International Journal of Advanced Computer Technology
ISSN: 2319-7900 www.ijact.org Volume 11, Issue 1, January-February 2022

Adaptive-Time Synchronization Algorithm for Semiconductor Superlattice Key Distribution

Chengyi Zhang[1], Jianguo Xie[2]
[1]University of California Los Angeles, CA 90095, America
[2]Beijing Electronic Science and Technology Institute, Beijing 100070, China.
[1]chriszhang09@ucla.edu, [2]xcharles@foxmail.com

DOI:- https://doi.org/10.5281/zenodo.5911209

Abstract: - *This paper presents a synchronisation algorithm for semiconductor superlattice key distribution symmetric encryption solution by optimising the Euclidian distance between the two chaotic waveforms generated in the receiver and the sender, respectively. This algorithm based on time synchronisation can reconstruct the generated waveforms in the receiver and the sender within the error of 5% to 6% (given that the waveforms were not perfectly congruent when they were created).*

Keywords: Synchronisation algorithm, symmetric key distribution, semiconductor superlattice key distribution.

1. INTRODUCTION

The motivation of this paper was to create a synchronisation algorithm for the semiconductor superlattice key distribution system invented by the Chinese Academy of Science (CAS). The semiconductor superlattice key distribution systems can create similar keys in the receiver and sender without time limitations, distance, and environment [1, 2]. The semiconductor superlattice key distribution system's keys are not replicable or predictable by a third party unless they gain access to one of these specific pairs of systems. However, semiconductor superlattice key distribution has a few downsides [3]:

- Semiconductor superlattice key distribution system creates keys that are not exactly congruent.
- Semiconductor superlattice key distribution system creates identical keys that are not synchronised in time.

Therefore, the semiconductor superlattice key distribution also needs information reconciliation, popularly used in physical key distribution, e.g., quantum key distribution, to ensure that the receiver and sender have identical keys [4]. The first step in the information reconciliation is to make the keys generated in the receiver and the sender as close as possible within the error of less than 10 %. In order to reach this target, this paper presented a synchronisation algorithm and demonstrated experimentally that the difference of the receiver and the sender in a semiconductor is a superlattice key distribution system could be within 5 % to 8 %, and this synchronisation algorithm was ready to practical application for semiconductor superlattice key distribution [5, 6]. The current synchronisation of semiconductor superlattice key distribution relies solely on the peak search algorithm, which is an algorithm that synchronises the keys by identifying the first peaks in the waveforms and matching the waveforms using first peaks [7]. In this algorithm, both the receiver and sender use the peak search algorithm to identify the position of the start of the key. However, the peak search algorithm is not accurate enough because the first peak could have different locations relative to the key. Although the peak search algorithm usually only results in the deviation of only one or two digits, the deviation would accumulate quickly since 64800 digits of keys were collected every time. If the hamming distance between the keys exceeds 10 %, they cannot be used for encryption. The synchronisation rate of the semiconductor superlattice key distribution by using the peak search algorithm was not good enough yet [8, 9]. Given such deficiencies of the peak search algorithm, an extra step needs to be adopted into the synchronisation algorithm. Here we proposed a new algorithm by using the first few digits of the key as the synchronisation sequence, which was sent to the other user for synchronisation calibration. Using this synchronisation sequence, the sender and the receiver can reach an agreement on the position of the start of the key that follows the end of the synchronisation sequence. We called this the adaptive-time synchronisation algorithm.

2. THEORETICAL ANALYSIS

The peak search and adaptive-time synchronisation algorithms can implement two synchronisation processes for the superlattice key distribution system. Section 2.1 represents the peak search algorithm for synchronisation for the superlattice key distribution system, and section 2.2 explains the adaptive time algorithm we proposed for better performance.

2.1 Peak Search

Peak Search is a time synchronisation algorithm in Superlattice Key Distribution. In a sequence of at least one full-cycle waveform, the front and back of the full-cycle waveform tend to be close to zero, as shown in Fig.1.

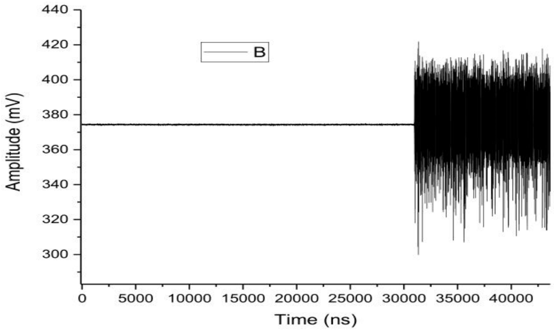

Figure 1. The waveform of the synchronisation sequence

The Peak Search algorithm aims to find the first peak of waveform B in Figure 1. First, we find the position of the peak by entering values for the parameter *ALIGN_THRESHOLD* and maximum height of the waveform and scan the waveform starting from zero. However, there are other cases where the starting point of the sequence is in waveform and cases where the starting point tends to be zero. As shown above, Figure 1 is an example where the starting point tends to be zero and Figure 2 is where the starting point is in the waveform. In the case of Figure 1, we designed an algorithm to find the start of the waveform. Find the start of the waveform by setting *ALIGN_WINDOW* and *SYNC_MARGIN* to scan the waveform from the starting point. After the start of the waveform is located, we can apply the Peak Search algorithm with the parameters of the location of the waveform to find the first peak in the waveform. In the case of Figure 2, we can directly apply the Peak Search algorithm.

Algorithm 1: Peak Search Algorithm

```
1.  Initialisation: - Initialise related parameters such as
    $ALIGN\_THRESHOLD, ALIGN\_WINDOW$
2.  Determine if the starting point of the sequence is a waveform.
    (a)  $if\ sum\ (rawdata\ (i{:}\ i + ALIGN\_WINDOW)) > ALIGN\_WINDOW * SYNC\_MARGIN;$
         i++;
    (b)  $else\ sync\_position = i + ALIGN\_WINDOW$
3.  Find the peak of the waveform
    (a)  $for\ i = sync\_position : length(rawdata)$
    (b)  $if\ rawdata(i) > ALIGN\_THRESHOLD * max(rawdata)$
    (c)  $if\ rawdata(i) > peakValue$
    (d)  $peakValue = rawdata(i), aligndPosition = i$
    (e)  $else\ break;$
```

After the peak location is determined, we can choose a certain waveform length following the first peak as the key for encryption. A similar waveform can be extracted on the receiver side by applying the peak search algorithm on the output from the semiconductor superlattice key distribution system, which can be used as the key for decryption. It is worth noticing that there is essentially no exchange of information for using the peak search algorithm even though the efficiency and error rate is poor.

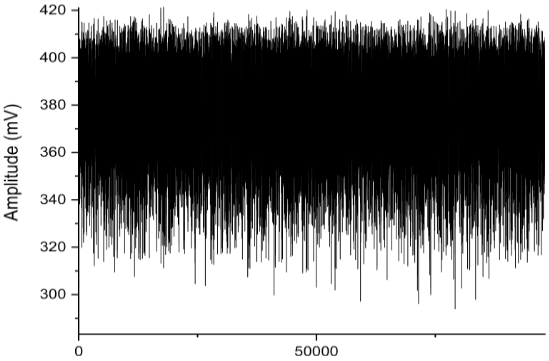

Figure 2. The original waveform

2.2 Adaptive-Time Synchronisation Algorithm

The first step of the adaptive time synchronisation algorithm is for the sender to select a sequence from its superlattice key distribution system's waveform output and send it to the receiver. For convenience and efficiency, we used the peak search algorithm to locate the start of the waveform and use the first 1200 digits as the synchronisation sequence. After selecting the sequence for synchronisation through the peak identification algorithm, we tried to align the synchronisation sequence to the corresponding parts in the other system in the superlattice key distribution system's waveform by calculating the Euclidean distance.

Algorithm 2: Adaptive Time Synchronization algorithm

```
1.  Initialise parameters like the synchronisation sequence sent from the sender.
2.  Find the position of the portion of the waveform that is the most similar to the synchronisation sequence:
    $arr = []$
    $for\ i = 1 : length(wave)$
    $diff = wave\ (i + length(sync\_sequence)) - sync\_sequence$
    $arr.\ append\ (diff)$
    $pos = min(arr)$
    The key starts at $pos + length(sync\_sequence)$ for the receiver and the key start right after synchronisation
    sequence for the sender.
```

The algorithm is essentially searching for a segment inside the waveform that has the least Euclidean distance to the synchronisation sequence. The algorithm would run from the start of the waveform, cut out a segment with the same length as the synchronisation sequence and calculate the Euclidean distance between the selected segment and the synchronisation sequence. Then, a segment is reselected by shifting one digit to the right, and its Euclidean distance is recorded from the synchronisation sequence. The previous step is repeated until the last digit can generate a key of 64800 digits too. After the algorithm has gone through all the possibilities, the algorithm selects the starting position, which leads to the least Hamming distance between the two waveforms as the decryption key for the receiver. The sender uses the waveform section following the synchronisation sequence as the encryption key.

3. EXPERIMENTAL DESIGN

In Section 2, we explained how to synchronise the sequence of the keys of the Superlattice Key Distribution System. We proposed an Adaptive Time Synchronisation algorithm. To further verify the accuracy of our algorithm and its effectiveness in practical applications, we designed the following experiment.

1. Since the sequence we send for synchronisation may affect the synchronisation accuracy, we tested three different driving sequences or inputs to test the reliability of our algorithm. The three driving sequences are pseudorandom, periodic, and superposition sequences.
2. We compare the advantages and disadvantages of the Adaptive Time Synchronisation algorithm and Peak Search algorithm through several experiments and three-time synchronisation sequences.

We evaluate the results by the control-variable method. Hamming distance has been used to characterise the errors between the keys after synchronisation in our experiment. The experiment is conducted in the following manner.

1. Generate pseudorandom, periodic, periodic driving sequences for synchronisation, followed by a normal sequence used as an excitation signal.
2. Texas Instrument's data acquisition software collects sequences. Since we were not sure if we would be able to collect a continuous and full-period of the waveform by only collecting one period-length of the output, we collected two-period-length of the output, which guarantees at least one whole period of the waveform. The waveform of the three driving sequences is collected. This process is gone through both devices.
3. Synchronise the two waveforms collected on the two machines mentioned above using the Adaptive Time Synchronisation algorithm. The synchronisation results are evaluated by the Hamming distance between the two waveforms after they are synchronised. In this case, a portion of the waveform is collected from the sender to the receiver. This process is conducted for all the waveforms collected through the three driving sequences.
4. Synchronise the three waveforms mentioned above using the Peak Search algorithm and evaluate the hamming distance in the waveforms.
5. The advantages and disadvantages of the Adaptive Time Synchronisation algorithm compared with Peak Search algorithm using Hamming distance.

For one driving sequence in the above experiment, both the sender and the receiver collect 30 output sequences for synchronisation and obtain 900 sets of Hamming distance. We compare and analyse the 900 sets of Hamming distance using statistical methods. For the specific experimental results, please refer to the next section.

4. EXPERIMENTAL RESULTS

For the sake of the symmetric encryption scheme, the difference (calculated by the hamming distance) should be less than 8%, which is the criterion we are using for the experiment. Table 1 shows the results of the percentages of time in which the respected algorithm failed to synchronise the keys.

Table 1. Results of two algorithms

Type	Adaptive time synchronisation	Peak Search
Periodic	5.78 %	31 %
Periodic superposition	6.11 %	15 %
Pseudo random	5.89 %	30.9 %

As shown above, the adaptive time synchronisation algorithm is a lot more accurate than the peak search algorithm, with only 5.78%, 6.11%, and 5.89% of failure rates compared to 31%, 15%, and 30.9% for the peak search algorithm. In addition to poor accuracy, the peak search algorithm failed for the synchronisation 30~40 times among 900 experiments, as shown below (Table 2 and Figure 3).

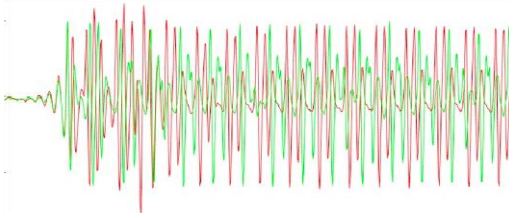

Figure 3. Result of Peak Search algorithm

Table 2 shows the peak search algorithm's performance when dealing with particular sets of periodic superposition sequences, in which the error is as high as 28%.

Table 2. Results of Peak Search algorithm

Sets	1	2	3	4
Hamming distance	28.51%	28.34%	28.48%	28.25%

We suspect that the problem with this algorithm is that we need to reset parameters like *ALIGN_WINDOW* for suspect data, or else the performance not be optimised. However, as it turns out, the problem is associated with the superlattice key distribution system itself. As shown below, one of the waveforms was significantly longer than the other. Even though this is already a severe error, such a problem is still a good result for the superlattice key distribution system, as it involves complicated physical devices.

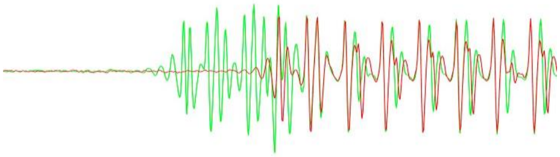

Figure 4. Result of Adaptive Time Synchronisation algorithm

Even in this extreme circumstance, the adaptive time synchronisation algorithm still successfully synchronised the two waveforms, with the results shown in Table 3.

Table 3. Results of Peak Search algorithm

Sets	1	2	3	4
Hamming distance	5.60%	2.58%	5.80%	3.01%

It shows that the adaptive synchronisation algorithm is more robust than the peek search algorithm for potential errors in the superlattice key distribution system.

CONCLUSION

We presented the adaptive-time synchronisation algorithm for superlattice key distribution, which has been demonstrated to be more accurate and robust to extreme circumstances than the previous peak search algorithm. When synchronising the three sets of data, the adaptive synchronisation algorithm maintained a success rate beyond 93%, while the conventional peak search algorithm can have only a success rate of 70%. Although this algorithm is better in terms of performance, we still have to improve its speed and efficiency, as it still runs slower than the peak search algorithm. The lack of efficiency with the adaptive synchronisation algorithm requires intensive calculations on the entire waveform, a tedious task compared to the peak search algorithm, which only requires going through a small portion of the waveform. Therefore, further improvements of the adaptive synchronisation algorithm will be attempts to reduce the number of iterations when determining the starting point of the key.

ACKNOWLEDGMENTS

National Cryptography Development Fund supported this work during the 13th Five-Year Plan Period (MMJJ20180112), China's National Defence Basic Scientific Research program (JCKY2019102C001).

REFERENCES

[1]. Liu, W., Yin, Z., Chen, X., Peng, Z., Song, H., Liu, P., Tong, X. and Zhang, Y., 2018. A secret key distribution technique based on semiconductor superlattice chaos devices. Sci. Bull, 63(16), pp.10341036.

[2]. Li, W., Aviad, Y., Reidler, I., Song, H., Huang, Y., Biermann, K., Rosenbluh, M., Zhang, Y., Grahn, H.T. and Kanter, I., 2015. Chaos synchronisation in networks of semiconductor superlattices. EPL (Euro physics Letters), 112(3), p.30007.

[3]. Huang, Y., Qin, H., Li, W., Lu, S., Dong, J., Grahn, H.T. and Zhang, Y., 2014. Experimental evidence for coherence resonance in a noise-driven GaAs/AlAs superlattice. EPL (Euro physics Letters), 105(4), p.47005.

[4]. Yin, Z., Song, H., Zhang, Y., Ruiz-García, M., Carretero, M., Bonilla, L.L., Biermann, K. and Grahn, H.T., 2017. Noise-enhanced chaos in a weakly coupled GaAs/ (Al, Ga) As superlattice. Physical Review E, 95(1), p.012218.

[5]. Einstein, A., 1905. Zur electrodynamic bewegter körper. Annalen der physik, 4,

[6]. Ho, C., Lamas-Linares, A. and Kurtsiefer, C., 2009. Clock synchronisation by remote detection of correlated photon pairs. New Journal of Physics, 11(4), p.045011.

[7]. Eddington, A.S., 1923. The mathematical theory of relativity. The University Press.

[8]. Xie, J., Wu, H., Xia, C., Ding, P., Song, H., Xu, L. and Chen, X., 2021. High throughput error correction in information reconciliation for semiconductor superlattice secure key distribution. Scientific Reports, 11(1), pp.1-9.

[9]. Wu, H., Yin, Z., Xie, J., Ding, P., Liu, P., Song, H., Chen, X., Xu, S., Liu, W. and Zhang, Y., 2021. Design and implementation of true random number generators based on semiconductor superlattice chaos, Microelectronics Journal, p.105119.

BIOGRAPHIES

CHENG YI ZHANG is pursuing a B.S. degree at the University of California Los Angeles, CA 90095, America. His teaching and research areas include mathematics and computer design.
chriszhang09@ucla.edu

JIANGUO XIE is pursuing an M.S. degree in Beijing Electronic Science and Technology Institute, Beijing 100070, China. His teaching and research areas include error correction codes, cryptography, and programming.
xcharles@foxmail.com