Vol.2015-MBL-77 No.5
Vol.2015-ITS-63 No.5
2015/12/2

リアルタイム制御システムの障害監視のための STAMP/STPA の適用検討

小林 良輔[†1] 鎌田 大貴[†1] 伊藤 信行[†2] 小林 幸彦[†2]
梶 克彦[†1] 内藤 克浩[†1] 水野 忠則[†1] 中條 直也[†1]

概要：リアルタイム制御システムは大規模化・複雑化の傾向にあり，システムの安全性・信頼性の低下が懸念されている．そのため安全性・信頼性を向上させる取り組みが必要となり，その一つとして障害診断手法がある． しかし，従来の診断手法はコンポーネントベース故障による事故を想定しているため大規模システムへの適用が困難である．本研究では，システム間の相互作用を診断対象とした STAMP/STPA 手法を適用し，障害監視を行う．今回の実験では，ACC を搭載した自動車を想定してハザード監視を行い，リアルタイムにハザード監視が行われているか検証する．

キーワード： STAMP/STPA，リアルタイム制御システム

1. はじめに

近年，自動車や航空機などに搭載されるリアルタイム制御システムのソフトウェアは大規模化・複雑化の傾向にある．その傾向は，システム障害時の原因特定を困難にするため，システムの安全性･信頼性を低下させる懸念がある．そのため安全性・信頼性を向上させる取り組みが必要である[1].

その手段として動作中のシステムのログデータを収集・解析して障害発生の原因をコンポーネントベースで診断するものがある．しかし，その障害診断手法には部品点数が増大すると，システム全体の障害診断が困難になるという問題点がある [2].

本研究では，システム間の相互作用に着目した障害診断手法である STAMP(Systems Theoretic Accident Model and Processes)[3]を自動車や FA 機器等のリアルタイム制御システムに適用し，障害監視を行う．今回の実験では ACC(Adaptive Cruise Control)を搭載した自動車を想定し，リアルタイムにハザード(障害を起こす要因)監視が行われているか検証する．

2. STAMP/STPA

STAMP はシステムを構成するコンポーネント間の相互作用に着目して，システム全体の流れを表す事故モデルである．システム設計段階では STPA (System Theoretic Process Analysis)という障害分析手法を適用することによりハザード分析を行っている[4]． STPA は次の手順である．

A.想定されるシステムの障害
B.CS(Control Structures)の構築
C.不適切な制御アクションの識別.
D.ハザードシナリオにつながる潜在原因の識別
E.潜在原因への対策検討，必要があれば設計見直し

今回の提案手法で必要になる B，D について説明する．

B. CS の構築

Control Structures は，システムの制御構造である．システムに関係するサブシステム間の制御アクションを識別し，図 1 のように展開して可視化する．

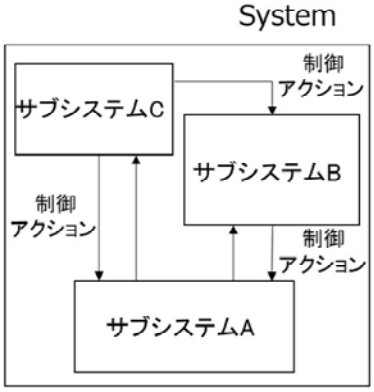

図 1 Control Structures

D. ハザードの識別

CS からハザードを起こす恐れのある制御アクションを抽出する．それらに対して 4 つのガイドワードを適用するガイドワードは以下の通りである．

・必要な制御アクションが供給されない
・誤った非安全な制御アクションが供給される
・意図しないタイミングで供給される
・途中で止まる(または必要以上に長く実施される)

これらのガイドワードによってハザードを識別する．

3. 提案手法

本研究では，リアルタイム制御システムの障害監視のための STAMP/STPA の適用を提案する．

STAMP/STPA 手法を応用して設計段階ではなく運用段階に適用する．現在考えている手順は以下の通りである．

†1 愛知工業大学
†2 三菱電機エンジニアリング

Vol.2015-MBL-77 No.5
Vol.2015-ITS-63 No.5
2015/12/2

(1)想定されるシステムの障害
(2)CS(Control Structures)の構築
(3)ハザードの識別.
(4)制約条件の識別
(5)センサネットワークによる整合性監視
この章では提案部分である(4)，(5)を説明する.

(4)　制約条件の識別と監視

3章のDで識別したハザードを検出するための制約条件(ハザードを見つけるために必要な条件)を識別する.識別した制約条件に沿った監視を行うことでハザードを検出する.

(5)　センサネットワークによる整合性監視

システムには多くのセンサが搭載されている．事象に対してセンサ値を取得している場合，センサ間で整合性がとれる.整合性を監視することでハザードの検出が可能である.

4.　実験

本章では実験について述べる．実験では，実験用リアルタイムシステムが正しく動作しないという障害を引き起こすハザードを想定して実験を行う.

4.1　実験用リアルタイム制御システム

リアルタイム制御システムの性質とは，処理要求をあらかじめ定められた時間制約内に出力する制御システムである[5]．今回の実験では，人々の身近にある自動車のシステムを想定している.

リアルタイム制御システムとして,ACCを採用した.ACCとは，センサにより先行車との車間距離を検知し，設定車速内で先行車の車速に合わせて速度を調整することで，一定の車間距離を保ちながら追従走行をするシステムである[6].

4.2　実験機器

実験機器として用いるZMP社のミニチュアカー RoboCar 1/10 for Automotive Platform(以下，RoboCarとする)の仕様を表1に示した．搭載しているOSはAUTOSARに準拠している．AUTOSARとは，実際の自動車も準拠している標準規格である[6]．ロータリーエンコーダでは自車の速度を取得する事ができる．赤外線測距センサにより先行車両との距離を検知する事ができる．これらの情報を基にサーボモータを制御しACCを実現している.

表1　実験機器で使用したRoboCarの仕様

構成	仕様
CPU	ルネサスエレクトロニクス社 V850E2
搭載OS	TOPPERS/ATK2 (AUTOSAR準拠)
内界センサ	ジャイロ1軸 加速度3軸 ロータリーエンコーダ (車輪*4 , 駆動モータ軸*1)
外界センサ	赤外線測距センサ *8
サーボモータ	ZMP製

4.3　実験システム

本研究の実験システムは故障が起きる対象であるRoboCarとRoboCarからBluetooth通信を利用して送信されるセンサデータをモニタリングするコンピュータから構成されている．　図2は実験システムの概要図である．TOPPERS/ATK2のアプリケーションとして3つのタスク(STAMPタスク，制御タスク，通信タスク)を用意し，これらを周期的に実行することでシステムを実現した．制御タスクはRoboCarに搭載されているセンサのデータ収集を行う．STAMPタスクは制御タスクが収集したセンサデータを基に障害監視を行う．通信タスクは収集したデータをコンピュータに送信する．実験では，その手段として無線通信であるBluetooth通信を利用した.コンピュータ上でRoboCarの情報をモニタリングするためBluetooth通信を利用しており，毎秒115，200bpsでデータ送信をする．　RoboCarのUARTに対応した変換機を使用してBluetooth通信を実装した.

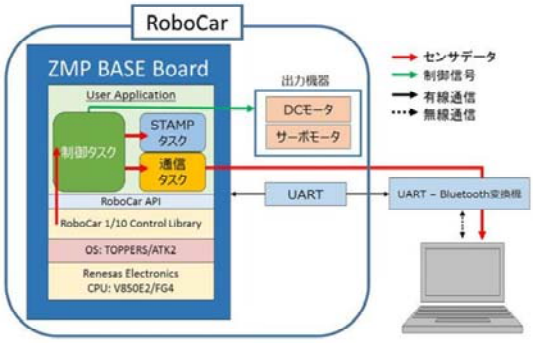

図2　実験システムの概要図

4.4　ACC動作確認

図3は,ACCが実装できているかを確認したものである.自車を固定し車間距離を変化させた実験で､先行車未検知､追いつき､追従､停止までの制御を確認できたことからACCが実装できたものとする.

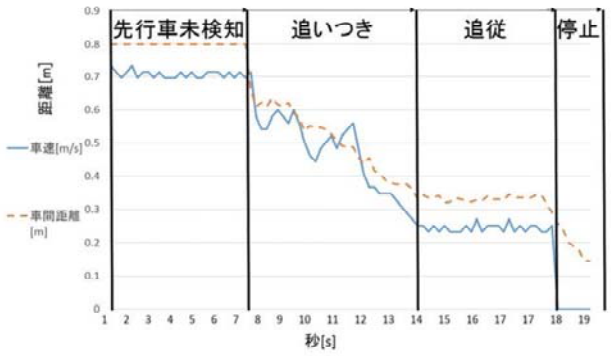

図3　RoboCarのACC実装評価結果

4.5　提案手法適用

ACCに関係するサブシステム間の制御アクションを識別するために図4のCS図を展開する．そして図中の制御アクションを3章の(2)で述べた4つのガイドワードを適用し

Vol.2015-MBL-77 No.5
Vol.2015-ITS-63 No.5
2015/12/2

てハザードを識別する．それらを検出するための制約条件も同時に識別する．今回は，赤外線測距センサと速度制御サブシステム間の距離情報を送信する制御アクションでのハザード，制約条件を表 2 として識別した．

今回は表 2 のセンサ値の時間的遷移の制約条件を監視する．

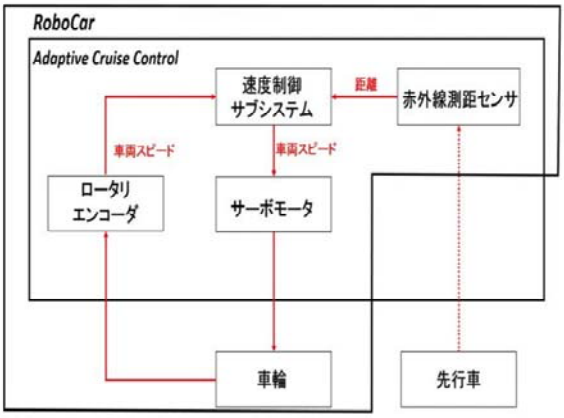

図 4　ACC の CS

表 2　ACC の CS に基づいたハザードと制約条件

ガイドワード / ハザード・制約条件	必要な制御アクションが供給されない	不適切なコントロールアクションが供給される	意図しないタイミングで供給される	アクションが途中で止まる
ハザード	センサから、速度制御サブシステムに距離情報が送られない	センサから誤った距離情報が速度制御サブシステムに送信される		一定周期で距離情報を送信していたが、送信が中断された
		前方車両がいないが距離情報が送信される		
制約条件	・センサのフラグ状態 ・速度制御サブシステムのフラグ状態	・センサの時間的遷移 ・センサの空間的遷移		・センサの送信周期

4.6 実験手順

実験ではセンサから誤った距離情報が速度制御サブシステムに送信されるというハザードを検出する．そのためにハザードを引き起こす故障を定義し注入する．発生したハザードがリアルタイムに監視出来ているかの検証を行う．実験は以下の手順である．

(1)ハザードを引き起こす故障を定義

RoboCar の最大スピードは 2.8m/sec であり，赤外線測距センサの送信周期は 100msec である．このことから 100msec の間に 0.28m 以上離れる事はない．そのため先行車が 100msec 間に 0.28m 以上移動した場合にハザードを引き起こす故障と想定する．

(2)ハザードを引き起こす故障の注入

故障注入には図 5 から図 6 のように実験開始から 20 秒後に自車の相対速度 4m/sec で先行車を移動させる．この操作によってハザードが発生し，制約条件に沿った監視をすることでハザードを検出する．

(3)制約条件の監視

センサ値の時間的遷移を監視し，ハザードの検出がセンサ値の送信周期(100msec)内であるかどうか確認した．

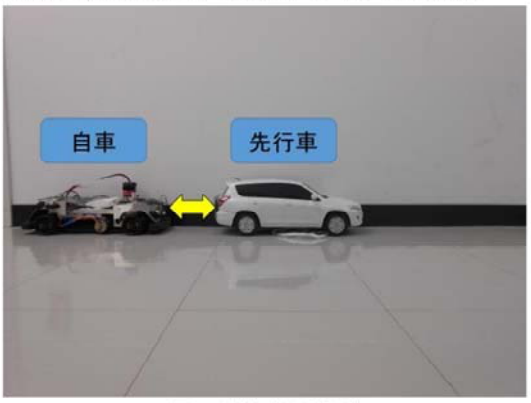

図 5 先行者遷移前

図 6 先行車遷移後

4.7 実験結果

実験では，センサ値の時間的遷移の制約条件を監視した．表 3 はセンサ値の時間的遷移である．先行車を 0.0 秒~19.9 秒まで車間距離を約 0.20m に固定し，20.0 秒の時に車間距離を約 0.60m まで移動させてハザードを引き起こした．このハザード監視に要する時間は検出前には 5~6 μ sec だったが，検出後には 14 μ sec まで上昇した．しかし，センサの送信周期(100msec)内でハザード監視ができたため，オーバーヘッドが小さいことがわかった．

表 3　0.1 秒毎のセンサ値の時間的遷移

時刻 [sec]	車間距離 [m]	監視に要する時間 [μ sec]	ハザード検出有無
0.0	0.203696	5.0	無
0.1	0.207810	6.0	無
0.2	0.201633	5.0	無
⋮			
19.9	0.213988	5.0	無
20.0	0.572580	14.0	有
20.1	0.593631	5.0	無
⋮			

Vol.2015-MBL-77 No.5
Vol.2015-ITS-63 No.5
2015/12/2

4.8 センサネットワークによる整合性監視

今後の課題として，次の実験も検討している．自車の前方のセンサだけでなく他方向のセンサも使用し，整合性を監視することでハザードを検出する．図 7 は割り込み時における先行車の相対的な遷移とその際に取得するセンサ値の規則性の一例である．正しく遷移した場合，センサは①，②，③と逐次的に値を取得する．遷移途中に値を取得できていないセンサ②がある場合，そのセンサ②は規則性から外れているためハザードとする．図 8 は想定されるセンサ値の遷移である．難しい点としてセンサ③が故障した場合，割り込みか追い越しかの判断が出来ない．

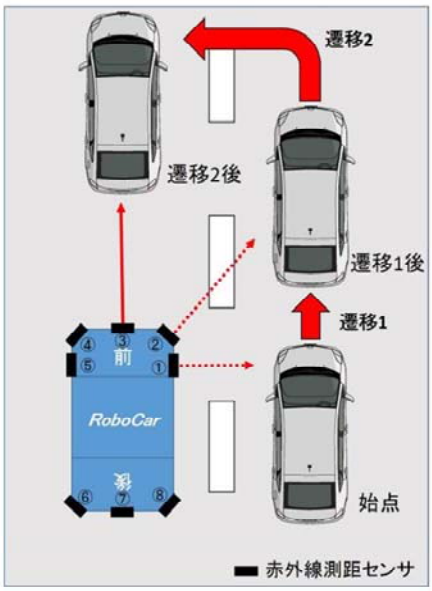

図 7 割り込み時による先行車の相対的な遷移

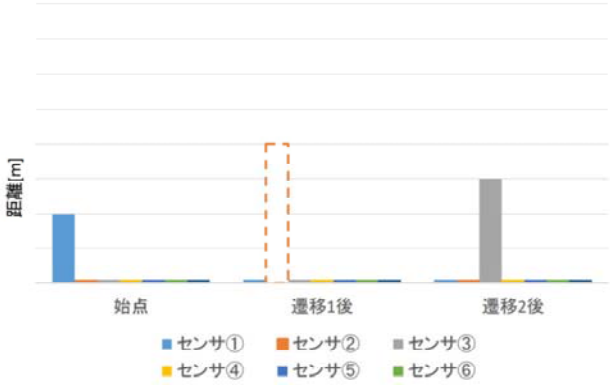

図 8　割り込み時による自車のセンサ値

5. おわりに

本研究では，リアルタイム制御システムにおける障害監視のための STAMP/STPA を適用する手法を提案した．

実験では，リアルタイム制御システムとして ACC を搭載した RoboCar を使用し，STAMP/STPA を適用した．このシステムの赤外線測距センサに故障注入し，ハザード監視を行った．

実験結果からセンサ値の時間的遷移のハザードがリアルタイムに監視できたことがわかった．

今後の課題として，ACC の制約条件をより多く網羅すること．センサ間の整合性によるハザードの検出等，検討している．

参考文献

1) JIS-Z8115 ディペンダビリティ（信頼性）用語：日本工業標準調査会(2000).
2) 北川裕貴，辻田和宏，山下昭裕，伊藤信行，小林幸彦，水野忠則，中條直也：リアルタイム制御システムにおける故障診断のためのログデータ収集，WiNF2014(2014).
3) 八山幸司：米国における STAMP(システム理論に基づく事故モデル)研究に関する取り組みの現状(前編)，ニューヨークだより，pp.3(2014).
4) 八山幸司：米国における STAMP(システム理論に基づく事故モデル)研究に関する取り組みの現状(前編)，ニューヨークだより，pp.6(2014).
5) 白川洋充，竹垣盛一，リアルタイムシステムとその応用，朝倉書店 (2001).
6) 小口泰平，ボッシュ自動車ハンドブック，日経 BP 社 (2011).