CAN向けグラフベース侵入検知システムの課題分析と特徴量拡張

佐々木 あかり[1,a)]　倉地 亮[1]　松原 豊[1]　高田 広章[1]

概要：近年，自動車システムのソフトウェア化・ネットワーク化の進展や新たな利用形態の登場に伴い，悪意ある攻撃機会や脅威が増加しており，自動車セキュリティ対策は急務となっている．対策の一つとして侵入検知システム（Intrusion Detection System, IDS）が注目されており，近年は特にグラフ理論を活用したアルゴリズムも研究されている．本研究では，主な車載ネットワークプロトコルである CAN（Controller Area Network）を対象とし，グラフ理論と決定木を組み合わせたグラフベース IDS に着目する．既存データセットを用いた実験により，従来手法の課題を明らかにする．さらに，CAN メッセージのデータ内容を考慮した新たな特徴量を提案し，その有効性を検証する．

キーワード：車載ネットワーク, CAN(Controller Area Network), IDS(Intrusion Detection System)

Challenges and Feature Augmentation in Graph-based Intrusion Detection Systems for CAN

Akari Sasaki[1,a)]　Ryo Kurachi[1]　Yutaka Matsubara[1]　Hiroaki Takada[1]

Abstract: In recent years, the increasing software-based and networked nature of in-vehicle systems, along with emerging usage patterns, has led to more opportunities for malicious attacks and heightened security threats, making automotive cybersecurity measures an urgent issue. Intrusion Detection Systems (IDSs) have gained increasing attention as a key countermeasure, and graph-theoretical approaches have become a growing focus of recent research. In this study, we focus on a graph-based IDS that combines graph theory and decision trees, targeting the Controller Area Network (CAN), a major in-vehicle network protocol. Through experiments on existing datasets, we analyze the limitations of conventional methods. Furthermore, we propose new features that incorporate the content of CAN message payloads and evaluate their effectiveness.

Keywords: in-vehicle network, CAN(Controller Area Network), IDS(Intrusion Detection System)

1. はじめに

近年，自動車システムのソフトウェア化 · ネットワーク化が進む中で，自動車セキュリティの重要性が高まっている [1]．車載ネットワークの事実上の標準プロトコルである CAN（Controller Area Network）[2] は，暗号化されていない，認証機構がないといったセキュリティ上の弱点が存在しており，その対策は急務となっている．実際に，2015 年には研究者による実証実験として，高速道路を走行中のジープ・チェロキーがクラックされ遠隔操作される事例が報告されている [3]．このような背景から，CAN 侵入検知システム（Intrusion Detection System, IDS）に関する研究が行われている．CAN IDS は，車載ネットワークにおける異常な通信を検知するためのシステムであり，様々な手法が提案されている．近年では，グラフ理論を用いた手法が注目されており，グラフ理論と決定木を組み合わせたグラフベース IDS も提案されている [4]．この手法では，

[1] 名古屋大学 大学院情報学研究科
Graduate School of Informatics, Nagoya University
[a)] a_sasaki@ertl.jp

© 2025 Information Processing Society of Japan

This paper is work in progress and not peer-reviewed.

CAN メッセージの遷移をグラフとして構造的に表現し，異常検知を行う．本研究は，既存のグラフベース IDS の一つに着目し，課題を明らかにした上で，新たな特徴量を提案する．本研究の前提は，以下の通りである．

(1) 侵入経路は問わず，CAN バスに攻撃が注入された後の検知を対象とする．
(2) CAN メッセージの設計情報 (どの CAN ID が何を意味するかなど) は利用しない．

また本研究の貢献は，以下の通りである．

(1) 公開データセットを用いた再現実験により，既存のグラフベース IDS ではマスカレード攻撃に対する検知性能が低いことを明らかにした．
(2) CAN メッセージのデータ内容を反映するウィンドウ単位の新規特徴量「逸脱割合」を提案した．具体的には，値の範囲・ハミング距離・マハラノビス距離に基づく 3 種の逸脱割合を定義した．
(3) 同データセットで提案手法の有効性を評価し，特にマハラノビス距離に基づく逸脱割合を特徴量に加えることで，マスカレード攻撃の検知性能においてベースラインのグラフベース IDS を上回ることを示した．

2. 背景知識

2.1 CAN

本論文では，車載ネットワークプロトコルの中でも特に CAN に着目するため，以下に CAN の説明を示す [5], [6]. CAN は 1983 年に Bosch 社が車載ネットワーク用に開発した通信プロトコルである．信頼性や確実性が求められる領域での活用が主であり，自動車の他にも，船舶や医療などの多方面で活用されている．CAN 通信はデータフレーム，リモートフレーム，エラーフレーム，オーバーロードフレーム，インターフレームスペースの 5 種類のフレームによって行われているが，このうち，本論文で対象とするデータフレームについて示す．データフレームには標準フォーマットと拡張フォーマットの 2 種類が存在しており，両者の違いは CAN ID フィールドの長さである．本論文では，標準フォーマットを前提とする．データフレームの標準フォーマットの構成について**図 1** に示す．各フィールドのうち，本研究に関連するものについてのみ以下に説明を示す．

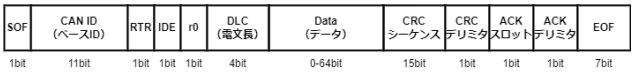

図 1 データフレームの標準フォーマット

- CAN ID
  CAN フレームの識別子を意味するビット列.
- DLC（電文長）
  データフィールドのデータサイズを示すビット列. 0 ～ 8 バイトの範囲の値を設定できる.
- Data（データ）
  データフィールドのデータを示すビット列.

CAN では，メッセージの送信権の調停に CAN ID が用いられる．バスが空いている状態のとき，すべてのユニットがメッセージの送信を始めることが可能である．バスに対して最初に送信したユニットが送信権を得ることができ，衝突が発生した場合には CAN ID によって送信権が決定される．その際，CAN ID の値が小さいほど優先順位が高くなる．ブロードキャスト型であること，およびプロトコルが認証機構を備えないことから，攻撃者による盗聴・なりすましが容易である点が脆弱性として指摘されている [7].

2.2 グラフベース侵入検知システム

グラフベース侵入検知システム（グラフベース IDS）は，ネットワークトラフィックをグラフとして表現し，異常なパターンを検出する手法である．複雑な攻撃パターンや相関関係を捉えることが可能であり，従来の IDS に比べて高い検知精度を持つとされている．メッセージ間の構造的な関係をモデル化することで，ネットワーク全体の挙動を包括的に表現できる点が特徴である．典型的には，観測時間やパケット数をウィンドウ単位に区切り，各ウィンドウから 1 つのグラフを構築する．これらのグラフは，ノード数やエッジ数といった単純な特徴量に基づいて解析されることが多い．本論文で着目するグラフ理論と決定木を組み合わせたグラフベース IDS では，CAN ID をノード，メッセージの遷移をエッジとして，CAN メッセージをグラフで表現する．さらに，グラフから特徴量を生成し，決定木を用いて異常検知を行う．

2.3 CAN バスにおける攻撃

CAN バスに対する攻撃は, 主に以下のようなものがある．

- **DoS（Denial of Service）攻撃**： 攻撃者は，CAN バスに優先度の高いメッセージを大量に注入することで，正規メッセージの通信を妨害する．これにより，車両の機能が停止する可能性がある．
- **ファジング攻撃**： 攻撃者は，ランダムな CAN ID やデータを持つ CAN メッセージを注入することで，車両の意図しない挙動を誘発する．これにより，車両の動作を不安定にさせる可能性がある．
- **なりすまし攻撃**： 攻撃者は，正規の CAN ID を持つ任意のデータをバスに注入することで，誤動作を誘発する．これにより，速度操作などの車両の挙動を操作することが可能となる．
- **リプレイ攻撃**： 攻撃者は，ある期間の正規の CAN 通信をキャプチャし，そのメッセージを再送信することで，状態の再現・制御を試みる．これにより，車両の挙動を意図しない状態にする可能性がある．

© 2025 Information Processing Society of Japan

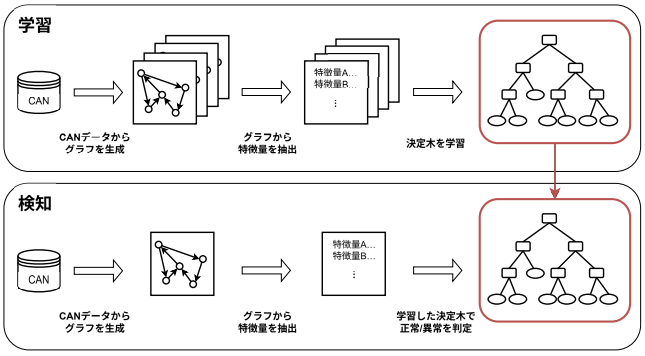

図 2　GDT-IDS の学習と異常検知の流れ

- **マスカレード攻撃**：攻撃者は，攻撃対象 CAN ID を持つメッセージの正規 ECU からの送信を停止させ，同じ CAN ID を持つ任意のデータを送信することで，正規の ECU になりすます．これにより，攻撃者は車両の挙動を操作する可能性がある．

3. 既存研究の概要

近年，グラフベース IDS が注目されている．Ye ら [4] は，グラフ理論と決定木を組み合わせたグラフベース IDS である「GDT-IDS」を提案している．論文では，一定のメッセージ数によってウィンドウに分割しており，ウィンドウサイズは 200 である．本研究でも，比較のためにウィンドウサイズを 200 とする．GDT-IDS の学習の流れと異常検知の流れを**図 2** に示す．

学習の流れは以下の通りである．

(1) **グラフの生成**：ウィンドウによって分割し，CAN ID をノード，メッセージの遷移をエッジとしたグラフを生成する．

(2) **特徴量の抽出**：グラフから特徴量を抽出する．具体的な特徴量は，以下の通りである．

- **ノード数**：グラフに含まれるノードの数をカウントする．
- **エッジ数**：グラフに含まれるエッジの数をカウントする．
- **最大出次数**：各ノードの出次数の最大値を求める．
- **時間差**：ウィンドウに含まれる最初のメッセージと最後のメッセージの時間差を計算する．実際には，グラフの構築に必要となる時間を計算する．
- **最大媒介中心性**：各ノードの媒介中心性を計算し，最大値を求める．媒介中心性は，あるノードが他のノード間の最短パスにどれだけ関与しているかを示す指標である．計算式は以下の通りである．

$$C_B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1}$$

ここで，$C_B(v)$ はノード v の媒介中心性，σ_{st} はノード s からノード t への最短パスの数，$\sigma_{st}(v)$ はそのうちでノード v を通る最短パスの数を表す．

- **グラフ密度**：グラフ理論における密度は，グラフの実際のエッジ数と可能な最大エッジ数の比率を示す指標である．計算式は以下の通りである．

$$D = \frac{R}{N(N-1)} \tag{2}$$

ここで，D はグラフ密度，R は実際のエッジ数，N はノード数を表す．

(3) **決定木の学習**：特徴量を用いて，決定木を学習する．決定木はノード，エッジ，リーフノードから構成される．各ノードは特徴量の値を表し，エッジはその値に基づく分岐を表す．リーフノードは最終的な分類結果を表す．学習の際は，CART（Classification and Regression Trees）アルゴリズムを用いる．決定木の構築において，各ノードでの分割を最適化するために，ジニ不純度を用いる．具体的には，以下の手順で学習を行う．

(a) 各特徴量において，候補しきい値を生成する．候補しきい値は，その特徴量のすべての値を昇順に並べたとき，隣接する値の平均をとることで生成される．

(b) 各特徴量の各候補しきい値に対して，学習データの分割を行う．

(c) 分割後の各場合において，ジニ不純度を計算する．ジニ不純度は，ノード内のクラスの不均一性を示す指標であり，計算式は以下の通りである．

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2 \tag{3}$$

ここで，$Gini(D)$ はデータセット D のジニ不純度，k はクラスの数，p_i はクラス i の割合を表す．

(d) ジニ不純度が最小となる特徴量，候補しきい値において，分割を行う．

(e) 分割を繰り返し，決定木を構築する．

また，異常検知の流れは以下の通りである．

(1) **グラフの生成**：ウィンドウによって分割し，CAN ID をノード，メッセージの遷移をエッジとしたグラフを生成する．

(2) **特徴量の抽出**：グラフから特徴量を抽出する．

(3) **異常検知**：学習済みの決定木を用いて分類結果を得て，異常か正常かを判定する．

GDT-IDS は，グラフの構造を利用することで，メッセージ間の構造的な関係をモデル化することが可能であり，従来の IDS では検知が難しいリプレイ攻撃やなりすまし攻撃を検知することができる．また，決定木を用いることで，特徴量の重要度を可視化することができ，異常検知の解釈性も向上している．

© 2025 Information Processing Society of Japan

4. 既存研究の課題を明らかにする実験

4.1 実験方法

本章では，既存のグラフベース IDS の課題を明らかにするために，以下の手順で実験を行う．ウィンドウのラベルを作成する際は，1 つのウィンドウに 1 個以上の攻撃メッセージが含まれていた場合はそのウィンドウを異常ウィンドウとし，それ以外の場合は正常ウィンドウとした．

(1) **GDT-IDS の実装**： Python を用いて，GDT-IDS を実装する．

(2) **公開データセットを用いた評価**： 3 種類の公開データセットに対して，学習および異常検知を行う．

(3) **評価指標の算出**： 結果を分析し，精度や F 値などの指標を算出する．

(4) **課題の抽出**： 結果を踏まえ，既存のグラフベース IDS の課題を明らかにする．

実験環境としては，以下を用いる．

- **ハードウェア**： CPU は Intel Core i9-13900KS，メモリは 128 GB.
- **ソフトウェア**： OS は Microsoft Windows 11 Pro, Python（Anaconda）は 3.9.19, 主要ライブラリは scikit-learn，NetworkX，NumPy，pandas.

また，評価指標としては，以下を用いる．

- **精度 (Accuracy)**: 正しく分類されたサンプルの割合を示す指標である．計算式は以下の通りである．

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

ここで，TP は True Positive（真陽性），TN は True Negative（真陰性），FP は False Positive（偽陽性），FN は False Negative（偽陰性）を表す．

- **再現率 (Recall)**: 異常なサンプルのうち，モデルが正しく異常であると予測した割合を示す指標である．計算式は以下の通りである．

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

- **適合率 (Precision)**: モデルが異常と予測したものの中で，実際に異常なサンプルである割合を示す．計算式は以下の通りである．

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6}$$

- **F 値 (F1 Score)**: 再現率と適合率の調和平均を示す指標である．計算式は以下の通りである．

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

本実験では，以下の 3 つの公開データセットを使用する．

- **car-hacking dataset**[8]: 自動車ネットワークにおける IDS の評価に広く使用されているデータセットである．4 種類の攻撃シナリオ（DoS 攻撃，ファジング攻撃，なりすまし攻撃 2 種）を含む．
- **car-hacking challenge 2020 dataset**[9]: 2020 年に開催された「Attack & Defense Challenge」において提供・収集されたデータセットである．4 種類の攻撃シナリオ (DoS 攻撃，ファジング攻撃，なりすまし攻撃，リプレイ攻撃) から成る混合攻撃を含む．
- **can-mirgu dataset**[10]: 現実的な攻撃を含む包括的なデータセットの不足という課題に対処するために，実車の CAN データから構築されたデータセットである．17 時間分の正常なデータと，36 種類の攻撃（DoS 攻撃，ファジング攻撃，リプレイ攻撃，なりすまし攻撃，マスカレード攻撃，サスペンション攻撃）を含む．

表 1 使用するデータセットの概要

データセット名	分類名	正常 メッセージ数	異常 メッセージ数
car-hacking	ノーマル	988,871	-
	DoS	3,078,250	587,521
	ファジング	3,347,013	491,847
	なりすまし	3,845,890	597,252
car-hacking challenge	ノーマル	180,686	-
	混合	3,026,591	341,056
can-mirgu	ノーマル	46,084,700	-
	マスカレード	2,192,359	25,183

これらのデータセットから，一部のファイルを使用して実験を行う．使用するファイルを以下に示す．同種のファイルがデータセット内に複数存在する場合に限り，括弧内に使用した具体的なファイル名を示す．

- **car-hacking dataset**: 正常メッセージのみから成る 1 種類のノーマルファイル，1 種類の DoS 攻撃，1 種類のファジング攻撃，1 種類のなりすまし攻撃 (gear_dataset.csv)
- **car-hacking challenge dataset**: 1 種類のノーマルファイル (Pre_train_S_0.csv)，2 種類の混合攻撃 (Pre_train_S_1.csv，Pre_train_S_2.csv).
- **can-mirgu dataset**: 6 種類のノーマルファイル (Benign_day1_file1.log∼Benign_day1_file6.log)，5 種類のマスカレード攻撃.

使用するファイルに関して，正常メッセージの数と異常メッセージの数を**表 1** に示す．

4.2 実験結果

car-hacking dataset, car-hacking challenge dataset, can-mirgu dataset の 3 つのデータセットに対して，GDT-IDS を適用し，異常検知を行った．その結果を**表 2** に示す．決定木を学習する際は，最大の木の深さは 5 とした．これは，car-hacking dataset を用いて木の深さを 1，3，5，7，9，11，13，15 と変化させてそれぞれ F 値を計算し，木の最大深さが 5 のときに F 値が最も高かったことから決定した．car-hacking dataset の DoS 攻撃，ファジング攻撃，な

© 2025 Information Processing Society of Japan

表 2　GDT-IDS の実験結果

データセット名	攻撃名	精度	再現率	適合率	F 値
car-hacking	DoS	0.998	0.998	0.998	0.998
	ファジング	0.998	0.998	0.998	0.998
	なりすまし	0.996	0.996	0.996	0.996
car-hacking challenge	混合	0.921	0.921	0.916	0.905
can-mirgu	マスカレード	0.588	0.588	0.501	0.512

りすまし攻撃に対する GDT-IDS の検知精度は既に論文 [4] で報告されている値とほぼ同等であり，非常に高い精度を持つことが確認できた．一方で，car-hacking challenge dataset の混合攻撃に対する検知精度は既に論文 [4] で報告されているが，今回の実験ではやや低下していることが確認できた．この理由としては，論文 [4] では学習やテストに用いたデータやその分割方法が詳細には記載されておらず，今回実験した場合と異なっていることが考えられる．can-mirgu dataset に対する GDT-IDS の検知精度は論文 [4] では報告されていないが，今回の実験では，精度は 0.588，再現率は 0.588，適合率は 0.501，F 値は 0.512 と，非常に低い値となった．

4.3　考察

car-hacking dataset や car-hacking challenge dataset においては，GDT-IDS は非常に高い検知精度を示した．一方で，can-mirgu dataset においては，マスカレード攻撃に対する検知精度が大きく低下した．**図 3** に，can-mirgu dataset のマスカレード攻撃における特徴量のペアプロット図を示す．ペアプロット図は，対角セルは各特徴量の分布を示し，非対角セルは 2 つの特徴量間の散布を示す．このペアプロット図は，左上から右下に向かって，ノード数（num_nodes），エッジ数（num_edges），最大出次数（max_out_degree），時間差（time_diff），最大媒介中心性（max_betweenness），グラフ密度（density）の順に並んでいる．青色の点は正常ウィンドウを，赤色の点は異常ウィンドウを示している．この結果から，すべての特徴量に関して正常ウィンドウと異常ウィンドウの分布が類似しており，単純な分離が難しいことが分かる．このことは，正規の CAN ID や送信頻度を模擬するマスカレード攻撃では，グラフの遷移構造が正常時とほとんど変化しないため，検知が困難であることを示唆している．GDT-IDS は，グラフの構造的な特徴量に依存しているため，構造が保たれたまま中身だけが置き換わる攻撃には弱いことが明らかになった．したがって，この課題に対処するには，構造だけでなく，データの内容そのものを捉える特徴量が必要となる．具体的には，以下のような特徴量が有効であると考える．

- 値が適切な範囲に収まっているかどうか（値の範囲）
- ビット列のパターンが崩れていないかどうか（ハミング距離）
- 複数バイトの組み合わせが統計的に異常でないかどうか（マハラノビス距離）

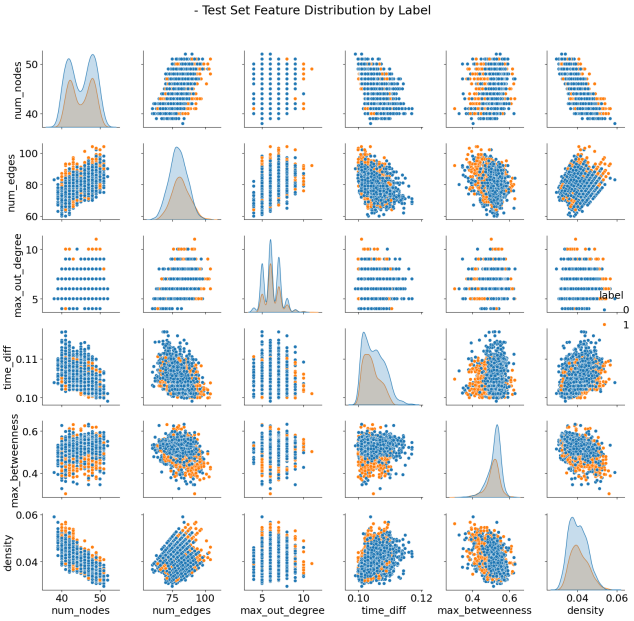

図 3　マスカレード攻撃を対象とした際のペアプロット図

これらの特徴量を用いることで，CAN メッセージのデータ内容をより詳細に捉えることが可能となり，異常検知の精度を向上させることができると考える．

5.　提案手法の概要

本研究では，CAN メッセージのデータ内容を考慮した新たな特徴量として，「逸脱割合（outlier ratio）」を提案する．逸脱割合とは，あるウィンドウ内に含まれるメッセージのうち，データフィールドの値が正常な範囲から逸脱しているメッセージの割合を表す指標である．より具体的には，ウィンドウサイズを 200 メッセージとした場合，逸脱しているメッセージが 20 存在すると，逸脱割合は 0.1 となる．ここでいう「正常な範囲」および「逸脱」は，以下の 3 通りで定義する．

- **値の範囲**：CAN メッセージのデータフィールド部分の数値的な値の範囲を計算する．各 CAN ID ごとに，正常なデータが取りうる値の範囲を学習し，その範囲外の値を取った場合に逸脱とみなす．
- **ハミング距離**：各 CAN ID ごとに，CAN メッセージのデータフィールド部分のビット列間のハミング距離を計算する．ハミング距離の計算式は以下の通りである．

$$d_H(x, y) = \sum_{i=1}^{n} (x_i \oplus y_i) \tag{8}$$

ここで，$d_H(x, y)$ はメッセージ x と y のハミング距離，x_i と y_i はそれぞれのメッセージの i 番目のビットを表す．各 CAN ID ごとに，正常なデータが取りうるハミング距離の範囲を学習し，その範囲外のハミング距離を持つメッセージを逸脱とみなす．

© 2025 Information Processing Society of Japan

- **マハラノビス距離**： 各 CAN ID ごとに，CAN メッセージのデータフィールド部分の統計的な分布を考慮し，マハラノビス距離を計算する．これにより，異常なメッセージを検出することができる．計算式は以下の通りである．

$$d_M(x, \mu, \Sigma) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)} \qquad (9)$$

ここで，$d_M(x, \mu, \Sigma)$ はメッセージ x のマハラノビス距離，μ は平均ベクトル，Σ は共分散行列を表す．各 CAN ID ごとに平均ベクトルと共分散行列を学習し，それを用いてマハラノビス距離を計算する．カイ二乗分布を用いて，マハラノビス距離がしきい値を超えるメッセージを逸脱とみなす．

これらの特徴量を用いることで，CAN メッセージのデータ内容をより詳細に捉えることが可能となり，異常検知の精度向上が期待できる． 実際の実装では，これらの特徴量を GDT-IDS に組み込み，決定木の学習と異常検知を行う．具体的には，提案手法の学習の流れは以下の通りである．

(1) **正常な範囲の学習**： 正常なデータセットから，「正常な範囲」を学習する．具体的には，各 CAN ID ごとに，値の範囲，ハミング距離の範囲，マハラノビス距離の平均と共分散行列を学習する．

(2) **グラフの生成**： ウィンドウによって分割し，CAN ID をノード，メッセージの遷移をエッジとしたグラフを生成する．

(3) **特徴量の抽出**： グラフから特徴量を抽出する．既存の GDT-IDS で使用されている特徴量に加えて，「逸脱割合」を抽出する．逸脱割合は，ウィンドウ内の逸脱メッセージ数の合計をウィンドウ内の全メッセージ数で割った値として定義する．

(4) **決定木の学習**： 特徴量を用いて，決定木を学習する．（以降の手順は，既存の GDT-IDS と同様である．）

6. 提案手法の有効性を評価するための実験

6.1 実験方法

提案手法の有効性を検証するために，以下の手順で実験を行う．

(1) **逸脱割合の実装**： 3 種類の方法で「逸脱割合」を計算し，それぞれ特徴量として追加した GDT-IDS を実装する．

(2) **学習と異常検知**： 3 種類の公開データセットに対して，それぞれ提案手法を適用し，学習および異常検知を行う

(3) **評価指標の算出**： 結果を分析し，精度や F 値などの指標を算出する．

(4) **有効性の評価**： 得られた結果をもとに，提案手法の有効性を評価する．

実験環境および評価指標，データセットについては，第 4 章で行った既存研究の実験と同様である．「正常な範囲」を学習する際は，決定木を学習する際に用いた攻撃を含むデータセットではなく，正常なデータのみから成るノーマルファイルを使用する．

表 3 値の範囲を特徴量として追加した GDT-IDS の実験結果

データセット名	攻撃名	精度	再現率	適合率	F 値
car-hacking	DoS	0.998	0.998	0.998	0.998
	ファジング	0.998	0.998	0.998	0.998
	なりすまし	0.996	0.996	0.996	0.996
car-hacking challenge	混合	0.924	0.924	0.920	0.909
can-mirgu	マスカレード	0.636	0.636	0.566	0.497

表 4 ハミング距離を特徴量として追加した GDT-IDS の実験結果

データセット名	攻撃名	精度	再現率	適合率	F 値
car-hacking	DoS	0.998	0.998	0.998	0.998
	ファジング	0.998	0.998	0.998	0.998
	なりすまし	0.996	0.996	0.996	0.996
car-hacking challenge	混合	0.927	0.927	0.924	0.910
can-mirgu	マスカレード	0.611	0.611	0.475	0.497

表 5 マハラノビス距離を特徴量として追加した GDT-IDS の実験結果

データセット名	攻撃名	精度	再現率	適合率	F 値
car-hacking	DoS	0.998	0.998	0.998	0.998
	ファジング	0.998	0.998	0.998	0.998
	なりすまし	0.996	0.996	0.996	0.996
car-hacking challenge	混合	0.925	0.925	0.920	0.909
can-mirgu	マスカレード	0.777	0.777	0.778	0.778

6.2 実験結果

car-hacking dataset, car-hacking challenge dataset, can-mirgu dataset の 3 つのデータセットに対して，値の範囲，ハミング範囲，マハラノビス距離からそれぞれ計算する新規特徴量である「逸脱割合」を追加した GDT-IDS を適用し，異常検知を行った．まず，値の範囲を特徴量として追加した際の結果を**表 3** に示す．さらに，ハミング距離を特徴量として追加した際の結果を**表 4** に示す．最後に，マハラノビス距離を特徴量として追加した際の結果を**表 5** に示す．正常な範囲の学習にはそれぞれ攻撃を含まないノーマルファイルを使用し，決定木学習には攻撃を含むファイルの一部を利用した．

car-hacking dataset，car-hacking challenge dataset については，どの特徴量を追加した場合でも，既存の GDT-IDS と同等の高い精度を示した．一方，can-mirgu dataset に対しては，追加する特徴量によって性能に差が見られた．より具体的には，値の範囲を特徴量として追加した場合は精度 0.636，再現率 0.636，適合率 0.566，F 値 0.497 となり，ハミング距離を特徴量として追加した場合は精度 0.611，再現率 0.611，適合率 0.475，F 値 0.497 となった．マハラノビス距離を特徴量として追加した場合は精度 0.777，再現率 0.777，適合率 0.778，F 値 0.778 となり，最も高い精度を示した．

© 2025 Information Processing Society of Japan

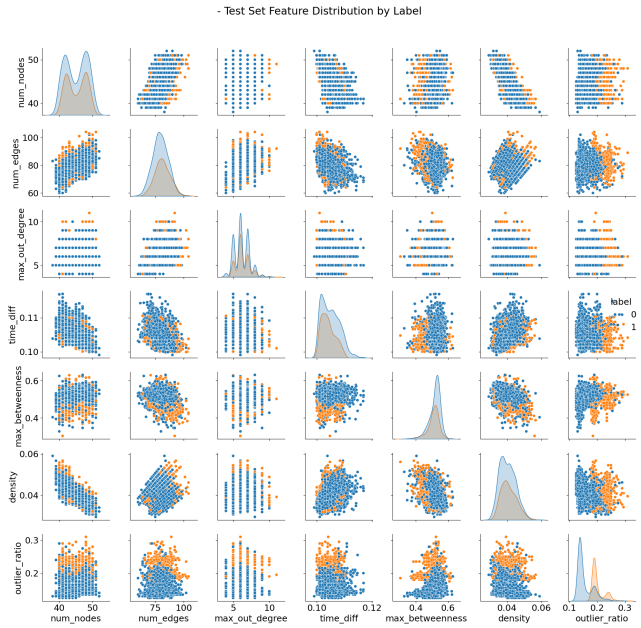

図 4 マスカレード攻撃を対象とした際のペアプロット図（マハラノビス距離）

6.3 考察と課題

表 3〜表 5 の結果より，car-hacking dataset，car-hacking challenge dataset ではいずれの追加特徴量でも高精度を維持した．一方で，can-mirgu dataset のマスカレード攻撃に対しては，マハラノビス距離を用いた「逸脱割合」の追加が最も有効であり，F 値が 0.778 へと大きく改善した．マハラノビス距離が有効であった理由は，複数バイトの組み合わせで統計的に異常がないかどうかを考慮できる点にあると考える．これにより，頻度や変化量が一見適切であっても，CAN メッセージのデータフィールド部分に潜む異常なデータをより詳細に捉えることが可能となったと考察する．**図 4** は，マスカレード攻撃を対象に，マハラノビス距離の逸脱割合を追加した場合の特徴量のペアプロットを示している．この結果から，マハラノビス距離を特徴量として追加した場合，正常ウィンドウと異常ウィンドウの分布がある程度分離されていることがわかる．特に，異常ウィンドウは正常ウィンドウよりもマハラノビス距離が大きいメッセージを多く含んでおり，異常なメッセージを検出しやすくなっていることが確認できる．

一方で，値の範囲やハミング距離を特徴量として追加した場合は，can-mirgu dataset のマスカレード攻撃に対しては F 値が低いままであり，検知精度が低いことがわかった．これは，値の範囲のみではマスカレード攻撃のような「正常な値の範囲に収まるが，異常なメッセージ」であるケースを検出できないためである．またハミング距離については，攻撃対象 CAN ID を持つデータのビット列の正常なパターン変化が比較的大きい範囲で行われていた場合，異常なメッセージのハミング距離が正常な範囲の中に収まることが多く，検知精度が低下したと考えられる．

表 6 マスカレード攻撃に対する FP と FN のウィンドウ数

特徴量	FP のウィンドウ数	FN のウィンドウ数
値の範囲	5	1,042
ハミング距離	96	1,021
マハラノビス距離	329	311

マスカレード攻撃に関して，FP と FN のウィンドウ数をカウントした結果を**表 6** に示す．これらの結果から，マハラノビス距離を特徴量として追加した場合，他の 2 つの特徴量を追加した場合に比べて FN が大幅に減少したことがわかる．これは，マハラノビス距離がデータの分布を考慮しているため，異常なメッセージをより正確に検出できたことを示している．一方で，FP は他の 2 つの特徴量を追加した場合に比べて増加している．これは，本実験ではマハラノビス距離のしきい値をカイ二乗分布の 90%点に設定しているが，90%点は一般的には厳しい境界であり，正常メッセージを異常と判定しやすくなることが影響していると考えられる．この結果から，マハラノビス距離から逸脱割合を計算した場合は，FN を大幅に減らす一方で，FP を増加させるというトレードオフを生むことが分かった．したがって，検知の真偽を確認しつつ，FP の影響を最小化する運用が必要であると考える．具体的には，(1) 追加の分析で確認する，(2) 単発の検知ではアラートを出さず連続して検知した場合に異常として通知する，(3) 異常検知時も車両の運行停止などの強制動作は行わず警告表示に留める，といった運用上の工夫が必要だと考える．

提案手法は，CAN メッセージのデータ内容を考慮することで，異常検知の精度を向上させることができた．特に，マハラノビス距離を特徴量として追加した場合，異常なメッセージをより正確に検出できることが確認できた．しかし，提案手法には以下の課題が残されている．

- **計算コストの増加**： 提案手法では，データフィールド部分の情報を詳細に捉えるため，計算コストが増加する．特に，ハミング距離やマハラノビス距離の計算は，データ量が増えると計算負荷が大きくなる．
- **データセットの限界**： 提案手法の有効性は，使用するデータセットに依存する．既存のデータセットは，限定的な攻撃シナリオに基づいており，実際の車両環境での多様な攻撃を網羅していないことが多い．今後は，より多様な攻撃シナリオを含むデータセットでの検証が求められる．
- **マハラノビス距離のデータ依存**： マハラノビス距離は，平均ベクトルと共分散行列に基づいて計算されるため，正常なデータの分布に依存する．したがって，正常なデータの分布が変化すると，マハラノビス距離の計算結果も変化する可能性がある．これにより，異常検知の精度が低下する可能性がある．そのため，オンラインでの平均ベクトルと共分散行列の更新方法など，対策の検討が必要である．

© 2025 Information Processing Society of Japan

- **異常検知のしきい値設定**：提案手法では，異常検知のためのしきい値を設定する必要がある．異常検知のしきい値が低すぎると，誤検知が増加し，逆に高すぎると，異常なメッセージを見逃す可能性がある．したがって，異常検知のしきい値を適切に設定するための方法の検討が必要である．

7. おわりに

本研究では，グラフ理論と決定木を組み合わせた CAN 向けグラフベース IDS（GDT-IDS）に着目し，公開データセットを用いた再現実験によって課題を明らかにした．さらに，課題の分析を通じて，グラフの構造的な特徴量に依存する GDT-IDS の限界を指摘し，CAN メッセージのデータ内容を考慮した特徴量である，「値の範囲」「ハミング距離」「マハラノビス距離」から求める「逸脱割合」を新規特徴量として提案した．提案手法を用いることで，特に「マハラノビス距離」を特徴量として追加した場合，CAN メッセージのデータ内容をより詳細に捉えることが可能となり，マスカレード攻撃に対する検知精度が向上した．一方で，FP の増加や計算コストの増加といった課題も残されている．今後は，これらの課題に対処するためのアルゴリズムの改良や，より多様な攻撃シナリオを含むデータセットでの検証が求められる．

参考文献

[1] IPA. 自動車の情報セキュリティへの取組みガイド 第 2 版. https://warp.da.ndl.go.jp/info:ndljp/pid/12446699/www.ipa.go.jp/files/000058198.pdf, 2017.

[2] Iso 11898-1:2015. https://www.iso.org/standard/63648.html, 2015.

[3] Andy Greenberg. Hackers remotely kill a jeep on the highway—with me in it. https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/, 2015.

[4] Pengdong Ye, Yanhua Liang, Yutao Bie, Guihe Qin, Jiaru Song, Yingqing Wang, and Wanning Liu. Gdt-ids: Graph-based decision tree intrusion detection system for controller area network. *The Journal of Supercomputing*, Vol. 81, No. 4, p. 591, 2025.

[5] はじめての can / can fd. https://cdn.vector.com/cms/content/know-how/VJ/PDF/For_Beginners_CAN_CANFD.pdf.

[6] 藤沢行雄. 特設 can 規格徹底解説. インターフェース, Vol. 50, No. 3, pp. 116–151, 3 2024.

[7] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage. Experimental security analysis of a modern automobile. In *2010 IEEE Symposium on Security and Privacy*, pp. 447–462, 2010.

[8] Hyun Min Song and Huy Kang Kim. Car hacking dataset. http://ocslab.hksecurity.net/Datasets/car-hacking-dataset, 2018.

[9] Hyunjae Kang, Byung Il Kwak, Young Hun Lee, Haneol Lee, Hwejae Lee, and Huy Kang Kim. Car hacking: attack and defense challenge 2020 dataset. https://ieee-dataport.org/open-access/car-hacking-attack-defense-challenge-2020-dataset, 2020.

[10] Sampath Rajapaksha, Garikayi Madzudzo, Harsha Kalutarage, Andrei Petrovski, and M. Omar Al-Kadri. Canmirgu: A comprehensive can bus attack dataset from moving vehicles for intrusion detection system evaluation. In *Symposium on Vehicles Security and Privacy. Internet Society*, 2024.

© 2025 Information Processing Society of Japan