T}_j = \text{normal}) \tag{6}$$

また，テレメトリの結果の組み合わせによって，各 Case になる確率 $P(\text{Case } i)$ が求められる．これを用いて式 (7) のように検証にかかるコマンド数の期待値が求められ，これを「検証コマンド総数」と定義する．ここで，$\mathbb{C}$ は検証が終了した結果の各場合 (Case) の集合である．

$$N(\mathrm{C}_k) = \sum_{\text{Case } i \in \mathbb{C}} P(\text{Case } i) N_{\text{Case } i} \tag{7}$$

検証コマンド総数が少ないコマンドを選択することによって，全体的にかかるコマンドの数の期待値が小さい検証プロセスを選択することができ，時間制約を考えた場合に重要な指標であると言える．

2.4.3　評価指標の使い分け

まず，安全重視で故障候補の切り分けを行う場合は，電力及び姿勢を考慮し，コマンド送信によって変化するテレメトリの数が少ないコマンドを選択すれば良い．また，効率重視で故障候補の切り分けを行う場合は，平均確認可能性及び確認可能リンク数が高く，検証コマンド総数が小さいコマンドを選択すれば良い．

この時，切り分け能力の高いコマンドは同時に，「変化するテレメトリの数」が多くなる傾向にあるので効率と安全性を両立させることは難しく，トレードオフを考えて選択する必要がある．

また，地上試験において電力や姿勢が制約になることはないため，考慮する必要はない．

3　本手法による不具合分析の実践と評価

3.1　問題設定

以下の図 6 のような構成の簡易的な衛星のモデルを対象にし以下の 2 つの故障状態の例に対し，本手法を用いた不具合分析を実践した．

- ヒータの接触不良
- 温度計故障 (断線)

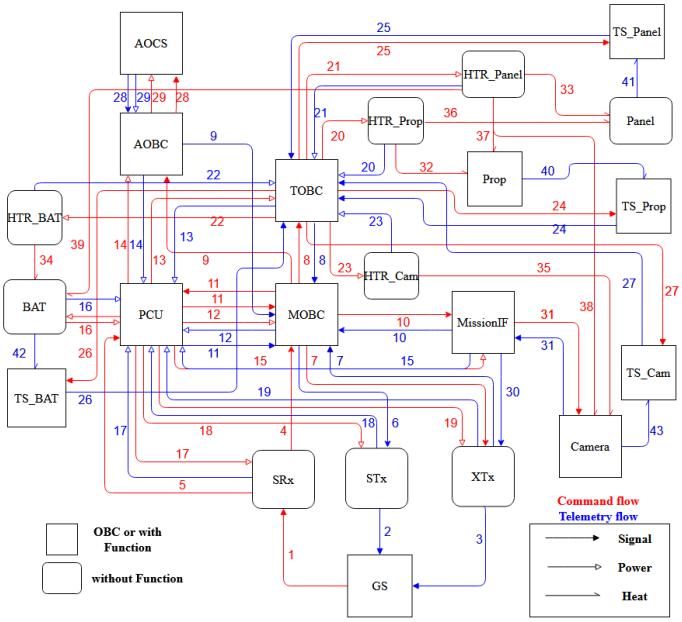

図 6　衛星内コンポーネント接続関係図 (数字はリンクの ID)

3.2　実践結果

3.2.1　ヒータの接触不良

図 6 に示す，推進系ヒータ (HTR_PROP) と推進系 (PROP) の間 (ID:32) の故障を考える．この時，現れる不具合事象は「推進系ヒータ ON コマンドを送った時，推進系温度が変化しない」である．本手法では，不具合発生時のコンポーネントの状態として，送信したコマンドによって変化した後の状態になっていると仮定して，初期状態を与えている．

上述した不具合分析の流れに従って，取得テレメトリでの切り分けから行い，検証用コマンドの探索した結果が図 7 のようになる．

```
COM 13 HTR_PANEL_ON
        Pm(C): 0.5 , E(C): 1.0 , N(C): 2.0
        Remain Power: 3.8 , Power Consume: 2 , Impacted TEL num: 8 , Attitude: Keep
COM 18 HTR_PROP_OFF
        Pm(C): 0.25 , E(C): 0.75 , N(C): 1.875
        Remain Power: 3.8 , Power Consume: -1 , Impacted TEL num: 6 , Attitude: Keep
```

図 7　検証用コマンド探索結果

選択肢として，パネルヒータ ON(13:HTR_PANEL_ON) と推進系ヒータ OFF(18:HTR_PROP_OFF) が提示されており，どちらから選択しても図 8 のように，故障箇所の特定ができた．

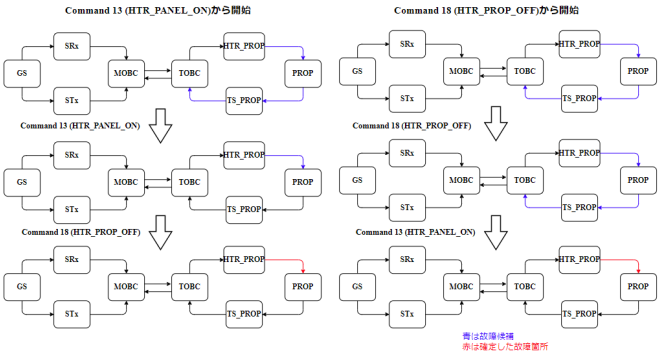

図 8　ヒータ接触不良時の検証プロセスによる違い

3.2.2　コマンドの指標に関する考察

図 8 に示したように，コマンドの選択順序によって切り分けを行う過程に違いが見られた．まず，平均確認可能性 (図 7 中の Pm(C)) が高い「パネルヒータ ON」から検証を行ったほうが一つ目のコマンドで大きく絞り込めている．一方で，「推進系ヒータ OFF」から検証した場合は 1 つ目の検証では故障候補を切り分けることができず，2 つ目のコマンド「パネルヒータ ON」によって推進系温度の上昇を確認できるため，「PROP-TS_Prop 間」，「TS_Prop-TOBC 間」の正常が確認できている．運用時，通信が不安定であり不具合分析に使える時間が明確でない時は，一度のコマンドで多くの確認ができることが望ましいため，図 8 の左側のように平均確認可能性が高いコマンドから検証を行うことで，1 度のコマンド送信により，多くの絞り込みができる可能性が高いと言える．

3.2.3　本手法と人間の不具合分析の違い

上で示した不具合事象「推進系ヒータ ON コマンドを送った時，推進系温度が変化しない」に関して，同様の条件を与えて不具合分析を行う際の過程（送信するコマンド，コマンドを選択する理由，確認するテレメトリ）を，本研究室の方々に対して調査した．

調査結果によると，多くの人が本手法での検証プロセスと同様な手順でコマンドの選択を行っており，人間の推論と近い形で検証用のコマンドを探索できていることが分かる．一方で，コマンドを選択した理由によると「推進系ヒータ OFF」コマンドを選択するのは故障可能性のあるコンポーネントによって二次故障の発生を防ぐために用いている．安全のために選択した「推進系ヒータ OFF」コマンドによって切り分けられる故障候補も存在し，確認事項を明示的に提示してくれるところが本手法の強みであると言える．

3.2.4　温度計故障

最後に，推進系温度計の断線 (テレメトリリンク 24) に対する実践例を示す．不具合事象としては上と同じものが考えられ，検証用コマンドの探索結果も同様になる．温度計故障時の検証プロセスを以下の図 9 に示す．

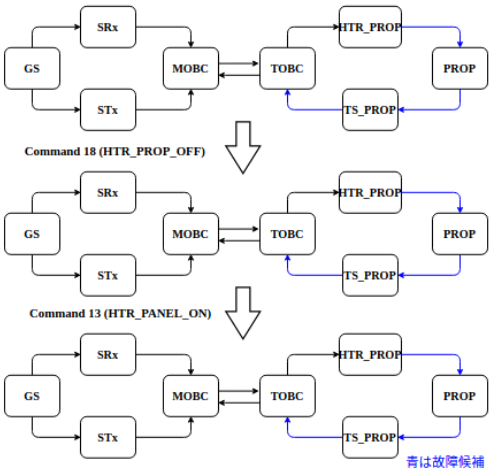

図 9　温度計故障時の検証結果

温度計の故障では，推進系への熱の伝わりを読み取る経路が 1 本しか無いため，熱の伝わりを観測することができす，本システムのみでは故障箇所の特定を行うことができなかった．

一方で，不具合検証のプロセスで得た情報と人間による推論の組み合わせによって，故障候補を絞り込むことは可能である．まず取得テレメトリ「パネル温度」の上昇を確認できれば推進系ヒータが正常に動作していることが分かる．また，「パネルヒータ ON」による推進系温度の変化が確認できなければ，故障箇所が確実に「PROP-TS_Prop 間」，「TS_Prop-TOBC 間」の中に存在することが分かる．

このように，提示された選択肢に従って検証を行うことで，故障箇所の推論に必要な情報を得ることができる．また，人間の推論を組み合わせても故障箇所の特定が行えなかった場合には，設計の不備を考えることができ，設計の正しさを確認するためにも利用できる．

4　結論

4.1　まとめ

本研究では，衛星の情報伝達経路モデルを用いてコマンドによる故障箇所特定の過程を体系化する手法，及びコマンドの安全性と故障候補切り分け能力の大きさを示す指標を提案し，テストケースで実践した．

実践例では，複数のコマンドの中から故障箇所特定のために適切なコマンドを提示し，想定した故障箇所を特定できることを示した．次に，提案した指標に基づいてコマンドを選択することで時間制約の厳しい不具合分析の際に効果的な分析が行える可能性があることを示した．また，システムのみでは特定を行えなかった場合も，人間の推論と組み合わせることで故障候補の絞り込みができ，提示された選択肢に従うことで推論に必要な情報を効率的に得ることができることが分かった．

一方で，本手法を用いて最終的な故障箇所の特定を行うことができる故障モードの多くは接続関係に関する故障であり，コンポーネント自体の故障に関して故障内容までを特定することはできない．安全性の評価として，不具合発生時の状態を保持する危険性を考慮できていなかった．一度不具合発生前の状態に戻すという操作に関して．

4.2　今後の課題

今後以下の様な課題に取り組む必要があると考えている．

- コンポーネントの機能の接続関係を組み込んだ，より粒度の細かい故障箇所特定
- リンクの正常確率に実機の情報を組み込むことによる，検証の効率化
- 設計情報からのモデル自動生成
- コマンドの安全性の評価・・・

参考文献

1) M Langer and J Boumeester. Reliability of CubeSats Statistical Data, Developers' Beliefs and the Way Forward. *Proceedings of 30th Annual*

AIAA/USU Conference on Small Satellites, pp. 1–12, 2016.

2) Catherine C Venturini. Improving Mission Success of CubeSats. Technical report, 2017.

3) Hirobumi SAITO. Secondary Analysis on On-Orbit Failures of Satellite. *JOURNAL OF THE JAPAN SOCIETY FOR AERONAUTICAL AND SPACE SCIENCES*, Vol. 59, No. 690, pp. 190–196, 2011.

4) Peter Struss and Oskar Dressier. "Physical Negation" - Integrating Fault Models into the General Diagnostic Engine. Vol. 89, pp. 1318–1323, 1989.

5) Kota Yamaguchi and Hori Koichi. Fault Network Analysis of Artificial Satellite Using Ontology. pp. 1–4, 2014.

6) 來村徳信, 西原稔人, 植田正彦, 池田満, 小堀聡, 角所収, 溝口理一郎. 故障オントロジーの考察に基づく故障診断方式：網羅的故障仮説生成. PhD thesis, sep 1999.

7) Yoshinobu Kitamura and Riichiro Mizoguchi. *A Framework for Systematization of Functional Knowledge based on Ontological Engineering*. PhD thesis.