

# シーケンス制御システムにおける理論保証付き攻撃検知手法

佐藤 和樹<sup>1,a)</sup> 村上 和羽<sup>1</sup> 儀保 駿<sup>1</sup> 渡邊 洋平<sup>1</sup> 田中 崇資<sup>1</sup> 澤田 賢治<sup>1</sup> 岩本 貢<sup>1</sup> 新 誠一<sup>2</sup>

**概要:** 制御システムに対する特定の攻撃に対する検知手法に対し、その検知精度は多くの場合実験によって解析及び評価され、理論的な保証が与えられていない。そのような背景の下で杉本ら (CSS 2023) は、Modbus TCP 通信を用いた制御システムへの DoS 攻撃に対する攻撃検知手法に対し、その検知精度を網羅的に評価するための理論的枠組みを提案した。本稿では、杉本らの研究に基づき、シンプルなシーケンス制御システムを対象とした任意の攻撃に対する検知手法の理論的な枠組みを提案する。具体的には、安全性モデルを定式化し、分布検定における最新の研究成果を適切に応用することで具体的な攻撃検知アルゴリズムを示す。提案検知アルゴリズムの性能は理論的に保証されているものの、検知器が観測すべきデータ長など、一部のパラメータは漸近的に導出されており、要求する検知精度を達成するために設定すべき適切な値が明らかではない。そこで、それらのパラメータを実装実験を通じて明らかにする。

## A Theoretical Framework of Attack Detection Methods in Sequence Control Systems

KAZUKI SATO<sup>1,a)</sup> KAZUHA MURAKAMI<sup>1</sup> HAYATO GIBO<sup>1</sup> YOHEI WATANABE<sup>1</sup> TAKASHI TANAKA<sup>1</sup>  
KENJI SAWADA<sup>1</sup> MITSUGU IWAMOTO<sup>1</sup> SEIICHI SHIN<sup>2</sup>

**Abstract:** The performance accuracy of detection methods for specific attacks against control systems is typically analyzed experimentally, and almost all of them have no theoretical guarantees. Sugimoto et al. (CSS 2023) formalized a theoretical framework to theoretically analyze detection methods for DDoS attacks against control systems using the Modbus TCP. In this paper, following Sugimoto's work, we give a theoretical framework to theoretically analyze detection methods for any attacks against simple sequence control systems. Specifically, we formalize a security model and show a concrete attack detection algorithm by appropriately applying the state-of-the-art method for the distribution testing. However, although the performance accuracy of the algorithm is theoretically guaranteed, some desirable parameters, such as data sizes to be observed for attack detection, are unclear since they were asymptotically derived. Thus, we clarify them via experiments.

### 1. はじめに

ガスや電気、水道といったインフラや、製造工場、交通管理、自動運転システム等、様々な制御システムにおけるセキュリティ問題が注目を浴びようになっている。2000 年頃までは制御システムはインターネットとは接続せずに独自 OS や独自プロトコルを用いて運用されていたが、それ

以降は制御システムのネットワーク化が加速した [8], [10]. サービスの可用性を優先するため、制御システムは一般的に 24 時間 365 日稼働しており、また 10 年以上の長期間にわたって運用されることから脆弱性が複数年単位でシステム内に残ってしまう [7], [9]. このように、制御システムにおける環境の変化はそのセキュリティ問題を深刻化させている。とりわけ、長年にわたって産業用制御システムに対するサイバー攻撃事例が数多く報告されていることから、ネットワーク経由の攻撃検知の重要性も高まっている。ネットワーク経由の攻撃検知では、正常パケットと攻撃パケットの分布的な特徴さに着目して検知を行うことが

<sup>1</sup> 電気通信大学

The University of Electro-Communications

<sup>2</sup> キヤノンメディカルシステムズ株式会社 先端研究所

Advanced Research Laboratory, Canon Medical Systems Corporation

<sup>a)</sup> sato-k@uec.ac.jp

多く、また検知性能の評価項目として誤検知率や検知漏れ確率があることから、相対エントロピー [1], [2], [5], [6] や尤度比検定 [4], [5] などの情報理論を用いた手法が研究されている。一方で、その安全性評価のほとんど全ては実装実験を通じて実施されており、評価可能なパラメータや設定は限られる。任意のパラメータについて、その検知精度に理論的保証を与えている攻撃検知アルゴリズムはほとんど知られていない。

杉本ら [11] は、Nishiuchi ら [6] が扱った Modbus TCP 通信を用いた制御システムへの DoS 攻撃に対する攻撃検知手法に対し、その検知精度を網羅的に評価するための理論的枠組みを提案した。具体的には、当該制御システムをモデル化し、仮説検定を用いて検知性能と攻撃成功確率の理論的評価を与えるための理論的枠組みを構築した。しかしながら、あくまで定式化に留まっており、具体的な検知アルゴリズムの提案には至っていない。

本稿では、杉本らの研究に基づき、シンプルなシーケンス制御システムを対象とした任意の攻撃に対する検知手法の理論的な枠組みを提案する。具体的には、安全性モデルを定式化した後、特定の分布検定で考えられる環境と同一視できることを示し、Diakonikolas ら [3] の最新の分布検定における結果を適用する。彼らはある分布から 2 種類のサンプルを得た上で、2 つのサンプルが同一の分布に従うものかどうかを判別する問題について、適切に判別するために必要なサンプル数の下界を示したと共に、その下界を達成する構成法も提案しており、その結果を我々の定式化した攻撃検知システムに適用する。結果として、攻撃検知システムに事前にどの程度のサンプル数を与えなければならないのか、また検知のためにはどの程度系列を観測しなくてはならないのかを明らかにし、具体的な攻撃検知アルゴリズムも示す。ただし、Diakonikolas らは漸近的な解析のみを実施しているため、検知器が観測すべきデータ長など、一部のパラメータに関して、要求する検知精度を達成するために設定すべき適切な値が明らかではない。そこで、それらのパラメータを実装実験を通じて明らかにする。

## 2. 準備

任意の自然数  $n \in \mathbb{N}$  に対して、 $\{1, \dots, n\}$  を  $[n]$  と書く。任意の有限集合  $\mathcal{X}$  に対し、その要素数を  $|\mathcal{X}|$  で表す。期待値が  $\lambda$  のポアソン分布を  $\text{Poi}(\lambda)$  と書く。

**定義 1** (全変動距離). 離散集合  $\Omega$  に対する離散確率分布  $p, q: \Omega \rightarrow [0, 1]$  が与えられたとき、全変動距離  $d_{TV}(p, q)$  は以下の式で定義される。

$$d_{TV}(p, q) := \frac{1}{2} \sum_{x \in \Omega} |p(x) - q(x)|.$$

## 3. システムモデルと安全性の定式化

本稿では、杉本らの研究 [11] 同様、PLC (Programmable

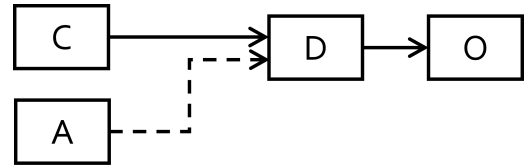


図 1 シーケンス制御システム Sys  
Fig. 1 Sequence Control System Sys

Logic Controller) 等の制御装置 C から制御対象の機器 O に対して入力が行われる、図 1 に示すようなシンプルなシーケンス制御システム Sys を考える。ここで、制御装置 C 以外の攻撃者 A から制御対象 O への入力が行われ得る状況を想定し、入力が正規の制御装置 C から行われたものかどうかを判定するための検知器 D を制御対象 O の手前に設置する。したがって、攻撃検知は検知器 D への入力によって判断するため、モデルとしては制御対象 O は省略して考える。

本来、制御装置 C から出力され、制御対象に渡るものは制御命令（を表すデータ）である一方で、検知器 D が検知に用いるデータは様々なものが有り得る。例えば、杉本ら [11] が対象にした Nishiuchi ら [6] の攻撃実験の検知器では、データが送られてくる遅延時間や処理時間の確率分布に基づき、正規のシステムから送られてきたデータかそうでないかを判断している。したがって、検知器が入力に取るのはデータの遅延時間や処理時間とし、必ずしもデータそのものを入力に取るわけではないことに注意されたい。本稿では、そのような検知器 D が入力に取り得る要素の集合に対し、その集合上の確率分布を考える。

**定義 2** (検知システム). 検知に用いる要素の集合を  $\mathcal{X}$  とし、その要素数を  $n$  とする。制御装置 C の出力を  $\mathcal{X}$  上の確率分布  $P_X$  に従う確率変数  $X$  とする。Sys に対する (擬) 距離  $\text{dist}$  に基づく攻撃検知システム  $\Pi_{\text{dist}}$  は、以下の確率的多項式時間アルゴリズムの組  $\Pi_{\text{dist}} = (\text{Sample}, \text{Derive}, \text{Detect})$  からなる。

- $(S, \text{aux}) \leftarrow \text{Sample}(\kappa, P_X)$ : サンプルングアルゴリズム  $\text{Sample}$  はセキュリティパラメータ  $\kappa$  および確率分布  $P_X$  を入力に取り、検知に必要な事前サンプルとして確率変数の系列  $S := (X_i)_{i \in [N]}$  及び補助情報  $\text{aux}$  を出力する。ここで、 $N$  はセキュリティパラメータに応じて定まる自然数である。
- $\text{cr} \leftarrow \text{Derive}(S, \text{aux})$ : 導出アルゴリズム  $\text{Derive}$  は、事前サンプル  $S$  及び補助情報  $\text{aux}$  を入力に取り、検知の判断基準となる情報  $\text{cr}$  を出力する。
- $0/1 \leftarrow \text{Detect}(L, \text{cr})$ : 検知アルゴリズム  $\text{Detect}$  は確率変数の系列  $L := (L_i)_{i \in [M]}$ 、判断基準  $\text{cr}$  を入力に取り、0 (正規の入力であると判断) または 1 (正規の入力ではないと判断) を出力する。ここで、 $M$  はセキュリティパラメータに応じて定まる自然数である。

ここで、上記モデルについて整理しておく。制御システム側、すなわち制御機器 C 側は  $P_X$  に基づいて確率変数を決定する一方、実用上はそのような分布を把握しているわけではなく、結果的に制御で実際に利用された系列から分布  $P_X$  が生成されると考えることができる。したがって、検知器 D が検知のために  $P_X$  を把握していると考えるのは自然ではないため、 $P_X$  から事前にサンプル  $S$  及び何らかの補助情報  $\text{aux}$  を得るものとし、Sample を定義している。一方で、検知器 D が検知の際にそれらの情報をそのまま用いるとは限らず、例えばサンプルから経験分布を推定しておく等、事前に何らかの検知基準となる情報  $\text{cr}$  に加工する可能性があることから、Derive アルゴリズムを定義した。上記の検知システム  $\Pi_{\text{dist}}$  は、Detect が正規の入力であれば 0 を、そうでなければ 1 を出力することが求められる。これをそれぞれ正当性と検知可能性と呼び、以下のように定義する。

**定義 3** ( $\alpha$ -正当性). 任意のセキュリティパラメータ  $\kappa$ 、任意の確率分布  $P_X$  に対し、検知システム  $\Pi_{\text{dist}}$  が以下を満たすとき、 $\alpha$ -正当性を満たすという。

$$\Pr\{\text{Detect}((X_i)_{i \in [M]}, \text{cr}) \rightarrow 0\} \geq 1 - \alpha.$$

ただし、 $\text{cr} \leftarrow \text{Derive}(\text{Sample}(\kappa, P_X))$  である。

すなわち、正規の入力であれば、Detect アルゴリズムは誤り確率  $\alpha$  を除いて正規の入力であると判断できる、ということである。

**定義 4** ( $(\varepsilon, \beta)$ -検知可能性). 任意のセキュリティパラメータ  $\kappa$ 、任意の確率分布  $P_X$ 、 $\text{dist}(P_X, P_Y) \geq \varepsilon$  であるような任意の確率分布  $P_Y$  に対し、検知システム  $\Pi_{\text{dist}}$  が以下を満たすとき、 $(\varepsilon, \beta)$ -検知可能性を満たすという。

$$\Pr\{\text{Detect}((Y_i)_{i \in [M]}, \text{cr}) \rightarrow 1\} \geq 1 - \beta.$$

ただし、 $\text{cr} \leftarrow \text{Derive}(\text{Sample}(\kappa, P_X))$  である。

すなわち、攻撃者による不正な入力が行われた場合、その確率変数が  $\text{dist}$  の意味で少なくとも  $\varepsilon$  離れているならば、Detect アルゴリズムは誤り確率  $\beta$  を除いて正規の入力でないと判断できる、ということである。

## 4. 検知システムの構成法

本節では、前節で定義した性質を満たす検知システムの実現手法として、分布検定アルゴリズムを応用した構成法を述べる。

$\alpha$ -正当性と  $(\varepsilon, \beta)$ -検知可能性を満たすような検知システム  $\Pi_{\text{dist}}$ 、特に検知アルゴリズム Detect はどのような要件を満たさなければならないのか、以下に整理する。

- 検知器は正規の入力の確率分布  $P_X$  も正規ではない入力の確率分布  $P_Y$  も知らない。ただし、 $P_X$  に従うサンプル  $S := (X_i)_{i \in [N]}$  を得ることができる。

- 入力された確率変数の系列  $L := (L_i)_{i \in [M]}$  が確率変数  $P_X$  に従うものであれば 0 を、そうではない、 $\text{dist}$  の意味で  $P_X$  と  $\varepsilon$  離れている確率変数  $P_Y$  に従うものであれば 1 を出力できなければならない。ただし、それぞれ高々確率  $\alpha$  及び  $\beta$  の失敗は許容される。

上記を踏まえると、分布検定と設定が非常に似ていることがわかる。そこで、Diakonikolas ら [3] による最新の分布検定におけるアルゴリズムを我々の検知アルゴリズム Detect に適用し、効率的な検知アルゴリズムを実現する。

Diakonikolas ら [3] は、ある性質  $\mathcal{P}$  を満たす離散確率分布群と、その性質  $\mathcal{P}$  から全変動距離の意味で  $\varepsilon$  遠い離散確率分布群を識別可能なアルゴリズムを提案した。以下では、最もシンプルな性質である  $\mathcal{P} = \{(p, q) : p = q\}$  を対象に、 $p = q$  であるか  $d_{TV}(p, q) \geq \varepsilon$  であるかを、少なくとも  $1 - \delta$  の確率で識別可能なアルゴリズムを考える。

**定義 5** ( $(\varepsilon, \delta)$ -分布同等性検定 [3]).  $0 < \varepsilon, \delta < 1$  とする。また、与えられた  $[n]$  上の離散確率分布  $p, q$  からそれぞれサンプル  $S_p$  及び  $S_q$  が得られる状況（ただし  $|S_p| > |S_q|$ ）において、少なくとも確率  $1 - \delta$  で以下の 2 つの状況を識別できるアルゴリズムを、 $(\varepsilon, \delta)$ -分布同等性検定アルゴリズム（または単に  $(\varepsilon, \delta)$ -テスター）と呼ぶ。

- (1) **完全性 (Completeness)** : 二つの分布が等しい場合、すなわち  $p = q$ 。
- (2) **健全性 (Soundness)** :  $p$  と  $q$  の全変動距離が  $\varepsilon$  以上異なる場合、すなわち  $d_{TV}(p, q) \geq \varepsilon$ 。

ここで、定義 2 における  $\text{dist}$  を全変動距離とした上で、正規の入力の確率変数を  $P_X := p$ 、攻撃者による入力の確率変数を  $P_Y := q$  とし、事前サンプルの個数  $N$  を  $N := |S_p|$ 、検知アルゴリズムが観測する系列のサイズ  $M$  を  $M := |S_q|$  とすることで、(1) 完全性が  $\delta$ -正当性を、(2) 健全性が  $(\varepsilon, \delta)$ -検知可能性を満たすことが分かる。したがって、シーケンス制御システムに対する攻撃検知システム（定義 2）は、定義 5 の意味での分布同等性検定問題として考えることができる。すなわち、攻撃検知システムの構成は以下の通りである。

- $(S, \text{aux}) \leftarrow \text{Sample}(\kappa, P_X)$ : セキュリティパラメータ  $\kappa$  によって定めた  $\varepsilon, \delta$  に対し、適切な  $N := |S_p|$  を設定し、 $P_X$  から  $S := (X_i)_{i \in [N]}$  と  $\text{aux} := (\varepsilon, \delta)$  を出力する。
- $\text{cr} \leftarrow \text{Derive}(S, \text{aux})$ :  $\text{cr} := (S, \text{aux})$  を出力する。
- $0/1 \leftarrow \text{Detect}(L, \text{cr})$ : 観測する  $L$  のサイズ  $M$  を  $M := |S_q|$  とし、後述するアルゴリズム 1,  $\text{Tester}(n, S, L, \varepsilon, \delta)$  を実行する。その出力が “ABORT” なら再度実行し、“YES” なら 0 を出力し、“NO” なら 1 を出力する。

したがって、提案する攻撃検知システムの性能は Diakonikolas らの結果に強く依存することから、以降は、Diakonikolas らが得た結果を解析する。

#### 4.1 サンプル数の下界

Diakonikolas ら [3] は,  $(\varepsilon, \delta)$ -テスターを実現するために必要なサンプル数の下界を導出した.

**定理 1** ([3]).  $p, q$  を  $d_{TV}(p, q) \geq \varepsilon$  であるような  $[n]$  上の離散確率分布とし,  $K > k$  を満たす  $K, k$  に対して,  $p$  から  $K$  個,  $q$  から  $k$  個のサンプルが得られたとする. このとき, すべての  $(\varepsilon, \delta)$ -テスターは, 以下の式を満たすような  $K, k$  を必要とする.

$$k = \Omega\left(\frac{n\sqrt{\log(1/\delta)}}{\sqrt{K}\varepsilon^2} + \frac{n\sqrt{\log(1/\delta)}}{\varepsilon^2} + \frac{\log(1/\delta)}{\varepsilon^2}\right).$$

#### 4.2 適用する $(\varepsilon, \delta)$ -テスターのアルゴリズム

Diakonikolas ら [3] は, 定理 1 のサンプル数の下界を漸近的に達成する  $(\varepsilon, \delta)$ -テスターのアルゴリズムを提案した (アルゴリズム 1 参照).

アルゴリズム 1 が取る入力は以下の通りである.

- $[n]$ : 分布の台.
- $S_p$ : 分布  $p$  からのサンプルの多重集合であり, その要素数は  $100(K + k)$  である. ここで,  $K > k$  である.
- $S_q$ : 分布  $q$  からのサンプルの多重集合であり, その要素数は  $100k$  である.
- $\varepsilon$ : 分布間の距離.
- $\delta$ : 誤り確率.

ただし, サンプル数に関するパラメータは以下の式で決まる.

$$k = C\left(\frac{n\sqrt{\log(1/\delta)/\min(n, K)}}{\varepsilon^2} + \frac{\log(1/\delta)}{\varepsilon^2}\right). \quad (1)$$

ここで,  $C$  は十分大きな定数である.

次の定理は, 十分大きなサイズのサンプルを入力に取れば, アルゴリズム 1 が  $(\varepsilon, \delta)$ -テスターであることを示している.

**定理 2** ([3]).  $k$  が以下の不等式を満たすとき,  $[n]$  上の分布  $p, q$  からそれぞれ  $O(K + k)$  個および  $O(k)$  個のサンプルを入力に取るアルゴリズム 1 が  $(\varepsilon, \delta)$ -テスターとなるような, 普遍的な定数  $C > 0$  が存在する.

$$k \geq C\left(\frac{n\sqrt{\log(1/\delta)/\min(n, K)} + \log(1/\delta)}{\varepsilon^2}\right).$$

ここで, より詳しく説明しなくてはならない点は, アルゴリズム 1 で用いている統計量  $Z$  と, 25 行目の閾値の求め方である. まず, 統計量  $Z$  について説明する.

統計量  $Z$  は分布間の全変動距離が離れるほど, その値が大きくなるような統計量であり, アルゴリズム 1 では  $Z$  を用いて閾値を設定し, 検定を実施している. 統計量  $Z$  を以下に定義する.

**定義 6** (統計量  $Z$  [3]).  $X_i, X'_i$  をそれぞれ, 離散確率分布

---

#### アルゴリズム 1 $\text{Tester}(n, S_p, S_q, \varepsilon, \delta)$

---

**Input:**  $n, S_p, S_q, \varepsilon, \delta$

**Output:** “YES”, “NO” or “ABORT”

```

1:  $F = \emptyset$ 
2: for each  $s \in S_p$  do
3:    $F = F \cup s$  with probability  $\min(n/100|S_p|, 1/100)$ 
4: end for
5: if  $|F| > n$  or  $|F| > 50K$  then
6:   return ABORT
7: end if
8:  $S'_p = S_p \setminus F$ 
9: Draw  $\ell, \ell' \sim \text{Poi}(2k)$ 
10: if  $\ell > \min(|S'_p|, 100k)$  or  $\ell' > |S_q|$  then
11:   return ABORT
12: end if
13: Let  $S''_p$  be a set of  $\ell$  random samples (taken without replacement) from  $S'_p$  and  $S'_q$  be a set of  $\ell'$  samples from  $S_q$ . Create  $S''_p, S'_q$  by assigning to corresponding sub-bins uniformly at random.
14: Let  $N_p$ : the number of samples in  $S''_p$  that collide with another sample in  $S''_p \cup S'_q$ 
15: Let  $N_q$ : the number of samples in  $S'_q$  that collide with another sample in  $S'_q$ 
16: if  $N_p > c \cdot \max(k^2/K, k^2/n)$  then
17:   return ABORT
18: end if
19: if  $N_q > 20N_p + C' \log(1/\delta)$  then
20:   return NO
21: end if
22: Flag each sample of  $S''_p, S'_q$  independently with probability  $1/2$ 
23: Let  $X_i, Y_i$  be the counts for the number of times element  $i$  appears flagged in each set  $S''_p, S'_q$  respectively and  $X'_i, Y'_i$  be the corresponding counts on unflagged samples.
24: Compute the statistic  $Z = \sum_{i=1}^n Z_i$ , where  $Z_i = |X_i - Y_i| + |X'_i - X_i| - |X_i - X'_i| - |Y_i - Y'_i|$ 
25: if  $Z < C'\sqrt{(\min(k, k^2/K + k^2/n) + \log(1/\delta)) \log(1/\delta)}$  then
26:   return YES
27: else
28:   return NO
29: end if
```

---

$p$  からのサンプルからなる多重集合  $S_p, S_p^*$  における, 要素  $i$  の出現回数とする.  $Y_i, Y'_i$  も同様に, それぞれ, 離散確率分布  $q$  からのサンプルからなる多重集合  $S_q, S_q^*$  における, 要素  $i$  の出現回数としたとき, 統計量  $Z$  を以下の式で定義する.

$$Z = \sum_{i=1}^n (|X_i - Y_i| + |X'_i - Y'_i| - |X_i - X'_i| - |Y_i - Y'_i|).$$

ここで, 統計量  $Z$  は以下の性質を満たす.

**補題 1** ([3]). 統計量  $Z$  は以下の 2 つの性質を持つ.

- $p = q$  のとき,  $E[Z] = 0$
- $d_{TV}(p, q) \geq \varepsilon$  のとき, 以下の式を満たす.

$$E[Z] = \Omega\left(\min\left(k\varepsilon, \frac{k^2\varepsilon^2}{n}, \frac{k^{3/2}\varepsilon^2}{n^{1/2}}\right)\right) \quad (2)$$

補題 1 では, 分布が等しいかそうでないかで統計量  $Z$  の

期待値が大きく変わることを示している。しかし、この統計量  $Z$  に関する性質は、あくまで期待値に関するものである。そこで、以下の補題 2 では統計量  $Z$  がどれだけ期待値に集中するかという集中不等式を示す。

**補題 2** ([3]). 任意の  $0 < \delta < 1$  に対して、統計量  $Z$  が以下の式を満たすような、ある定数  $C > 0$  が存在する。ここで、 $N$  は定義 6 におけるサンプルの多重集合全体  $S_p \cup S_p^* \cup S_q \cup S_q^*$  内で 2 回以上現れるような要素の個数である。

$$\Pr \left\{ |Z - E[Z]| > C \sqrt{(N + \log(1/\delta)) \log(1/\delta)} \right\} \leq \delta \quad (3)$$

アルゴリズム 1 の 25 行目では期待値が集中する範囲外の値を閾値とすることで、より正確な分布の同等性判定を行っている。

まず、 $p = q$  の場合を考える。この時、 $E[Z] = 0$  となるので、式 (3) より、

$$\Pr \left[ Z > C \sqrt{(N + \log(1/\delta)) \log(1/\delta)} \right] \leq \delta$$

となり、 $\sqrt{(N + \log(1/\delta)) \log(1/\delta)}$  よりも十分大きな値を閾値としたい。

一方で、 $d_{TV}(p, q) \geq \varepsilon$  の場合は、式 (2) の下界を示す値  $\left( \min \left( k\varepsilon, \frac{k^2\varepsilon^2}{n}, \frac{k^{3/2}\varepsilon^2}{n^{1/2}} \right) \right)$  と同じかそれより少しでも小さな値を取ればよいことがわかる。しかし、 $Z < E[Z]$  の場合、式 (3) より、

$$\Pr \left[ Z < E[Z] - C \sqrt{(N + \log(1/\delta)) \log(1/\delta)} \right] \leq \delta$$

となるため、式 (2) の下界を示す値よりも十分小さな値を閾値としたい。以下の主張 1 では、 $d_{TV}(p, q) \geq \varepsilon$  のときに閾値となるような値を示している。

**主張 1** ([3]).

$$\begin{aligned} & \min \left( k\varepsilon, \frac{k^2\varepsilon^2}{n}, \frac{k^{3/2}\varepsilon^2}{n^{1/2}} \right) \\ & \gg \sqrt{\left( \min \left( k, \left( \frac{k^2}{K} + \frac{k^2}{n} \right) \right) + \log(1/\delta) \right) \log(1/\delta)}. \end{aligned}$$

この解析に基づき、アルゴリズム 1 では、25 行目の閾値に対して  $\sqrt{(N + \log(1/\delta)) \log(1/\delta)}$  よりも十分大きく、 $\min \left( k\varepsilon, \frac{k^2\varepsilon^2}{n}, \frac{k^{3/2}\varepsilon^2}{n^{1/2}} \right)$  よりも十分小さい値になるような  $C'$  を用いている。

上記のように、アルゴリズム 1 では、いくつかの定数、具体的には、 $S_p$  や  $S_q$  を決める  $k$  の計算に必要な定数  $C$ 、16 行目の閾値に用いられている定数  $c$ 、そして 25 行目の閾値に用いられる上記の定数  $C'$  を適切に設定する必要がある。しかしながら、Diakonikolas らは漸近的な議論のみ行っており、具体的な値を与えていない。そこで、次節では実装実験を通じて適切な  $C, c, C'$  の値の導出を試みる。

## 5. 実装実験

### 5.1 目的

Diakonikola ら [3] が提案したアルゴリズム 1 は、漸近的な解析が行われているが、入力長や出力の判定に用いるパラメータもまた漸近的に定義されている。しかし、実際に制御システムにおいてアルゴリズム 1 を用いるには、これらのパラメータを具体的な数値として定める必要がある。そこで本研究では、アルゴリズム 1 をソフトウェアとして実装することにより、入力長や出力を決定する上で必要となる定数  $C, c, C'$  の具体的な値を明らかにすることを目的とする。

### 5.2 実験方法

本実験では、アルゴリズム 1 における  $C, c, C'$  の具体的な値を明らかにすることを目的としている。よって以下の完全性と健全性について、実験を行うことで、アルゴリズムが  $(\varepsilon, \delta)$ -テスターとなる  $C, c, C'$  の具体的な値を明らかにする。

**完全性** 正規分布からのサンプル二つに対して、 $1 - \delta$  以上の確率で “YES” を返すか。

**健全性** 正規分布からのサンプルと、一様分布からちょうど  $\varepsilon$  離れた分布からのサンプルに対して、 $1 - \delta$  以上の確率で “NO” を返すか。

具体的には以下の実験を行う。

**実験 1** 適当な  $\varepsilon, \delta$  として  $\varepsilon = 0.5, \delta = 0.1$  で固定したときに、 $C, c, C'$  を 1 から  $10^9$  まで、10 倍ずつ  $(1, 10, 100, \dots, 10^9)$  変化させたときにアルゴリズム 1 が  $(\varepsilon, \delta)$ -テスターとなるような、 $C, c, C'$  の値の具体的な値を調べ、適切な  $c$  の値を決定する。

**実験 2**  $\varepsilon$  を 0.1 から 0.9 まで 0.1 ずつ、 $\delta$  を 0.01 から 0.10 まで 0.01 ずつ、 $C, C'$  を実験 1 の結果から適切に設定した範囲（具体的には 1 から 10 まで 1 ずつ）で変化させ、実験 1 で求めた適切な  $c$  の値（具体的には  $c = 1$ ）を固定したときにアルゴリズム 1 が  $(\varepsilon, \delta)$ -テスターとなるような、 $C, C'$  の値の具体的な値を決定する。

また、アルゴリズムの入力として与えられる分布の台  $n$  は 100、 $S_p$  のサンプル数にかかわるパラメータである  $K$  は実験 1 は  $10^{14}$ 、実験 2 は 100000 とし、各  $C, c, C'$  に対して 100 回実験を行い、正答率を評価した。

### 5.3 実験環境

本研究における実験環境を表 1 に示す。

### 5.4 実験 1 の結果

図 2 および図 3 に、アルゴリズム 1 における  $C, c, C'$  の完全性および健全性に対する実験結果を示す。

完全性に関しては、 $C' = 1$  の時を除き、すべての  $(C, C')$

表 1 実験環境

Table 1 Experimental Environment

CPU	Intel Xeon Gold 6240 CPU @ 2.60GHz
メモリ	791 GB
OS	Ubuntu 22.04.5 LTS
言語	Python 3.11.13
使用ライブラリ	numpy 2.3.1

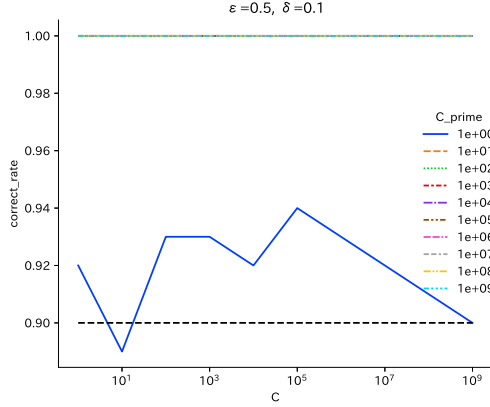


図 2 完全性 (実験 1)

Fig. 2 Completeness(Experiment1)

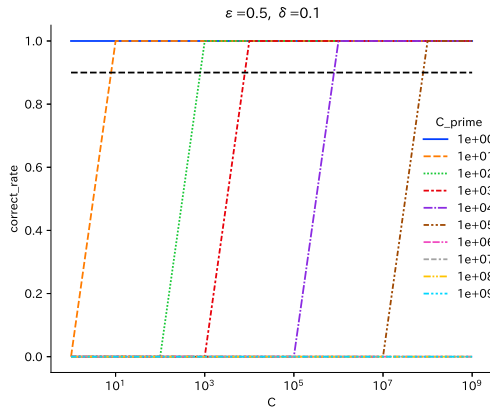


図 3 健全性 (実験 1)

Fig. 3 Soundness(Experiment1)

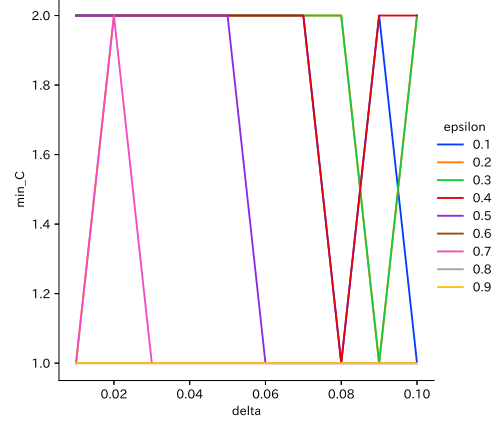
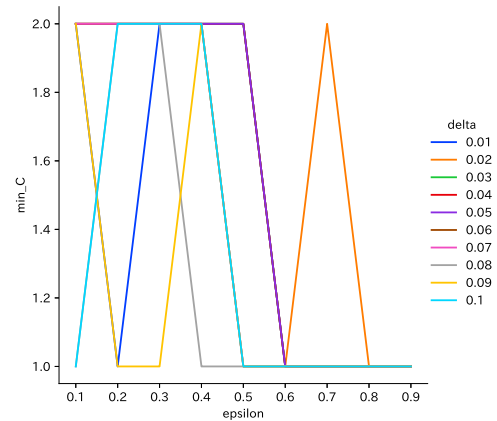
の組み合わせで正しく判別ができていた。したがって、完全性を常に満たすためには  $C' > 1$  となるような  $C'$  で十分であることが確認できた。

健全性に関しては、 $C' = 1$  の場合は常に判別可能である一方、 $C' \geq 1,000,000$  の場合はまったく正しく判別できていなかった。それ以外の  $C'$  においては、 $C'$  の値が大きくなるほど、それに依存してより大きな  $C$  が健全性を満たすために必要であることがわかった。なお、 $C'$  に対する  $C$  の変化率は一定である。したがって、図 2 で確認した完全性を満たす最小の  $C'$  (この実験においては  $C' = 10$ ) に対して、健全性を保証するために必要な最小の  $C$  は  $C = 10$  であった。

最後に、いずれの設定においても、ABORT が出ること

表 2 各  $\varepsilon, \delta$  に対して完全性を満たす最小の  $C'$ Table 2 Minimum  $C'$  that satisfies completeness for each  $\varepsilon$  and  $\delta$ 

$\varepsilon$	$\delta$	$C'$
0.1	0.1	1
0.5	0.1	1
それ以外		2
0.9	0.07	3

図 4  $C' = 3$  における最小の  $C$  の遷移 (横軸:  $\delta$ )Fig. 4 Minimum  $C$  under  $C' = 3$  (horizontal axis:  $\delta$ )図 5  $C' = 3$  における最小の  $C$  の遷移 (横軸:  $\varepsilon$ )Fig. 5 Minimum  $C$  under  $C' = 3$  (horizontal axis:  $\varepsilon$ )

はなく、 $c$  の値は  $c = 1$  で十分であると確認できた。

以上の結果から、完全性と健全性の双方を保証するためには、 $C = 10, c = 1, C' = 10$  とすれば十分であり、この設定でアルゴリズム 1 は  $(\varepsilon, \delta)$ -テスターとして機能することが確認できた。

## 5.5 実験 2 の結果

実験 1 で求めた  $C, c, C'$  の値をより細かく解析し、それらをより小さく設定可能かを調べた。表 2 および図 4, 図 5 にアルゴリズム 1 におけるパラメータ  $C, c, C'$  の完全性および健全性に対する影響を示す。

表 2 は、各  $(\varepsilon, \delta)$  に対して、完全性を満たす最小の  $C'$  を

示している。90 個の組み合わせのうち、3 個を除き、 $C' = 2$  が完全性を満たす最小の  $C'$  であった。したがって、 $C' = 3$  であれば必ず完全性を満たすことが分かる。

健全性に関しては、どんな  $\varepsilon, \delta$  に対しても完全性を満たすように、 $C' = 3$  を固定した上で、 $\varepsilon$  および  $\delta$  を変化させたときの、健全性を満たす最小の  $C$  の遷移を図 4、図 5 に示す。いずれの場合においても、 $\varepsilon$  と  $\delta$  の変化によって、必要な  $C$  に大きな変化は見られず、 $C = 2$  であれば必ず健全性を満たすことが分かった。

以上の結果から、 $C = 2, c = 1, C' = 3$  とすることでアルゴリズム 1 が  $(\varepsilon, \delta)$ -テスターとして問題なく機能することが確認できた。

## 5.6 考察

### 5.6.1 実験 1 について

完全性の実験結果（図 2）において、 $C' = 1$  の正答率にばらつきが見られるのは、閾値として用いる  $\sqrt{(\min(k, k^2/K + k^2/n) + \log(1/\delta)) \log(1/\delta)}$  が、集中不等式 (2) で示される  $|Z - E[Z]|$  の範囲内に存在しているためであると考えられる。集中不等式 (2) は、 $Z$  が期待値付近に高い確率で集中することを保証するが、 $Z$  の具体的な値までは示していない。このため、実験ごとに  $Z$  の値が変動し、その結果として  $Z$  が閾値の前後どちらにも位置することがあり、正答率のばらつきが発生していると考えられる。

健全性の実験結果（図 3）において、 $C'$  が大きくなるにつれて必要となる  $C$  も増加する傾向が見られるのは、 $C'$  を大きくすると、 $E[Z]$  よりも閾値が大きくなり、判定が誤りとなる一方で、 $C$  を大きくすると式 (2) により  $E[Z]$  も増加し、結果として  $E[Z]$  が閾値を上回るためであると考えられる。

### 5.6.2 実験 2 について

完全性の実験結果（表 2）において、 $C' = 2$  である組み合わせが 87 個、それ以外であった組み合わせが 3 個だったことから、最小の  $C'$  が若干変動しているのは誤差だと考えることができる。特に、 $C' = 1$  の場合は、正答率のばらつきがたまたま正答率の範囲内に収まったためだと推測される。なお、 $C' = 3$  となったのは  $\varepsilon$  が極端に大きい場合であることから、実用的な範囲においては  $C' = 2$  であれば必ず完全性を満たすと言える。

健全性の実験結果（図 4、図 5）において、 $\varepsilon = 0.8, 0.9$  の時に、 $C = 1$  で十分であったのは、2 つの分布が大きく離れていることによるものだと考えられる。全変動距離  $\varepsilon$  が十分に大きい場合、 $\varepsilon$  の値による入力長の変化以上に明確な差異が生じ、比較的少ないサンプル数で判別が可能となり、その結果として  $C = 1$  でも高い正答率が得られたと推測される。

## 6. まとめと今後の課題

本稿では、シンプルなシーケンス制御システムを対象とし、任意の攻撃に対する検知手法の理論的な枠組みを提案した。具体的には、当該制御システム及び攻撃検知システムをモデル化して安全性を定式化すると共に、分布検定の最新の結果を応用することで、十分な検知精度を達成するために必要なサンプル数を示すと共に、その下界を達成する具体的な検知アルゴリズムを示した。また、漸近的な評価のため不明瞭であったパラメータの定数部分を、実装実験を通じて実用的なパラメータに対して適切な定数パラメータを導出した。今回設定したパラメータにおいては、少なくとも  $C = 2, c = 1, C' = 3$  と設定した上で提案アルゴリズムを実行すれば良い。

今後の課題として、定数パラメータの理論的導出が挙げられる。提案検知アルゴリズムの精度については漸近的な理論保証が与えられているものの、今回、実装に際して特定すべき定数パラメータについては実験的に導出した。理想的には、あらゆるパラメータに対する適切かつ具体的な定数パラメータが理論的に導出されるべきである。

**謝辞** 本研究はキャノンメディカルシステムズ株式会社の助成を受けたものである。記して感謝を示す。

## 参考文献

- [1] Bereziński, P., Jasiul, B. and Szpyrka, M.: An Entropy-Based Network Anomaly Detection Method, *Entropy*, Vol. 17, No. 4, pp. 2367–2408 (2015).
- [2] Day, L., Ghosh, T., Bagui, S. and Bagui, S.: Entropy Analysis for Modbus Traffic over TCP/IP in Industrial Control Systems, *CATA 2022* (Gupta, B., Bandi, A. and Hossain, M., eds.), EPIc Series in Computing, Vol. 82, EasyChair, pp. 63–71 (2022).
- [3] Diakonikolas, I., Gouleakis, T., Kane, D. M., Peebles, J. and Price, E.: Optimal testing of discrete distributions with high probability, *STOC 2021*, New York, NY, USA, Association for Computing Machinery, p. 542–555 (2021).
- [4] Harrou, F., Bouyeddou, B., Sun, Y. and Kadri, B.: A Method to Detect DOS and DDOS Attacks based on Generalized Likelihood Ratio Test, *ICASS 2018*, pp. 1–6 (2018).
- [5] Nishiuchi, T., Abe, Y., Watanabe, Y., Iwamoto, M., Sawada, K. and Shin, S.: On the Attack Detection Performance of Information-theoretic Method in Industrial Control System, *IECON 2024*, pp. 1–6 (2024).
- [6] Nishiuchi, T., Fujita, S., Watanabe, Y., Iwamoto, M. and Sawada, K.: Packet Analysis and Information Theory on Attack Detection for Modbus TCP, *IECON 2023*, pp. 1–6 (2023).
- [7] Takano, M.: Sustainable cyber security for tility facilities control system based on defense-in-depth concept, *SICE Annual Conference 2007*, pp. 2910–2913 (2007).
- [8] Trifonov, R., Tsochev, G., Manolov, S., Yoshinov, R. and Pavlova, G.: Cyber Trends in Industrial Control Systems, *CSCC 2021*, pp. 41–45 (2021).

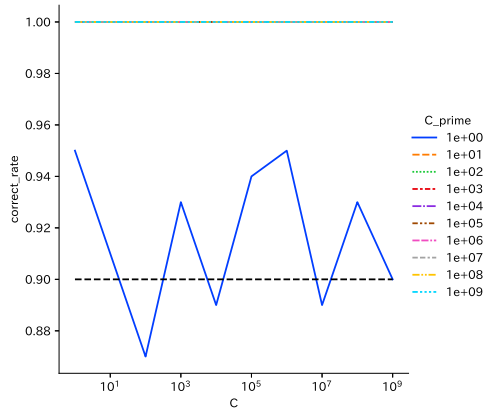


図 A.1 完全性 (実験 1)

Fig. A.1 Completeness(Experiment1)

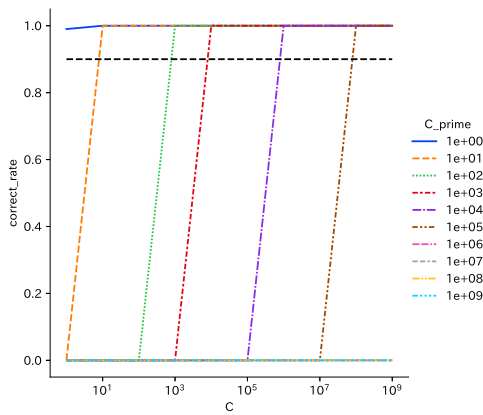


図 A.2 健全性 (実験 1)

Fig. A.2 Soundness(Experiment1)

- [9] Ude, O. and Swar, B.: Securing Remote Access Networks Using Malware Detection Tools for Industrial Control Systems, *ICPS 2021*, pp. 166–171 (2021).
- [10] Zhang, D. and Wang, J.: Research on Security Protection Method of Industrial Control Boundary Network, *2021 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS)*, pp. 560–563 (2021).
- [11] 杉本航太, 安部芳紀, 西内達哉, 渡邊洋平, 澤田賢治, 岩本 貢: 制御システムにおける攻撃検知手法の理論的かつ網羅的評価の一検討, *CSS 2023 予稿集*, pp. 407–414 (2023).

## 付 録

### A.1 一様分布に対する実験結果

アルゴリズム 1 において, 確率分布  $p$  を一様分布とした場合の実験も行った. その結果, 実験 2 における健全性の結果を除き, ほぼ同様の傾向が見られた. 図 A.1) 及び図 A.2 に実験 1 における完全性と健全性の結果を, 図 A.3 及び図 A.4 に実験 2 における最小の  $C$  の遷移を, 表 A.1 に各  $\varepsilon, \delta$  に対して完全性を満たす最小の  $C'$  を記す.

紙面の都合上, 正規分布で実験を行った場合と類似する結果が表れた箇所について詳細は省略するが, 健全性の実

表 A.1 各  $\varepsilon, \delta$  に対して完全性を満たす最小の  $C'$

Table A.1 Minimum  $C'$  that satisfies completeness for each  $\varepsilon$  and  $\delta$

$\varepsilon$	$\delta$	$C'$
0.1	0.09	1
その他		2
0.9	0.05	3

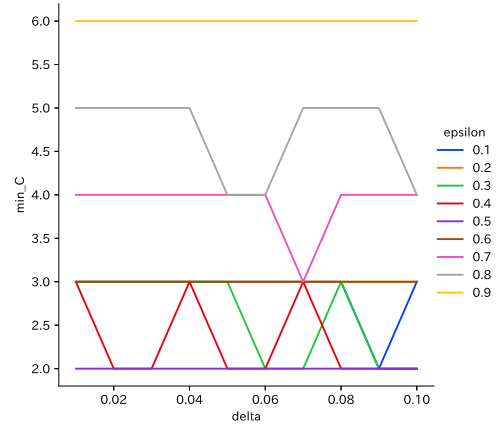


図 A.3  $C' = 3$  における最小の  $C$  の遷移 (横軸:  $\delta$ )

Fig. A.3 Minimum  $C$  under  $C' = 3$  (horizontal axis:  $\delta$ )

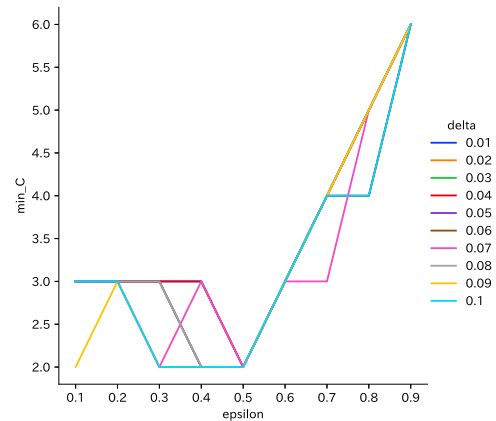


図 A.4  $C' = 3$  における最小の  $C$  の遷移 (横軸:  $\varepsilon$ )

Fig. A.4 Minimum  $C$  under  $C' = 3$  (horizontal axis:  $\varepsilon$ )

験結果 (図 A.3, 図 A.4) において,  $\varepsilon$  が大きいほど必要な  $C$  が大きくなっていることが確認できる. この原因については不明瞭であるが, 実用的に重要なのは  $\varepsilon$  が小さい場合であり,  $\varepsilon \leq 0.5$  においては正規分布に対する実験と同様の結果が得られていることに留意されたい.