Piecewise Multi-Parameter Tangent/Cotangent Function and Its Application in Chaotic Pseudorandom Bit Generator

Qi Wu*
Department of E-Commerce, Jiangxi University of Finance & Economics, China

*Corresponding author email: wuqiocjzjd@126.com

Abstract. To increase the number of parameters of 1-dimensional discrete chaotic mappings, employing translation & scaling, this letter creatively designs a piecewise 5-parameters Tangent/Cotangent Function. Both bifurcation diagram and Lyapunov exponent spectrum illustrate that the proposed function seems to be inappropriate for cryptographic applications, due to its narrow chaotic area. However, after applying it to devising chaotic pseudorandom bit generator, it outperforms skew tent mapping, whose chaotic area is full. It might be applied to some other fields as well, such as cryptographic hash functions, which will possibly be paid attention to in the future.

Keywords: 1-Dimensional Discrete Chaotic Mapping; Piecewise 5-Parameters Tangent/Cotangent Function; Pseudorandom bit generator.

1. Introduction
Among chaotic mappings, 1-Dimensional Discrete Chaotic Mappings (abbreviated as 1DDCM hereafter) are of the simplest form and highest efficiency [1]. However, in terms of our experiments [2-12], classic 1DDCM, such as piecewise linear mapping (skew tent mapping in most cases) and Logistic mapping, are defective: The chaotic area of Logistic mapping is highly narrow and incontinuous, which makes it difficult to select strong parameters; Although skew tent mapping owns broad chaotic area, when applied to devising pseudorandom bit generators, its strong cipher space is confined in a small adjacent area of 0.5.

Novices in chaotic theory, especially those who have learnt Set Theory, usually pay little attention to the size of parameter space. For example, many deem that, for a certain parameter, it’s useless to enlarge its strong parameter space from interval (0,1) to (0,15), due to Card((0,1))=Card((0,15))=C1(Card(A) stands for the cardinality of a set A.). In the world of math, it’s true; Nonetheless, in the world of computers, it’s not. For a computer, the number of numbers in interval (0,1) is finite. While strong parameter space is enlarged from (0,1) to (0,15), there are many more strong parameters for selection, and the system could withstand brute force attack much more easily.

Nowadays, researchers in chaotic theory pay little attention to 1DDCM [13-16], as it’s widely believed that 1DDCM owns too few parameters and is apt to reveal its phase trajectory. Nevertheless, we persist that research on 1DDCM is significant. On one hand, owing to its simple form and lucid chaotic properties, 1DDCM is the best tutorial for novices in chaos. On the other hand, thanks to its high efficiency, 1DDCM provides a sound base for high-dimensional chaotic mappings, for instance, coupling several 1DDCMs [2-4, 6-10].

Recently, we have been working at designing multi-parameter 1DDCMs, whereas most of our experiments end in failure. “Unimodal mappings usually possess good chaotic properties.” [1] is not
universal. According to our experiments, many 1-dimensional multi-parameter chaotic mappings, such as piecewise parabola, piecewise inverse proportional mapping, general cubic mapping, piecewise exponential mapping, piecewise logarithmic mapping, piecewise arc-tangent/arc-cotangent mapping, piecewise cubic mapping, illustrate no chaotic property at all.

Next, we propose our only recent success, i.e. Piecewise Multi-Parameter Tangent/Cotangent Function (abbreviated as PMPTCF henceforth). This function is unimodal, possessing 5 parameters, and overwhelms Logistic mapping and skew tent mapping.

The upcoming parts are as follows: Section 2 designs and analyses PMPTCF. Section 3 designs and analyses a PRBG based on PMPTCF. Section 4 concludes.

2. Design & Analysis of PMPTCF

2.1. Design of PMPTCF

Given a unimodal mapping defined on interval [0,1], demanding its peak at the point (t,1), its left segment ascends and right segment descends. Then, its left segment and right segment could be set as tangent function and cotangent function, respectively, as these two are constantly ascendent/descendent in a single period. Employing 2 elementary transformations, i.e. translation and scaling (Rotation will introduce sine and cosine functions. It might enlarge the strong parameter space, but will certainly decrease the efficiency, so it’s out of our consideration in this paper.), we could obtain the formula for PMPTCF:

\[
f(x) = \begin{cases} 
  d_1 \tan(c_1 x + a_1) + b_1, & x \leq t \\
  d_2 \cot(c_2 x + a_2) + b_2, & x > t 
\end{cases}
\]

In Equ. (1), \(t\) designates the x-coordinate of the peak of the unimodal mapping; \(a_1\) and \(b_1\) represent the amount of translation for tangent function, \(c_1\) and \(d_1\) stand for the extent of scaling for tangent function; \(a_2\) and \(d_2\) represent the amount of translation for cotangent function, \(c_2\) and \(d_2\) stand for the extent of scaling for cotangent function.

It should be noted that, the 9 parameters in Equ. (1) couldn’t be selected at will. They have to ensure that the left segment passes through points (0,0) and (t,1), the right segment passes through points (t,1) and (1,0). Therefore, it could be got:

\[
\begin{align*}
  d_1 \tan(a_1 + b_1) &= 0 \\
  d_1 \tan(c_1 t + a_1) + b_1 &= 1 \\
  d_2 \cot(c_2 t + a_2) + b_2 &= 1 \\
  d_2 \cot(c_2 + a_2) + b_2 &= 0
\end{align*}
\]

Via some simple formulas for trigonometric functions on Equ. (2), it’s easy to know that we could select parameters of PMPTCF in a way: pick \(t\) at will from interval (0,1), then

1. select \(c_1\) and \(d_1\) at will from interval \((0, +\infty)\), calculate \(a_1 = \frac{\arccos[2d_1 \sin(c_1 t) - \cos(c_1 t)] - c_1 t}{2}\) and \(b_1 = -d_1 \tan a_1;\)
2. select \(c_2\) and \(d_2\) at will from interval \((0, +\infty)\), calculate \(a_2 = \frac{\arccos[2d_2 \sin(c_2(t-1)) + \cos(c_2(t-1))] - c_2(t-1)}{2}\) and \(b_2 = -d_2 \cot(c_2 + a_2).\)

Till now, readers could understand why we always say that PMPTCF possesses 5 parameters: among the 9 parameters, only 5 could be selected at will; the other 4 are settled once the 5 are selected, which means there’s no freedom of choice for them. Even though, the parameter space of PMPTCF has overwhelmed skew tent mapping (only 1 parameter) and Logistic mapping (Nominally 1 parameter. In fact, there’s no strong parameter for it except \(\mu = 4\), indicating it owns 0 parameter.) It’s known that parameter space is the keypoint for withstanding brute force attacks, and should be taken into account for all cryptographic fields, such as PseudoRandom Bit Generator (abbreviated as PRBG hereafter).
2.2. Analysis of PMPTCF

There’re many approaches for analyzing chaotic mappings. Due to the limitation of length, this letter analyzes PMPTCF only via bifurcation diagram and Lyapunov exponent spectrum on a small fraction of parameter values.

Given $x_0=0.1$, $c_1=1.85$, $d_1=1$, $c_2=1.05$, $d_2=1.8$, let $t$ go from 0 to 1 with step 0.001, iterate PMPTCF 500 times, filtering the first 200 times, the value of $x$ for the last 300 times is depicted, as shown in figure 1.

![Bifurcation diagram of PMPTCF](image1)

**Figure 1.** Bifurcation diagram of PMPTCF ($t$ as horizontal axis).

Figure 1 is far different from bifurcation diagrams of most papers: The value of $x$ often goes out of the interval [0,1]. The reason is, in the definition of PMPTCF, it’s not guaranteed that PMPTCF is a purely unimodal mapping. For instance, many segments of tangent function might appear in the domain $[0,t]$, in this way, the image set corresponding to domain $[0,t]$ will probably go out of interval [0,1]. This phenomenon is not what we want. Considering translation and scaling, discussing the parameters for tangent and cotangent functions respectively, it’s not difficult to evade this phenomenon. Nevertheless, in this way, the 5 parameters of PMPTCF are not mutually independent, which makes the computation of parameter space for PMPTCF harder. Thus, in this paper, we don’t evade the phenomenon deliberately. We just select parameters as prudently as possible.

From figure 1, it could be seen that PMPTCF confines the range of $t$ heavily: only in a small adjacent area of 0.5 are dots relatively dense. Intuitively, only this area might supply chaotic area for cryptographic applications.

Given $x_0=0.1$, $t=0.49$, $d_1=1$, $c_2=1.05$, $d_2=1.8$, let $c_1$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 500 times, filtering the first 200 times, the value of $x$ for the last 300 times is depicted, as shown in figure 2.
From figure 2, it could be seen that PMPTCF confines the range of $c_1$ loosely: for most scope, no periodic area could be seen, which is quite beneficial to cryptographic applications. Given $x_0=0.1$, $t=0.49$, $c_1=1.85$, $c_2=1.05$, $d_2=1.8$, let $d_1$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 500 times, filtering the first 200 times, the value of $x$ for the last 300 times is depicted, as shown in figure 3.

From figure 3, it could been that PMPTCF confines the range of $d_1$ strictly: most of interval [1,2] is not applicable, which is adverse to cryptographic applications. Given $x_0=0.1$, $t=0.49$, $c_1=1.85$, $d_1=1$, $d_2=1.8$, let $c_2$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 500 times, filtering the first 200 times, the value of $x$ for the last 300 times is depicted, as shown in figure 4.
In figure 4, cases similar to figure 1 emerge: the value of \( x \) goes out of interval \([0,1]\), the reason for which is the same as that aforementioned. From figure 4, it could be seen that PMPTCF confines the range of \( c_2 \) strictly: most of interval \([1,2]\) is not applicable, which is to the disadvantage of cryptographic applications.

Given \( x_0=0.1 \), \( t=0.49 \), \( c_1=1.85 \), \( d_1=1 \), \( c_2=1.05 \), let \( d_2 \) go from 0.5 to 2 with step 0.001, iterate PMPTCF 500 times, filtering the first 200 times, the value of \( x \) for the last 300 times is depicted, as shown in figure 5.

From figure 5, it could be seen that PMPTCF confines the range of \( d_2 \) loosely: for most scope, no periodic area could be seen, which is to the benefit of cryptographic applications.
In all, judging from bifurcation diagram, PMPTC F performs unsatisfactorily: among the 5 parameters, only 2 are in favor of cryptographic applications; the other 3, especially $t$, possess a heavily confined range.

Next, let’s analyze the Lyapunov exponent spectrum of PMPTCF. Given $x_0=0.1$, $c_1=1.85$, $d_1=1$, $c_2=1.05$, $d_2=1.8$, let $t$ go from 0 to 1 with step 0.001, iterate PMPTCF 2000 times, filtering the first 1000 times, the Lyapunov exponent spectrum is depicted in terms of the $x$ value in the last 1000 iterations, as shown in figure 6.

![Figure 6. Lyapunov exponent spectrum of PMPTCF ($t$ as horizontal axis).](image)

From figure 6, it could be seen that, PMPTCF dwells in chaotic area only when $t$ is near 0.5. Other scope basically could be ignored for cryptographic applications.

Given $x_0=0.1$, $t=0.49$, $d_1=1$, $c_2=1.05$, $d_2=1.8$, let $c_1$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 2000 times, filtering the first 1000 times, the Lyapunov exponent spectrum is depicted in terms of the $x$ value in the last 1000 iterations, as shown in figure 7.

![Figure 7. Lyapunov exponent spectrum of PMPTCF ($c_1$ as horizontal axis).](image)
From figure 7, it could be seen that, for most values of $c_1$, PMPTCF dwells in chaotic area, which is quite suitable for cryptographic applications. Given $x_0=0.1$, $t=0.49$, $c_1=1.85$, $c_2=1.05$, $d_2=1.8$, let $d_1$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 2000 times, filtering the first 1000 times, the Lyapunov exponent spectrum is depicted in terms of the $x$ value in the last 1000 iterations, as shown in figure 8.

![Figure 8. Lyapunov exponent spectrum of PMPTCF ($d_1$ as horizontal axis).](image)

From figure 8, it could be seen that, for most of interval $[1,2]$ from which $d_1$ is selected, PMPTCF dwells in periodic area, which is adverse to cryptographic applications. Given $x_0=0.1$, $t=0.49$, $c_1=1.85$, $d_1=1$, $d_2=1.8$, let $c_2$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 2000 times, filtering the first 1000 times, the Lyapunov exponent spectrum is depicted in terms of the $x$ value in the last 1000 iterations, as shown in figure 9.

![Figure 9. Lyapunov exponent spectrum of PMPTCF ($c_2$ as horizontal axis).](image)

From figure 9, it could be seen that, for most of interval $[1,2]$ from which $c_2$ is selected, PMPTCF dwells in periodic area, which is to the disadvantage of cryptographic applications.
Given $x_0=0.1$, $t=0.49$, $c_1=1.85$, $d_1=1$, $c_2=1.05$, let $d_2$ go from 0.5 to 2 with step 0.001, iterate PMPTCF 2000 times, filtering the first 1000 times, the Lyapunov exponent spectrum is depicted in terms of the $x$ value in the last 1000 iterations, as shown in figure 10.

Figure 10. Lyapunov exponent spectrum of PMPTCF ($d_2$ as horizontal axis)

From figure 10, it could be seen that, for most value range of $d_2$, PMPTCF dwells in chaotic area, which is superb for cryptographic applications.

In all, in the light of Lyapunov exponent spectrum, PMPTCF performs unsatisfactorily: for the 5 parameters, only 2 are with wide chaotic area, which is suitable for cryptographic applications; the other 3, especially $t$, are of strictly confined scope. Generally, it seems that PMPTCF is not suitable for applying to cryptography: Although it owns many parameters and wide value range, a considerable part of the range is periodic area, not chaotic area, which is demanded by cryptographic applications. This seems far poorer than skew tent mapping, whose chaotic area is full. Is it true? Next, we try to apply PMPTCF to PRBG.

3. Design & Analysis of PRBG Based on PMPTCF

Here, we follow the framework of Ref. [12] to design a simple PRBG. Given $x_0$, $t$, $c_1$, $d_1$, $c_2$, $d_2$, compute the corresponding $a_1$, $b_1$, $a_2$, $b_2$, after each iteration of PMPTCF, emit a bit $s_i$ via comparing $x_i$ and 0.5:

$$s_i = \begin{cases} 0, & x_i < 0.5 \\ 1, & x_i \geq 0.5 \end{cases}$$

Set $x_0=0.1$, when $t$ goes from 0.488 to 0.505 with step 0.001, $c_1$ goes from 0.5 to 2 with step 0.1, $d_1$ goes from 0.5 to 2 with step 0.1, $c_2$ goes from 0.5 to 2 with step 0.1, $d_2$ goes from 0.5 to 2 with step 0.1, for these $18 \times 16 \times 16 \times 16 = 1179648$ kinds of different parameter values, generate binary sequences of length 50000 respectively. Through 5 categories of pseudorandom tests (monobit, serial, poker, runs and auto-correlation tests, level of significance set to 0.05), we find out there’re 437 parameter tuples passing all the 11 tests. This result seems unsatisfactory, but it has overwhelmed skew tent mapping thoroughly (when the peak of the tent $a$ goes from 0 to 1 with step 0.001, only 2 values could pass all the 11 tests). Due to the limited performance of our computer, we only let $c_1$, $d_1$, $c_2$, $d_2$ go through a tiny part of their possible scope (Theoretically, these 4 parameters could traverse interval $(0, +\infty)$). Thus, the strong parameter space of the PRBG could be far beyond this.
Next, owing to the limitation of length, we give a small part of results for the pseudorandom tests (table 1-6). Readers unfamiliar with those tests could refer to Ref. [2-12] for some basic knowledge.

**Table 1. Results of monobit test.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | $X^2$ | Critical Value |
|------|-------|-------|-------|-------|-------|----------------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 0.1037| Σ                |
| 0.495| 2     | 0.9   | 1.8   | 1     | 2.3667|                |
| 0.504| 1.8   | 1     | 2     | 0.9   | 2.8577|                |

**Table 2. Results of serial test.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | $X^2$ | Critical Value |
|------|-------|-------|-------|-------|-------|----------------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 3.0070|                |
| 0.495| 2     | 0.9   | 1.8   | 1     | 2.9659|                |
| 0.504| 1.8   | 1     | 2     | 0.9   | 2.8684|                |

**Table 3. Results of poker test.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | $X^2(m=4)$ | Critical Value |
|------|-------|-------|-------|-------|------------|----------------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 18.0314    | 25             |
| 0.495| 2     | 0.9   | 1.8   | 1     | 16.7667    |                |
| 0.504| 1.8   | 1     | 2     | 0.9   | 23.9680    |                |

**Table 4. Results of runs test.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | $X^2$ | Critical Value |
|------|-------|-------|-------|-------|-------|----------------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 26.2145| 31.4          |
| 0.495| 2     | 0.9   | 1.8   | 1     | 29.4775|                |
| 0.504| 1.8   | 1     | 2     | 0.9   | 22.6755|                |

**Table 5. Results of auto-correlation test.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | $|X|(d=10000)$ | Critical Value |
|------|-------|-------|-------|-------|---------------|----------------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 0.81          | 1.96           |
| 0.495| 2     | 0.9   | 1.8   | 1     | 0.39          |                |
| 0.504| 1.8   | 1     | 2     | 0.9   | 0.38          |                |

**Table 6. Results of linear complexity.**

| $t$  | $c_1$ | $d_1$ | $c_2$ | $d_2$ | Linear Complexity | $N/2$ |
|------|-------|-------|-------|-------|-------------------|-------|
| 0.49 | 1.85  | 1     | 1.05  | 1.8   | 499               | 500   |
| 0.495| 2     | 0.9   | 1.8   | 1     | 500               |       |
| 0.504| 1.8   | 1     | 2     | 0.9   | 500               |       |
As Berlekamp-Massey algorithm is too time-consuming, when calculating the linear complexity, we reduce the length of the 3 sequences to 1000 bits, with other conditions unchanged. From table 1-6, it could be seen that, all these 3 sequences have passed the 5 categories of pseudorandom tests and their linear complexity are close to half of their length (similar to the output of BSS). They’re quite suitable for cryptographic scenarios needing pseudorandom numbers, such as image encryption and hash function.

4. Conclusion
Different from former 1DDCM [17-18], which increase the number of parameters solely by increasing the number of segments of piecewise mapping (usually called “sawtooth mapping”), this paper maintains 2 segments, employing translation and scaling, designs a piecewise 5-parameter tangent/cotangent function. Bifurcation diagram and Lyapunov exponent spectrum demonstrate that, the function doesn’t possess a wide chaotic area and is not quite suitable for cryptographic applications. Nonetheless, when applied to design of PRBG, its performance overwhelms that of skew tent mapping, whose chaotic area is full. Moreover, this is just the performance within a small scope of value range for the 5 parameters (owing to the humble performance of our computer). While the value range is enlarged, even better performance could be expected.

In the future, we tend to combine PMPTCF and our proposed ways of coupling, try to construct coupled chaotic systems with better chaotic properties than our proposed ones based on skew tent mapping [2-4, 6-10], and apply them to all kinds of cryptographic scenarios.

Acknowledgments
This research is financially supported by the Science and Technology Project of Provincial Education Department of Jiangxi for Youth (GJJ180288). Thanks to Professor Zhihong Guan, Meng Jia and Jian Shu for much helpful advice.

References
[1] Hao B L 2003 Starting with Parabolas (Shanghai: Shanghai Science and Technology Education Press)
[2] Tan Z W and Wu Q 2008 Proc. Int. Conf. on Computational Intelligence and Security (Suzhou)
[3] Tan Z W and Wu Q 2008 Proc. Int. Symp. on Intelligent Information Technology Application (Shanghai) p 224
[4] Wu Q, Tan Z W and Wan C X 2011 Journal of Chinese Computer System vol. 32 p 639
[5] Wu Q 2015 Proc. Int. Conf. on Cyber-Enabled Distributed Computing and Knowledge Discovery (Xi’an)
[6] Wu Q 2016 Computer Engineering & Science vol. 38 p 2197
[7] Wu Q 2016 Proc. Int. Conf. on Industrial Informatics – Computing Technology, Intelligent Technology, Industrial Information Integration (Wuhan) p 341
[8] Wu Q 2018 Proc. Int. Conf. on Smart Materials, Intelligent Manufacturing and Automation (Hangzhou)
[9] Wu Q 2018 Proc. Int. Conf. on Network and Information Systems for Computers (Wuhan)
[10] Wu Q 2018 Proc. Int. Conf. on Intelligent Information Processing (Guilin)
[11] Wu Q 2019 Proc. Int. Symp. on Advances in Electrical, Electronics and Computer Engineering (Zhuhai)
[12] Wu Q 2019 Proc. Int. Symp. on Cyberspace Safety and Security (Guangzhou)
[13] Zhao X 2012 Application Research of Computers vol. 29 p 913
[14] Liu L D and Song H S 2014 Electronic Design Engineering vol. 22 p 123
[15] Li C H, Li Y B and Zhao L 2014 Application Research of Computers vol. 31 p 1403
[16] Chen W J 2015 Journal of Sichuan Normal University (Natural Science) vol. 38 p 391
[17] Luo S J, Qiu S S and Chen X 2012 Journal of Shenzhen University Science and Engineering vol. 29 p 335
[18] Su X and Yang G 2014 Computer Technology and Development vol. 24 p 167