Volume 5, Issue 2, Nov. 01, 2020

TC-CPS Newsletter

IEEE TC-CPS Cyber-Physical Systems

A New Permutation-based Write Strategy for Skyrmion Racetrack Memory

Tsun-Yu Yang,[1] Ming-Chang Yang,[1] and Wang Kang[2]
[1]Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong SAR
[2]School of Microelectronics, Beihang University, Beijing, China

1 Introduction

Racetrack memory (RM) is an emerging non-volatile memory (NVM) technology that offers the promising feature of shifting/moving data along a racetrack by current. RM delivers not only high-density data storage as conventional hard disk drive (HDD) but also efficient read/write performance as dynamic random-access-memory (DRAM) [15]. The first generation of RM, namely *domain-wall racetrack memory (DW-RM)*, was demonstrated by IBM in 2008 [15]. However, DW-RM suffers great challenges of reducing the DW size and the critical current density for DW motion [3]. Another promising generation of RM, namely Skyrmion racetrack memory (Sky-RM), was first found in 2009 [13] and has drawn increasing attention recently. Sky-RM leverages magnetic Skyrmions to carry information, where Skyrmions are particle-like spin textures. Specifically, compared with DW-RM, Sky-RM is more energy-saving and has higher stability.

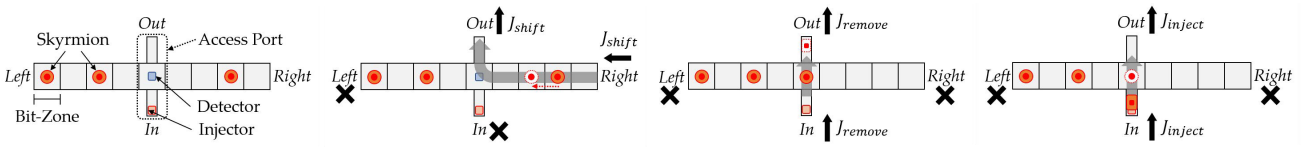
Till now, lots of efforts have been involved in developing Sky-RM [20, 10, 14, 18, 8]. Nevertheless, it was reported that writing/injecting new Skyrmions into Sky-RM is not only time-consuming but also energy-hungry [18]. Although there are some works focusing on minimizing the latency and energy cost of Skyrmion injection from the aspect of physical design [8], none of the existing studies, to date, tries to minimize the number of expensive Skyrmion injections during data writes for Sky-RM from the system point of view. Therefore, this article presents a new permutation-based write strategy, called Permutation-Write (PW) strategy, to optimize the write performance and energy by leveraging the unique features of Sky-RM. The key idea of the PW strategy is to exploit the topologically-protected and particle-like feature of Skyrmions, so that the existing Skyrmions in the racetrack can be "re-permuted" for circumventing the expensive Skyrmion injections upon writing new data into the Sky-RM.

2 Background and Motivation

2.1 Basics of Skyrmion Racetrack Memory (Sky-RM)

The basic structure of Sky-RM is shown in the Figure 1 [20]. There are mainly three components in a Sky-RM. The first one is *injector*, which can generate a local spin-polarized current for creating (i.e., injection) a Skyrmion [6]. It can be either a spin-value or magnetic tunnel junction (MTJ) device [8], [17]. The second one is *detector*. It's able to detect the presence of a Skyrmion by leveraging the magnetoresistance effect with an MTJ device [5], [2]. The final one is the *cross-shaped racetrack* (with four terminals denoted as *Left*, *Right*, *In*, and *Out*) for carrying and manipulating Skyrmions. By applying a driving current through a heavy metal between any two terminals, the Skyrmions can be freely shifted along the racetrack by spin Hall effect [11]. On the other hand, the racetrack can be logically partitioned into several equal-sized *bit-zones*, where each bit-zone can hold at most one Skyrmion. If the bit-zone contains a Skyrmion, the bit will be interpreted as "1"; otherwise, bit "0" will be interpreted [7]. Besides, the intersected bit-zone is often referred to as *access port* since it's together with the injector, detector, and the terminals *In* and *Out*.

Based on the basic structure and data representation, there are four basic operations in Sky-RM: `detect`, `shift`, `remove`, and `inject` operations [20]. Firstly, the `detect` operation is to "read out" the value in the



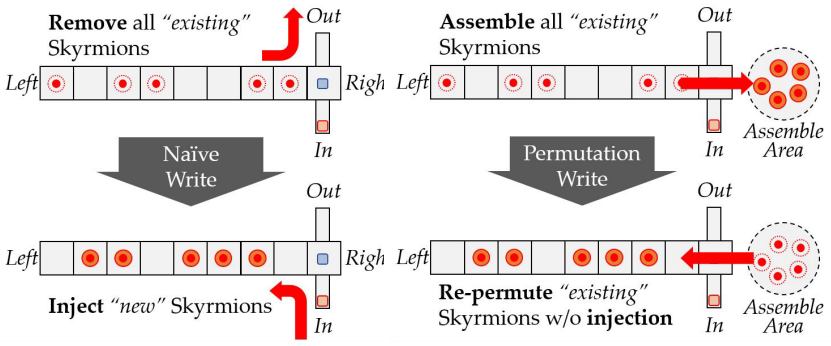
(a) `Detect` Operation (b) `Shift` Operation (c) `Remove` Operation (d) `Inject` Operation

Figure 1: Basic Structure and Basic Operations of Sky-RM.

intersected bit-zone of the access port. That is, it is to detect the presence of a Skyrmion at the access port. Figure 1(a) shows an example when the bit of "0" can be read out through the access port. Secondly, the `shift` operation is to move Skyrmions along the track for one bit-zone by applying a driving current between two terminals. For example, as shown in Figure 1(b), while a driving current J_{shift} is applied from terminal *Right* to terminal *Out*, all the Skyrmions on the right of the access port will be shifted for one bit zone from terminal *Right* to terminal *Out* while all the Skyrmions on the left of the access port will simply stay. Thirdly, the `remove` operation is to get rid of the Skyrmion (if any) at the access port. As shown in Figure 1(c), similar to the `shift` operation, the `remove` operation can be completed by applying a current J_{remove} from terminal *In* to terminal *Out*, so that the Skyrmion will be "shifted out" and get disappeared from the track. Fourthly, the `inject` operation is the most complicated one. As shown in Figure 1(d), there are two major steps involved. The first step is to generate a Skyrmion through the injector at the access port, and the second step is to apply a current J_{inject} from terminal *In* to terminal *Out*, so that the newly generated Skyrmion can be "shifted into" the track.

2.2 Motivation: Write Inefficiency of Sky-RM

As mentioned in the previous section, among the four basic operations of Sky-RM, the `inject` operation is the most expensive one not only in latency but also in energy [18]. As a result, naïvely exploiting the costly `inject` operations to write new data into Sky-RM may be quite time-consuming and energy-hungry. Figure 2(a) shows an example of naïve write strategy. As illustrated, when writing a new data, all the existing Skyrmions need to be firstly removed by `shift` and `remove` operations. Then, `inject` and `shift` operations will be used to form the bit pattern of new data.



(a) Naïve Write Strategy (b) Permutation-Write Strategy

Figure 2: Naïve Write versus Permutation-Write Strategies.

In fact, write reduction has been widely discussed for other types of NVM technologies such as Phase-Change Memory (PCM) [16]. This is because, like Sky-RM, PCM also suffers time-consuming and energy-costly data writes [9]. One of the famous works is data-comparison write (DCW) [19], which introduces an effective write reduction technique for PCM. The key idea of DCW is to use the "read-compare-write" operation instead of a simple write operation so that it can avoid unnecessary bit writes when the corresponding bits are identical in the old and new written data. Furthermore, Flip-N-Write [1] inherits the design of DCW and further reduces the writes by introducing a single "flip bit". That is, if the flip bit is set, every bit in the written data will be interpreted as its

opposite value. By doing so, Flip-N-Write can limit the actual number of bit writes to less-than-half of the data size. Although these two designs gain great success in write reduction, they are mainly developed for PCM-based NVM, while they may greatly overlook and under-utilize the unique features of Sky-RM for injection minimization.

Therefore, this article proposed a new write strategy to exploit the unique features of Sky-RM. We are particularly interested in the possibility of "re-using" the existing Skyrmions of the old data for minimizing the number of new Skyrmion injections on writing new data. Our design concept is shown in Figure 2(b). As illustrated, if we can find a way to "assemble" and "re-permute" the existing Skyrmions of the old data into the new data, we are possible to totally circumvent (or minimize the use of) the expensive Skyrmion injections for data writes. The main technical challenge of this new write strategy lies in how to leverage those less-expensive operations of Sky-RM to efficiently assemble and re-permute the existing Skyrmions in the old data to form the new data, so as to ultimately minimize the number of expensive Skyrmion injections for write performance and energy optimization.

3 Permutation-Write Strategy

3.1 Word-based Structure for Sky-RM

In this article, we consider a "word-based structure" for Sky-RM as illustrated in Figure 3, which contains M words of data and N bit-zones inside a word. The rationale behind this is that our design mainly treats the Sky-RM as a promising replacement of memory, and the modern CPU usually accesses the memory content in the unit of a word (e.g., 32 bits or 64 bits). On the other hand, every word is associated with an access port to manipulate its own data by detecting, removing, and injecting Skyrmions.

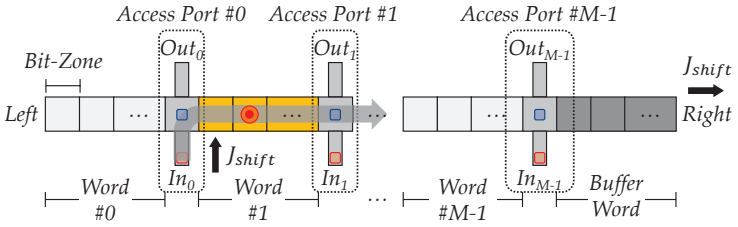


Figure 3: Word-based Structure of Sky-RM.

3.2 Permutation-Write Strategy for Sky-RM

Based on the word-based structure discussed in the previous section, this section presents the Permutation-Write strategy to optimize both write latency and write energy consumption for Sky-RM. Nevertheless, please note that, the Permutation-Write strategy can be easily adapted to structures of different access port configurations.

Here shows the pseudo-code of PW strategy, which is composed of three major steps. Consider a new data word D is received to be written into word w of the Sky-RM, where the number of bits in a word is N. The first major step (i.e., *assembling step*, Lines 1–11) is to assemble the existing Skyrmions in the word w. A counter namely *sky_count*, which is initialized with 0, will be used to count the total number of existing Skyrmions in the word w. Then, PW will shift all bits of word w to right one by one so that the right-most bit of word w will be located at the access port #w.

The second major step (i.e., *re-permuting step*, Lines 12–25) is to re-permute the existing Skyrmions in word w (and to inject insufficient Skyrmions if needed), so as to form the new data. The re-permuting step will iterate every bit, from the leftmost one to the rightmost one, in the new written data D. If the value of the iterated bit is "0", a shift operation from In_w to $Out_w - 1$ will be given to create a bit "0" into the word w. Otherwise, PW will write a bit of "1" into word w by shifting one of the assembled Skyrmions to left; nevertheless, when the existing Skyrmions are exhausted, a new Skyrmion will be injected instead.

Finally, when there is an excess number of assembled Skyrmions, the third major step (i.e., *removing step), Lines 26–30* will be conditionally performed to remove all the assembled Skyrmions from the track one-by-one.

Algorithm 1: PERMUTATION-WRITE Strategy

Input: w: The address (index) of the written word.
Input: D: The new data to be written in.
Input: N: The number of bits in a word.
`// 1) Assemble the Existing Skyrmions`
$sky\_count \leftarrow 0$;
`Shift` right from In_{w-1} to Out_w;
for $i = N : 1$ **do**
  $b \leftarrow$ `Detect` the binary value of i^{th} bit in word w;
  if $b == 0$ **then**
    `Shift` out the bit 0 from In_{w-1} to Out_w;
  else
    `Shift` right an existing Skyrmion from In_{w-1} to $Right$;
    $sky\_count \leftarrow sky\_count + 1$;
  end
end
`// 2) Re-permute the Existing Skyrmions & Inject New Skyrmions (if needed)`
for $i = 1 : N$ **do**
  if $D[i] == 0$ **then**
    `Shift` left from In_w to Out_{w-1} for creating a bit "0" into the word w;
  else
    if $sky\_count > 0$ **then**
      `Shift` left an existing Skyrmion from $Right$ to Out_{w-1};
      $sky\_count \leftarrow sky\_count - 1$;
    else
      `Shift` left from In_w to Out_{w-1};
      `Inject` a new Skyrmion from In_w;
    end
  end
end
`Shift` left from In_w to Out_{w-1};
`// 3) Remove Excess Skyrmions (if any)`
while $sky\_count > 0$ **do**
  `Shift` left from $Right$ to Out_w;
  `Remove` an excess Skyrmion from Out_w;
  $sky\_count \leftarrow sky\_count - 1$;
end

4 System-Level Evaluation

To demonstrate the effectiveness of the proposed Permutation-Write strategy, we compare the proposed PW strategy with the naïve write strategy (denoted as "Naïve") and two state-of-the-art write reduction strategies for NVM (i.e., "DCW" [19] and "Flip-N-Write" [1]). However, since the physical design of Sky-RM is still maturing, instead of simply showing the total performance latency and energy consumption, we tend to show the numbers of `shift`, `detect`, `remove`, and `inject` operations required by different strategies on processing the data writes by words.

The realistic workloads that we used is `MiBench` suite [4]. Also, we develop an in-house trace-driven simulator to model the word-based structure of Sky-RM and measure the operational costs of the evaluated strategies. We instrument Intel `PIN` [12] to generate and feed the traces collected from the benchmark programs into our simulator. The word size is set to be 64 bits, since the 64-bit CPU is commonly used in contemporary computer systems.

Figure 4 shows our evaluation under various benchmark programs. In each sub-figure of Figure 4, the y-axis denotes the average number of operations per 64-bit-word write introduced by different write strategies and the x-axis further denotes the operation type. First of all, we can observe that the number of `shift` operations are all

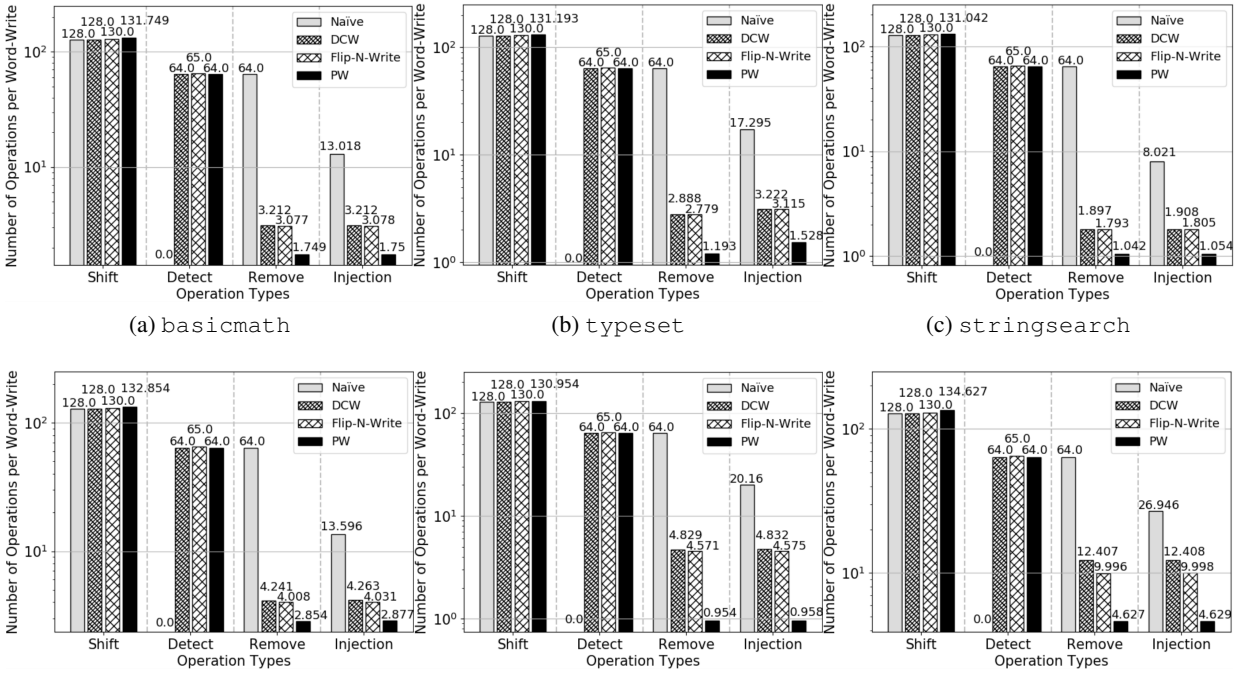


Figure 4: The average number of operations per 64-bit-word write introduced by the evaluated write strategies under various `MiBench` programs.

very close to each other. Secondly, although Naïve require 0 `detect`, it will always require 64 `remove` operations whenever a new data word is written. On the contrary, although PW, DCW, and Flip-N-Write all need around 64 `detect` operations, the number of `remove` required by them are all few. Therefore, the trade-off between the costs of `detect` and `remove` operations makes none of the evaluated write strategies have distinct advantage in these two operations. As for the most expensive `inject` operation, the proposed PW strategy significantly reduces the number of inject by 85%, 54%, and 49% than that of Naïve, DCW, and Flip-N-Write, respectively. Such a great improvement is attributed to that PW re-uses the existing Skyrmions in the racetrack for minimizing the number of new Skyrmion injections. Moreover, the reduction in Skyrmion injection can also contribute to the reduction in unnecessary Skyrmion removals. This fact reveals why the PW strategy can also reduce the number of `remove` operations by about 50% than that of both DCW and Flip-N-Write.

5 Conclusion

Although Skyrmion racetrack memory (Sky-RM) is a new promising generation of racetrack memory (RM), the time-consuming and energy-hungry Skyrmion injection makes the data writes to Sky-RM inefficient. Therefore, in this article, we propose a new permutation-based write strategy, called *Permutation-Write (PW)*, to optimize the write performance and energy for Sky-RM by re-permuting the existing Skyrmions in the old data word to form the new written data word. Our realistic evaluations show that the PW strategy can significantly reduce the number of expensive injections by $50-80\%$ than other evaluated write strategies.

References

[1] S. Cho and H. Lee. Flip-n-write: A simple deterministic technique to improve pram write performance, energy and endurance. In *2009 42nd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Dec 2009.

[2] D. M. Crum, M. Bouhassoune, J. Bouaziz, B. Schweflinghaus, S. Blügel, and S. Lounis. Perpendicular reading of single confined magnetic skyrmions. *Nature Communications*, 2015.

[3] A. Fert, V. Cros, and J. Sampaio. Skyrmions on the track. *Nature Nanotechnology*, 2013.

[4] M. R. Guthaus, J. S. Ringenberg, D. Ernst, T. M. Austin, T. Mudge, and R. B. Brown. Mibench: A free, commercially representative embedded benchmark suite. In *Proceedings of the Fourth Annual IEEE International Workshop on Workload Characterization. WWC-4 (Cat. No.01EX538)*, Dec 2001.

[5] C. Hanneken, F. Otte, A. Kubetzka, B. Dupé, N. Romming, K. von Bergmann, R. Wiesendanger, and S. Heinze. Electrical detection of magnetic skyrmions by tunnelling non-collinear magnetoresistance. *Nature Nanotechnology*, 2015.

[6] W. Jiang, P. Upadhyaya, W. Zhang, G. Yu, M. B. Jungfleisch, F. Y. Fradin, J. E. Pearson, Y. Tserkovnyak, K. L. Wang, O. Heinonen, S. G. E. te Velthuis, and A. Hoffmann. Blowing magnetic skyrmion bubbles. *Science*, 2015.

[7] W. Kang, X. Chen, D. Zhu, X. Zhang, Y. Zhou, K. Qiu, Y. Zhang, and W. Zhao. A comparative study on racetrack memories: Domain wall vs. skyrmion. In *2018 IEEE 7th Non-Volatile Memory Systems and Applications Symposium (NVMSA)*, Aug 2018.

[8] W. Kang, Y. Huang, X. Zhang, Y. Zhou, and W. Zhao. Skyrmion-electronics: An overview and outlook. *Proceedings of the IEEE*, Oct 2016.

[9] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger. Architecting phase change memory as a scalable dram alternative. *SIGARCH Comput. Archit. News*, June 2009.

[10] Z. Liang, G. Sun, W. Kang, X. Chen, and W. Zhao. Zuma: Enabling direct insertion/deletion operations with emerging skyrmion racetrack memory. In *2019 56th ACM/IEEE Design Automation Conference (DAC)*, June 2019.

[11] L. Liu, C.-F. Pai, Y. Li, H. W. Tseng, D. C. Ralph, and R. A. Buhrman. Spin-torque switching with the giant spin hall effect of tantalum. *Science*, 2012.

[12] C.-K. Luk, R. Cohn, R. Muth, H. Patil, A. Klauser, G. Lowney, S. Wallace, V. J. Reddi, and K. Hazelwood. Pin: Building customized program analysis tools with dynamic instrumentation. In *Proceedings of the 2005 ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '05, New York, NY, USA, 2005. ACM.

[13] S. Mühlbauer, B. Binz, F. Jonietz, C. Pfleiderer, A. Rosch, A. Neubauer, R. Georgii, and P. Böni. Skyrmion lattice in a chiral magnet. *Science*, 2009.

[14] N. Nagaosa and Y. Tokura. Topological properties and dynamics of magnetic skyrmions. *Nature Nanotechnology*, 2013.

[15] S. S. P. Parkin, M. Hayashi, and L. Thomas. Magnetic domain-wall racetrack memory. *Science*, 2008.

[16] S. Raoux, G. W. Burr, M. J. Breitwisch, C. T. Rettner, Y. . Chen, R. M. Shelby, M. Salinga, D. Krebs, S. . Chen, H. . Lung, and C. H. Lam. Phase-change random access memory: A scalable technology. *IBM Journal of Research and Development*, July 2008.

[17] J. Sampaio, V. Cros, S. Rohart, A. Thiaville, and A. Fert. Nucleation, stability and current-induced motion of isolated magnetic skyrmions in nanostructures. *Nature Nanotechnology*, 2013.

[18] Y. Tchoe and J. H. Han. Skyrmion generation by current. *Physical Review B*, May 2012.

[19] B. Yang, J. Lee, J. Kim, J. Cho, S. Lee, and B. Yu. A low power phase-change random access memory using a data-comparison write scheme. In *2007 IEEE International Symposium on Circuits and Systems*, May 2007.

[20] D. Zhu, W. Kang, S. Li, Y. Huang, X. Zhang, Y. Zhou, and W. Zhao. Skyrmion racetrack memory with random information update/deletion/insertion. *IEEE Transactions on Electron Devices*, Jan 2018.

A novel detection method combined with VMD and GRU for cyberattacks in cyber-physical system

Wenfeng Deng, Zili Xiang, Keke Huang, Chunhua Yang
Central South University

Abstract

As the foundation and core of the 'Industrial 4.0', cyber-physical systems (CPSs) is experiencing unprecedented development. With deep integration of information technology, concerns for cybersecurity in CPSs have always been a vital research issue due to the expansion of system complexity and the enhancement of system openness. Recently, threats of cyberattacks have raised vigorously attentions from interdisciplinary and aroused massive efforts in the countermeasures against cyberattacks. Considering the multi-level relevant factors and the potential temporal correlation in the data collected by the SCADA system in CPS, this paper presents a hybrid data-based detection method combining with the variational mode decomposition (VMD) technology and the gate recurrent unit (GRU) network for the promote of cybersecurity. Case studies based on the IEEE-30 bus system, CSTH process and TE process have been conducted to verify the detection performance of the proposed method.

1 Introduction and Motivation

Recent years have witnessed great progresses in the development of Cyber-Physical Systems, termed as CPSs, including the broad applications in smart grids [1, 2, 3], modern water and chemical plants [4], transportation systems [5], oil/gas distribution systems [6], etc. Due to the use of uncountable smart meters, the integration of large-scale advanced information and intelligence technologies has deepened the mutual coupling between cyberspace and physical process, significantly promoting the real-time system states monitoring and supporting intelligent scheduling, but also making CPSs vulnerable to external attacks [7], especially the cyberattacks[8]. The cyberattacks injected by adversaries captures the control rights of control center by launching the network domain, and further affect the normal work of sensors, actuators and other components with the typical method of tampering with measurements and manipulating some specific system state estimations. Generally, types of cyberattacks can be divided into categories of denial-of-service (DOS) attacks [9], replay attacks [10], and deception attacks, also called by the false data injection attacks (FDIA) [11], which is an extremely deceptive attack with high concealment against State Estimation (SE) and can bypass traditional bad data detection methods. Once a cyberattack occurs, the failure to detect the attack will have a serious impact on CPS. For example, in 2014, a steel plant in Germany was attacked by a cyberattack, which forced the control components of the industrial control system and the entire production line to stop operation, causing huge economic losses [12]. So, with the increasing demand for security in national economy and people's lives, the protection of CPSs is attracting considerable attentions and has become one of the hot research issues in current focus [13].

In the past decades, there have been great efforts in the security protection of CPS against malicious attacks [14]. Most of the attack detection methods are based on system state and measurement data. Based on the system sates, the auto-regressive (AR) models is used to predict the short-term state of power system, and then an attack detection method is proposed based on it [15]. Reference [16] use the Kalman Filter estimator combined with the χ^2-detector and Euclidean detectors to detect the DOS attacks, replay attacks and FDIA. However, these methods have certain limitations, that is, it is difficult to estimate the states for some specific CPSs. On the other hand, to improve the applicability of attack detection technology, some detection methods based on measurement data have been widely developed. For instance, the problem of FDIA detection is transformed into a matrix separation in literature [17], and the convex optimization method was used for detection. Moreover, in order to improve the attack detection performance, the temporal correlation of system state is taken into account. [18] using wavelet transform and deep neural network techniques to detect FDIA. Considering that the wavelet transform is limited by the choice of mother wavelets and the advance of a new decomposition technique namely the variational mode decomposition (VMD),

[19] proposed a hybrid detection method of VMD and online sequential extreme learning machine. Besides, a new FDIA detection method based on long short-term memory Recurrent Neural Network (LSTM-RNN) is proposed in [20], and it proved to be able to detect different types of attacks with the simulation on Tennessee Eastman (TE) process. However, the limitation of ponderous model structure with complexity lacks of ideal solutions in the abovementioned algorithms. Practically, it is of great significance to predict the normal operation state of CPS accurately, but the realistic model is much more complicated due to various factors. Therefore, it is undoubtful that how to extract the operating features of CPS and provide support for system attack monitoring is urgent to be solved.

This paper presents a novel hybrid detection method for cyberattacks combining with method of variational mode decomposition (VMD) and gate recurrent unit (GRU), and the main procedure is summarized as follows. Firstly, in the offline training stage, the historical multi-source data derived from the SCADA system in CPS are collected and then are decomposed by the VMD method to extract potential features from the perspective of modes. Then, the deep learning method in the field of machine learning, GRU network is introduced to predict each mode and aggregate to the operating state of each device in CPS precisely. Afterwards, a kernel density estimation (KDE) method is utilized to determine the monitoring limit in accordance with the historical normal data. Thus, with the trained hybrid prediction model, the real-time attacked state of the system can be identified by our detection engine rapidly in the online detecting stage. To verify the effectiveness of the proposed method, three types of cyberattacks are analyzed and case studies based on the IEEE-30 bus system, CSTH process and TE process have been conducted.

The main contributions of this work include: (1) Considering the temporal correlation of dynamical states of CPS, this paper present a novel hybrid method of variational mode decomposition (VMD) and gate recurrent unit (GRU) in CPS against cyberattacks, which is applicable to various specific practical detection scenarios. (2) To balance the detection performance between the false alarm rate and false negative rate, a kernel density estimation method is employed to determine the monitoring limit of for attack detection. (3) Simulation experiments based on three kinds of datasets indicate the better detection performance of the proposed method and the distinguished advantages compared with other detection methods.

The reminder of this article is structured as follows. Section 2 introduces a brief overview of the preliminary knowledge, such as the basic attack model, the variational mode decomposition (VMD) technology and the gate recurrent unit (GRU) network, and illustrates the process of our proposed detection method. Section 3 demonstrates the simulation results with three types of datasets. Finally, Section 4 concludes this paper and provides some further discussions.

2 Methodology

2.1 System Model and Cyberattack

In CPS, a large number of intelligent instruments or sensors are installed in the physical layer to sense the whole system. Then the measurement data is transmitted to the control center for state estimation which is an important part of CPS. The estimated state provides real-time information and effective monitoring for the system to ensure the normal and safe operation of the system. At the same time, the information can be used for production management, energy management, etc. For example, in the power system, the results of state estimation can be applied to emergency analysis, optimal power flow analysis, economic dispatch, etc.

In this paper, to describe the system model more generally, we assume that the cyber-physical system is a nonlinear dynamic system. The nonlinear measurement equation can be constructed as follows [21]:

$$y = h(x) + e \tag{1}$$

where the measurement vector y denote the data collected by the field sensor. x represents the system states. e is the measured Gaussian noise with zero mean and covariance R. $h(\cdot)$ is the nonlinear relationship between the measurement and the state.

The traditional method of system state estimation is the weighted least square method [22]. The basic principle is to find state x based on measurement y by solving the following optimization function:

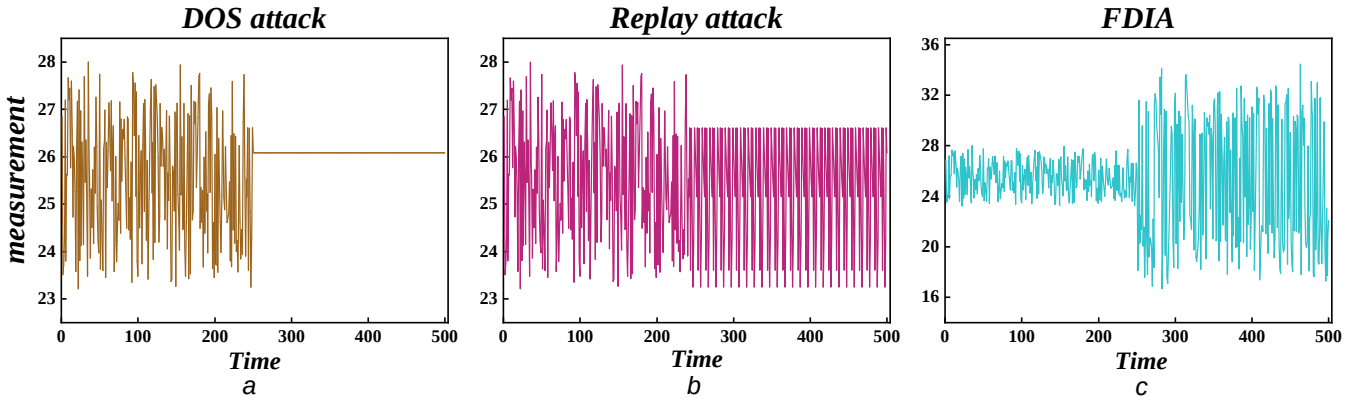


Figure 1: Typical types of cyberattacks in CPS. (a) State awareness under DOS attack, (b) State awareness under replay attack, (c) State awareness under replay attack.

$$\hat{x} = \arg\min_{x} [y - h(x)]^T W [y - h(x)] \tag{2}$$

where the weight matrix W is the diagonal matrix whose elements are the reciprocal of the variance of the noise e. Because of the vulnerability of CPS, it is more sensitive to malicious attacks, especially cyberattacks. As mentioned above, the cyberattacks can be divided into DOS attacks, replay attacks and false data injection attacks.

i) DOS Attacks

The purpose of DOS attack is to make the control center unable to obtain data and thus make system resources unavailable. Attackers usually jam the communication channels or change communication protocols to prevent data collected by field sensor from being transmitted to the control system. This will result in the failure of the system to collect real-time data so as to monitor the system in real time and lead to the safety accidents, economic losses and other serious consequences. Nowadays, the representative attack methods are PingofDeath, TearDrop, etc.

To describe DOS attacks, we assume that the system is attacked in time period T_a. The sensor data collected by the control center will be represented as follows [23]:

$$y_i'(t) = \begin{cases} y_i(t), & t \notin T_a \\ y_i^a(t), & t \in T_a \end{cases} \tag{3}$$

where $y_i(t)$ is the measurement of sensor i at time t and $y_i^a(t)$ is the collected modified data because of DOS attack. If the attack starts at time t_a, the $y_i^a(t)$ is defined as $y_i^a(t) = y_i(t_a - 1)$, which means the control center collected stale data namely the last value received by controller before attack, rather than new sensor measurements.

ii) Replay Attacks

The strategy of the replay attack is to maliciously repeat or delay the transmission of valid data to the control center. Therefore, two assumptions are necessary. One is that the attacker has the ability to obtain data form the smart meters over a period of time, and the other is that the attacker has the privilege to change the data collected by the sensors and transmit the false data to the control center. The attackers do not need to know the network structure information, communication protocol and so on, only need to modify the data of the smart meters to achieve the attack, which means the cost of attack is low. The attack may affect the transmission delay of the normal data stream, consume the bandwidth of the communication line, and cause physical damage to the system, etc.

The replay attack process is divided into two stages. In the first stage, it is assumed that the attacker acquires the sensor measurements and choose the data of time window size T (in time interval $[t_o, t_f]$) as replay attack data. In the second stage, the attacker decides whether to change the data collected by the sensor to achieve the attack at each time-step, and the data received by the control center can be described as:

$$y'_i(t) = y_i(t - k + t_0) \tag{4}$$

where $y_i'(t)$ is the data received by the control center at time t of sensor i. $y_i(t-k+t_0)$ means that when attack occurs at time k, the modified data received by the controller at time t in a time-step.

iii) False Data Injection Attacks

As a kind of malicious attack with concealment, false data injection attack (FDIA) can bypass the traditional bad data detection method. This means that FDIA is more harmful than the above two kinds of attacks, causing serious impacts such as economic losses and security accidents, and therefore receive a lot of attention, especially in the field of smart grid [24]. It requires the attacker to obtain the permission of the smart meter to inject the set attack data into the normal data. Of course, designing undetected attack vector is very challenging for attackers. First, the attacker needs to attack the system to obtain the complete model or part of the model, and then construct the attack vector a based on the system model. In order to describe the impact of the FDIA on the system state, the deviation of system state c is introduced. When the attack vector meets the condition $a = h(c)$, the attack can bypass the bad data detection method. In this case, the measurement residual can be expressed as:

$$\begin{aligned} \|y_a - h(x_{bad})\|_2 &= \|y + a - h(x+c)\|_2 \\ &= \|y - h(x) + a - h(c)\|_2 \\ &\leq \|y - h(x)\|_2 + \|a - h(c)\|_2 \\ &= r + \tau \end{aligned} \tag{5}$$

where y_a is the measurement data collected by control center after attack. x_{bad} is the result of state estimation based on y_a. If τ=0, or $\tau < r - \|y - h(x)\|_2 - \|a - h(c)\|_2$, the attack can bypass the bad data detection, which misleads the control center that the true state of the power system is x_{bad}.

The attack mechanisms and the corresponding effects of three types of attacks are shown in Fig. 1. The subfigure (a) in Fig. 1 represents the state awareness of CPS under DOS attack, while subfigures (b) and (c) under replay attack and FDIA respectively, where the system is affected at the 251th moment. It can be seen that all kinds of attacks will cause the measurements of the system to deviate from the normal state.

2.2 Mode Decomposition with VMD

Cyberattack initiated by adversities, no matter the DOS attack, replay attack or FDI attack, are ultimately achieved by breaking through the authority identification in the cyberspace [25], such as protocol vulnerabilities. It modifies the command content between the control center and the physical process and the attacking effect can be eventually reflected in the measurement sequence from the SCADA system, where the inherent temporal sequential correlation will be destroyed under normal operation. To some extent, identifying the behavior of cyberattack is essentially a special form of pattern recognition, which can be potentially mined through the collected data [26]. In recent years, the signal decomposition technology derived from the signal processing community is widely used in the field of pattern recognition, including the anomaly diagnosis [27], sequence prediction [28], etc., promoting the decomposition of measurement data to be an effective tool.

Actually, with signal decomposition technology, the original signal can be decomposed into a series of band-limited sub-signals, also called by modes, in different frequency domain, which is represent by IMFs (Intrinsic Mode Functions) with specific bandwidth. The abnormal behavior on the sequence can always be reflected in the temporal correlation in a certain mode and it is suitable for the analysis of aggressive behaviors with strong concealment. Therefore, a non-recursive signal processing algorithm of VMD (variational mode decomposition) first proposed by Dragomiretskiy and Zosso [29] is employed to our detection framework owing to the advantages in dealing with non-linear dynamical signal. During the decomposition process, to compact each mode u_k around a corresponding center pulsation ω_k, the bandwidth of each mode can be fulfilled by the following steps:

Step 1: Adopt Hilbert transform to attain the unilateral frequency spectrum for each analytic IMF u_k.

Step 2: Transform the frequency spectrum to the baseband regions by applying an exponential tuned to the respective estimated center frequency.

Step 3: Estimate the each bandwidth through the H^1 Gaussian smoothness of the demodulated signal.

Thus, searching for the IMF u_k and the center pulsation ω_k can be transformed to the constrained variational problem constructed as follows:

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) \otimes u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \tag{6}$$

Subject to,

$$s.t. \sum_{k=1}^{K} u_k = f(t) \tag{7}$$

where $f(t)$ represents the original signal to be decomposed, and $u_k := \{u_1, u_2, ..., u_K\}$ and $\omega_k := \{\omega_1, \omega_2, ..., \omega_K\}$ represents the sets of all modes and the corresponding center frequencies, wherein K denotes the total number of modes. Besides, δ is the Dirac distribution, $\otimes$ is the convolution operator and t stands for the time stamp. Classically, it is assumed that the IMF with high order denotes low-frequency component.

For complicity, the above mentioned constrained optimization problem can be resolved by combining the quadratic penalty term and the Lagrangian multipliers, which is converted into an unconstrained optimization problem expressed as follows [30]:

$$\begin{aligned} L(u_k, \omega_k, \lambda) := \alpha \sum_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) \otimes u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \\ + \left\| f(t) - \sum_{k=1}^{K} u_k(t) \right\|_2^2 \\ + \left\langle \lambda(t), f(t) - \sum_{k=1}^{K} u_k(t) \right\rangle \end{aligned} \tag{8}$$

here $\lambda(t)$ stands for the Lagrangian multipliers for tightening restraint while $f(t) - \sum_{k=1}^{K} u_k(t)$ stands for the quadratic penalty term for accelerating the convergence velocity, and α is a penalty parameter.

The solution to the unconstrained problem in Eq. (8) can be found by employing the so-called Alternate Direction Method of Multipliers (ADMM) [31] to find the saddle point of the previously derived augmented Lagrangian, where the mode u_k and the center frequency ω_k can be updated in two directions to achieve the analysis of VMD. To be concrete, the updating iterative form of u_k and ω_k is shown as follows:

$$u_k^{n+1} := \arg\min_{u_k} L\left(\{u_{i<k}^{n+1}\}, \{u_{i\geq k}^{n+1}\}, \{\omega_i^n\}, \{\lambda^n\} \right) \tag{9}$$

$$\omega_k^{n+1} := \arg\min_{\omega_k} L\left(\{u_i^{n+1}\}, \{\omega_{i<k}^{n+1}\}, \{\omega_{i\geq k}^{n}\}, \{\lambda^n\} \right) \tag{10}$$

$$\lambda^{n+1} := \lambda^n + \tau \left(f(t) - \sum_{k=1}^{K} u_k^{n+1} \right) \tag{11}$$

A convergence condition should be defined in advance as

$$\sum_{k=1}^{K} \frac{\left\| u_k^{n+1} - u_k^n \right\|_2^2}{\left\| u_k^n \right\|_2^2} < \varepsilon \tag{12}$$

herein n represents the number of iteration steps and τ denotes the time stamp of the dual ascent. Thus, the solutions for u_k and ω_k are represent by:

$$\hat{u}_k^{n+1} = \frac{\hat{f}(\omega) - \sum_{i\neq k} \hat{u}_i(\omega) + \frac{\hat{\lambda}(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \tag{13}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega}{\int_0^\infty \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega} \tag{14}$$

where $\hat{f}(\omega)$, $\hat{\lambda}(\omega)$, $\hat{u}_i(\omega)$ and $\hat{u}_k^{n+1}(\omega)$ are the Fourier transforms of $f(\omega)$, $\lambda(\omega)$, $u_i(\omega)$ and $u_k^{n+1}(\omega)$ respectively.

The diagram of the VMD decomposition method is shown in Fig. 2, which can intuitively display the performance of multiple modes decomposed by the time series. It is noted that the number of modes depicted as K in the formula should be predefined in the application of VMD during the decomposition process. An appropriate selection of K is of great significance, which is also part of our experimental discussion.

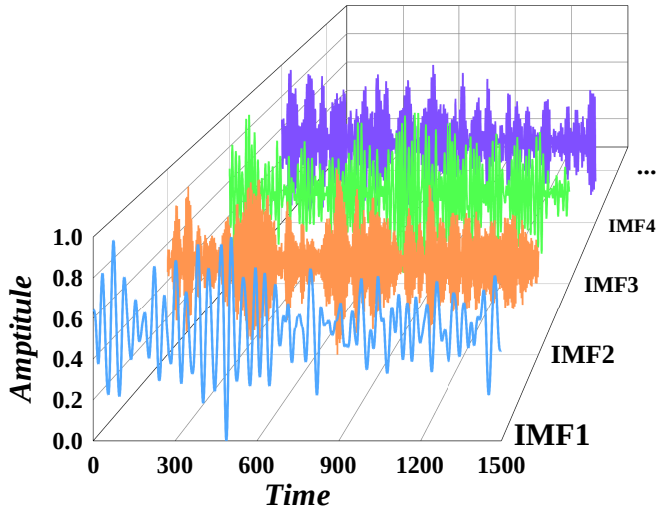


Figure 2: Diagram of VMD decomposition method.

2.3 Prediction with GRU

As an important branch of machine learning, deep learning algorithm has been widely used in short and long-term forecasting in recent years. Deep learning algorithm aims to extract deeper features hidden in large amounts of data through deep neural network structure to study the future development trend. Recurrent Neural Network (RNN) is an artificial neural network model with memory ability, which is very suitable for describing the relationship between current moment data and historical sequence data. However, this model has certain limitations, that is, when the predicted results are far away from the relevant information, gradient vanishing or gradient exploding problem will occur during the network training. In order to solve the problems of long-term memory and gradient explosion, LSTM was proposed in literature [32]. As a variant of LSTM, GRU is widely used for predicting because it performs similarly to LSTM but is computationally cheaper [33].

Unlike the LSTM, GRU introduces reset gate r_t and update gate z_t to solve the problem of long-term dependence of data. The two gates are obtained by the precious hidden state h_{t-1} and the current input x_t:

$$r_t = \sigma\left(W_r \cdot [x_t, h_{t-1}] + b_r\right) \tag{15}$$

$$z_t = \sigma\left(W_z \cdot [x_t, h_{t-1}] + b_z\right) \tag{16}$$

where σ (sigmoid) is a type of activation function that maps data to a range of 0-1 to generate gate control signals. W_r and W_z are the weight matrix, b_r and b_z are the bias vector.

Then the candidate hidden state is introduced to prepare for the next hidden state calculation. The candidate hidden state $\tilde{h}_t$ at time point t is calculated as follows:

$$\tilde{h}_t = \tanh\left(W_h \cdot [x_t, r_t \odot h_{t-1}] + b_h\right) \tag{17}$$

where $\odot$ is the element-wise product of vectors. The reset gate r_t can control the flow of the hidden state of the previous time to the candidate hidden state at the current time point, thus discarding historical information that is not related to prediction.

Finally, the update gate z_t is used to selectively remember and forget historical and current information at same time, and the hidden state at time point t, namely the h_t is updated by:

$$h_t = (1 - z_t) \odot \tilde{h}_t + z_t \odot h_{t-1} \tag{18}$$

The framework of the GRU prediction model is shown in Fig. 3.

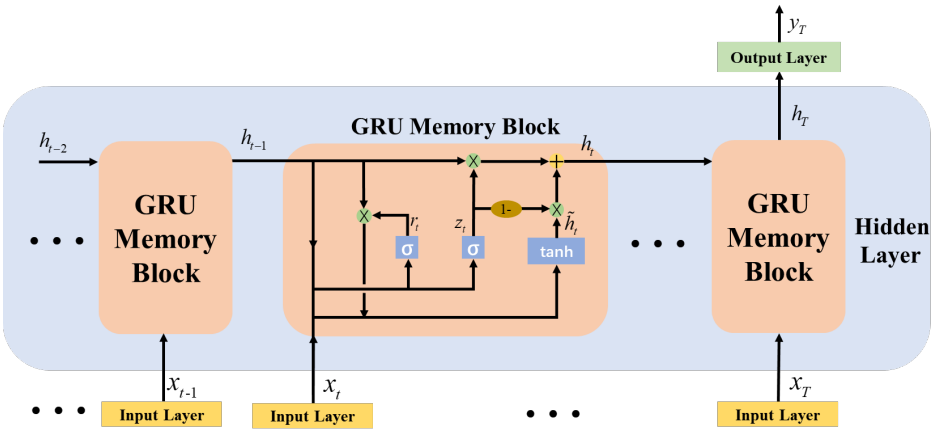


Figure 3: The framework of GRU prediction model.

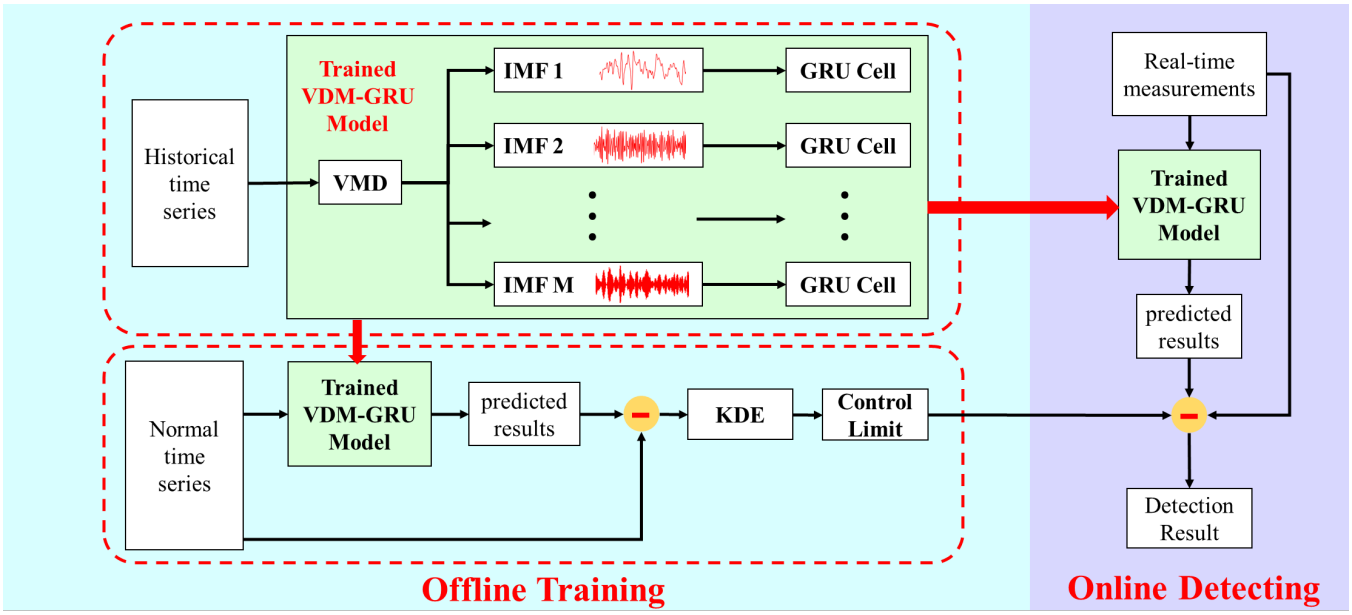


Figure 4: Overall flow chart of VMD-GRU attack detection method

2.4 VMD-GRU detection method

Because the measurements in CPS have the characteristics of nonlinear, time correlation and high fluctuation, it is difficult to detect attack behaviors according to the measurements directly. In view of the advantages of VMD and GRU, this paper proposed a novel attack detection method combining both of them. For simplicity, we refer to it as the hybrid VMD-GRU detection model. We analyze the data collected by each sensor independently, which can reduce the computational complexity caused by large network and can quickly detect the attack location. The proposed hybrid VMD-GRU attack detection model is shown in Fig. 4.

Assume the historical time series data of sensor i without attacks $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iT}\}$ as input for prediction. Then the VMD method decomposes the original time series data x_i into several different modes u_{ik}, $k = 1, 2, \cdots, M$, where M is the total number of decomposition modes. By using VMD to decompose time series data into different modes, the characteristics of normal data can be analyzed from different scales. In this way, the data characteristics affected by various factors can be mined more deeply, so as to improve the accuracy of attack detection. For the different modes obtained by VMD decomposition technology, the GRU network was used for model training respectively. In view of the excellent performance of GRU in processing time series data prediction and the cheaper

computational cost, its model framework is suitable for the prediction work here. The next work is to set the monitoring limit to prepare for detecting whether the test data is attacked. The most common method to set the limit is kernel density estimation (KDE). Firstly, a period of normal historical time series data should be selected and input into the trained GRU model to output the prediction results. Then the absolute value of the residual between the actual value and the predicted value is use to obtain the monitoring limit by KDE. The next step is to use the VMD-GRU prediction method for attack detection. Only the residual error between the predicted result and the actual measured value is calculated, and compared with the monitoring limit to determine whether there is an attack. Besides, as we analyzed the measurement of each sensor separately, and for sensor i, the specific algorithm description is presented in algorithm 1.

Algorithm 1 The hybrid VMD-GRU attack detection method

Input: Historical time series measurements without attack x_i, a period of time series data x_{icl} without attack used to calculate the monitoring limit, the detected time series measurements x_{itest}
Output: The attack detection results
#Execution
Step 1: The initial value of the number of modes K is set randomly, and the intrinsic mode functions is calculated based on Eq. (13) and (14) and VMD decomposition steps shown in section 2.2, so as to obtain each mode of the original data x_i.
Step 2: Calculate the ratio of residual energy according to Eq. (19) to determine the appropriate number of modes K, and then get the final decomposition mode u_{ik}, $k = 1,2,...,K$.
Step 3: Each mode u_{ik} is slid into different data blocks according to the set sliding window size as the input of the training model.
Step 4: For each mode, GRU models will be trained respectively according to the data blocks obtained in Step 3.
#Monitoring Limit Calculating
Step 5: According to procedures mentioned Step 1 and Step 2, x_{icl} is decomposed into different modes u_{iclk} by VMD.
Step 6: For each mode, input the modal data to the prediction model and get the modal forecasted results y_{ik}, $k = 1,2,\cdots,M$.
Step 7: The prediction results of each mode are summarized to generate the final prediction results $y_i = \sum_{k=1}^{M} y_{ik}$.
Step 8: Calculated the residual between the predicted result and the actual measurement and input it to Kernel Density Estimation (KDE) to obtain the monitoring limit.
#Attack Detection
Step 9: Calculate the predicted results for the test data x_{itest} in the same way as Steps 5 to 7.
Step 10: Calculate the residual between the predicted result and the actual measurement, and compare with the monitoring limit to determine whether there is an attack.

3 Experiments

3.1 Dataset Statement and Evaluation Index

To verify the outstanding performance of the proposed method in cyberattack detection, three representative datasets from different industries are utilized in our simulation, i.e., the IEEE standard-30 bus dataset, CSTH and TE process.

i) IEEE Standard Power System

With the development of power system CPS, it becomes the development trend of power system in the future because CPS can greatly improve the level of automatic management [34, 35]. However, it also brings a series of problems. Due to frequent transmission of information, the power system becomes more vulnerable to various attacks. Cyberattack may bring economic loss, security accident and other serious consequences, so the attack detection in the power system has been widely concerned and studied. Most of the work focuses on FDIA in power systems. Here we will analyze the detection performance of the proposed methods under three different attacks. In

this paper, the IEEE standard power system in MATPOWER toolbox is selected as the research object, which was developed by the Power System Engineering Research Center at the Cornell University [36]. This paper used the IEEE standard 30 bus network for experiments. In order to simplify the model, the DC power flow model is used for system state analysis. We will analyze the detection accuracy of the power data under attack on each bus. Because our method is to detect each sensor data separately, we randomly select the attack position to carry out three attacks and analyze the detection accuracy.

ii) CSTH process

The continuous stirred tank heater (CSTH) process is a typical industrial reaction process which is widely used as a dataset to study industrial process control, state recognition and fault diagnosis [37]. The continuous stirred tank heater is located in the Department of Chemical and Materials Engineering at the University of Alberta, which has a circular cross section with a volume of 8l and height of 50 cm. In the stirrer tank heater, the hot and cold water are stirred thoroughly, and the mixture is heated by steam to the set temperature before draining off from the bottom of the autoclave through a long pipe. Due to the complexity of the reaction process and the diversity of data distribution, it is difficult to detect after being attacked. If the system is attacked, the feedback data in the control process will deviate too much from the actual state, which will lead to a serious deviation from the stable state in the reaction process, thus causing security accidents. In this paper, the CSTH simulation model established on the Simulink platform in reference [38] was used for experiments. A mode of CSTH is selected here for the simulation experiment, in which the mode parameter is set as the level setpoint 12, the temperature setpoint 12 and the hot water valve setpoint 5.5. In order to ensure the authenticity of the simulation model, a certain amount of noise was added into the simulation model, namely, the oscillation interference to the cold water flow, the random interference to the liquid level and the measurement noise of the temperature. These three types of noise are the measurements collected from the CSTH laboratory at the University of Alberta, thus adding these noises to the simulation model can make the operation of the system more credible. Firstly, the simulation was carried out according to the normal state parameter setting in the set mode. Then, one of the three different attacks are designed in the temperature and liquid level dimension to obtain attacked data.

iii) TE Process

Tennessee Eastman (TE) platform is a standard simulation platform developed by Eastman Chemical Company based on actual chemical reaction processes [39]. On this platform, it has five main operating units: an exothermic two-phase reactor, a vapor-liquid flash separator, a product condenser, a recycling compressor, and a product stripper. The gaseous reactant is first cooled into a liquid by a condenser and then sent to a vapor-liquid flash separator. The steam from the separator is recycled into the reactor through a centrifugal recycling compressor, and the separated liquid components are further purified by the stripping tower. And then products G and H are obtained at the bottom of the stripping tower, while the remaining reactants at the top enter the reactor for recycling. Due to the time-varying, strong coupling and nonlinear characteristics of the generated data, a large number of researches used its data and models for control, optimization, process monitoring and fault diagnosis. There are several reasons why this article chooses the TE process to study cyberattacks. Firstly, the research on TE process has been very mature. Secondly, the TE model process is a typical industrial model, so it is of practical significance to study on its attack detection. Finally, TE process has the characteristics of high complexity, nonlinearity and time correlation, which is suitable to verify the advantages of the proposed method. A large number of sensors, including temperature sensors, pressure sensors, liquid level sensors, etc. monitor and control TE process. The TE process contains 12 manipulated variables, and 41 measured process variables, of which 22 are continuously monitored process variables and 19 are composition measurements. The normal data is obtained through simulation on the Simulink platform, and the design of three different cyberattacks are designed to attack the 7th, 8th, and 9th dimensions of the continuously monitored process variables, namely reactor pressure, reactor level and reactor temperature.

Here, in order to quantify the performance of cyberattack detection, four indicators derived from the confusion matrix of the two-class classification problem shown in Table. 1 are employed in the part, including the Accuracy, Precision, Sensitivity, and Specificity, which are referred as ACC, PPV, TPR and TNP and are defined as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}$$

$$PPV = \frac{TP}{TP+FP} \tag{20}$$

$$TPR = \frac{TP}{TP+FN} \tag{21}$$

$$TNR = \frac{TN}{TN+FP} \tag{22}$$

where TP, TN, FP, FN are specifically stated in Table. 1. Consequently, with these quantitative indicators, the detection performance for the model can be evaluated from different perspectives. The specific usage will be described in detail in the following experiments.

Table 1: Confusion matrix of two-class classification problem

		Actual Label	
		Target Class	Negative Class
Predicted Label	Target Class	TP	FP
	Negative Class	FN	TN

3.2 Simulation Results

The VMD decomposition method is employed in the construction of the proposed detection model. Modes decomposed into the IMFs which consist of significant information is critical to mitigate the non-stationary and non-linear characteristics and extract the important features of abnormal behaviors. Before the formal decomposition via VMD, the number of modes decomposed in VMD should be confirmed first as it is crucial to balance the optimal modal analysis and the unnecessary computational burden in the following experiments. Here, in order to determine the optimal number of IMFs, the ratio of residual energy r_res presented in [40] is introduced, which is defined as follows:

$$r_{res} = \frac{1}{M}\sum_{t=1}^{M}\left|\frac{f(t)-\sum_{k=1}^{K}u_k(t)}{f(t)}\right| \tag{23}$$

where $f(t)$ represents the original measurement series of devices in CPS while $u_k(t)$ represents the k-th IMF at time t. Besides, K stands for the number of IMFs to be decomposed and M stands for the total number of measurements. For clarity, the power measurement sequence in the 4-th bus of IEEE30 bus system is decomposed into the range of 2 to 20 and the corresponding ratio of residual energy is present therein. Obviously, the residual energy under different number of modes shown in Fig. 5 demonstrated that with the increasing decomposed components, the information of the original sequence is preserved more fully in the independent components of different frequency domains, which is very helpful for signal decomposition and analysis. However, the computational overhead of complex decomposition process can not be ignored. With fitting results shown in the subgraph of Fig. 5, it is believed that an appropriate balance lies on the number of 10 since a large amount of critical information in the sequence have been contained with an acceptable calculation requirements, which is beneficial to the training process

In order to comprehensively verify the detection performance of the proposed model against cyberattacks, three kinds of datasets will be analyzed in this subsection. Among them, the attack scenarios are selected according to the attacks that the system is vulnerable to, such as those vulnerable to false data injection attack in the power grid, but not limited to this. In fact, as the mean of malicious behavior, cyberattack is easy to invade to CPS through the cyberspace. In order to illustrate the situation of attack detection, the residual error between the predicted value and the measured value of the system operation state as well as the monitoring limit obtained by off-line training are used for comparison and judgment. System sequence containing 300 sampling data are selected, among which the first 250 are real-time normal data. The cyberattack starts to invade in the data sample in the 251th operation instant, and continues to destroy in the subsequent sequence. As can be seen in Fig. 6, the method proposed in this paper is

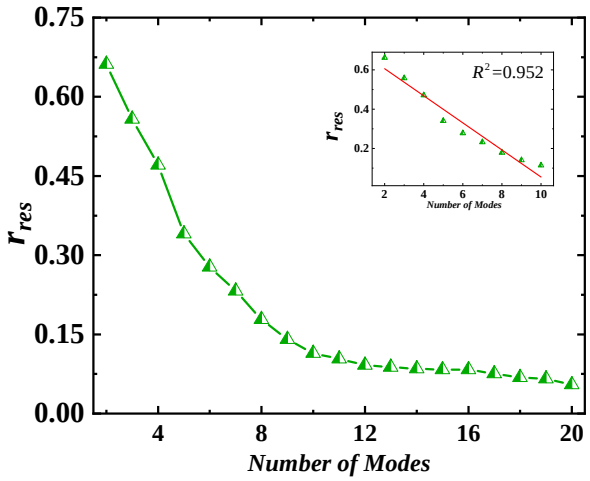


Figure 5: The ratio of residual energy with different number of modes.

excellent at attack detection level, and can reflect the impact of abnormal behavior on the system at the first moment under different datasets. Although there are missed judgments in the following 20 or even 30 moments, in the actual system, the operator usually pays more attention to the first moment of the attack, so as to make timely mitigation measures and improve preparedness. Therefore, by achieving a perfect balance between false positive rate and false negative rate, the proposed model can be used in the actual attack detection of CPS.

Furthermore, in order to quantitatively demonstrate the detection accuracy of the proposed detection method for attacks, we introduce four detection indexes which are described above. Compared with the other two kinds of attacks, FDIA is more threatening due to its great concealment and destruction ability, which has attracted extensive attention and research. Therefore, here FDIA is selected as the research object to analyze the detection effect. Of course, the detection effect of the other two attacks is also verified. We select 80% of the normal historical data as the training set to train the model, and calculated the monitoring limit with 20% of the data. Two different types of test set data are collected, which are normal data and attacked data, to analyze whether our detection method can have low false positive rate and false negative rate. The results are shown in Fig. 7. It is obvious from the histogram that our method can detect the attacked data accurately and avoid large false positives in the normal dataset. Taking IEEE 30 standard bus power network dataset as an example, the four indexes of our method are: 1, 0.988, 0.97656, 0.976. That is because our detection method can accurately predict the data of the next moment under normal

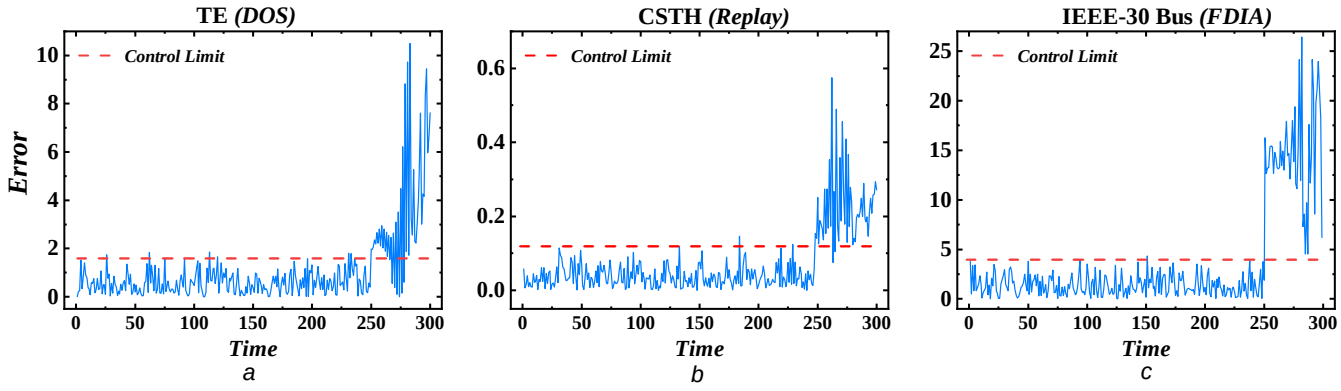


Figure 6: Detection performance against three types of cyberattacks on three datasets with the proposed method. (a) Detection performance in the TE process where the DOS attack invades the system at 251th moment. (b) Detection performance in the CSTH process where the replay attack invades the system at 251th moment. (c) Detection performance in the IEEE-30 bus system where the FDIA invades at 251th moment.

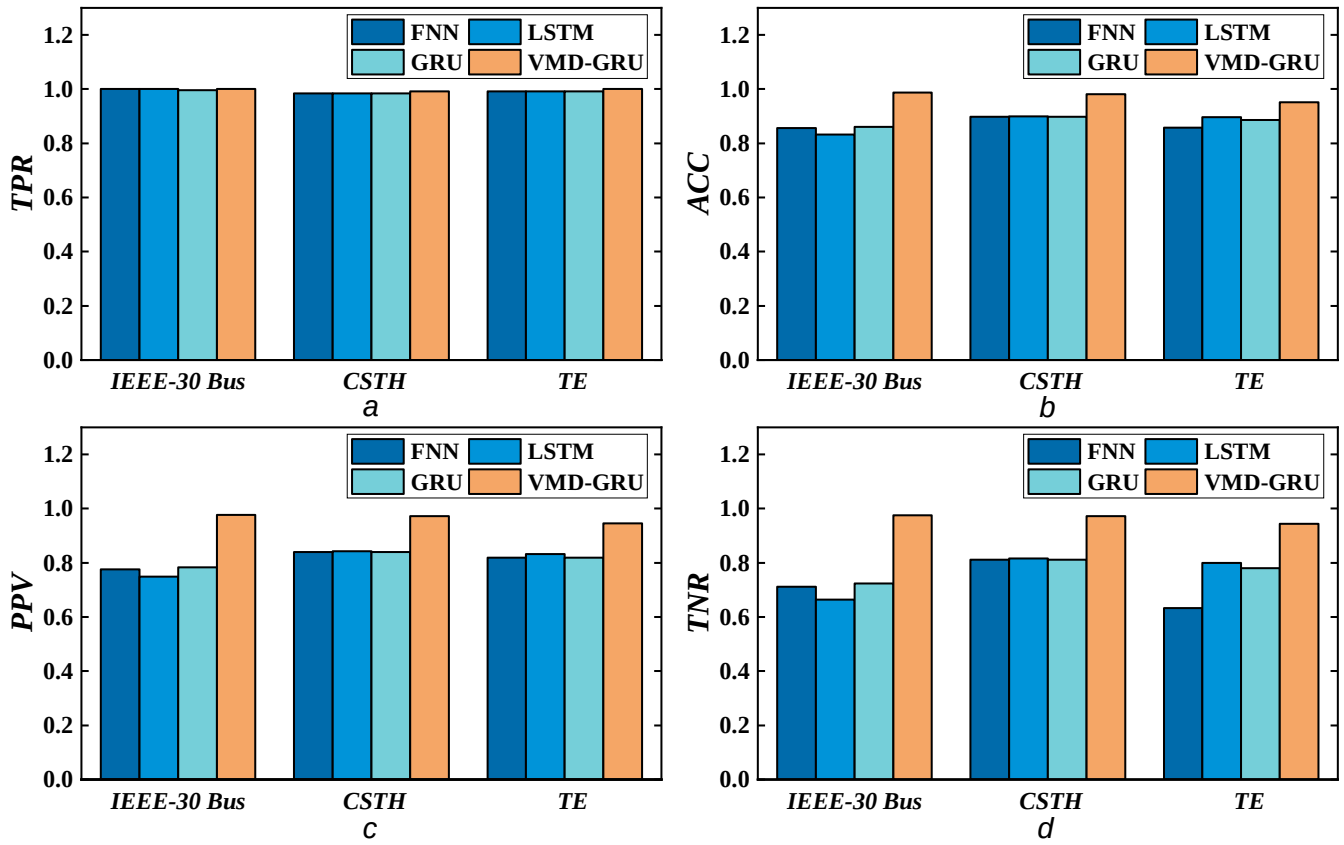


Figure 7: Comparison of different detection algorithms under the cyberattack of FDIA on three datasets. (a) Sensitivity of different detection algorithms on three types of datasets. (b) Accuracy of different detection algorithms on three types of datasets. (c) Precision of different detection algorithms on three types of datasets. (d)Specificity of different detection algorithms on three types of datasets.

circumstances based on the normal historical data, so as to determine whether the system has been attacked. To illustrate the generality of this approach, three different datasets are used, namely the IEEE 30 standard bus power systems, CSTH, and TE processes. Similarly, it can be seen from the histogram that the proposed method has accurate detection for three kinds of datasets, whether attack detection or normal data detection. The specific reason is that although the three industrial models are different, they all have the characteristics of high complexity, nonlinear and time correlation, etc. Considering these characteristics, the proposed hybrid detection model has excellent predictive ability and can detect attacks well. Moreover, in order to better demonstrate the advantages of the proposed method, the common learning prediction model is selected for comparison. Through the comparison of the four indexes, it can be found that the proposed method has certain advantages. This is because the original signal is easily affected by various factors and fluctuates sharply, it is difficult to use a single mode to directly extract and analyze the law of change for prediction. The analysis of different modes through VMD decomposition can reduce the interference between different scales of information in the original data, thus making the prediction better, so as to improve the detection accuracy.

4 Conclusion and Discussion

The cybersecurity of CPS has become an important but challenging issue, owing to the expansion of association complexity and the enhancement of system openness. Practically, the increasing prominent coupling between the cyberspace and physical process have forced CPS to be vulnerable and easily exposed to the threat of cyberattack. Generally, cyberattacks launched by adversaries aim to intercept the control authority of control center illegally and

further invade to the physical world and process from the cyberspace, so as to achieve malicious target. Usually, attacks can lead to catastrophic cascading consequences, resulting in a significant loss of CPS. Fortunately, the historical measurement data can play a valuable role in attack detection since the affect of abnormal behaviors can be reflected in the data level. Considering the potential temporal correlation of the system state during the normal operation as well as the highly complexity and dynamics of external factors in reality, a hybrid data-based detection method combining variational mode decomposition (VMD) technology and gate recurrent unit (GRU) network for identifying the cyberattacks in CPS, where VMD is used for mode decomposition and GRU for state prediction. Specifically, in the offline training stage, the VMD method is first employed to decompose the historical system operation data from devices in SCADA system into reasonable modes in different frequency domains. Then, the deep learning method in the field of machine learning, GRU network is introduced to predict each mode and aggregate to the operating state of each device in CPS precisely. Next, a kernel density estimation (KDE) method is utilized to determine the monitoring limit in accordance with the historical normal data. Thus, with the trained hybrid prediction model, the real-time attacked state of the system can be identified rapidly in the online detecting stage, helping operators detect attacks in advance and take preventive measures timely. Three types of cyberattacks are analyzed and case studies based on the IEEE-30 bus system, CSTH process and TE process have been conducted to verify the detection performance of the proposed method. On the whole, this paper provides in-depth insights for cybersecurity of CPS.

The main contribution of this work can be summerized as follows:

(1) Considering the temporal correlation of dynamical states of CPS, this paper present a novel hybrid method of variational mode decomposition (VMD) and gate recurrent unit (GRU) in CPS against cyberattacks, which is applicable to various specific practical detection scenarios.

(2) To balance the detection performance between the false alarm rate and false negative rate, a kernel density estimation method is employed to determine the monitoring limit of for attack monitoring.

(3) Simulation experiments based on three kinds of datasets indicate the better detection performance of the proposed method and the distinguished advantages compared with other detection methods.

However, the method proposed in this paper still has some limitations and is worth further discussion and improvement. The determination of the number of mode in VMD decomposition technology is an open problem. Although there have parts of determination principles for modes in Refs. [19, 40, 41], it presents strong bias in the definition according to the affecting factors understood by individuals and lacks of a consistent support of interpretation. In addition, the role of deep learning framework as a black box has always been a hot topic among scientific community, and it is worth pondering how to provide appropriate theoretical support to explain its predictive ability. Thirdly, although the detecting operation of the proposed model can be completed quickly in this paper, it inevitably requires a lot of computing resources for CPS with larger scale. Finally, for the future research, it is necessary to study the specific impact mechanism of cyberattacks on physical processes with integration method. Besides that, considering the impact of multi-source heterogeneous data and mining the spatio-temporal relationship of multidimensional sequence data for attack detection is an valuable guidance for our future research.

References

[1] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—the new and improved power grid: A survey," *IEEE communications surveys & tutorials*, vol. 14, no. 4, pp. 944–980, 2011.

[2] Z. Xiang, K. Huang, W. Deng, and C. Yang, "Blind topology identification for smart grid based on dictionary learning," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2019, pp. 1319–1326.

[3] S. Karnouskos, "Cyber-physical systems in the smartgrid," in *2011 9th IEEE international conference on industrial informatics*. IEEE, 2011, pp. 20–23.

[4] D. Gollmann, P. Gurikov, A. Isakov, M. Krotofil, J. Larsen, and A. Winnicki, "Cyber-physical systems security: Experimental analysis of a vinyl acetate monomer plant," in *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*, 2015, pp. 1–12.

[5] L. Yongfu, S. Dihua, L. Weining, and Z. Xuebo, "A service-oriented architecture for the transportation cyber-physical systems," in *Proceedings of the 31st Chinese Control Conference*. IEEE, 2012, pp. 7674–7678.

[6] Y. Wadhawan and C. Neuman, "Evaluating resilience of oil and gas cyber physical systems: A roadmap," in *Annual Computer Security Application Conference (ACSAC) Industrial Control System Security (ICSS) Workshop*, 2015.

[7] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE transactions on automatic control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[8] M. S. Mahmoud, M. M. Hamdan, and U. A. Baroudi, "Modeling and control of cyber-physical systems subject to cyber attacks: a survey of recent advances and challenges," *Neurocomputing*, vol. 338, pp. 101–115, 2019.

[9] S. Liu, X. P. Liu, and A. El Saddik, "Denial-of-service (dos) attacks on load frequency control in smart grids," in *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*. IEEE, 2013, pp. 1–6.

[10] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *2009 47th annual Allerton conference on communication, control, and computing (Allerton)*. IEEE, 2009, pp. 911–918.

[11] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, pp. 1–33, 2011.

[12] R. M. Lee, M. J. Assante, and T. Conway, "German steel mill cyber attack," *Industrial Control Systems*, vol. 30, p. 62, 2014.

[13] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—a survey," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1802–1831, 2017.

[14] D. Ding, Q.-L. Han, Y. Xiang, X. Ge, and X.-M. Zhang, "A survey on security control and attack detection for industrial cyber-physical systems," *Neurocomputing*, vol. 275, pp. 1674–1683, 2018.

[15] J. Zhao, G. Zhang, M. La Scala, Z. Y. Dong, C. Chen, and J. Wang, "Short-term state forecasting-aided method for detection of smart grid general false data injection attacks," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1580–1590, 2015.

[16] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Combating false data injection attacks in smart grid using kalman filter," in *2014 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2014, pp. 16–20.

[17] L. Liu, M. Esmalifalak, and Z. Han, "Detection of false data injection in power grid exploiting low rank and sparsity," in *2013 IEEE international conference on communications (ICC)*. IEEE, 2013, pp. 4461–4465.

[18] J. James, Y. Hou, and V. O. Li, "Online false data injection attack detection with wavelet transform and deep neural networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3271–3280, 2018.

[19] C. Dou, D. Wu, D. Yue, B. Jin, and S. Xu, "A hybrid method for false data injection attack detection in smart grid based on variational mode decomposition and os-elm," *CSEE Journal of Power and Energy Systems*, 2020.

[20] W. Wang, Y. Xie, L. Ren, X. Zhu, R. Chang, and Q. Yin, "Detection of data injection attack in industrial control system using long short term memory recurrent neural network," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2018, pp. 2710–2715.

[21] Y. Li, W. Huo, R. Qiu, and J. Zeng, "Efficient detection of false data injection attack with invertible automatic encoder and long-short-term memory," *IET Cyber-Physical Systems: Theory & Applications*, vol. 5, no. 1, pp. 110–118, 2020.

[22] J. Mandel, *The statistical analysis of experimental data*. Courier Corporation, 2012.

[23] M. Krotofil, A. A. Cárdenas, B. Manning, and J. Larsen, "Cps: driving cyber-physical systems to unsafe operating conditions by timing dos attacks on sensor signals," in *Proceedings of the 30th Annual Computer Security Applications Conference*, 2014, pp. 146–155.

[24] S. Wang, S. Bi, and Y.-J. A. Zhang, "Locational detection of false data injection attack in smart grid: a multi-label classification approach," *IEEE Internet of Things Journal*, 2020.

[25] N. Ye, Y. Zhang, and C. M. Borror, "Robustness of the markov-chain model for cyber-attack detection," *IEEE Transactions on Reliability*, vol. 53, no. 1, pp. 116–123, 2004.

[26] H. Karimipour, A. Dehghantanha, R. M. Parizi, K.-K. R. Choo, and H. Leung, "A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids," *IEEE Access*, vol. 7, pp. 80 778–80 788, 2019.

[27] J. Sanz, R. Perera, and C. Huerta, "Fault diagnosis of rotating machinery based on auto-associative neural networks and wavelet transforms," *Journal of sound and vibration*, vol. 302, no. 4-5, pp. 981–999, 2007.

[28] Z. Tian, Y. Ren, and G. Wang, "Short-term wind power prediction based on empirical mode decomposition and improved extreme learning machine," *Journal of Electrical Engineering & Technology*, vol. 13, no. 5, pp. 1841–1851, 2018.

[29] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE transactions on signal processing*, vol. 62, no. 3, pp. 531–544, 2013.

[30] R. Glowinski and P. Le Tallec, *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*. SIAM, 1989.

[31] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[34] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid technologies: Communication technologies and standards," *IEEE transactions on Industrial informatics*, vol. 7, no. 4, pp. 529–539, 2011.

[35] K. Huang, Z. Xiang, W. Deng, X. Tan, and C. Yang, "Reweighted compressed sensing-based smart grids topology reconstruction with application to identification of power line outage," *IEEE Systems Journal*, 2019.

[36] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on power systems*, vol. 26, no. 1, pp. 12–19, 2010.

[37] S. Sehgal and V. Acharya, "Design of pi controller for continuous stirred tank heater process," in *2014 IEEE Students' Conference on Electrical, Electronics and Computer Science*. IEEE, 2014, pp. 1–5.

[38] N. F. Thornhill, S. C. Patwardhan, and S. L. Shah, "A continuous stirred tank heater simulation model with applications," *Journal of process control*, vol. 18, no. 3-4, pp. 347–360, 2008.

[39] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & chemical engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[40] Y. Liu, C. Yang, K. Huang, and W. Gui, "Non-ferrous metals price forecasting based on variational mode decomposition and lstm network," *Knowledge-Based Systems*, vol. 188, p. 105006, 2020.

[41] Z. Wang, J. Wang, and W. Du, "Research on fault diagnosis of gearbox with improved variational mode decomposition," *Sensors*, vol. 18, no. 10, p. 3510, 2018.

Counteracting Adversarial Attacks in Autonomous Driving

Qi Sun[1], Arjun Ashok Rao[1], Xufeng Yao[1], Bei Yu[1], Shiyan Hu[2]

[1]CSE Department, The Chinese University of Hong Kong
[2]University of Southampton

1 Introduction

With the arrival of the artificial intelligence era, autonomous driving systems based on deep neural networks (DNN) have triggered a new revolution in traveling, and have a high potential to change the development of cities. An autonomous driving system needs to complete the following tasks: sensing, decision-making, planning, and control. Among these, sensing is considered as the most fundamental task and of vital importance. In recent years, vision and LiDAR-based 3D object detection systems which utilize deep neural networks have been widely used as the sensing systems [1].

Stereo-based 3D object detection is a vision-based system which fully exploits sparse, dense, semantic, and geometrical information in stereo imagery. Most of these models, *e.g.*, Faster R-CNN [2], utilize large feature networks as their backbone to extract features and use region proposal networks (RPNs) to generate object proposals which are then refined in subsequent modules to get the exact bounding boxes and class labels. With this rich information, we can get more accurate keypoints, viewpoints, object dimensions, and bounding boxes [3, 4, 5, 6]. Usually, the left and the right images cooperate with each other in the stereo-vision system, as shown in Figure 1. 3D spatial knowledge is highly dependent on the left and right stereo-pair images. Contrary to stereo systems, monocular 3D object detection approaches suffer from the lack of accurate depth information, and as a result, cannot provide comparable performance [5]. In addition to vision-based systems, complex real environments make manufacturers adopt LiDAR-based systems at the same time. LiDAR systems generate 3D point cloud data to model the 3D structures of scenes, either by projecting them into a bird's view or directly learning the 3D representations for classification and regression [7, 8, 9, 10].

Although deep learning algorithms have demonstrated superior performance in many circumstances, it has been recently shown that these algorithms are vulnerable to perturbations. This security risk is especially dangerous for 3D object detection in autonomous driving. Consequently, the concept of adversarial attacks [11] came into being to measure these perturbations. Typically, adversarial perturbations are crafted to be imperceptible to human observers and indistinguishable from the original image. This is achieved by constraining the ℓ_p norm of the adversarial image

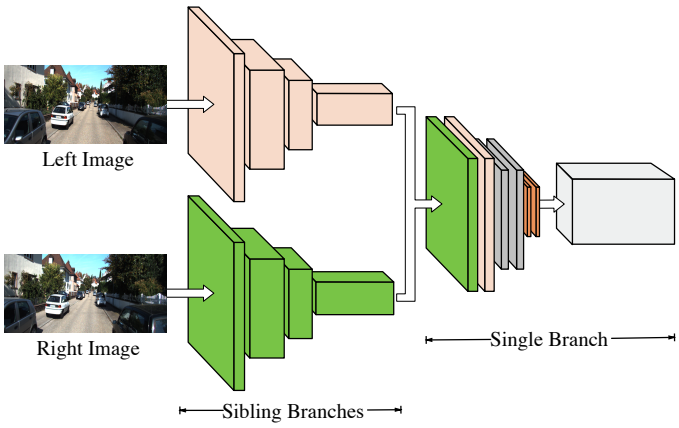


Figure 1: The structure of a typical stereo-based multi-task object detection model. There are two sibling branches, *e.g.*, RPN modules. Each branch takes left and right images as input respectively. The extracted object proposals are concatenated or reshaped into a single feature map for further processing, *e.g.*, regressing 3D boundary cube, and predicting viewpoints.

to a pre-defined value that ensures human imperceptibility from a pixel-difference perspective. However, adversarial examples can cause large errors in the detection model when added to images. To date, several adversarial attack algorithms have been designed to attack DNN models [11, 12, 13, 14, 15, 16, 17, 18, 19]. [11] first demonstrated the existence of perturbations to natural images which can fool DNN models into misclassification. To generate adversarial images more efficiently, [12] proposed a novel method termed 'fast gradient sign method' (FGSM) to generate the perturbations by computing the gradient of the loss function. Intuitively, this means optimizing each input image pixel through its gradient to maximize the loss while model parameters are kept unchanged. FGSM utilizes the linearity hypothesis of DNN models, *i.e.*, designs of deep learning models encourage linear behavior for computational gains. The basic iterative method (BIM) [16] extended FGSM by iteratively take multiple small steps to adjust the perturbation direction. Projected gradient descent (PGD) [17] further studied the adversarial perturbations from the perspective of optimization. PGD initializes the search for an adversarial image at a random point within the perturbation range. The noisy initialization creates a stronger attack than previous methods. Attacking the object detection model is more challenging compared to attacking the classification model as it needs to mislead the multiple region proposals. [20] attacks detectors via expectation over transformation (EOT) technique – a method that computes the perturbation by adding random distortions (*e.g.*, resizing, rotation, *etc.*) to natural images. [18] attacks the shapes of bounding boxes and classification labels simultaneously. [19] and [21] focus on attacking more relevant objects by splitting the whole image into subregions, *e.g.*, foreground and background, or several superpixels. Adversarial examples also exist in the physical world. Some adversarial images and road signs are printed to fool deep vision models [16, 22]. Adversarial T-shirts can evade person detection systems, even with only a few adversarial patches on the clothing [23, 24]. [25] generates adversarial 3D objects via transformation-based methods.

Correspondingly, to improve robustness against attacks, certain adversarial defense algorithms have been proposed. Currently, defense methods develop along three directions: using modified training or modified inputs, modifying networks, and using external add-on networks [26]. A majority of the literature that introduced new adversarial attack methods simultaneously train the models with their attacked inputs [12, 16, 17] - a practice termed as adversarial training. Some modified inputs by conducting preprocessing operations, *e.g.*, random resizing [27] and data compression [28]. SafetyNet [29] proposed to append an SVM classifier to the models such that SVM can use the discrete codes computed by ReLUs. For an input image, its discrete codes are compared against the codes of training data to determine whether it is an adversarial image. Generative adversarial networks (GANs) [30, 31], composed of a generator and a discriminator, add two novel modules to help generate perturbations and discriminate adversarial inputs. Outside of these outstanding works, to the best of our knowledge, there has been no work done on defending against attacks on stereo-based 3D object detection models. Although we can directly impart reasonable robustness via brute-force adversarial training with adversarial images as inputs, this strategy ignores the physical characteristics of stereo vision.

In this paper, we propose a defense method based on adversarial training with a novel and physically meaningful regularization term. Stereo-based detection models normally utilize the implicit spatial information from the left and right images to regress proposals independently, *i.e.*, the sibling branches in Figure 1. Meanwhile, the concatenated features from these two images are further fused to learn model information, *i.e.*, the single branch in Figure 1. Considering these two types of mechanisms that can be modeled as univariate and multivariate functions, a novel stereo-based regularizer is proposed. The regularizer is further relaxed to its upper bounds which ease the optimization process. To maximize the local smoothness of the loss surface, the upper bound is further approximated by the remainders of Taylor expansions. With these features, our novel defense method can counteract adversarial attacks efficiently.

The rest of our paper is organized as follows. Section 2 introduces the problem to be addressed and preliminaries. Section 3 explains our proposed defense techniques in detail. Section 4 summarizes the overflow defense flow. Section 5 demonstrates the experiments and results, followed by conclusion in Section 6.

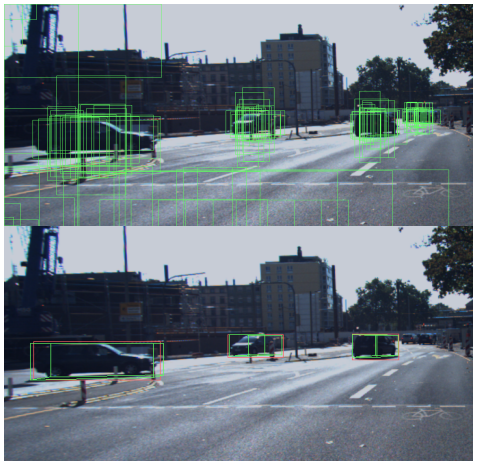


Figure 2: The generated object proposals and the final detected objects.

2 Preliminaries

2.1 Adversarial Training

Adversarial training can be traced back to the rise of adversarial attack algorithms. The typical form of most adversarial training algorithms involve training the target model on adversarial images generated via the attack method. Notably, most adversarial training methods perform the following min-max training strategy shown in Equation (24).

$$\begin{aligned} &\min_{\theta} \max_{\delta} L(x+\delta, \theta; y), \\ &\text{s.t. } \|\delta\|_p \leq \varepsilon, \end{aligned} \tag{24}$$

where θ represents the model parameters, δ is the perturbation, y is the ground truth and $L(x+\delta, \theta; y)$ is the loss function. $\|\cdot\|_p$ is the ℓ_p-norm, which constrains the perturbation within ε such that the perturbation is imperceptible to cameras and human eyes.

2.2 Stereo-based 3D Object Detection

Stereo-based 3D object detection [5, 6] has proved a success in object detection in autonomous driving systems. Stereo-based systems can detect and associate objects simultaneously using the left and right images through exploiting semantic and geometric information in stereo imagery. The network architecture can be briefly divided into two parts [5], as shown in Figure 1. The first module contains two sibling Stereo RPNs which extract features and generate object bounding proposals for the left and right images independently. Relying on the sibling features extracted in the first module, the subsequent module fuses the features and predicts the boundary cube, keypoint, and other related spatial information. The final detection results are jointly determined by the region proposals. An example is shown in Figure 2.

2.3 Problem Formulation

Denote x_l and x_r as the input left and right images respectively. The object bounding boxes in the left and right images are b_l and b_r respectively and the object class label is y. Given a stereo 3D object detection model with parameters θ and loss function L, our task is to solve the following min-max problem:

$$\begin{aligned} &\min_{\theta} \max_{\delta_l, \delta_r} L(x_l+\delta_l, x_r+\delta_r, \theta; b_l, b_r, y), \\ &\text{s.t. } \|\delta_l\|_p \leq \varepsilon, \ \ \|\delta_r\|_p \leq \varepsilon, \end{aligned} \tag{25}$$

Figure 3: Bounding boxes regressed from the left and right images.

where δ_l and δ_r represent the perturbations on the left and right images. δ_l and δ_r are both constrained within the manipulation budget ε. In the following sections, we use L_o to denote the above original $L(\cdot)$ loss function for brevity.

3 Defense Algorithm

As previously mentioned, the stereo-based 3D object detection model can handle various tasks. Different tasks can be modeled as different forms of functions. For example, the sibling RPN modules generate bounding boxes for the left and right images respectively (as shown in Figure 3). Therefore we can model this part as two independent univariate functions. The regularization term should constrain both of these two functions. Regressing the 3D bounding box or predicting the viewpoint can be represented as a multivariate function. The embedded features which are learned from the left-right stereo pair are jointly used as inputs to the multivariate function. Consequently, the regularization term should be able to handle multivariate functions. Both of the two regularization terms are optimized by relaxation and approximation, to improve local smoothness of the loss surface.

3.1 Stereo-based Regularizer

The two regressed bounding boxes from the left and right images share a high intersection over union (IoU). This phenomenon is consistent with the pre-existing understanding that stereo cameras capture the same field of view from a rectified stereo-pair with a small level of disparity. However, the two resulting bounding boxes contain differences that are influenced by physical factors such as the distance between the car and the object, the object orientation with respect to the stereo camera, *etc*. These physical factors vary with environments, which make them expensive to be measured accurately. For simplicity, we compute the distance between the two bounding boxes to characterize the effects of the practical physical factors.

Let $f_l(x_l)$ and $f_r(x_r)$ represent two univariate functions, to represent the features extracted from the left image x_l and right image x_r respectively. Therefore, the distance between the bounding boxes predicted from the two images is defined as:

$$d\ (x_l, x_r) = \| f_l(x_l) - f_r(x_r) \|_n. \tag{26}$$

As mentioned before, the physical characteristics are measured with $d\ (x_l, x_r)$. After attacking the images, the corresponding distance is computed as:

$$d\ (x_l + \delta_l, x_r + \delta_r) = \| f_l(x_l + \delta_l) - f_r(x_r + \delta_r) \|_n. \tag{27}$$

To improve the robustness of the detection system, we hope that these physical characteristics are well reserved. Therefore, the regularization term for the sibling branches is defined as:

$$L_b = |\ d\ (x_l + \delta_l, x_r + \delta_r) - d\ (x_l, x_r)\ |, \tag{28}$$

where $|\cdot|$ computes the absolute value. With L_b and the original loss function L_o, the updated optimization objective function is $L = L_b + L_o$. Note that as shown in Equation (25), the regularization term is minimized with respect to θ. Minimizing Equation (28) would possibly result in inflexible optimization and ambiguous convergence status [32].

The straightforward hazard is that pushing $d\ (x_l, x_r)$ close to zero makes the model confuse the left and right images. So is for $d\ (x_l + \delta_l, x_r + \delta_r)$. For example, $d\ (x_l, x_r) = 0$ would result in $f_l(x_l) = f_r(x_r)$. Although the original loss term L_o would alleviate this hazard as it computes the errors between the predicted bounding boxes and ground truths, L_b would no longer be a helper and would become a burden. This contradicts our initial intuition.

We add a margin m to reinforce the optimization of the distance functions [33, 32]. Take $d\ (x_l, x_r)$ as an example. $f_l(x_l)$ and $f_r(x_r)$ are in symmetric positions in $d\ (x_l, x_r)$. This means that adding a positive margin to $f_l(x_l)$ is equivalent to adding a negative margin to $f_r(x_r)$. The margin-based distance function is shown in Equation (29).

$$\begin{aligned} d\ (x_l, x_r) &= \|f_l(x_l) - f_r(x_r) + m\|_n, \\ d\ (x_l + \delta_l, x_r + \delta_r) &= \|f_l(x_l + \delta_l) - f_r(x_r + \delta_r) + m\|_n. \end{aligned} \tag{29}$$

The same margin m is shared in the two distance metrics because we believe that the model should be able to recover the same results after been attacked.

The tasks which use the fused features learned from the early module can be modeled as multivariate functions. For example, the viewpoint prediction function can be represented as $f_m(x_l, x_r)$, and the resultant vector with perturbation becomes $f_m(x_l + \delta_l, x_r + \delta_r)$. We hope the model can get the same result after the images are attacked, therefore the regularization term L_m to be minimized is defined as:

$$L_m = \|f_m(x_l + \delta_l, x_r + \delta_r) - f_m(x_l, x_r)\|_n. \tag{30}$$

Different from Equation (29), we do not add a margin here because the features learned from the perturbated images should be equal to the original features. With L_m, the update optimization objective function is $L = L_o + L_b + L_m$.

3.2 Local Smoothness Optimization

Recent work has demonstrated that the robustness of models usually suffers from the non-linearity of loss surface and gradient obfuscation. Many methods have been proposed to improve the local smoothness [34, 35, 36]. Equation (29) and Equation (28) is transformed to a nested $\|\cdot\|$ formulation shown in Equation (31).

$$L_b = \| \ \|f_l(x_l + \delta_l) - f_r(x_r + \delta_r) + m\|_n - \|f_l(x_l) - f_r(x_r) + m\|_n \ \|_1. \tag{31}$$

Equation (31) with nested norm parameters is challenging to be solved. Moreover, m is a hyper parameter that needs to be determined through adversarial training. Besides, the difference between two terms in $\|\cdot\|$ is at a high magnitude, while the loss surface usually has a low magnitude. Inspired by recent work which approximates the regularization term by the remainder of its Taylor expansion [34, 35], we propose to relax Equation (31) as Equation (32). The detailed relaxation process is attached in Appendix A.

$$\begin{aligned} L_b &= \| \ \|f_l(x_l + \delta_l) - f_r(x_r + \delta_r) + m\|_n - \|f_l(x_l) - f_r(x_r) + m\|_n \ \|_1 \\ &\leq \| \ f_l(x_l + \delta_l) - f_r(x_l) \ \|_n + \| \ f_l(x_r + \delta_r) - f_r(x_r) \ \|_n \\ &\leq \| \ \delta_l \nabla_{x_l} f_l(x_l) \ \|_n + \gamma_l(\varepsilon, x_l) + \| \ \delta_r \nabla_{x_r} f_r(x_r) \ \|_n + \gamma_r(\varepsilon, x_r), \end{aligned} \tag{32}$$

where $\delta_l \nabla_{x_l} f_l(x_l)$ is the first-order term in the Taylor expansion of $f_l(x_l)$, and $\delta_r \nabla_{x_r} f_r(x_r)$ is the first-order term in the Taylor expansion of $f_r(x_r)$. $\gamma_l(\varepsilon, x_l)$ and $\gamma_r(\varepsilon, x_r)$ are the maximums of the high-order remainders of the Taylor expansions. According to the inner maximization operation in Equation (25), they are defined as:

$$\begin{aligned} h_l(\varepsilon, x_l) &= \| \ f_l(x_l + \delta_l) - f_l(x_l) - \delta_l \nabla_{x_l} f_l(x_l) \ \|_n, \\ h_r(\varepsilon, x_r) &= \| \ f_r(x_r + \delta_r) - f_r(x_r) - \delta_r \nabla_{x_r} f_r(x_r) \ \|_n, \\ \gamma_l(\varepsilon, x_l) &= \max_{\|\delta_l\|_p \leq \varepsilon} h_l(\varepsilon, x_l), \quad \gamma_r(\varepsilon, x_r) = \max_{\|\delta_r\|_p \leq \varepsilon} h_r(\varepsilon, x_r), \end{aligned} \tag{33}$$

where h_l and h_r represent the high-order remainders for the left and right images respectively.

With Equation (32), we can not only erase m, but also relax Equation (31) to its upper bound. Considering the trade-off between computational workload and model accuracy, the higher order remainders, *e.g.*, the 2-nd gradient

is not computed. The insights behind Equation (32) is straightforward: the difference between $f_l(x_l+\delta_l)$ and $f_l(x_l)$ is constrained by the first-order gradient term and the high-order remainder of the Taylor expansion of $f_l(x_l+\delta_l)$. γ_l and γ_r are good measures of how linear the surfaces are within the perturbation range ε. This kind of quality is called *local smoothness measure*. By minimizing the smoothness term, we will maximize the smoothness of the loss surface and therefore improve the model robustness.

As to the classification regularizer L_m, it follows a similar relaxation strategy. $f_m(x_l+\delta_l, x_r+\delta_r)$ is approximated by:

$$f_m(x_l+\delta_l, x_r+\delta_r) \approx f_m(x_l,x_r) + \delta_l \nabla_{x_l} f_m(x_l,x_r) + \delta_r \nabla_{x_r} f_m(x_l,x_r). \tag{34}$$

Thus we can form the following bound:

$$\begin{aligned} L_m &= \parallel f_m(x_l+\delta_l, x_r+\delta_r) - f_m(x_l,x_r) \parallel_n \\ &\leq \parallel \delta_l \nabla_{x_l} f_m(x_l,x_r) + \delta_r \nabla_{x_r} f_m(x_l,x_r) \parallel_n + \gamma_m(\varepsilon, x_l, x_r), \end{aligned} \tag{35}$$

where $\gamma_m(\varepsilon, x_l, x_r)$ is the maximum of the high-order remainder $h_m(\varepsilon, x_l, x_r)$. They are defined as follows:

$$\begin{aligned} h_m(\varepsilon, x_l, x_r) &= \parallel f_m(x_l+\delta_l, x_r+\delta_r) - f_m(x_l,x_r) \\ &\qquad - \delta_l \nabla_{x_l} f_m(x_l,x_r) - \delta_r \nabla_{x_r} f_m(x_l,x_r) \parallel_n, \\ \gamma_m(\varepsilon, x_l, x_r) &= \max_{\|\delta_l\|_p \leq \varepsilon, \|\delta_r\|_p \leq \varepsilon.} h_m(\varepsilon, x_l, x_r). \end{aligned} \tag{36}$$

Combining Equation (33) and Equation (36) together, we define the regularization term for high-order remainder as L_h, as shown in Equation (37).

$$L_h = h_l(\varepsilon, x_l) + h_r(\varepsilon, x_r) + h_m(\varepsilon, x_l, x_r). \tag{37}$$

Similarly, we combine all of the first-order gradient term together, and then we have the regularization term L_∇ defined as follows:

$$\begin{aligned} L_\nabla = &\parallel \delta_l \nabla_{x_l} f_l(x_l) \parallel_n + \parallel \delta_r \nabla_{x_r} f_r(x_r) \parallel_n \\ &+ \parallel \delta_l \nabla_{x_l} f_m(x_l,x_r) + \delta_r \nabla_{x_r} f_m(x_l,x_r) \parallel_n \end{aligned} \tag{38}$$

The overall stereo-based regularizer is $L_h + L_\nabla$. Together with the original loss function L_o in Equation (25), we can derive the following min-max problem formulation:

$$\begin{aligned} &\min_\theta L_a = L_o + L_\nabla + [\max_{\delta_l,\ \delta_r} L_h] \\ &\text{s.t. } \|\delta_l\|_p \leq \varepsilon, \ \ \|\delta_r\|_p \leq \varepsilon, \end{aligned} \tag{39}$$

where L_a is defined as the summation of the training error together with the regularization terms.

4 Overall Flow

In the previous section, we discuss the stereo-based regularizer in detail. Afterward, local smoothness is considered and the originally proposed regularizer is relaxed to obtain the local smoothness. An adversarial training strategy is adopted in this paper.

We iteratively optimize perturbations δ_l, δ_r, and model parameters θ. The pseudo-code of the overall optimization training flow is shown in Algorithm 2. The advantages of using ℓ_1-norm over ℓ_2-norm in terms of robustness analysis procedures are largely recognized across the scientific literature [37]. To improve model robustness, ℓ_1-norm is used as the norm in Equation (29) and Equation (30).

Algorithm 2 Adversarial Training of Stereo-based Object Detection Model

Require: Training set $\{(x_l^1, x_r^1, b_l^1, b_r^2, y^1), \ldots, (x_l^N, x_r^N, b_l^N, b_r^N, y^N)\}$, batch size B, # of iterations for outer optimization I_o, # of iterations for inner optimization I_i, model parameters θ, learning rate η, perturbation range ε.

for $i = 1 \rightarrow I_o$ **do**
  Sample a batch B from the training set;
  Generate initial δ_l and δ_r for B, in perturbation range ε;
  for $j = 1 \rightarrow I_i$ **do**
    Calculate L_h in Equation (37) for the batch B;
    Update δ_l and δ_r via back-propagation with L_h as the loss function;
  end for
  Compute L_∇ with δ_l and δ_r, according to Equation (38);
  Compute $L_a = L_o + L_\nabla + L_h$ with δ_l and δ_r;
  Update θ via back-propagation with L_a as the loss function;
end for

Table 2: Statistical Results of Adversarial Attacks

Model	AP_{2d} (%)			AOS (%)			AP_{3d} (%)			AP_{bv} (%)		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
`No Attack` [5]	99.28	91.09	78.62	98.42	89.43	76.94	54.10	34.44	28.15	68.24	46.84	39.34
`FGSM`, $\varepsilon = 0.7$	88.29	76.45	62.39	87.54	74.11	60.36	40.52	32.94	27.56	15.52	12.19	10.05
`FGSM`, $\varepsilon = 2$	76.82	60.49	49.67	74.73	57.84	47.35	26.21	21.35	16.81	13.64	7.7	6.14
`PGD`, $\varepsilon = 0.7$	69.55	58.94	48.04	66.72	56.04	45.59	22.52	18.88	15.32	7.02	5.53	4.29
`PGD`, $\varepsilon = 2$	53.01	43.11	34.21	51.48	40.23	31.80	9.60	7.61	6.23	3.82	2.22	1.95

Table 3: Statistical Results of Adversarial Defenses

Testing Images	Defense Method	AP_{2d} (%)			AOS (%)			AP_{3d} (%)			AP_{bv} (%)		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
`FGSM`, $\varepsilon = 0.7$	`Direct + FGSM`	87.58	81.54	71.53	87.25	80.11	62.42	41.95	30.62	**28.89**	21.57	19.62	16.56
	`SmoothStereo`	**88.38**	**82.74**	**73.94**	**88.89**	**81.87**	**63.63**	**45.51**	**31.01**	26.61	**24.50**	**20.88**	**18.26**
`FGSM`, $\varepsilon = 2$	`Direct + FGSM`	84.73	70.82	57.90	**84.13**	69.19	55.61	40.15	30.57	**24.42**	16.21	13.03	10.54
	`SmoothStereo`	**85.95**	**72.64**	**61.22**	81.65	**74.83**	**60.00**	**41.43**	**31.63**	23.79	**18.25**	**14.76**	**12.53**
`PGD`, $\varepsilon = 0.7$	`Direct + PGD`	73.37	**61.82**	56.66	73.04	60.46	50.04	**27.47**	20.08	**18.74**	**13.77**	7.10	9.30
	`SmoothStereo`	**75.67**	61.58	**59.73**	**73.43**	**62.27**	**52.82**	24.88	**20.90**	16.99	12.44	**11.73**	**9.46**
`PGD`, $\varepsilon = 2$	`Direct + PGD`	54.46	49.11	40.44	53.37	46.23	38.07	14.39	10.38	9.32	5.84	**4.65**	3.29
	`SmoothStereo`	**55.29**	**49.38**	**41.92**	**53.47**	**47.27**	**40.60**	**18.11**	**12.42**	**9.43**	**6.82**	4.52	**3.94**

Figure 4: Examples of results on FGSM attacks. The images from left to right are: original detection results (ground-truth), adversarial images generated via FGSM with $\varepsilon = 2$, defense results via direct adversarial training, and defense results via our `SmoothStereo`.

Figure 5: Example of results on PGD attacks. The images from left to right are: original detection results (ground-truth), adversarial images generated via PGD with $\varepsilon = 2$, defense results via direct adversarial training, and defense results via our `SmoothStereo`.

5 Experimental Results

In this section, we evaluate our defense method on the challenging KITTI object detection benchmark [38]. KITTI set is divided into three categories: Easy, Moderate, and Hard, which reflect the difficulties of the object detection tasks. The state-of-the-art Stereo-based 3D object detection model from [5] is used as the target detection model. Two popular and powerful attack methods are implemented to attack the detection model, *i.e.*, FGSM [12] and PGD [17]. Direct adversarial training, proposed in [17] is used to defend against the adversarial attacks, and results are compared with our novel defense method. For brevity, our method is shorted as `SmoothStereo`.

The result statistics are listed in Table 2. AP_{2d} represents the average detection precision of the 2D bounding box. AOS represents the average orientation similarity of the joint 3D detection [38]. AP_{3d} represents the average detection precision of the 3D bounding box. AP_{bv} represents the average localization precision of bird's eye view. The error statistics are computed according to boxes with IoU ≥ 0.7. Note that in real environments, the perturbations are usually not overly abnormal. Greater perturbation ranges lead to stronger attacks. For balance, in our experiments, we take two perturbation values as examples, *i.e.*, $\varepsilon = 0.7$ and $\varepsilon = 2$. In each experiment, the left and right images share the same perturbation range ε. The optimization iteration of PGD images is 2. It is evident from Table 2 that PGD produces much lower accuracy rates, and is hence a much stronger adversary compared to FGSM. This phenomenon is consistent with people's experience. Even with a moderate $\varepsilon = 2$, PGD can degrade the model performance by nearly half.

5.1 Defense against FGSM Attacks

Figure 4 shows an example of the FGSM attack and the results of different defense methods. The adversarial image misleads the model to misclassify one car and incorrectly predict the object orientation. Direct adversarial training still loses that car, while misclassifying the granite steps as a car. In comparison, our method can correctly predict the locations and directions of the cars. Moreover, our regularization and smoothness terms are also able to outperform natural detections in some cases. This is shown in Figure 4 where our robust model correctly detects a car which was previously misclassified on the unperturbed model. This proves the local smoothness of our method. The statistical results are listed in Table 3.

5.2 Defense against PGD Attacks

Figure 4 shows an example of the PGD attack and the results of defense methods. The original model loses the car in adversarial images. Intuitively, this can be considered a misclassification and the model incorrectly perceives class 'Car' as class 'Background'. Direct adversarial training [17] can correct the model and predict the car successfully. In comparison, our `SmoothStereo` method not only predicts the car, but also finds the nearest object which hinders the car. The results prove that our method can also improve robustness of the model while improving the local

smoothness. The statistical results are listed in Table 3.

In summary, the results show that our method can efficiently improve local smoothness of the detection model and improve prediction results. It is also shown that out novel regularization, which considers local smoothness and stereo information, can significantly boost detection performance of the original model as well.

6 Conclusion

To counteract adversarial attacks and improve the robustness of object detection models for autonomous driving systems, a novel defense method which specifically considers the physical meaning of the Stereo-based 3D object detection model is proposed in this paper. Our regularizer can help the model learn the relative relationship of the bounding boxes between the left and right images, which can be modeled as two univariate functions. The regularizer is also capable of handling the branch which is modeled as a multivariate function. These regularizers are further relaxed to their upper bounds and approximated by first-order remainders of Taylor expansions. With this relaxation and approximation, we can maximize the local smoothness of the loss surface to improve the robustness. It is also shown in the results that our novel regularization considering local smoothness and stereo information can boost the detection performance of the original model as well.

7 Acknowledgment

This work is partially supported by Tencent Technology, SmartMore, and The Research Grants Council of Hong Kong SAR (Project No. CUHK14209420).

References

[1] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3d object detection methods for autonomous driving applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3782–3795, 2019.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[3] X. Chen, K. Kundu, Y. Zhu, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals using stereo imagery for accurate object class detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 5, pp. 1259–1272, 2017.

[4] P. Li, T. Qin *et al.*, "Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 646–661.

[5] P. Li, X. Chen, and S. Shen, "Stereo r-cnn based 3d object detection for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7644–7652.

[6] Y. Chen, S. Liu, X. Shen, and J. Jia, "Dsgn: Deep stereo geometry network for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 536–12 545.

[7] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in neural information processing systems*, 2017, pp. 5099–5108.

[8] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1907–1915.

[9] W. Luo, B. Yang, and R. Urtasun, "Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3569–3577.

[10] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[11] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *International Conference on Learning Representations (ICLR)*, 2014.

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *International Conference on Learning Representations (ICLR)*, 2015.

[13] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2574–2582.

[14] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," *arXiv preprint arXiv:1705.07204*, 2017.

[15] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, "Adversarial examples for semantic segmentation and object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1369–1378.

[16] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

[17] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *International Conference on Learning Representations (ICLR)*, 2018.

[18] Y. Li, D. Tian, X. Bian, S. Lyu *et al.*, "Robust adversarial perturbation on deep proposal-based models," *British Machine Vision Conference (BMVC)*, 2018.

[19] Y. Li, X. Bian, M. Chang, and S. Lyu, "Exploring the vulnerability of single shot module in object detectors via imperceptible background patches," in *British Machine Vision Conference (BMVC)*, 2019.

[20] S.-T. Chen, C. Cornelius, J. Martin, and D. H. P. Chau, "Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 52–68.

[21] X. Dong, J. Han, D. Chen, J. Liu, H. Bian, Z. Ma, H. Li, X. Wang, W. Zhang, and N. Yu, "Robust superpixel-guided attentional adversarial attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 895–12 904.

[22] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1625–1634.

[23] K. Xu, G. Zhang, S. Liu, Q. Fan, M. Sun, H. Chen, P.-Y. Chen, Y. Wang, and X. Lin, "Adversarial t-shirt! evading person detectors in a physical world," *arXiv*, pp. arXiv–1910, 2019.

[24] Z. Wu, S.-N. Lim, L. Davis, and T. Goldstein, "Making an invisibility cloak: Real world adversarial attacks on object detectors," *European Conference on Computer Vision (ECCV)*, 2020.

[25] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *International conference on machine learning*, 2018, pp. 284–293.

[26] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[27] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, "Mitigating adversarial effects through randomization," *International Conference on Learning Representations (ICLR)*, 2018.

[28] N. Das, M. Shanbhogue, S.-T. Chen, F. Hohman, L. Chen, M. E. Kounavis, and D. H. Chau, "Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression," *arXiv preprint arXiv:1705.02900*, 2017.

[29] J. Lu, T. Issaranon, and D. Forsyth, "Safetynet: Detecting and rejecting adversarial examples robustly," in *IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[30] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[31] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1222–1230.

[32] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6398–6407.

[33] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks." in *ICML*, vol. 2, no. 3, 2016, p. 7.

[34] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli, "Adversarial robustness through local linearization," in *Advances in Neural Information Processing Systems*, 2019, pp. 13 847–13 856.

[35] J. Xu, Y. Li, Y. Bai, Y. Jiang, and S.-T. Xia, "Adversarial defense via local flatness regularization," *arXiv preprint arXiv:1910.12165*, 2019.

[36] B. Yu, J. Wu, J. Ma, and Z. Zhu, "Tangent-normal adversarial regularization for semi-supervised learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 676–10 684.

[37] S. A. Flores, "Robustness of ℓ_1-norm estimation: From folklore to fact," *IEEE Signal Processing Letters*, vol. 25, no. 11, pp. 1640–1644, 2018.

[38] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 3354–3361.

[39] R. Kress, *Numerical Analysis*, ser. Graduate Texts in Mathematics. Springer New York, 1998. [Online]. Available: https://books.google.com.hk/books?id=e7ZmHRIxum0C

A Relaxation of Equation (31)

According to the triangle inequality:

$$||a|+|b|| \leq |a \pm b| \leq |a|+|b|, \tag{40}$$

which is one of the defining property of the normed vector space [39], Equation (31) can be relaxed to a upper bound:

$$\begin{aligned} L_b &= \| \, \|f_l(x_l+\delta_l) - f_r(x_r+\delta_r)+m\|_n - \|f_l(x_l)-f_r(x_r)+m\|_n \, \|_1 \\ &\leq \| \, f_l(x_l+\delta_l) - f_r(x_r+\delta_r)+m-(f_l(x_l)-f_r(x_r)+m) \, \|_n \\ &= \| \, (f_l(x_l+\delta_l)-f_l(x_l)) - (f_r(x_r+\delta_r)-f_r(x_r)) \, \|_n \\ &\leq \| \, f_l(x_l+\delta_l)-f_l(x_l) \, \|_n + \| \, f_r(x_r+\delta_r)-f_r(x_r) \, \|_n. \end{aligned} \tag{41}$$

The left and right images are in the symmetric positions in Equation (41), *i.e.*, $f_l(x_l+\delta_l)-f_r(x_r+\delta_r)$ leads to the same deduced results with $f_r(x_r+\delta_r)-f_l(x_l+\delta_l)$. Further, $f_l(x_l+\delta_l)$ can be approximated by its first-order Taylor expansion $f_l(x_l)+\delta_l\nabla_{x_l}f_l(x_l)$. Thus we can have the following bound:

$$\begin{aligned}
&\| f_l(x_l+\delta_l)-f_l(x_l) \|_n \\
\approx&\| \delta_l\nabla_{x_l}f_l(x_l)+f_l(x_l+\delta_l)-f_l(x_l)-\delta_l\nabla_{x_l}f_l(x_l) \|_n \\
\leq&\| \delta_l\nabla_{x_l}f_l(x_l) \|_n+\| f_l(x_l+\delta_l)-f_l(x_l)-\delta_l\nabla_{x_l}f_l(x_l) \|_n \\
\leq&\| \delta_l\nabla_{x_l}f_l(x_l) \|_n+\gamma_l(x_l,\varepsilon),
\end{aligned} \tag{42}$$

where $\gamma_l(x_l,\varepsilon)$ is defined as the maximum of the remainder of the first-order Taylor expansion of $f_l(x_l+\delta_l)$, *i.e.*:

$$\gamma_l(x_l,\varepsilon)=\max_{\|\delta_l\|_p\leq\varepsilon}\| f_l(x_l+\delta_l)-f_l(x_l)-\delta_l\nabla_{x_l}f_l(x_l) \|_n. \tag{43}$$

Similarly, the term for the right image is relaxed as follow:

$$\| f_r(x_r+\delta_r)-f_r(x_r) \|_n\leq\| \delta_r\nabla_{x_r}f_r(x_r) \|_n+\gamma_r(x_r,\varepsilon). \tag{44}$$

Given Equation (42) and Equation (44), L_b is further relaxed to it upper bound, as shown in Equation (45).

$$\begin{aligned}
L_b&=\| \|f_l(x_l+\delta_l)-f_r(x_r+\delta_r)+m\|_n-\|f_l(x_l)-f_r(x_r)+m\|_n \|_1 \\
&\leq\| \delta_l\nabla_{x_l}f_l(x_l) \|_n+\gamma_l(x_l,\varepsilon)+\| \delta_r\nabla_{x_r}f_r(x_r) \|_n+\gamma_r(x_r,\varepsilon).
\end{aligned} \tag{45}$$

An Introduction to Software Defined Network for Industrial Internet of Things

Jin Du, Xianghui Cao
School of Automation, Southeast University, Nanjing, China

A Introduction

Due to the pursuit of personalized real consumption trends and the proposal of strategies like Industry 4.0, flexible production and mass customization have become the future development trend of manufacturing [1]. A new generation of flexible network structure is required to construct a more digital, networked, and intelligent manufacturing system with ubiquitous perception and adaptive adjustment capabilities to meet these needs. However, most of the current network systems are based on the conventional OSI architecture, which has underperformed under the current situation of traffic explosion and network topology expansion. The main reasons are as follows: 1) It is difficult to deploy, manage, operate and maintain network systems because of the diversity of equipment and differences of debugging and configuring devices produced by different manufacturers [2], which is an enormous challenge especially for networks with thousands of equipment. 2) Network devices are basically closed like black boxes with working mode mostly fixed and it is hard for users to customize them in depth according to business requirements [3]. 3) Usually, it takes several years to complete the process of communication protocol standardization to actual deployment because network devices are high coupling and other network elements also need to make corresponding adjustments when a network element changes the communication protocol. 4) The distributed architecture makes it difficult to realize global visualization, management and control of the network.

SDN (Software Defined Network) can be a potential solution to these challenges. SDN is a new generation of network architecture, evolving from Clean Slate academic project at Stanford University [4] and receiving widespread attention in recent years. The general idea behind SDN is to separate the control plane from the forwarding plane, thus facilitating centralize communication strategies for packet forwarding, channel assignment and traffic control into SDN controllers for execution, which means switches only have forwarding functions without independent brains [5]. Consequently, SDN controllers obtain a global perspective and achieve a better traffic monitoring and supervision, thus improving network resource utilization. A typical example is Google's B4 network, which increased link utilization to nearly 100% by introducing SDN as its architecture [6]. More importantly, API (Application Program Interface) is open and network is programmable [2], so users can customize the network by programming to response to business requirements better and faster, and researchers can participate in network studies preferably. Therefore, network deployment and management can be automatic in a programmable way instead of manual way in conventional network architecture. It is also worth noting that adopting SDN brings an effective solution to the energy consumption problem [7] due to energy saving by moving communication control tasks from network nodes to controllers.

B SDN for Industrial Internet of Things

Introducing SDN into Industrial Internet of Things (IIoT) can bring many benefits. In an industrial network system, plenty of equipment (such as sensors, controllers and actuators) needs to be deployed and communicate with each other in the production field because the production involves numerous parameters, but these devices usually do not all come from the same manufacturer. SDN can provide a low-cost and easy way to deploy and manage such networks, and the network programmable characteristic makes it capable to refactor or adjust the production system change. With system's global view, optimal strategies can be obtained and adopted.

An architectural view of SDN for IIoT is shown as Figure 1. The architecture can be divided into four planes: application plane, control plane, forwarding plane and end node plane. Located at the application plane are some scripts and programs, which provides applications and services such as network deployment and management, security strategies like firewall and production-related applications [8, 9]. At the control plane are SDN controllers, which provide centralized network management and communication control services [5]. Situated at the forwarding plane are SDN switches, the only function of which is forwarding packets under SDN controllers' direction.

The interaction between controllers and switches is via the southbound interface protocol, the most widely accepted and deployed of which is OpenFlow [2]. Strategies (such as routing algorithm, packets forwarding strategy, traffic shaping, etc.) are computed and implemented in the SDN controllers, which further distribute flow tables to SDN switches to direct them to forward packets [9]. The hardware of SDN switches can be special devices supporting OpenFlow or existing equipment retrofitted.

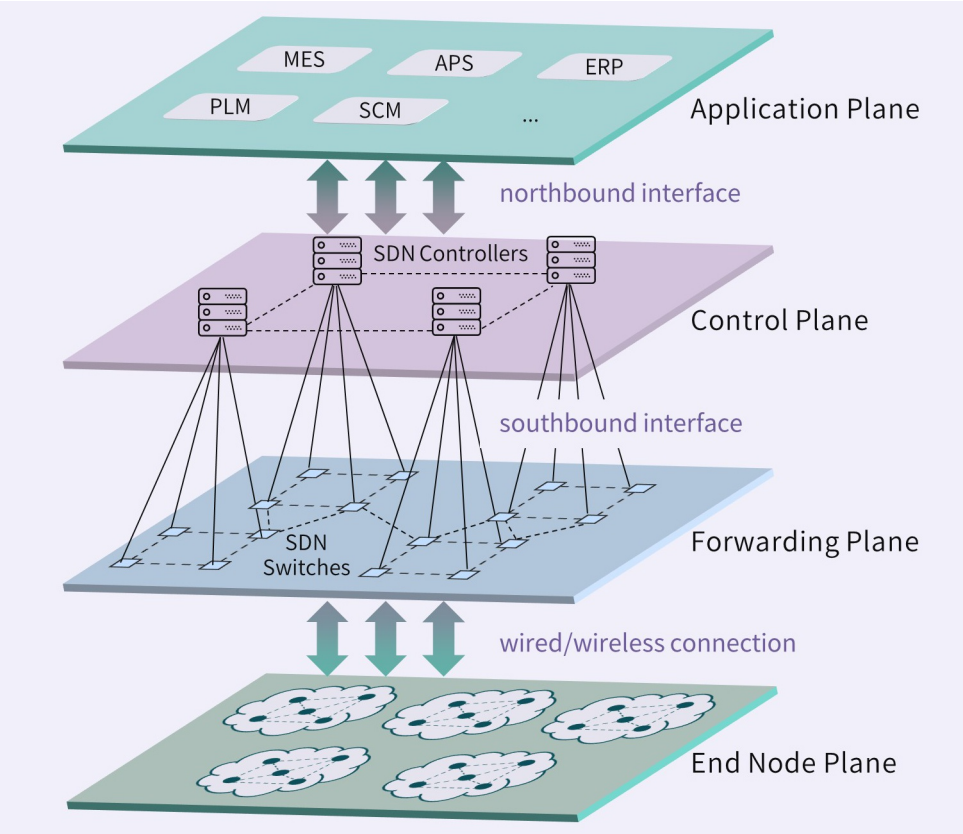


Figure 1: SDN

C Network Function Virtualization

NFV (Network Function Virtualization) technology concentrates the physical computing resources into a large resource pool of structured virtual machines, thus achieving complete decoupling of hardware and software. Thus, dedicated hardware devices can basically be replaced by X86 generic servers. Currently, NFV technology is mainly used in the core network.

NFV and SDN are regarded as excellent partners with each other, among which NFV represents the future of computing while SDN represents the future of the network. Ojo et al. [10] designed an SDN-IoT architecture with NFV implementation, which improved the agility and flexibility of network. Yousaf et al. [5] described how SDN and NFV complement each other in driving the future network like 5G. Luo et al. [11] proposed a new energy efficient scheme utilizing SDN and NFV, which can effectively extend life of industrial wireless sensor networks. Piedrahita et al. [12] proposed an automatic event response mechanism utilizing SDN and NFV against malicious attacks aiming at industrial control system.

D Time Sensitive Networking with SDN

IIoT has a critical demand of transmission reliability and low and bounded end-to-end delay. Traditional OT (Operation Technology) network mainly adopts industrial Ethernet, such as EtherCAT, PROFINET, POWERLINK and CC-LINK, which already has a high real-time performance. But industrial Ethernet has poor compatibility and interoperability, which makes it difficult to achieve fusion of OT and IT (Information Technology). And the transmission of mixed data is still a great challenge.

TSN (Time-Sensitive Networking) is an advanced technology aiming at hybrid transmission of multiple types of data in the same network and guaranteeing QoS at the same time. TSN achieves bounded delay with core technologies such as time synchronization, scheduling and traffic shaping, path and bandwidth reservation, etc.

SDN is also considered as an excellent partner of TSN. Schriegel et al. [13] introduced a distributed heterogeneous IIoT architecture which utilized time sensitive transmission capability of TSN and SDN's advantages in deploying, operating and managing networks. Combination of SDN and TSN is also termed TSSDN (Time-sensitive Software-Defined Networking). As Pang et al. [14] stated, TSSDN is a new paradigm with both deterministic real-time communication ability and network flexibility. And a flow scheduling mechanism in TSSDN was proposed in their work [14], which achieved zero frame loss when new flow schedule was updated to switches. Gerhard et al. [15] presented a Software-Defined Flow Reservation (SDFR) method, which achieved TSN configuration specified in IEEE 802.1Qcc in current SDN controller.

E Resource Allocation assisted with SDN

Industrial applications have strict demands on transmission QoS (Quality of Service), especially end-to-end latency and reliability. In order to take effect in using network resources, improve transmission quality, and then obtain better system performance and improve production benefit, there have been plenty of researches on network optimization strategies. But these works are mostly based on conventional distributed network, which makes it difficult to gain global optimal results and perform cross-layer optimization.

As a centralized architecture, SDN has a global view of the network and a centralized allocation of network resources [16], which can bring new options to network optimization strategies. Guck et al. [17] proposed a network solution based on SDN which achieved hard real-time QoS for industrial networks through function split between routing and resource allocation. Cheng et al. [18] presented an SDN-based link quality-aware routing mechanism for industrial wireless sensor networks and achieved global optimization. Jhaveri et al. [19] designed a QoS-aware routing mechanism based on SDN for robotic CPS, which adopts jitter, packet loss and link utilization as link weight to calculate optimal path. Their simulation result showed that their method performed better than existing solutions in terms of throughput, latency and jitter.

As Xie et al. [20] stated, in the general sense machine learning technology is difficult to be applied in conventional network due to its distributed nature, and some features of SDN (such as logically centralized, global view of the network, etc.) make machine learning technologies workable in networked system. They conducted a comprehensive survey of machine learning algorithms for SDN and outlined the application of machine learning in SDN. Tang et al. [21] proposed a traffic load prediction and intelligent channel allocation algorithm based on deep learning in SDN architecture, and the result showed that the prediction accuracy in centralized SDN control system is better than that in semi-centralized control system and distributed control system.

F Conclusion

This paper provided an introduction of SDN for IIoT. Firstly, we summarized the challenges of current network architecture that make it difficult to meet the needs of smart manufacturing, introduced the concept and main features of SDN, and explained the motivation to introduce SDN into IIoT. And then we expounded the structure of SDN for IIoT. Subsequently, we outlined combination of SDN and some emerging technologies, such as NFV, TSN and network resource allocation.

Despite numerous advantages of introducing SDN into IIoT, there still exist some important issues that remain challenging, for example: 1) It costs a lot to transform existing structure into SDN architecture completely. 2) There is a concern that this transition may severely affect the current production activities. A potential solution is hybrid SDN network [22], which means SDN devices and conventional devices coexist in the IIoT and the transformation is completed step by step on the precondition of insuring normal manufacture. 3) When the network changes dynamically due to the movement of devices, it is a challenge to update network strategies such as forwarding rules adaptively in real time [2]. Finally, there still needs more practice to make it from theory to maturity.

References

[1] Z. Kai, Q. Ting, P. Yanghua, L. Hao, L. Congdong, and G. Q. Huang. Cell-level production-logistics synchronization for multi-variety and small-batch production: a step toward industry 4.0. In *IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, pages 419–424, 2017.

[2] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig. Software-defined networking: a comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015.

[3] S. Bera, S. Misra, and A. V. Vasilakos. Software-defined networking for Internet of things: a survey. *IEEE Internet of Things Journal*, 4(6):1994–2008, 2017.

[4] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008.

[5] F. Z. Yousaf, M. Bredel, S. Schaller, and F. Schneider. NFV and SDN-key technology enablers for 5G networks. *IEEE Journal on Selected Areas in Communications*, 35(11):2468–2478, 2017.

[6] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S.Stuart, and A. Vahdat. B4: experience with a globally deployed software defined WAN. In *Proceedings of the ACM SIGCOMM Conference*, Hong Kong, China, 2013.

[7] S. Tomovic and I. Radusinovic. Performance analysis of a new SDN-based WSN architecture. In *2015 23rd Telecommunications Forum Telfor (TELFOR)*, pages 99–102, 2015.

[8] L. J. Jagadeesan and V. Mendiratta. Programming the network: application software faults in software-defined networks. In *2016 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 125–131, 2016.

[9] C. Martinez, R. Ferro, and W. Ruiz. Next generation networks under the SDN and OpenFlow protocol architecture. In *2015 Workshop on Engineering Applications - International Congress on Engineering (WEA)*, pages 1–7, 2015.

[10] M. Ojo, D. Adami, and S. Giordano. A SDN-IoT architecture with NFV implementation. In *2016 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, 2016.

[11] S. Luo, H. Wang, J. Wu, J. Li, L. Guo, and B. Pei. Improving energy efficiency in industrial wireless sensor networks using SDN and NFV. In *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, pages 1–5, 2016.

[12] A. F. Murillo Piedrahita, V. Gaur, J. Giraldo, Á. A. Cárdenas, and S. J. Rueda. Leveraging software-defined networking for incident response in industrial control systems. *IEEE Software*, 35(1):44–50, 2018.

[13] S. Schriegel, T. Kobzan, and J. Jasperneite. Investigation on a distributed SDN control plane architecture for heterogeneous time sensitive networks. In *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*, pages 1–10, 2018.

[14] Z. Pang, X. Huang, Z. Li, S. Zhang, Y. Xu, H. Wan, and X. Zhao. Flow scheduling for conflict-free network updates in time-sensitive software-defined networks. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2020.

[15] T. Gerhard, T. Kobzan, I. Blöcher, and M. Hendel. Software-defined flow reservation: configuring IEEE 802.1q time-sensitive networks by the use of software-defined networking. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 216–223, 2019.

[16] Y. Bi, G. Han, C. Lin, Y. Peng, H. Pu, and Y. Jia. Intelligent quality of service aware traffic forwarding for software-defined networking/open shortest path first hybrid industrial Internet. *IEEE Transactions on Industrial Informatics*, 16(2):1395–1405, 2020.

[17] J. W. Guck, M. Reisslein, and W. Kellerer. Function split between delay-constrained routing and resource allocation for centrally managed QoS in industrial networks. *IEEE Transactions on Industrial Informatics*, 12(6):2050–2061, 2016.

[18] D. Cheng, X. Wang, S. Zhang, and M. Huang. SDN-based routing mechanism for industrial wireless sensor networks. In *2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 1274–1281, 2018.

[19] R. H. Jhaveri, R. Tan, and S. V. Ramani. Real-time QoS-aware routing scheme in sdn-based robotic cyber-physical systems. In *2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR)*, pages 18–23, 2019.

[20] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, C. Wang, and Y. Liu. A survey of machine learning techniques applied to software defined networking (SDN): research issues and challenges. *IEEE Communications Surveys Tutorials*, 21(1):393–430, 2019.

[21] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato. An intelligent traffic load prediction-based adaptive channel assignment algorithm in SDN-IoT: a deep learning approach. *IEEE Internet of Things Journal*, 5(6):5141–5154, 2018.

[22] R. Amin, M. Reisslein, and N. Shah. Hybrid SDN networks: a survey of existing approaches. *IEEE Communications Surveys Tutorials*, 20(4):3259–3306, 2018.

Technical Activities

A Conferences and Workshops

- IEEE International Conference on Cyber Physical and Social Computing (CPSCom) 2020
- IEEE Sensors Council Summer School 2020

B Special Issues in Academic Journals

- IEEE Internet of Things Journal special focus on Security, Privacy, and Trustworthiness in Intelligent Cyber-Physical Systems and Internet-of-Things (Submission deadline: Jan. 15, 2021)
- IEEE Transactions on Automation Science and Engineering special issue on Machine Learning for Resilient Industrial Cyber-Physical Systems (Submission deadline: Oct.01, 2020)
- SCIENCE CHINA Information Sciences special focus on Cyber-Physical Systems (Submisssion deadline: Sep. 15, 2020)

Call for Contributions

Newsletter of Technical Committee on Cyber-Physical Systems (IEEE Systems Council)

The newsletter of Technical Committee on Cyber-Physical Systems (TC-CPS) aims to provide timely updates on technologies, educations and opportunities in the field of cyber-physical systems (CPS). The letter will be published twice a year: one issue in February and the other issue in October. We are soliciting contributions to the newsletter. Topics of interest include (but are not limited to):

- Embedded system design for CPS
- Real-time system design and scheduling for CPS
- Distributed computing and control for CPS
- Resilient and robust system design for CPS
- Security issues for CPS
- Formal methods for modeling and verification of CPS
- Emerging applications, e.g. automotive system, smart energy system, biomedical device, etc.

Please directly contact the editors and/or associate editors by email to submit your contributions.

Submission Deadline:

All contributions must be submitted by **Jan. 1st, 2021** in order to be included in the February issue of the newsletter.

Editor:

- Bei Yu, Chinese University of Hong Kong, Hong Kong byu@cse.cuhk.edu.hk

Associate Editors:

- Xianghui Cao, Southeast University, China xhcao@seu.edu.cn
- Long Chen, Sun Yat-Sen University, China chenl46@mail.sysu.edu.cn
- Keke Huang, Central South University huangkeke@csu.edu.cn
- Wuling Huang, Chinese Academy of Science wuling.huang@ia.ac.cn
- Yier Jin, University of Florida, USA yier.jin@ece.ufl.edu
- Abhishek Murthy, Philips Lighting Research, USA abhishek.murthy@philips.com
- Rajiv Ranjan, Newcastle University, United Kingdom raj.ranjan@ncl.ac.uk
- Muhammad Shafique, Vienna University of Technology, Austria mshafique@ecs.tuwien.ac.at
- Yiyu Shi, University of Notre Dame, USA yshi4@nd.edu
- Ming-Chang Yang, Chinese University of Hong Kong, Hong Kong mcyang@cse.cuhk.edu.hk