Regular Paper

Communications of the CCISA
Vol. 28 No. 3 Aug. 2022

應用於物聯網安全之滑動視窗細胞自動機串流加密器

吳錫聰[1,*]、吳國華[2]
[1,2]國立宜蘭大學電子工程學系
[1]stwu@niu.edu.tw、[2]miuone16@gmail.com

摘要

在物聯網時代通訊安全是相當重要的一個部分，資料傳輸時對敏感資料進行加密是不可或缺的，其中串流加密法為即時通訊廣為使用的加密方法。近來相續有細胞自動機(Celluar Automata) 在串流加密器的應用研究被發表，透過不同的細胞自動機規則產生串流加密器所需的金鑰流，細胞自動機具運算簡單與快速的優點，適於資源較缺乏的物聯網。本論文中我們結合滑動視窗、細胞自動機與位元置換的提出金鑰流產生器，輸出的金鑰流透過 NIST SP800-22 的亂度測試則有 91%以上的通過率。

關鍵詞：細胞自動機、金鑰流產生器、滑動視窗、串流加密器

* 通訊作者 (Corresponding author.)

Regular Paper

Communications of the CCISA
Vol. 28 No. 3 Aug. 2022

The Stream Cipher Based on Sliding Window Cellular Automata in Secure IoT

Shyi-Tsong Wu[1*], Guo-Hua Wu[2]
[1,2]Department of Electronic Engineering, National Ilan University, Taiwan, R.O.C.
[1]stwu@niu.edu.tw, [2]miuone16@gmail.com

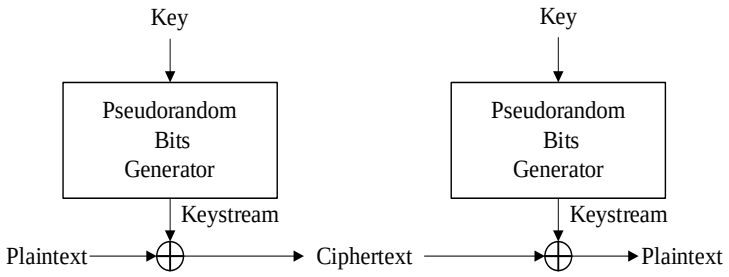
Abstract

In today's Internet of Things era, communication security is a very important issue. Encrypting sensitive data during data transmission is essential. Stream cipher is one of the most widely used encryption methods in real time communications. Recently, there have been many published studies on the applications of Cellular Automata in stream ciphers, and the keystream produced by different cellular automata rules. Cellular automata is optimized for simple and fast computation, suitable for the Internet of Things with limited resources. In this paper, we propose a keystream generator that combines sliding window, cellular automata and bit permutation. The generated keystream produced by the proposed scheme has the pass rate at least 91% under the randomness test of NIST SP800-22.

Keywords: Celluar Automata, Keystream Generator, Sliding Window, Stream Cipher

壹、簡介

現在的密碼系統為求安全考量，一般需要相當複雜的運算，然而物聯網 (Internet of Things，IoT) 在設備有限制的情況下，太過於複雜的系統將會導致通訊效率的降低[7][13][16]，在效率的考量下細胞自動機 (Celluar Automata，CA) 是一個非常適合的方法[13]。目前細胞自動機已經被應用在密碼學領域中，很多基於細胞自動機的加密系統相繼被提出[1][5][12][13][14]，在細胞自動機簡單的邏輯運算中，不同的細胞自動機規則互相結合，可以產生安全性高的輸出[9][15]。本文中我們結合滑動視窗、細胞自動機與位元置換，提出適用於資源較缺乏的物聯網系統的金鑰流產生器 (keystream Generator)，透過疊代與細胞自動機的邏輯計算，產生安全性高的金鑰流 (keystream)，以運用在串流加密器 (stream cipher)上。串流加密器是一種對稱式的加密法，如圖一所示，利用明文與隨機金鑰流進行互斥或(XOR)運算後取得密文，解密方法只要密文再與相同的隨機金鑰流進行互斥或運算就可以取得明文，以保護使用者的安全。



圖一：串流加密流程圖

貳、細胞自動機

金鑰流產生器在物聯網通訊安全中，扮演著舉足輕重的腳色，資料要在各種不同的裝置中進行通訊[13]，所選擇的金鑰流產生器就非常重要，邏輯簡單的方法可以使得在加密的過程中，提升整體通訊的效率，Neumann 提出細胞自動機的概念[11]，細胞自動機擁有非線性與運算簡單的特性，根據自己本身的狀態與鄰域的狀態，透過不同的規則，產生各種不同的結果，以一維空間的細胞自動機來說，一次疊代輸出由 0 或 1 的邏輯值所組成，在數學上，可以定義為下列方程式 (1)：

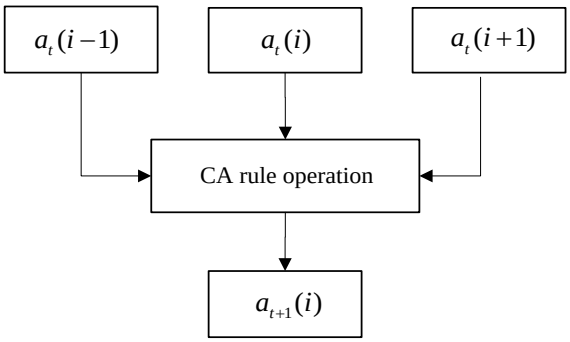
$$a_{t+1}(i) = f(a_t(i-1), a_t(i), a_t(i+1)) \quad (1)$$

其中 $a_t(i)$代表一個位元的狀態，i 代表位元在序列中的位置，t 代表當前序列狀態，$a_{t+1}(i)$代表下次序列的狀態，一個位元進行運算疊代到下個位元時，會根據細胞本身與鄰域狀態進行邏輯運算，也就是說自己本身與左側、右側的三個位元根據不同的規則，產生下

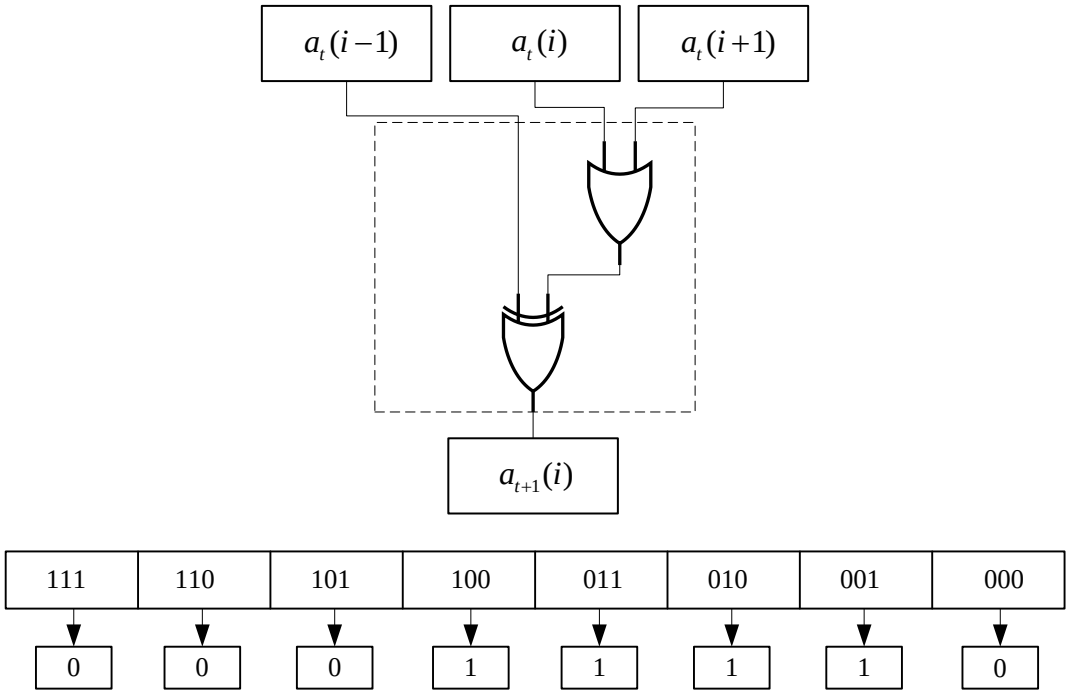
一次疊代的位元狀態，如圖二所示。細胞自動機依不同的邏輯運算有不同的規則編號，運算有方程式 (2) 代表規則 30 的邏輯運算規則：

$$a_{t+1}(i) = a_t(i-1) \oplus (a_t(i) \bigcup a_t(i+1)) \tag{2}$$

圖三代表細胞自動機規則 30 運算的輸入與輸出結果。根據各種不同的邏輯運算以制定不同的細胞自動機運算規則，Stephen Wolfram 對一維空間的細胞自動機進行研究後整理並且提出了 256 種規則[18]，表一列出數個細胞自動機規則。



圖二：細胞自動機



圖三：細胞自動機規則30之輸入與輸出

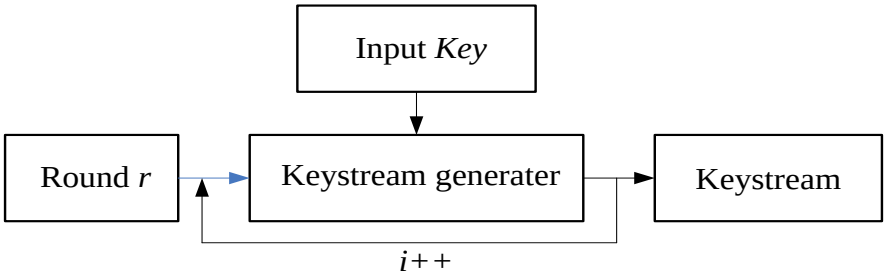
表一為不同細胞自動機規則之邏輯運算，根據不同的規則與輸入資料以進行疊代產生不同的輸出序列，將這些序列當作串流加密器所需之金鑰流。細胞自動機產生的金鑰流擁有簡單化、規則化、區塊化特性[4][9]，且易於模擬在各種不同的平台上[13]，所以以細胞自動機是產生金鑰流的一種相當有效率方法，滿足在現今通訊上金鑰流產生器的特性。

表一：細胞自動機規則與其邏輯運算

細胞自動機規則	邏輯運算
Rule 30	$a_t(i-1)\oplus(a_t(i)\bigcup a_t(i+1))$
Rule 45	$a_t(i-1)\oplus(a_t(i)\bigcup\overline{a_t(i+1)})$
Rule 60	$a_t(i-1)\oplus a_t(i)$
Rule 86	$(a_t(i-1)\bigcup a_t(i))\oplus a_t(i+1)$
Rule 90	$a_t(i-1)\oplus a_t(i+1)$
Rule 150	$a_t(i-1)\oplus a_t(i)\oplus a_t(i+1)$
Rule 165	$a_t(i-1)\oplus\overline{a_t(i+1)}$

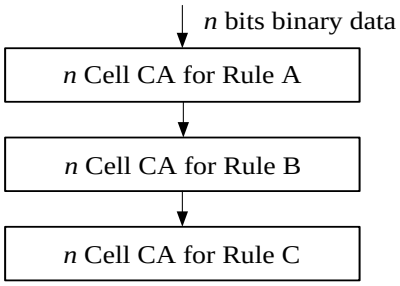
參、滑動視窗之細胞自動機金鑰流產生器

本節中，我們將提出一個基於滑動視窗之細胞自動機金鑰流產生器。細胞自動機的初始狀態在跟相鄰近的細胞進行疊代的過程中，每次的狀態展現出近乎是具多樣性與不可預測性，相對於簡單的規則，展現出的是複雜的交互現象，這些狀態的數值非常適合用來當作金鑰流產生器， 圖四為基本金鑰流產生器方塊圖，其中輸入金鑰為 128-bit，r 為回合數，Keystream 為輸出。

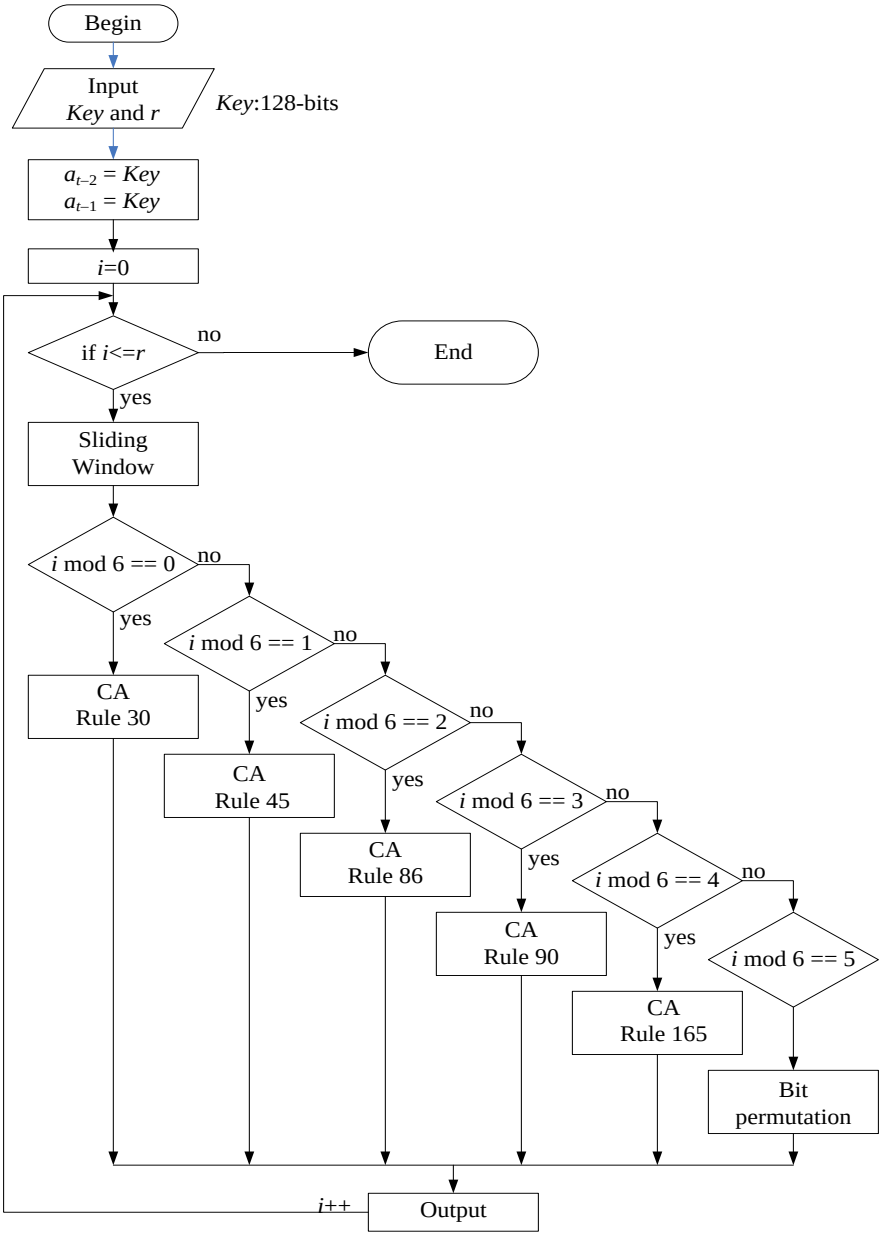


圖四：金鑰流產生流程

以單一細胞自動機的規則疊代的序列要做為金鑰流安全性低，為了要增加輸出的金鑰流的亂度與提高隨機性，我們使用不同的規則進行疊代，圖五所示為複合式地使用多個細胞自動機規則[3]，這樣的方法可以增加金鑰流的亂度。



圖五：結合多個不同規則之一維細胞自動機



圖六：基於滑動視窗之細胞自動機金鑰流產生器流程圖

3.1 滑動視窗 (Sliding Window)

細胞自動機金鑰流產生器是以前次輸出資料為本次細胞自動機疊代運算的輸入，透過滑動視窗的概念乃使擷取前次輸出資料的範圍為可變，等於本次疊代的輸入資料有可變的範圍，可使本次疊代的輸入資料增加可變性，致使輸出更具不可預測性。進行一維細胞自動機之前，要進行滑動視窗 (Sliding Window) 以擷取資料，當正常執行一維細胞自動機以疊代至下個狀態時，序列最左邊的數與最右邊的數會沒有鄰域，而使得的運算過後序列縮減[12]，為了使數列不會縮減，並且加入過去的資料會使得金鑰流的安全性更高，圖七代表滑動視窗的運算流程。資料 a_{t-2} 與 a_{t-1} 代表前次疊代的輸出資料，如方程式 (3)、 (4) 表示：

$$a_{t-2} = \{a_{t-2}(0), a_{t-2}(1), a_{t-2}(2), \cdots a_{t-2}(127)\} \tag{3}$$

$$a_{t-1} = \{a_{t-1}(0), a_{t-1}(1), a_{t-1}(2), \cdots a_{t-1}(127)\} \tag{4}$$

a_{t-2} 與 a_{t-1} 串接，如方程式 (5) 所示:

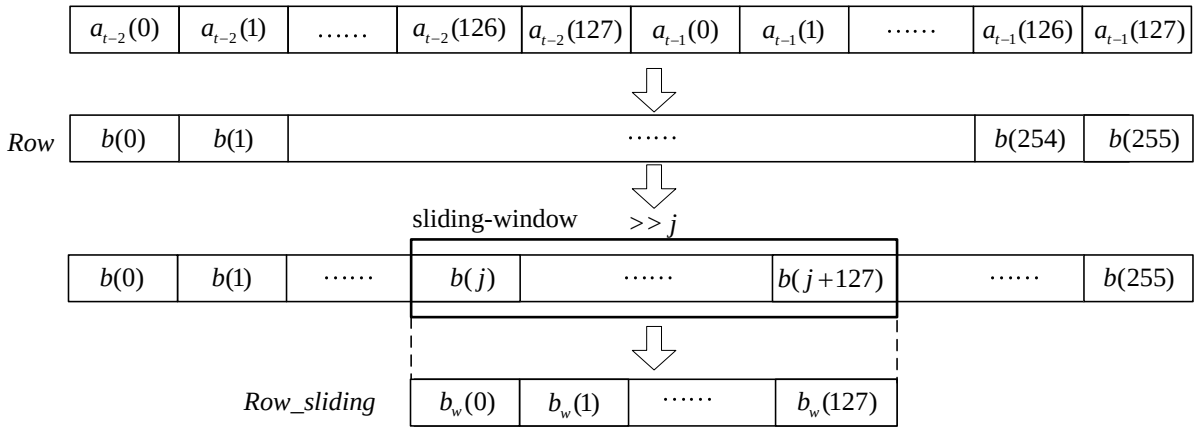
$$Row = a_{t-2} \,||\, a_{t-1} = \{b(0), b(1), b(2), \cdots b(255)\} \tag{5}$$

根據序列$\{b(0),b(1)\ldots b(7)\}$決定視窗位移位置 j，如方程式 (6)：

$$j = b(0)\times 2^0 + b(1)\times 2^1 + b(2)\times 2^2 + b(3)\times 2^3 + b(4)\times 2^4 + b(5)\times 2^5 + b(6)\times 2^6 \tag{6}$$

根據起始位置 j 來圈選每次要進行細胞自動機規則運算的序列，j 代表視窗起始位置，如方程式 (7) 所示，共 128-bit，擷取資料的視窗位置會依 j 而變，就像一個可以滑動的視窗。

$$Row\_sliding = \{b(j), b(j+1), \cdots b(j+127)\} \tag{7}$$



圖七：滑動視窗 *Sliding*(*Row*)

以輸出序列為 128-bit 再進行下一回合細胞自動機運算時，序列最兩側的數值會因為沒有鄰域可以進行運算，數列將會逐漸縮減，為了得到 128-bit 的輸出，滑動視窗最左、最

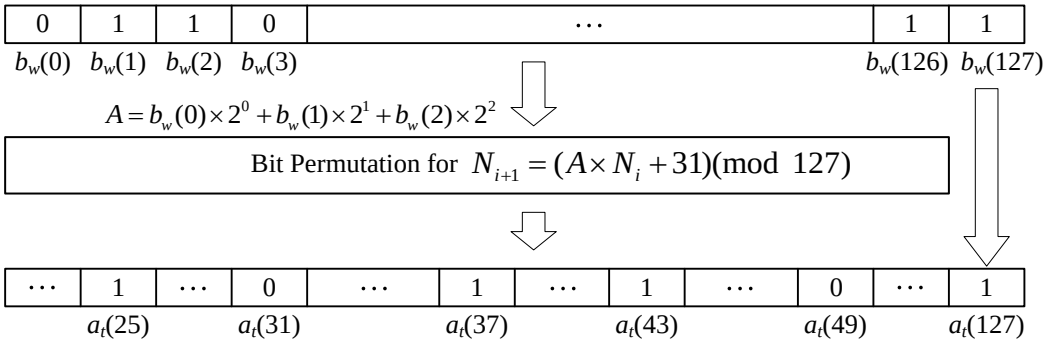
右各加 1-bit，即最左邊加 $b_w(127)$，最右邊加 $b_w(0)$，以構成要 130-bit 的序列 $\{b_w(127),b_w(0),b_w(1)\ldots b_w(127),b_w(0)\}$，並以之進行一維細胞自動機運算，輸出將會得到 128-bit 的輸出序列。

3.2 位元置換 (Bit Permutation)

位元置換基本是由線性同餘法(LCG)來進行數列的交換位置，我們利用線性同於法來進行位元置換，線性同餘法如以方程式 (8) 所示：

$$N_{i+1} = (A \times N_i + B)(\mathrm{mod}\, M) \tag{8}$$

其中 A、B、M 為參數，N_i 代表原始位置，N_{i+1} 代表經過置換的位置，我們根據序列的前 3-bit 決定 A 值，B 為 31，M 為 127，B 與 M 互質，整個數列會為最大週期，圖八為數列位移的一個例子，若 $b_w(0), b_w(1), b_w(2)$等於 011，即 $A = 6$，則 $b_w(0)$位元置換至 $a_t(31)$ 的位置，其餘類推，最後 1 個位元 $a_t(127) = b_w(127)$。位元置換階段目的是打散整個 128-bit 序列，使得細胞自動機產生的金鑰流亂度更好。



圖八：位元置換 Bit Permutation()

3.3 金鑰流產生

我們提出之金鑰流產生器結合一維細胞自動機、滑動視窗、位元置換，來產生串流加密法所需要的金鑰流，圖九為基於滑動視窗之細胞自動機金鑰流產生器的演算法。首先，輸入資料為 128-bit 的金鑰 *Key*，與回合次數 r，每次回合經過滑動視窗，再輪流選擇要進行的運算，一維細胞自動機與位元置換都會輸出 128-bit 長度的輸出金鑰流，接著把輸出的資料取代序列 a_{t-2} 與 a_{t-1}，更新下回合的輸入序列，輸入值 r 決定整體的回合數，其影響最後輸出的金鑰流長度，也就代表會輸出 $r \times 128$ bits 金鑰流。

Input: *Key* = 128 bits (key) and *r* (number of rounds)
Output:k_o = 128 × *r* bits key stream
Begin
$a_{t-2} = a_{t-1} = Key$
for($i = 0$; $i <= r$; i++)
//Sliding step
$Row = a_{t-2} \,||\, a_{t-1}$;
$Row\_sliding = Sliding(Row)$;
$s = i$ mod 6 ; //Select stage function
if $s == 0$, $f_s() = f_0()$; $f_0()$: CA rule 30
if $s == 1$, $f_s() = f_1()$; $f_1()$: CA rule 45
if $s == 2$, $f_s() = f_2()$; $f_2()$: CA rule 86
if $s == 3$, $f_s() = f_3()$; $f_3()$: CA rule 90
if $s == 4$, $f_s() = f_4()$; $f_4()$: CA rule 165
if $s == 5$, $f_s()$ = Bit permutation() ;
for($jj = 0$; $jj <= 127$; jj ++)
$a_t(jj) = f_s(Row\_sliding)$;
k_o += a_t ;
return k_o ; //output keystream
$a_{t-2} = a_{t-1}$; //replace a_{t-2}
$a_{t-1} = a_t$; //replace a_{t-1}
End

圖九：基於滑動視窗之細胞自動機金鑰流產生器演算法

肆、安全性分析與實驗結果

本節將針對我們所提出的金鑰流產生器進行安全性的分析與金鑰流亂度測試的統計結果，我們所提的金鑰流產生器每次疊代產生 128-bit 資料輸出，捨棄一開始的前 64 回合的輸出，捨棄資料是為了避免初始值疊代之後進行反向運算的攻擊。

4.1 安全性分析

我們所要進行的安全性分析方面包含：暴力攻擊、選擇密文攻擊，金鑰流的平衡性分析、差分密碼分析，分述如後。

● 暴力攻擊

暴力攻擊 (Brute-force attack) 又稱蠻力攻擊，是相當常見的一種攻擊方式[17]，幾乎所有的加密系統都有可能被暴力攻擊而破解，主要是透過逐一搜查可能的金鑰直到找

到真正的金鑰為止，而抵抗這個攻擊的方法則是金鑰的空間要足夠大，一般而言只要金鑰長度超過 128 位元就能夠提供足夠的安全性。

對於我們所提出基於滑動視窗之細胞自動機金鑰流產生器的輸入金鑰長度為 128 位元，而金鑰的組合空間約有 2^{128} 種可能性，因此我們所提出的金鑰流產生器的安全強度能夠抵抗暴力攻擊的威脅。

●選擇密文攻擊

選擇密文攻擊 (Chosen ciphertext attack) 是一種密碼分析攻擊[22]，首先假設攻擊者能取得各種資料，可以選擇一些密文並從系統中獲得相對應的明文，也就是說攻擊者可能透過蒐集到的密文推測出明文的結果。

假如有兩個相同的密文，串流加密法與區塊加密法不同的是串流加密法加密後的相同密文對應的可能是不同的明文，以串流加密法 $m_i \oplus k_i = c_i$ 為例，相同的 c_i 對應的可能是不同的明文 m_i，串流密碼最重要的就是金鑰流 k_i 的強度，我們所提出的基於滑動視窗之細胞自動機金鑰流產生器透過滑動視窗與多層細胞自動機的非線性轉換，能夠提高金鑰流的強度，抵抗選擇密文的攻擊。

●平衡性分析

平衡性分析是一種檢測金鑰流是否遵循均勻分布的統計測試[8]，如下方程式 (9) 所示：

$$e = \frac{|q_1 - q_0|}{N} \tag{9}$$

其中 e 為不平衡度 (Disequilibrium degree)，q_1 是序列中 1 的數量，q_0 是 0 的數量，N 是序列的總數，序列的 e 值越低，則代表序列的平衡性越好。

由表二為我們所提出的金鑰流產生器所產生的金鑰流的不平衡值 Disequilibrium degree e 統計資料，由表中可以發現到 0 和 1 的數量幾乎是平衡的，證明我們所提之基於滑動視窗之細胞自動機金鑰流產生器具有良好的平衡性。

表二：金鑰流測得之不平衡度e

Length of sequence	10000	50000	100000
Number of 0's	5011	25087	50032
Number of 1's	4989	24931	49968
Disequilibrium degree (%)	0.22	0.31	0.064

●差分密碼分析 (Differential Cryptanalysis)

差分密碼分析是一種用於研究輸入訊息如何影響輸結果的密碼分析方式，主要適用於分組密碼，也適用於串流密碼與單向雜湊函數，常見的攻擊方式是透過轉換網路跟蹤差異，發現密碼系統可能的非隨機行為的位置，並且利用這些非隨機行為來找出密碼系統的原始金鑰[19]。

對於串流密碼系統最重要的是原始金鑰對於金鑰流的敏感度與擴散效應，而串流密碼系統可以藉由計算金鑰流差異率(*kdr*)來觀察金鑰流產生器敏感度，金鑰流差異率 *kdr* 如方程式 (10) 所示：

$$kdr(k) = \frac{\mathrm{Diff}(k, k_1) + \mathrm{Diff}(k, k_2)}{2 \times N} \times 100\% \tag{10}$$

其中 N 是輸出金鑰流的二進制長度，而 $\mathrm{Diff}(k, k_1)$ 與 $\mathrm{Diff}(k, k_2)$ 則來自金鑰 k 與 k_1、k_2 的差異，也就是輸入的原始金鑰輸出金鑰流之間的不同位數，接下來的測試中我們的輸入參數為 $N = 10^6$，並且將原始金鑰 k_1、k_2 只有 1 位元的差異如下所示：

$N = 10^6$
k = 0x00000000000000000000000000000000
k_1 = 0x00000000000000000000000000000001
k_2 = 0x00000000000000000000000000000010

表三為我們提出之基於滑動視窗細胞自動機金鑰流產生器的金鑰流差異率 *kdr*，由 *kdr* 顯示產生的金鑰流有良好的差分密碼分析抵抗力。

表三：金鑰流的差異率(*kdr*)

The Proposed Key Stream Generator	*kdr* (%)
A Key Stream Generator Based on Sliding Window Cellular Automata	50.05 %

4.2 金鑰流亂度測試的統計結果

本小節中我們透過 FIPS SUB 140-1 與 SP800-22 分別對輸出金鑰流進行亂度測試[20][21]。

●FIPS SUB 140-1 金鑰流測試結果

我們選擇 100 個不同的金鑰以產生 100 組 20,000 位元的二進制輸出金鑰流進行測試並且捨棄金鑰流產生器前 64 次疊代的輸出，也就是捨棄前 64×128 位元資料輸出，以保原輸入密碼。

表四為 FIPS SUB 140-1 的測試結果，我們可以發現 FIPS SUB 140-1 的通過率為 100%，這表示它們具有良好的亂度特性。

表四：FIPS SUB 140-1金鑰流測試之通過率

Monobit Test	Poker Test	Runs Test	Long Run Test
100%	100%	100%	100%

●SP800-22 金鑰流測試

SP800-22 是美國國家標準研究院 (NIST) 所制定的金鑰流測試標準，是目前國際上最具代表性的金鑰流測試方法之一，對於金鑰流產生器所產生的二進制資料進行統計測試，SP800-22 共有 15 項測試，我們使用隨機的 100 個初始值來產生 100 個長度為 1,000,000-bit 的二進制資料，表五為 15 項測試的實驗結果，實驗結果可以發現到，每項測試的結果至少是 91%以上。

表五：SP800-22 金鑰流測試的實驗結果

Statistical Tests	*p*-Value	Pass rate Under 10^6 bits/sample
Frequency	0.7003567	98 %
Block Frequency	0.8070172	100 %
Runs	0.1194802	98 %
Longest Runs of Ones	0.5466953	100 %
Rank	0.6644652	99 %
Discrete Fourier Transform	0.9144634	99 %
Non-overlapping Templates Matching	0.9999994	100 %
Overlapping Templates Matching	0.0510680	97 %
Universal Statistical	0.0784289	94 %
Linear Complexity	0.7791804	100 %
Serial	0.0482862	95 %
Approximate Entropy	0.0490072	97 %
Cumulative sums	0.4550819	98 %
Random Excursions	0.0347417	93 %
Random Excursions variant	0.0259393	91 %

伍、結論

本文中我們結合細胞自動機、滑動視窗與位元置換，提出基於滑動視窗之細胞自動機金鑰流產生器，以之應用於物聯網安全通訊之串流加密器，在細胞自動機相對簡單的規則交互作用下，以產生快速、安全性高的金鑰流，可以運用在資源相對較少的物聯網

通訊上。我們提出之金鑰流產生器可以抵抗暴力攻擊、選擇密文攻擊等密碼分析攻擊，經 FIPS SUB 140-1 的測試有 100%的通過率，在 SP800-22 的金鑰流測試則有至少 91%以上的通過率。

參考文獻

[1] G. Bansod, N. Raval and N. Pisharoty, "Implementation of a New Lightweight Encryption Design for Embedded Security", *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 142-151, Jan. 2015.

[2] Rong-Jian Chen and Jui-Lin Lai, "Data encryption using non-uniform 2-D Von Neumann cellular automata", *2005 9th International Workshop on Cellular Neural Networks and Their Applications*, pp. 77-80, 2005.

[3] R. Dogaru and I. Dogaru, "Efficient and cryptographically secure pseudorandom number generators based on chains of hybrid cellular automata maps", *2014 10th International Conference on Communications (COMM)*, pp. 1-4, 2014.

[4] E. Göncü, A. Koçdoğan and M. E. Yalçin, "A High Speed True Random Number Generator with Cellular Automata with Random Memory," *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2018.

[5] R. Kumar, P. Khan and S. Kumar, "A Cellular Automata-based Healthcare Data Encryption Technique for IoT Networks", *2019 IEEE 16th India Council International Conference (INDICON)*, pp. 1-4, 2019.

[6] A. Kumaravel and O. N. Meetei, "An application of non-uniform cellular automata for efficient cryptography", *2013 IEEE Conference on Information & Communication Technologies*, pp. 1200-1205, 2013.

[7] D. Ma and Y. Shi, "A Lightweight Encryption Algorithm for Edge Networks in Software-Defined Industrial Internet of Things", *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, pp. 1489-1493, 2019.

[8] Jinqiu, Lin, and Si Xicai, “A New Stream Cipher Based on Coupled Map,” Computer Science and Information Engineering, 2009 WRI World Congress on IEEE, Vol. 1, 2009.

[9] S. Nandi, B. K. Kar and P. Pal Chaudhuri, "Theory and applications of cellular automata in cryptography", I*EEE Transactions on Computers,* vol. 43, no. 12, pp. 1346-1357, Dec. 1994.

[10] Nandi, Subrata and Roy, Satyabrata, "Application of cellular automata in symmetric key cryptography", *International Conference on Communication and Signal Processing*, ICCSP 2014 – Proceedings, 2014.

[11] Neumann, John von, "Theory of self-reproducing automata" Edited by Arthur W. Burks, 1966.

[12] S. Roy, N. Bhatia and U. S. Rawat, "A novel cryptosystem using cellular automata", *2017 International Conference on Communication and Signal Processing (ICCSP)*, pp. 1781-1785, 2017.

[13] S. Roy, U. Rawat and J. Karjee, "A Lightweight Cellular Automata Based Encryption Technique for IoT Applications", *IEEE Access*, vol. 7, pp. 39782-39793, 2019.

[14] Satyabrata Roy, Jyotirmoy Karjee, U.S. Rawat, Dayama Pratik N., Nilanjan Dey, "Symmetric Key Encryption Technique: A Cellular Automata based Approach in Wireless Sensor Networks", *Procedia Computer Science*,Volume 78, pages 408-414, 2016.

[15] K. Rajeshwaran and K. Anil Kumar, "Cellular Automata Based Hashing Algorithm (CABHA) for Strong Cryptographic Hash Function", *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pp. 1-6, 2019.

[16] Tripathy, Somanath, and Sukumar Nandi, "LCASE: Lightweight Cellular Automata-based Symmetric-key Encryption", *Int. J. Netw. Secur,* 8.3, 2009.

[17] William Stallings, Cryptography and Network Security: Principles and Practice, 3rd Edition, Prentice-Hall Inc., Upper Saddle River, N.J., 2003.

[18] Wolfram Stephen, "Cryptography with cellular automata",*Conference on the Theory and Application of Cryptographic Techniques*. Springer, Berlin, Heidelberg, 1985.

[19] Shiguo Lian, Jinsheng Sun, Zhiquan Wang, “A block cipher based on a suitable use of the chaotic standard map,” Chaos Solitons & Fractals, Berlin, 2005.

[20] A. Rukhin, J.Soto, et al. “Special Publiccation 800-22 Revision 1: A Statistical Test Suite for Random and Pseudom Number Generators for Cryptographic Applications,” National Institute of Standards and Technology, Aug. 2008.

[21] FIPS, P.140-1.Security Requirements for Cryptographic, 140-1, 1994.

[22] 結城浩、吳嘉芳譯，圖解密碼學與比特幣原理，臺北市，碁峰資訊，2016。

[作者簡介] Biography

吳錫聰，國立宜蘭大學電子工程學系副教授
吳國華，國立宜蘭大學電子工程學系碩士