CANのイベント送信付き周期メッセージの検出と攻撃検知への応用

矢嶋 純[1,2,a)]　森川 郁也[1]　長谷部 高行[1]　大久保 隆夫[2]

概要：近年の自動車は，走る，曲がる，止まるなどの動作制御が車載ネットワークで行われている．一方，インターネットなどの外部ネットワークに接続するコネクテッドカーが増加しており，外部ネットワーク経由での遠隔操作攻撃の危険が指摘されている．遠隔操作攻撃では遠隔操作メッセージを車載ネットワークに注入することで，攻撃者が自動車を自在に操る．遠隔操作攻撃への対策として，車載ネットワーク上で攻撃を検知し，ユーザに通知して安全に停車させたり，通信を遮断するなどの方法が考えられる．攻撃の検知手段の一つとして，CAN メッセージの内容に着目した検知が考えられる．この手法では平常時のデータの振る舞いを事前に抽出しておき，抽出した振る舞いと異なる内容のデータを受信した際に攻撃であると検知する．本研究では，従来あまり着目されていなかったイベント送信付き周期メッセージに対する攻撃を検知することを目指し，大量の CAN ログのデータ内容などからイベント送信付き周期メッセージを抽出する方法，及び，抽出後の攻撃検知手法について考察する．

キーワード：CAN，攻撃検知，特徴抽出，イベント送信付き周期送信メッセージ

Extraction Method of Event Based Periodic Messages for CAN Anomaly Detection

Jun Yajima[1,2,a)]　Ikuya Morikawa[1]　Takayuki Hasebe[1]　Takao Okubo[2]

Abstract: Recently, the greater part of cars are electronically controlled by the in-vehicle networks like CAN. In addition, the number of connected cars that connected with external networks (for example, the Internet etc.) increases. Therefore, it is pointed out that there are remote-controlled attacks from physically outside of a car. In this attack, they control a car by injecting remote control messages into the in-vehicle networks. As one countermeasure, there is attack detection on in-vehicle network. If an attack can be detected, many emergency activity can be applied. As one of the attack detection method, there are detection methods focusing on CAN payload. In the pre-extraction phase of these methods, the detector extracts behaviors of data in CAN payload at normal situation. In the detection phase, when the detector detects a different behavior from normal situation, the situation is detected as an attack. In this paper, we show an attack detection method against event-based periodic messages that were not paid attention so much so far. We also show a method for pre-extraction phase. This method extracts event-based periodic messages from all messages, and event-based part of it.

Keywords: CAN, Anomaly Detection, Feature Extraction, Event-Based Periodic Messages

1. はじめに

近年の自動車は，走る，曲がる，止まるといった自動車の制御から，パワーウィンドウの開閉に至るまで，多くの箇所が車載ネットワークで電子制御されている．一方，イ

[1] 富士通研究所 セキュリティ研究所
Security Laboratory, FUJITSU Laboratories Ltd.
[2] 情報セキュリティ大学院大学
Institute of Information Security
[a)] jyajima@fujitsu.com

ンターネットなどの外部ネットワークに接続するコネクテッドカーが増加しており，コネクテッドであることは自動運転でも必須の要件となっている．外部ネットワーク接続機器に脆弱性があった場合，脆弱性を悪用して車載ネットワークに侵入し，自動車を遠隔操作する攻撃が懸念されている．実際にこのような攻撃は示されており [1]，140 万台のリコールになる問題へと発展している．このようなセキュリティ問題に対処するには，脆弱性を作りこまないことが重要であるが，パソコンやスマートフォンの例を見ても分かる通り，脆弱性が全くない製品を開発することは極めて困難であり，脆弱性が入ってしまうことを念頭においたセキュリティ運用が必要と考えられる．このセキュリティ運用の一つとして，車載ネットワークでの攻撃検知があげられる．車載ネットワークでの攻撃検知は，データの受信時刻が平常時から逸脱したことを利用して検知する手法や，送受信されるデータの内容が平常時から逸脱したことを利用して検知する手法が知られている．前者については，周期的にメッセージ送信が行われるネットワーク，後者については周期的では無いメッセージ送信が行われるネットワークで利用可能である．本論文では主に後者について，従来はあまり検知で利用されてこなかった，イベント発生に関連して送信タイミングが変わるメッセージ，とりわけ我々がイベント送信付き周期送信メッセージと呼ぶメッセージを検知対象に含めることができるような検知手法の紹介を行う．この方式では，検知を行う前に，大量の車載ネットワークログを元に，どのような種類のデータが，車載ネットワークメッセージ内のどのビット位置にどれ位の長さのデータとして格納されているのかを知っておく必要がある．このデータの種類，ビット位置，データ長の組み合わせのことを本論文では振る舞いと呼ぶことにする．本論文では，イベント送信付き周期送信メッセージの検出，及び，メッセージ内のどこにイベント依存部があるかを検出する方法を提案する．また，従来から知られているデータの種類も含めて，大量ログを元に，振る舞いを推定する手法を提案する．完成した推定手法を実装し，いくつかの種類のメッセージに対して推定を行ったところ，推定成功確率は約 69.0% であった．我々は，この実験において提案手法の二つの問題点を発見した．一つ目はサポートするメッセージ種類が不足している問題，二つ目は実験に用いるログの長さの不足の問題である．一つ目についてはサポートするメッセージの種類数を増やせば解決できると考えられ，仮にサポートするメッセージの種類を増やした場合，推定成功確率は約 92.5% になると考えられる．さらに，二つ目について長いログを準備した場合，推定成功確率は 100% になると考えられる．本論文では，これらの結果について考察するとともに，今後の展開についても説明する．

本論文の構成は以下のとおりである．§2 では，車載ネットワークで用いられる CAN について説明する．§3 では，CAN でよく用いられる攻撃検知手法を紹介する．そして §4 では攻撃検知を実行するために必要な事前調査で使用するデータ振る舞い抽出技術を説明する．§5 にてイベント送信付き周期送信メッセージを含む複数の送信形式に対応した攻撃検知方式を紹介し，本論文の提案である検知前の事前調査で利用するデータの振る舞い抽出手法を提案する．§6 では，提案手法の効果を評価するための実験を行い，§7 にてまとめを行う．

1.1 本論文の貢献

本論文は，我々がイベント送信付き周期送信メッセージと呼ぶメッセージを検知対象に含めることができるようにするための事前準備手法を提案し，具体的な検知手法を紹介する．提案手法を用いることで，従来は異なるメッセージ種類と判定されていたイベント送信付き周期送信メッセージを検知対象にすることができるため，このメッセージについて正確に攻撃検知を行うことができるようになる．これにより，攻撃検知全体としての検知精度を従来よりも向上することができると考えられる．従来のイベント送信付き周期送信メッセージを用いた攻撃検知手法では，複数のイベント送信付き周期送信メッセージ間の関連性を用いた検知が提案されている [2] が，本論文では単独のイベント送信付き周期送信メッセージを検出することができる．従来は単独のイベント送信付き周期送信メッセージの発見手法は提案されていないため，従来手法と組みわ合わせることで攻撃検知精度の向上が期待できる．

2. CAN

2.1 概要

CAN (Controller Area Network) は，独 BOSCH 社によって開発され，ISO 標準化 [3] されたネットワーク仕様であり，多くの自動車の車載ネットワークで利用されている．CAN では，2 本の通信線の電圧の差分値で通信上のデータ (0, 1) を表現する．このため，2 本の線に同時にノイズが乗ったとしても，差分値としては変化が少なく，ノイズの影響を受けにくい．CAN に対して送受信を行うネットワークノードは ECU (Electronic Control Unit) と呼ばれる．CAN はブロードキャスト型の通信仕様であり，一つの ECU が送信した CAN メッセージは，同一のバスに繋がる全ての ECU に到達する．CAN ではデータを送受信する際に，データフレームというフォーマットのデータを送信する．データフレームにはメッセージの種類を示す ID と，データの内容を示すデータ部が含まれている．CAN で通信される各ビットはドミナント (0) とリセッシブ (1) があり，複数の ECU が同時に CAN メッセージをネットワークに送信すると，ドミナントが優先される．これを CSMA/CA (Carrier Sense Multiple Access with Collision

Avoidance) 方式という．各 ECU に搭載された CAN コントローラでは常にネットワーク側を監視しており，CAN メッセージを送信する際には，自身が送信した CAN メッセージが確かにネットワークに流れたかを確認している．つまり，リセッシブを送信しようとした際に，別の ECU がドミナントを送信した場合にはドミナントで上書きされるため，自身の送信が失敗したことに気付く．このときは送信を一旦あきらめる．あるいは，ECU は常に CAN のメッセージを監視しているため，自身が送信しようとした際に別の ECU が CAN メッセージを送信中であった場合にも送信を一旦あきらめる．そして，別の ECU による CAN メッセージの送信が終わってから，再度送信を試みる．メッセージの受信の際には，車載ネットワーク側からブロードキャストされて到達した CAN メッセージが自身の ECU で必要とする CAN メッセージであった場合のみ各 ECU に搭載された CAN コントローラにて受信を行い，それ以外は無視して取り込まない仕様となっていることが多い．本論文ではデータフレームによるデータの送受信についての攻撃検知を行うため，データフレームについて詳細に説明する．

2.2 データフレーム

データフレームはデータの送受信に用いられ，標準フォーマットと拡張フォーマットが存在する．データフレームには ID 部とデータ部があり，ID 部については，標準フォーマットでは 11 ビット，拡張フォーマットでは 29 ビットの ID が格納される．データ部については最大 64 ビットのデータを格納できる．データ部に格納される各データは，ID の値毎に格納のされ方が予め決められているような運用になっている．例えばある ID(A) については，先頭 8 ビットが固定値，次の 8 ビットが連続的に少しずつ変化する値，次に 8 ビットがチェックサムなどという風に格納され，この格納のされ方は ID(A) のメッセージ全てで共通となっているが，ID(B) のメッセージでの格納のされ方とは同じとは限らないというものである．本論文では標準フォーマットのみを扱うが，本論文での議論はどちらのフォーマットにも適用可能である．

2.3 メッセージの送信形式

CAN メッセージにはいくつかの送信形式が存在する．送信形式の例を以下に示す．メッセージの送信形式はメッセージの種類，つまり，データフレームの ID フィールドの値毎に決められている．どのメッセージがどの送信形式なのかはメーカー毎，及び，車種毎に異なっている．

- 周期送信メッセージ
  メッセージが周期的に送信される．周期が乱れることはあるが，1 メッセージ/1 周期という関係が保たれている．
- イベント送信付き周期送信メッセージ
  平常時には周期的に送信されるが，イベントが発生した際にはそのタイミングでメッセージが送信され，以降は周期送信に戻る．このメッセージの例を表 1 に示す．

表 1　イベント送信付き周期メッセージの例
Table 1　An example of event-based periodic messages

timestamp	value
177.821529	29 00 00 00 00 00 28 56
187.822498	29 00 00 00 00 00 28 56
197.821295	29 00 00 00 00 00 28 56
200.641073	29 80 00 00 00 00 28 57
210.641196	29 00 00 00 00 00 28 57
214.960909	29 80 00 00 00 00 28 58
224.960940	29 00 00 00 00 00 28 58

- 非周期送信メッセージ
  周期性無くメッセージが送信される．

3. CAN に対する代表的な検知手法

3.1 メッセージの周期性に着目した検知

周期送信メッセージに対する攻撃を検知するためによく用いられる手法である．メッセージの周期性に着目し，メッセージの受信間隔が平常時と異なることを検知した際に攻撃であると判断するのが基本戦略である．この手法の代表例として，周期検知 [4]，シフト判定式周期検知 [5]，累積和検知 [6] などがある．

3.2 メッセージのデータ内容に着目した検知

周期送信メッセージだけでなく，メッセージの周期性を利用するのが困難な，イベント送信付き周期メッセージと非周期送信メッセージにも利用可能な手法である．平常時におけるメッセージ内の各データの振る舞いを事前調査しておき，検知フェーズにて平常時とは異なる振る舞いをするデータを検出したら，攻撃として検知する検知戦略である．この手法の代表例として，メッセージの受信時の状態遷移に着目した検知 [7] がある．

3.3 イベント送信付き周期送信メッセージに対する攻撃検知

3.3.1 濱田らの手法

濱田らはイベントメッセージに対する攻撃検知手法を提案している [8]．この手法では，メッセージのデータ内容，及び周期性について，相関関係を利用して検知を行う．我々の手法もデータ内容と周期性を利用するが，彼らのものとはイベントメッセージの定義が異なっており，それぞれの手法のカバー範囲は異なると考えられる．

3.3.2 矢嶋らの手法

矢嶋らもイベント送信付き周期送信メッセージに対する攻撃検知手法を提案している [2]．彼らの手法は，複数のデータ間の関連性とメッセージの受信間隔の関連性を元に検知を行う．今回提案する手法では，単一のデータにおけるデータの変化，及び，周期性を利用して検知を行うため，この手法ともカバー範囲が異なっている．

4. CAN のフィールド分類

4.1 概要

本論文では，周期送信メッセージだけでなく，イベント送信付き周期メッセージも検知対象とする検知手法を紹介する．イベント送信付き周期送信メッセージは，イベント発生時には周期性が保証されないため，メッセージの周期性のみに着目した手法をそのまま利用することはできない．本論文では，周期性とデータ内容の両方に着目した手法を紹介する．§3.2 で説明したデータ内容に着目した検知を行うためには，データ内容の平常時の振る舞いを事前に検知者が知っておく必要がある．この振る舞いを調査する方法として，長い CAN のログを準備して，振る舞いを調べる方法がいくつか提案されている [9], [10]．これらの手法ではデータフレーム内のデータの分割パターンを全パターン考慮し，各パターンでの振る舞いのもっともらしさから，分割パターンと分割した後の各データの振る舞いを予測する．以下にこれらの手法をもう少し詳しく説明する．

4.2 Markovitz らの手法

Markovitz らの手法 [9] では，固定値，カウンタ/センサ値，マルチ値の 3 つの種類のデータを考える．CAN のデータフィールド（最大 64 ビット）は，多数のデータの組み合わせで構成されていることが一般的であり，各データそれぞれがどの種類になっているかを ID 毎に予測する．

Markovitz らの手法では，まず全てのデータ分割のパターンを考える．例えば先頭から，(1 ビット-2 ビット-1 ビット-4 ビット-16 ビット-32 ビット-8 ビット) というような格納パターンがありうる．このような分割のパターンは，先頭からのビット位置とデータ長で表現することができる．この例では，(0,1), (1,2), (3,1), (4,4), (8,16), (24, 32), (56, 8) となる．分割のパターン数は全部で 64 箇所 $\times 1$ ビット $+63$ 箇所 $\times 2$ ビット $+ \ldots + 1$ 箇所 $\times 64$ ビット $= 2080$ パターンとなる．この 2080 パターンの組み合わせで一つの振る舞いが表現できるが，同じビット位置のものは排他で考えてよい．そこで 2080 パターン全部について独立でどのデータの種類が適切かを予測する．固定値はログに現れたデータのユニーク数が 1 であること，マルチ値はユニーク数が小さくてそれほど短くはないデータであること，カウンタ/センサ値はその他のデータのときというのが条件となっている．この手法では，種類を決めると同時に，評価値と呼ばれる値も決定される．各分割パターンでの予測が終了したら，先頭から順に，評価値スコアの高いものを分割，及び，データのパターンとして確定させていく．

4.3 岸川らの手法

岸川らの手法 [10] では，データの種類を 3 種類から 5 種類に増やし，また，連続値（Markovitz らの手法におけるセンサ値）であることの判断に出現回数だけでなく，データの変化の仕方を用いるのが特徴である．この変化の仕方はデータの時系列変化に関するものであり，具体的には変化量の分散である．変化量の分散が小さければ，データ受信のたびにデータが変化する度合いは小さいと判断できる．確定の際の優先順位は，固定値，カウンタ，センサ，チェックサム，マルチ値とする．Markovitz らの手法よりも種類数が多いため，より多くのデータの振る舞いに対応した検知が可能となり，全体として攻撃検知精度が向上する．

5. 提案方式

5.1 概要

本論文では，岸川らの手法の 5 種類に加えて，イベント付き周期送信メッセージのデータの振る舞いを追加する．本節では攻撃検知手法を紹介し，提案手法で必要となるデータの振る舞いの事前抽出手法について提案する．

5.2 準備 –イベント送信付き周期送信メッセージに対する攻撃検知–

5.2.1 イベント送信付き周期送信メッセージ

表 1 に実車から取得したイベント送信付き周期メッセージの例を示す．イベント送信付き周期メッセージでは，基本的に周期的にメッセージ送信が行われるが，イベント発生時には追加でメッセージが送信される．この追加メッセージでは，データの値が平常時から変化して特定値になったり，変化したりしている (下線部)．平常時には値は固定値である．攻撃検知ではこの特性を利用できると考えられる．

5.2.2 攻撃検知手法

本論文で紹介する攻撃検知手法は，データの振る舞いと，メッセージ送信の周期性の両方を利用する．具体的には，データが固定値であるとき，つまり平常時の送信であったときに，データ送信の周期性が崩れていたら攻撃として検知する．また，データの振る舞いが平常時から逸脱したときに，逸脱の仕方が通常とは異なっていたら攻撃として検知する．アルゴリズムを §5.2.2.1 に示す．

5.2.2.1 攻撃検知アルゴリズム

具体的な攻撃検知アルゴリズムは以下のとおりである．

(1) メッセージを受信する．監視対象の ID でなければ破棄して (1) へ．監視対象の ID であった場合，ID の値毎に (2) 以降を実行する．

(2) イベント発生時に変化する箇所について，変化しているかを確認する．変化していなかったときは (3) へ．変化していたら (4) へ．
(3) 一つ前の同一 ID の受信時刻と今回の受信時刻の時間間隔がほぼ 1 周期と同じであれば正常として (1) へ．異なっていれば攻撃として検知して (5) へ．
(4) 通常のデータ変化と同じかどうかを確認する．具体的には，イベント発生時に特定値になるような箇所では特定値なっているかどうかを確認する．特定値となっていたら正常と判断して (1) へ．異なっていれば攻撃として検知して (5) へ．
(5) ユーザに警告したり，外部ネットワークを遮断するなどの緊急措置を取って自動車を安全にしたのち (1) へ戻る．

5.3 提案手法 –イベント送信付き周期メッセージの検出–

§5.2.2.1 のアルゴリズムを実行するにあたり，イベント発生時に変化している箇所，および、変化の仕方を事前に知っておく必要がある．本節ではこの振る舞いの事前調査を，他のデータ種類の検出も同時に行いながら行う手法を提案する．

5.3.1 イベント送信付き周期メッセージの検出

ここでは，他のデータの種類も含めた全体アルゴリズムの設計については一旦保留し，イベント送信付き周期メッセージの検出，及び，イベント発生時の変化箇所を検出する方法について説明する．

§5.2.1 で説明した通り，イベント送信付き周期送信メッセージでは基本的に周期的にメッセージ送信が行われるが，イベント発生時には追加でメッセージが送信され，この追加メッセージでは，データの値が平常時から変化して特定値になったり，変化したりしている．そこでこのメッセージの検出方法としては以下のようになる．ここで，以下のアルゴリズムを実行するときには平常時の周期の情報は知っているものとする．以下は，イベント発生時にデータが特定値になるようなものの検出と，イベント発生時にデータが変化するようなものの検出を同時に行うアルゴリズムである．

5.3.1.1 イベント送信付き周期送信メッセージの検出アルゴリズム

以下のアルゴリズムで検出できる．本アルゴリズムでは，「イベント送信付き周期送信メッセージである可能性の判定」と「イベント送信付き周期メッセージの判定」の処理を呼び出す．このアルゴリズムは，可能性のあるすべてのビット位置とデータ長について，初期状態としてはイベント送信付き周期送信メッセージである可能性が高い (フラグ=1) と設定しておく．そして「イベント送信付き周期送信メッセージである可能性の判定」において，CAN ログを分析しながら，各ビット位置，データ長について，イベント発生していないときの周期が 1 周期に近いかどうか，及び，イベントが発生したときに周期が全く崩れなかったかを調査する．最後に，調査結果を元に，そのビット位置，データ長について，イベント送信付き周期送信メッセージであるかどうかを判定する．

- イベント送信付き周期送信メッセージの検出アルゴリズム

(1) 大量の事前調査用 CAN ログを準備する．
(2) ID 毎に以下を実行する．ID 毎に Markovitz の手法と同様の 2080 箇所 (全ての位置と長さの組み合わせパターン) 全てについて，「イベント送信付き周期送信可能性フラグ（特定値）」=1,「イベント送信付き周期送信可能性フラグ（変化）」=1 とする．
(3) CAN ログからデータを 1 個リードする．調査対象の ID のメッセージがリードされるまで繰り返す．
(4) 「可能性の低いデータ種類の判定」を実行する．
(5) 全てのデータのリードが終わっていなければ (3) へ．終わっていたら以下へ．
(6) 「イベント送信付き周期メッセージの判定」を実行する．

- イベント送信付き周期送信メッセージである可能性の判定

(1) For $i = 1$ to 64 do (データの開始位置のループ)
  (a) For $j = 1$ to $65 - i$ do (データ長のループ)
    (i) リードしたメッセージが，その ID についての 1 個目であった場合，$p_1(i,j) = data_1(i,j)$, $NUnique(i,j) = 1, pp_1(i,j) = data_1(i,j)$ として保存する．
    (ii) リードしたメッセージが，その ID についての 2 個目以降 (k 個目) であった場合，以下を実行する．
      (A) $data_k(i,j) = p_1(i,j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ「イベント送信付き周期送信可能性フラグ（特定値）(i,j)」=0 とする．
      (B) $data_k(i,j) \neq p_1(i,j)$ であったときには，$p_2(i,j)$ が定義されていないときには $p_2(i,j) = data_k(i,j), NUnique(i,j) = NUnique(i,j) + 1$ とする．定義されていたときには，$data_k(i,j) \neq p_2(i,j)$ なら「イベント送信付き周期送信可能性フラグ（特定値）(i,j)」=0，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ周期性無しフラグ'(特定値)=1 とする．
      (C) $data_k(i,j) = pp_1(i,j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ「イベント送信付き周期送信可能性フラグ（変化）(i,j)」=0 とする．
      (D) $data_k(i,j) \neq pp_1(i,j)$ であったときには，$pp_1(i,j) = data_k(i,j)$，1 個前の受信との時間間

隔 $t_k - t_{k-1}$ が 1 周期に近くなければ周期性無しフラグ (変化)=1 とする．

- イベント送信付き周期メッセージの判定

(1) For $i = 1$ to 64 do (データの開始位置のループ)
  (a) For $j = 1$ to $65 - i$ do (データ長のループ)
    (i) イベント送信付き周期送信可能性フラグ（特定値）(i, j)=1 かつ，周期性無しフラグ (特定値)=1 なら，その ID のメッセージをイベント送信付き周期送信メッセージ（特定値）と確定する．
    (ii) イベント送信付き周期送信可能性フラグ（変化）(i, j)=1 かつ，周期性無しフラグ (変化)=1 なら，その ID のメッセージをイベント送信付き周期送信メッセージ（変化）と確定する．

5.3.2 全体アルゴリズム

ここでは，§5.3.1.1 のアルゴリズムをベースとして改造し，他の種類のデータの検出も含めた全体アルゴリズムを提案する．我々の手法では，全部で 7 種類のデータを検出する．これにより従来よりも扱えるデータの種類が増加するため，全体の検知精度が向上することが期待できる．まず全てのビット位置，データ長について，全ての種類の可能性を「高い」（フラグ=1）としておく．そして，「可能性の低いデータ種類の判定」にて，各種類のデータの振る舞いとして妥当であるかを判断し，可能性が低いと判断される種類についてはフラグを 0 にする．そして最後に，固定値，イベント送信付き周期メッセージ（特定値），イベント送信付き周期メッセージ（変化），カウンタ，センサ，チェックサム，マルチ値の優先順位でデータの種類を確定する．

5.3.2.1 メッセージ内のデータ振る舞い推定アルゴリズム

- メッセージ内のデータ振る舞い推定アルゴリズム

(1) 大量の事前調査用 CAN ログを準備する．
(2) ID 毎に以下を実行する．ID 毎に Markovitz の手法と同様の 2080 箇所 (全ての位置と長さの組み合わせパターン) 全てについて，「固定値可能性フラグ」=1，「イベント送信付き周期送信可能性フラグ（特定値）」=1，「イベント送信付き周期送信可能性フラグ（変化）」=1，「カウンタ可能性フラグ」=1，「センサ可能性フラグ」=1，「チェックサム可能性フラグ」=1，「マルチ値可能性フラグ」=1 とする．
(3) CAN ログからデータを 1 個リードする．調査対象の ID のメッセージがリードされるまで繰り返す．
(4) 「可能性の低いデータ種類の判定」を実行する．
(5) 全てのデータのリードが終わっていなければ (3) へ．終わっていたら以下へ．
(6) 「リード終了後の判定」を実行する．
(7) 「振る舞いの確定処理」を実行する．

- 可能性の低いデータ種類の判定

(1) For $i = 1$ to 64 do (データの開始位置のループ)
  (a) For $j = 1$ to $65 - i$ do (データ長のループ)
    (i) リードしたメッセージが，その ID についての 1 個目であった場合，$p_1(i, j) = data_1(i, j)$, $NUnique(i, j) = 1, pp_1(i, j) = data_1(i, j)$ として保存する．
    (ii) リードしたメッセージが，その ID についての 2 個目以降 (k 個目) であった場合，$diff(k, k-1), (i, j) = data_k(i, j) - data_{k-1}(i, j)$ を計算し，以下を実行する．
      (A) $data_k(i, j) = p_1(i, j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ「イベント送信付き周期送信可能性フラグ（特定値）(i, j)」=0 とする．
      (B) $data_k(i, j) \neq p_1(i, j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ周期性無しフラグ (特定値)=1 とし，$p_2(i, j)$ が定義されていないときには $p_2(i, j) = data_k(i, j), NUnique(i, j) = NUnique(i, j) + 1$ とする．定義されていたときには，$data_k(i, j) \neq p_2(i, j)$ なら「イベント送信付き周期送信可能性フラグ（特定値）(i, j)」=0 とする．
      (C) $data_k(i, j) = pp_1(i, j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ「イベント送信付き周期送信可能性フラグ（変化）(i, j)」=0 とする．
      (D) $data_k(i, j) \neq pp_1(i, j)$ であったときには，1 個前の受信との時間間隔 $t_k - t_{k-1}$ が 1 周期に近くなければ周期性無しフラグ (変化)=1 とし，$pp_1(i, j) = data_k(i, j)$ とする．
      (E) $NUnique(i, j) > 1$ ならば，「固定値可能性フラグ (i, j)」=0 とする．
      (F) $diff(k, k-1) = 0$，または，$diff(k, k-1) \neq diff(2, 1), (i, j)$ のとき，「カウンタ可能性フラグ (i, j)」=0 とする．
      (G) データの分散と一様性を計算するための処理を実行 (平均導出のためにデータ値の累積和を計算するなど)．

- リード終了後の判定

(1) For $i = 1$ to 64 do (データの開始位置のループ)
  (a) For $j = 1$ to $65 - i$ do (データ長のループ)
    (i) 分散値 (i, j) が一定以上になっていたら，「センサ可能性フラグ (i, j)」=0 とする．
    (ii) データ (i,j) が一様分布していなければ，「チェックサム可能性フラグ (i, j)」=0 とする．
    (iii) 可能性フラグのいずれかが 1 となっている (i,j) については，「マルチ値可能性フラグ (i, j)」=0 とする．

- 振る舞いの確定処理

(1) For $i = 1$ to 64 do (データの開始位置のループ)
  (a) For $j = 1$ to $65 - i$ do (データ長のループ)

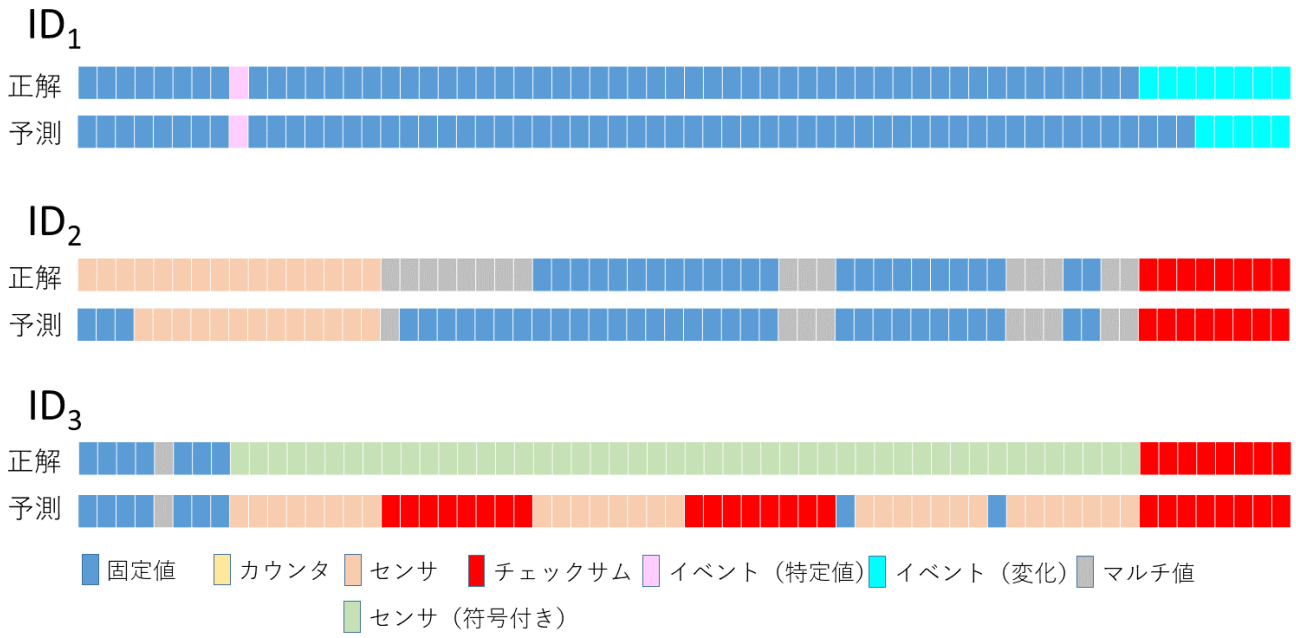

図 1 実験結果例
Fig. 1 Results of Experiment

(i)「固定値可能性フラグ (i,j)」=1 となっている箇所は全て「固定値」と確定する．

(2) For $i = 1$ to 64 do (データの開始位置のループ)

(a) For $65 - i$ to 1 do (データ長のループ)

(i)「イベント送信付き周期送信可能性フラグ（特定値）(i,j)」=1 かつ，周期性無しフラグ (特定値)=1 なら，その ID の箇所を「イベント送信付き周期送信（特定値)」と確定する．

(ii)「イベント送信付き周期送信可能性フラグ（変化）(i,j)」=1 かつ，周期性無しフラグ (変化)=1 なら，その ID の箇所を「イベント送信付き周期送信（変化)」と確定する．

(iii)「カウンタ可能性フラグ (i,j)」=1 なら，その ID のメッセージを「カウンタ」と確定する．

(iv)「センサ可能性フラグ (i,j)」=1 なら，その ID のメッセージを「センサ」と確定する．

(v)「チェックサム可能性フラグ (i,j)」=1 なら，その ID のメッセージを「チェックサム」と確定する．

(vi)「マルチ値可能性フラグ (i,j)」=1 なら，その ID のメッセージを「マルチ値」と確定する．

6. シミュレーション実験

6.1 シミュレーション実験と結果

提案手法の効果を確かめるため，シミュレーション実験を行った．今回実験を行ったのは提案手法である，CAN ログからデータの振る舞いを推定し，各データのビット位置と長さを推定する部分である．我々は準備として，実車の CAN データを取得し，手作業で分析して正解と思われるデータの振る舞いを推測した．そして，提案手法で推定を行い，手作業の結果と合致するかを検証した．今回用いた CAN のログは全体で 314 秒程度の長さであり，約 37 万個の CAN メッセージが含まれている．

検証は 8 種類の CAN ID に対して行った．今回調査を行った 8 種類についてのビット単位での推定成功確率は約 69.0% であった．ただし，この試行実験において 2 個の問題点が確認され，この問題点の内の一つを解決したと仮定して再見積もりをした実質の推定精度は約 92.5% であることを確認した．また，問題点を二つとも解決すると，推定精度は 100% になることを確認した．この点については次節で説明する．また，代表的な 3 例についての結果を図 1 に示す．

6.2 考察

試行実験の結果，多くの ID にて高精度に振る舞いを推定できていることが確認できたものの，いくつかの ID では正確に推定できていないことが確認された．問題点は以下の通りと考えられる．

- サポートしていないデータ種類の存在
  CAN ログを分析して正解データを作成している際に，提案方式でサポートしていないデータ種類の存在を確認した．例えば，図 1 の ID_3 では明らかに正解と異なるデータと推定された．このデータはログを良く眺めると符号付きのセンサ値であることが確認された．しかし，今回の提案手法においてはセンサデータは全て符号なしで処理を行っており，これが原因で異なる振る舞いとして推定されたと考えられる．他にもサポー

トしていないデータ種類をいくつか発見した．

- データの不足による本来変化するデータの不変箇所の存在
  センサ値などの場合，データが増加したり減少したりする．ログを分析すると，このようなデータは多くの場合においてバイト単位のデータであるように見える．しかし，実際に提案手法で推定すると，バイトデータの最上位に近い部分について不変で，固定値となっていることが確認された．これは本当に固定値であるか，あるいは本来は変化するビット位置であるが今回使用したログの量が不足していて変化が最上位まで到達していないかのどちらかであると考えられる．この問題についてはログを大量に取得することで精度を向上することができると考えられる．

上記二つの原因の内，一つ目の問題点についてはサポートするデータの種類を増やすことで解決可能であると考えられる．この問題点を解決したと仮定した場合，推定精度は約 92.5% となる．二つ目については，分析対象の CAN ログの長さを伸ばすことで対応可能と考えらえれる．仮に CAN ログの中に全ての振る舞いが入っていると仮定した場合，推定精度は 100% になると考えられる．

6.2.1 各 ID における考察

- ID_1 について
  最下位バイトにおいて，3 ビットが正解と推定結果が異なっている．これは上で説明したデータの不変箇所の問題であると考えられる．
- ID_2 について
  最上位バイトの不正解については，上で説明したデータの不変箇所の問題であると考えられる．3 バイト目の不正解については，ログにおいてはデータは 2 値をとっており，人間の目にはバイト単位のデータに見える．実際には 2 値の中に不変のビットが多数存在するため，正解と推定は多くのビットで異なった結果となっている．
- ID_3 について
  中央付近のデータの推定結果が正解とは大きく異なっている．これは上で説明した通り，ログ上では符号付きデータとなっているが，提案手法では全て符号なしで推定しているため，他の振る舞いと推定されてものと考えられる．

7. まとめと今後の課題

本論文では，イベント付き周期送信メッセージに着目した新しい振る舞い推定方式を提案した．推定方式の試行実験においては，本提案方式の問題点を 2 点発見した．そのうち一方が解決できたと仮定すると，推定精度は約 92.5% を達成した．また，両方が解決できたと仮定すると推定精度は 100% を達成できる見込みであることが判明した．本提案方式を使用した検知により，従来よりも検知精度の向上が期待できる．今後は考察の節で検討した課題点の内，前者であるサポートしていないデータの種類を可能な限り増やすことが考えられる．

参考文献

[1] Miller, C. and Valasek, C.: Remote Exploitation of an Unaltered Passenger Vehicle, *BLACKHAT2015* (2015).

[2] Yajima, J., Hasebe, T. and Okubo, T.: Data Relation Analysis Focusing on Plural Data Transition for Detecting Attacks on Vehicular Network, *The 22nd International Conference on Network-Based Information Systems (NBiS-2019)* (2019).

[3] ISO11898: Road vehicles – Controller area network (CAN) – (2003).

[4] Kishikawa, T., Matsushima, H., Haga, T., Maeda, M., Umigami, Y. and Ujiie, Y.: In-Vehicle Network System, Electronic Control Unit, and Irregularity Detection Method, *Publication Number WO/2015/170451, International Applications.* (2015).

[5] Otsuka, S., Ishigooka, T., Oishi, Y. and Sasazawa, K.: CAN Security: Cost-Effective Intrusion Detection for Real-Time Control Systems, *SAE Technical Paper 2014-01-0340* (2014).

[6] Yajima, J., Abe, Y. and Hasebe, T.: Proposal of Anomaly Detection Method "Cumulative Sum Detection" for In-Vehicle Networks, *Embedded Security in Cars (escar Asia 2018)* (2018).

[7] Tsurumi, J., Kishikawa, T., Sasaki, T., Takahashi, R., Haga, T. and Matsushima, H.: Proposal of Anomaly Detection Method for In-Vehicle Network based on Relation between Flag type Data, *2017 Symposium on Cryptography and Information Security* (2017).

[8] Hamada, Y., Yoshida, K., Adachi, N., Kamiguchi, S., Ueda, H., Miyashita, Y., Isoyama, Y. and Hata, Y.: Intrusion Detection for Acyclic Messages in In-Vehicle Network: A Proposal, *Computer Security Symposium 2018* (2018).

[9] Markovitz, M. and Wool, A.: Field classification, modeling and anomaly detection in unknown CAN bus networks, *escar Europe 2015* (2015).

[10] Kishikawa, T., Maeda, M., Tsurumi, J., Haga, T., Takahashi, R., Sasaki, T., Anzai, J. and Matsushima, H.: A Generic CAN Message Field Extraction Method to Construct Anomaly Detection Systems for In-Vehicle Networks, *2017 Symposium on Cryptography and Information Security (SCIS 2017)* (2017).