Geração de Sequências Aleatórias usando Mapas Caóticos Unidimensionais

José A. P. Artiles, Daniel P. B. Chaves, Cecilio Pimentel

***Resumo*— Geradores de números aleatórios são amplamente utilizados em comunicações e criptografia. Este trabalho propõe uma metodologia para o projeto destes geradores a partir de mapas caóticos unidimensionais. Esta é baseada em duas técnicas: discretização codificada variante no tempo e pós-processamento empregando sequências-m para eliminar a correlação residual na sequência codificada. A técnica de discretização codificada variante no tempo produz sequências binárias com função autocorrelação com um padrão bem definido que é explorado pelo bloco de pós-processamento para reduzir seu requerimento de memória em relação à codificação padrão fixa no tempo. Validam-se os geradores propostos empregando a bateria de teste NIST.**

***Palavras-Chave*— Mapas caóticos, sistemas dinâmicos, geração de números aleatórios, função autocorrelação, teste NIST, sequências-m.**

***Abstract*— Random number generators are widely used in communications and cryptography. This work presents a methodology for the design of these generators from unidimensional chaotic maps. This is based on two techniques: Time-varying coded discretization and post-processing employing m-sequences to eliminate the residual correlation of the coded sequence. The time-varying coded discretization technique produces binary sequences with an autocorrelation function with a well-defined pattern that is exploited by the block of post-processing to reduce its memory requirement in relation to the required by the standard time-invariant discretization. The effectiveness of this procedure is verified through the NIST test.**

***Keywords*— Chaotic maps, dynamical systems, random numbers generator, autocorrelation function, NIST test, m-sequences.**

I. Introdução

Os geradores de números pseudo-aleatórios são largamente usados em comunicações [1]–[3] e criptografia [4], [5]. Na área de criptografia são utilizado em sistemas criptográficos de chave privada, geradores de números pseudo-aleatórios (PRNG, *pseudo-random number generators*), e em sistemas criptográficos de chave pública [4], [5].

A geração de PRNG usando mapas caóticos tem sido considerada na literatura [6]–[11]. Usualmente, o projeto destes geradores é composto de três blocos: geração de uma sequência caótica a partir da iteração um mapa caótico, codificação da amostra caótica com um símbolo binário a partir da quantização do espaço de fase em duas regiões, gerando-se assim uma sequência binária e finalmente emprega-se uma unidade de pós-processamento para quebrar a correlação existente nesta sequência binária. Se esta unidade tiver taxa unitária, a taxa de taxa de geração de bits por amostra caótica deste gerador é igual a 1. Um dos objetivos destes trabalhos é a proposição de circuitos para implementação do mapa caótico a fim de que o gerador possa ser implementado em hardware.

Em [12], foi apresentada uma técnica alternativa para o projeto de PRNG empregando mapas caóticos. O gerador proposto é baseado em três blocos. Gera-se uma sequência discreta a partir de um mapa caótico unidimensional. Em seguida, utiliza-se a técnica de saltos de amostras [13] para a quebra da correlação da sequência caótica. Por fim, especifica-se um particionamento do espaço de fase com a proposição de um esquema de codificação que garanta uma distribuição de probabilidade uniforme da sequência gerada, denominado de discretização codificada variante no tempo (CVT). Usando parâmetros apropriados, a sequência binária na saída do codificador variante no tempo passa nos testes de aleatoriadade NIST [14] indicando que a metodologia proposta produz sequências binárias com boas propriedades criptográficas. Uma característica desta proposta é que a taxa de geração de bits por amostra caótica é no máximo igual a 1. O objetivo deste trabalho é propor um gerador alternativo que propicie o aumento desta taxa para um inteiro arbitrário q. A ideia central é substituir a unidade de saltos de amostras por uma unidade de pós-processamento de taxa unitária que quebre a correlação residual da sequência binária na saída do codificador.

Neste trabalho, propomos uma metodologia para projeto de PRNG baseado em três blocos: geração da sequência caótica, codificação CVT com 2^q níveis de quantização e um bloco de pós-processamento baseado na soma módulo-2 da sequência binária na saída do codificador e uma sequência-m gerada por um registrador de deslocamento com realimentação linear (LFSR, *linear feedback shift register*). Seja N_{min} o grau mínimo do polinômio gerador do LFSR para que a sequência binária gerada seja aprovada na bateria de testes NIST. Inicialmente, observamos que a função autocorrelação da sequência na saída do codificador CVT apresenta um padrão com picos em posições específicas que resulta em uma redução do valor de N_{min} do LFSR em relação ao requerido por uma codificação fixa no tempo. Baseado neste padrão da função autocorrelação, propomos uma nova unidade de pós-processamento com a adição de um bloco de controle e chaveamento do LFSR que propicia uma redução de aproximadamente 30 % no valor de N_{min}. Apesar do métodos considerados independer do mapa caótico, empregamos para estudo de caso três classes de mapas caóticos: mapa tangente hiperbólica, mapa cúbico e mapa de Hénon.

O artigo está organizado em seis seções. Na Seção II descrevemos os mapas caóticos utilizados neste trabalho. Na Seção III apresentamos o PRNG proposto e introduzimos a

J. A. P. Artiles, D. P. B. Chaves, C. Pimentel, Departamento de Eletrônica e Sistemas, Universidade Federal de Pernambuco, Recife-PE, Brasil, E-mails: joseantonio.artiles@ufpe.br, daniel.chaves@ufpe.br, cecilio@ufpe.br.
Este trabalho foi parcialmente financiado pelo CNPq e pela FACEPE.

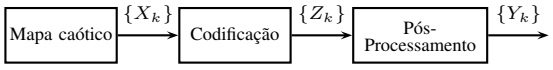

Fig. 1. Diagrama em blocos do esquema de geração de um PRNG.

técnica de codificação variante no tempo. Os resultados do teste NIST e tabelas com valores de N_{min} são mostradas na Seção IV. A codificação Gray variante no tempo é introduzida na Seção VI.

II. Mapas Caóticos

O comportamento dos mapas caóticos unidimensionais é observado mediante uma série temporal discreta $\{x_i\}_{i=0}^{\infty}$, obtida pela iteração de uma função não-linear $f(x)$, sob uma condição inicial x_0, da seguinte forma:

$$x_n = f(x_{n-1}), n = 1, 2, 3, \ldots \tag{1}$$

Denomina-se $\{x_n\}_{i=0}^{\infty} = \{x_0, f(x_1), f(x_2), \ldots\}$ uma órbita de f iniciando em x_0. Os mapas $f : [-1, 1] \to [-1, 1]$ usados neste trabalhos são descritos a seguir.

- O mapa tangente hiperbólica [15] é baseado na função tangente hiperbólica e é definido por:

$$f(x) = \begin{cases} e \cdot \tanh\big(r \cdot (x+1)\big) - 1, & x < 0 \\ (-1)^b \cdot \big[e \cdot \tanh\big(-r \cdot (x-1)\big) - 1\big], & x \geq 0 \end{cases} \tag{2}$$

  em que o fator de escala e é dado por

$$e = \frac{2}{\tanh(r)}. \tag{3}$$

  Este mapa tem dois parâmetros de controle especificados pela dupla (b, r). O parâmetro b define a simetria do mapa, que pode apresentar simetria par ($b = 0$), quando é denominado e-tanh, ou simetria ímpar ($b = 1$), quando é denominado o-tanh. O parâmetro r é um número real positivo que controla a extensão da região planar em torno do eixo de simetria. Neste trabalho fixamos $r = 3$, o mesmo utilizado em [12], dado que os resultados são semelhantes para outros valores de r.
- O mapa cúbico (MC) é definido como [1]:

$$f(x) = 4x^3 - 3x. \tag{4}$$

- O Mapa de Hénon (MH) é dado por [1]:

$$x_{k+1} = 1 + 0,3x_{k-1} - 1,4x_k^2. \tag{5}$$

A próxima seção descreve o procedimento de geração de um PRNG proposto neste trabalho.

III. Geração de um PRNG usando mapas caóticos

A geração de um PRNG proposta neste trabalho envolve de três etapas, conforme ilustra a Fig. 1. Inicialmente, uma órbita finita $\{X_k\}$ é gerada a partir de um mapa caótico. Em seguida, a técnica de discretização codificada gera uma sequência binária $\{Z_k\}$. A correlação residual entre os símbolos da sequência binária requer a utilização de uma unidade de pós-processamento para geração de uma sequência pseudo-aleatória $\{Y_k\}$.

A primeira etapa consiste na obtenção de uma órbita para um mapa caótico específico a partir de uma condição inicial usando (1). Os primeiros 400 pontos gerados são descartados devido ao transiente inicial da órbita. Para a discretização da sequência caótica, particiona-se o espaço de fase em 2^q regiões, $\mathcal{R}_i$, para $i = 1, 2, \cdots, 2^q$ e satisfazendo $\Pr(X_k \in \mathcal{R}_i) = 1/2^q$, $\forall\, i$. Cada região $\mathcal{R}_i$ é codificada com uma sequência binária de q *bits* (denominada de sequência código). Usualmente, esta rotulação é fixa no tempo e uma das contribuições deste trabalho é propor novas codificações variante no tempo. Por exemplo, para a codificação fixa no tempo (CFT) com $q = 2$, as quatro regiões são mapeadas pelas sequências $(00, 01, 10, 11)$. Se $X_k \in \mathcal{R}_1$ então a sequência binária gerada neste intervalo é 00, se $X_k \in \mathcal{R}_2$ a sequência gerada é 01, e assim sucessivamente. Esta codificação de regiões não varia com k. Define-se a taxa de codificação por $R = q$ bits por amostra caótica. Cada sequência código na saída do codificador é mapeada pela unidade de pós-processamento em uma sequência de mesmo comprimento, portanto, esta unidade tem taxa unitária. Assim, a taxa total do sistema é R.

A seguir estudaremos o comportamento da função autocorrelação da sequência codificada binária $\{Z_k\}$ usando a codificação CFT. A função autocorrelação de um processo estacionário $\{Z_k\}$ é dada por $R_Z[m] = E[Z_k\, Z_{k+m}]$, em que $E[\cdot]$ denota valor esperado. Para uma sequência aleatória, $R_Z[m] = 1/4$ para todos os valores de $m \neq 0$.

A Fig. 2 mostra $R_Z[m]$ versus m para uma sequência binária $\{Z_k\}$ gerada por simulação para o mapa o-tanh utilizando a codificação CFT e diferentes valores de q ($q = 1, 3, 5, 7$). O aumento de q conduz a um aumento da taxa R, mas introduz um espalhamento da função autocorrelação, devido à correlação existente nas sequências código que rotulam cada amostra caótica. Esta figura destaca o valor mínimo de m para o qual $R_Z[m]$ se estabiliza em 1/4. Este comportamento também é observado para todos os mapas considerados neste trabalho.

Na próxima seção, mostraremos que a codificação CVT muda significantemente o comportamento de $R_Z[m]$, com implicação direta no requerimento de memória da unidade de pós-processamento, conforme será analisado na Seção IV.

A. Codificação variante no tempo

Em [12], foi introduzida uma codificação CVT que consiste em um deslocamento cíclico para esquerda de sequências código que rotulam regiões adjacentes. Por exemplo, se para a k-ésima amostra caótica a codificação para $q = 2$ é $(00, 01, 10, 11)$, então para a próxima amostra a codificação passa a ser $(01, 10, 11, 00)$. As Figs. 3, 4, 5 e 6 ilustram a função autocorrelação para os mapas utilizados neste trabalho para $q = 3$ e $q = 5$ com codificação CVT. Diferentemente da codificação CFT, percebe-se nestas figuras um padrão de concentração para cada valor de q. Existem picos desta função em valores específicos de m, dados por múltiplos de q. Observa-se que o número de picos, denotado por P, é praticamente constante com q.

Este comportamento indica que existe correlação entre *bits* de sequências código distintas separadas de q posições. Por

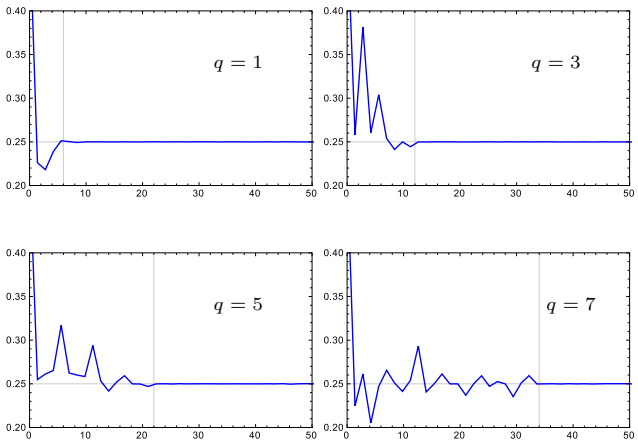

Fig. 2. $R_Z[m]$ versus m para o mapa o-tanh com $q = 1, 3, 5, 7$ e codificação CFT.

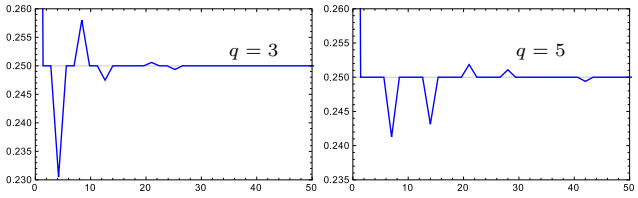

Fig. 3. $R_Z[m]$ versus m para o mapa o-tanh com $q = 3, 5$ e codificação CVT.

exemplo, três amostras caóticas consecutivas $X_1X_2X_3$ geram três sequências código $(Z_1Z_2Z_3)$, $(Z_4Z_5Z_6)$ e $(Z_7Z_8Z_9)$. Portanto existe nestas sequências código correlação apenas entre os *bits* de cada um dos três conjuntos $\{Z_1, Z_4, Z_7\}$, $\{Z_2, Z_5, Z_8\}$, e $\{Z_3, Z_6, Z_9\}$. O comportamento de $R_z[m]$ indica que não existe correlação entre os *bits* de uma mesma sequência código, ou em *bits* de sequências código distintas que estão em posições distintas. Este comportamento implica em uma concentração da correlação em conjuntos reduzidos de bits quando comparado ao caso fixo no tempo.

IV. Pós-processamento baseado em LFSR

A sequência codificada $\{Z_k\}$ com codificação CVT apresenta uma memória finita sendo necessário então o emprego de um bloco de pós-processamento para a quebra desta memória.

Em decorrência da estatística local quase ideal [16] apresentada pelas sequências-m, empregamos neste trabalho um processamento que realiza a soma módulo-2 da sequência codificada com uma sequência-m gerada por um LFSR, conforme é ilustrado na Fig. 7. Seja $\{Z_k\}$ a sequência na saída de um codificador e $\{W_k\}$ a sequência-m gerada pelo LFSR, então a k-ésima amostra de saída é dada por $Y_k = Z_k \oplus W_k$, em que $\oplus$ indica adição módulo-2. Uma sequência $\{W_k\}$ é gerada por um LFSR de comprimento N se satisfaz a seguinte relação:

$$W_k = c_1W_{k-1} \oplus c_2W_{k-2} \oplus ... \oplus c_{N-1}W_{k-(N-1)} \oplus W_{k-N} \quad (6)$$

em que $c_i \in \{0, 1\}$. A malha de realimentação do LFSR pode ser representada por um polinômio de conexão

$$p(x) = 1 + c_1x + c_2x^2 + ... + c_{N-1}x^{N-1} + x^N. \quad (7)$$

O polinômio de conexão define o período e o comportamento estatístico da sequência $\{W_k\}$. Para garantir o período máximo, $2^N - 1$, o polinômio $p(x)$ deve ser primitivo [17]. A sequência gerada por um polinômio primitivo é chamada de sequência-m. Estas sequências apresentam N *bits* com característica local quase ideal [16], que serão utilizados eliminar a correlação residual da sequência $\{Z_k\}$.

Neste trabalho empregaremos a bateria de teste NIST (versão 800-22) [14] para testar se a sequência $\{Y_k\}$ é adequada para aplicações criptográficas. Os testes são utilizados para determinar a aceitação ou rejeição da hipótese de aleatoriedade ideal. Neste trabalho adotamos, $\alpha = 0,01$, ou seja, uma sequência é aprovada com nível de significância maior do 99%. Nas simulações realizadas, consideramos que a sequência binária de entrada no conjunto de teste NIST tem comprimento 524288000 (formada pela concatenação de

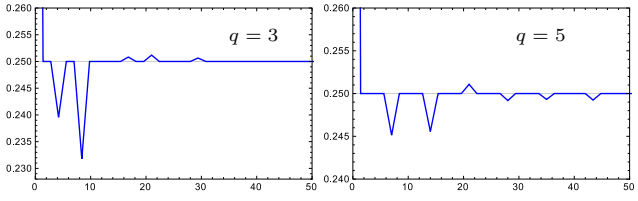

Fig. 4. $R_Z[m]$ versus m para o mapa MC com $q = 3, 5$ e codificação CVT.

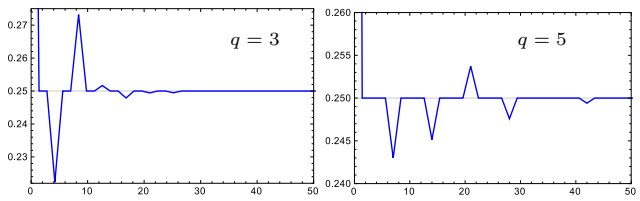

Fig. 5. $R_Z[m]$ versus m para o mapa e-tanh com $q = 3, 5$ e codificação CVT.

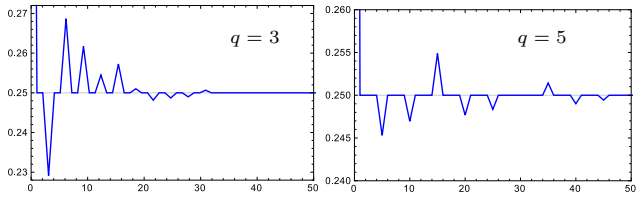

Fig. 6. $R_Z[m]$ versus m para o mapa MH com $q = 3, 5$ e codificação CVT.

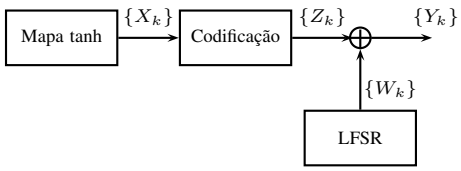

Fig. 7. Diagrama de blocos do esquema de geração de um PRNG usando sequências-m.

TABELA I

N_{min} PARA DIFERENTES VALORES DE q COM CODIFICAÇÃO CFT

$R = q$	e-tanh	o-tanh	Cúbico	Hénon
1	7	5	8	9
2	13	10	12	18
3	18	12	14	27
4	26	17	18	33
5	34	22	23	36
6	40	27	28	44
7	47	33	35	50

TABELA II

N_{min} PARA DIFERENTES VALORES DE q COM CODIFICAÇÃO CVT

$Rx = q$	e-tanh	o-tanh	Cúbico	Hénon
1	7	5	8	8
2	9	7	11	15
3	9	11	12	15
4	10	10	12	15
5	10	9	12	15
6	11	10	12	15
7	10	10	12	15

500 subsequências de comprimento 1048576, geradas com condições iniciais escolhidas aleatoriamente).

Define-se N_{min} como o menor valor do grau N do polinômio de realimentação do LFSR para que a sequência $\{Y_k\}$ passe no teste NIST. A Tabela I mostra os valores de N_{min} para vários valores de q obtidos para os mapas utilizados neste trabalho com codificação CFT. Observa-se uma dependência aproximadamente linear entre os parâmetros q e N_{min}. A Tabela II mostra os valores de N_{min} obtidos com codificação CVT. Observa-se que o valor de N_{min} permanece praticamente constante com o aumento de q. Este comportamento deve-se ao fato que a função autocorrelação apresenta uma quantidade de picos praticamente constante para todos os níveis de quantização.

Particularmente, a sequência $\{Z_k\}$ apresenta conjuntos curtos de bits correlacionados que são espaçados homogeneamente por sequências de bits de comprimento múltiplo de q. Consequentemente, são necessárias sequências mais curtas com estatística local quase-ideal, provenientes da sequência-m, para eliminar a correlação de $\{Z_k\}$. Pode-se reduzir os valores de N_{min} apresentados na Tabela II a partir de um projeto cuidadoso da unidade de pós-processamento que explore a presença de picos da função autocorrelação da codificação CVT, conforme é mostrado na próxima seção.

V. UM NOVO BLOCO DE PÓS-PROCESSAMENTO

Para eliminar a correlação existente na sequência $\{Z_k\}$ gerada pela codificação CVT, utiliza-se a característica apresentada nas Figs. 3 e 4, que é a presença de P picos. A Fig. 8 mostra a implementação de um bloco de pós-processamento que explora esta característica. A ideia central consiste em projetar um LFSR que reduza a correlação entre amostras (Z_k e Z_j) em posições relativas indicadas pelos P picos, isto é, separadas por kq amostras, $k = 1, 2..., P$.

Mostra-se no digrama de blocos da Fig. 8 uma chave I_1; quando esta está aberta, os *bits* do processo de codificação são iguais aos *bits* da saída do sistema. Quando a chave está

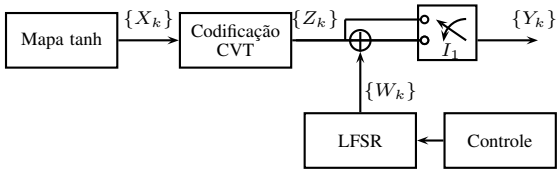

Fig. 8. Diagrama em blocos do esquema de geração de um PRNG usando sequências-m para codificação CVT.

fechada os *bits* da saída $\{Y_k\}$ são obtidos da soma módulo-2 das sequências $\{Z_k\}$ e $\{W_k\}$. Os *bits* da sequência-m são utilizados de tal forma que para os *bits* correlacionados da sequência $\{Z_k\}$ utilizam-se *bits* descorrelacionados da sequência $\{W_k\}$ que apresentam estatística local quase-ideal entre eles, garantindo que a sequência $\{Y_k\}$ seja descorrelacionada.

A operação da chave depende dos valores de q e P como é exemplificado na Fig. 9 para os parâmetros $q = 3$ e $P = 6$. Define-se uma janela de comprimento $J = q(P + 1)$ *bits* para garantir que todos os picos estejam inseridos na janela. Os primeiros q *bits* são os mesmos do processo de quantização do mapa caótico, $Y_1 = Z_1$, $Y_2 = Z_2$ e $Y_3 = Z_3$, os demais *bits* da janela são obtidos da soma módulo-2 entre a sequência $\{Z_k\}$ e a sequência-m, como ilustra a Fig. 9. Na próxima janela faz-se o mesmo procedimento, os primeiros q *bits* da janela são obtidos diretamente do codificador CVT, $Y_{22} = Z_{22}$, $Y_{23} = Z_{23}$ e $Y_{24} = Z_{24}$, os demais *bits* novamente somam-se módulo-2 com a sequência-m. Janelas adjacentes começam por W_1 ou W_2 de forma alternada, o que é realizado pelo bloco de controle. Por exemplo, a primeira janela começa com W_1 até W_{18}, na segunda utiliza-se de W_2 até W_{19} e a terceira volta a ser de W_1 até W_{18}. O grau do polinômio que gera uma sequência $\{Y_k\}$ com propriedades de uma sequência descorrelacionada deve cumprir dois requerimentos:

- O grau do polinômio deve ser maior ou igual ao número de picos existentes da função autocorrelação da sequência $\{Z_k\}$.
- O período da sequência-m deve ser maior que o comprimento da janela.

A primeira condição indica que o número de *bits* com estatística local quase-ideal na sequência-m deve ser maior ou igual à quantidade de bits correlacionados na janela. A segunda condição garante que não existam *bits* correlacionados da sequência-m no intervalo de uma janela. Uma propriedade relevante da sequência-m é que sua dizimação pode formar uma nova sequência-m, conforme é descrito no Teorema 1 [18].

Teorema 1: Seja $\{W_k\}$ uma sequência-m gerada por um polinômio primitivo de grau N. Uma sequência de dizimação $\{W'_k\}$ de parâmetro s obtida a partir de $\{W_k\}$ é da forma $W'_k = W_{sk}$. Esta sequência de dizimação também é uma sequência-m se, e somente se, $\gcd(s, 2^N - 1) = 1$.

Observa-se que os bits de $\{Z_k\}$ correlacionados na primeira janela da Fig. 9 são, por exemplo, $Z_1, Z_4, Z_7, Z_{10}, \cdots$. Estes são somados com uma dizimação de $\{W_k\}$ de parâmetro q para formar a sequência de saída $Z_1, Z_4 \oplus W_1, Z_7 \oplus W_4, Z_{10} \oplus W_7, \cdots$. De acordo com o Teorema 1, para que a sequência dizimada seja uma sequência-m, q e $2^N - 1$ devem ser primos

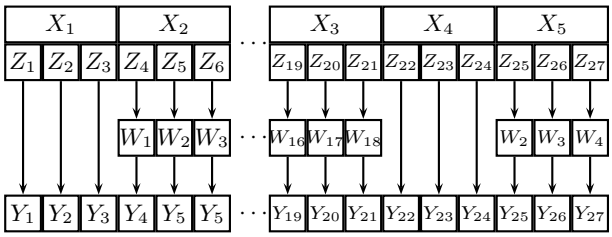

Fig. 9. Diagrama de blocos com pós-processamento para geração de um PRNG para $q = 3$ e $P = 6$.

TABELA III

N_{min} PARA DIFERENTES VALORES DE q PARA O NOVO BLOCO DE PÓS-PROCESSAMENTO COM CODIFICAÇÃO CVT

$Rx = q$	**e-tanh**	**o-tanh**	**Cúbico**	**Hénon**
1	7	6	7	8
2	7	7	7	10
3	7	7	7	11
4	6	6	7	11
5	6	6	7	11
6	7	7	7	11
7	7	7	7	11

entre si.

A Tabela III apresenta os valores de N_{min} para o bloco de pós-processamento da Fig. 8 com diferentes valores de q e codificação CVT. A utilização deste bloco leva a uma diminuição de aproximadamente 30% da memória do LFSR em relação aos valores da Tabela II, para todos os mapas, entretanto a complexidade da unidade de pós-processamento é maior com a introdução da chave I_1 e do bloco de controle.

VI. CONCLUSÕES

Apresentamos uma metodologia para o projeto de PRNG baseada em mapas caóticos que considera a taxa de bits por amostra caótica como um parâmetro do projeto. A sequência binária é obtida mediante um processo de discretização codificada das amostras caóticas. Implementamos dois métodos de codificação: CFT e CVT. Mostra-se a importância do processo de codificação na otimização da memória do bloco de pós-processamento. Baseado no comportamento da função autocorrelação da codificação CVT, um novo método de pós-processamento para esta codificação é proposto. Empregando o teste NIST, concluímos que a sequência obtida pela metodologia proposta apresenta boas propriedades criptográficas. Um prosseguimento natural deste trabalho é um estudo teórico do comportamento da função autocorrelação da codificação CVT proposta que permita uma otimização do pós-processamento em termos de memória do LFSR.

REFERÊNCIAS

[1] F. Lau and C. Tse, *Chaos-Based Digital Communication Systems*. Engineering online library, Springer, 2010.

[2] M. Eisencraft, R. Attux, and R. Suyama, *Chaotic Signals in Digital Communications*. Electrical Engineering & Applied Signal Processing Series, Taylor & Francis, 2013.

[3] P. Stavroulakis, *Chaos Applications in Telecommunications*. Taylor & Francis, 2005.

[4] L. Kocarev and S. Lian, *Chaos-based Cryptography: Theory, Algorithms and Applications*. Studies in Computational Intelligence, Springer, June 2011.

[5] L. Kocarev, J. Makraduli, and P. Amato, "Public-key encryption based on chebyshev polynomials," *Circuits, Systems and Signal Processing*, vol. 24, no. 5, pp. 497–517, October 2005.

[6] M. Yalcin, J. Suykens, and J. Vandewalle, "True random bit generation from a double-scroll attractor," *IEEE Transaction on Circuit and System*, vol. 51, no. 7, pp. 1395–1404, 2004.

[7] G. Setti, G. Mazzini, R. Rovatti, and S. Callegari, "Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos,," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 793–805, 2005.

[8] S. Callegari, M. Fabbri, and A. Beirami, "Very low cost chaos-based entropy source for the retrofit or design augmentation of networked devices," *Analog Integr Circ Sig Process*, vol. 87, pp. 155–167, 2016.

[9] A. Beirami and H. Nejati, "A framework for investigating the performance of chaotic-map truly random number generators," *IEEE Transactions on circuits and systems -II: Express Briefs*, vol. 60, no. 7, pp. 446 – 450, 2013.

[10] F. Pareschi, G. Setti, and R. Rovatti, "A fast chaos-based true random number generator for cryptographic applications," *Proceedings of the 32nd European Solid-State Circuits Conference*, pp. 130–133, 2006.

[11] A. Beirami, H. Nejati, and W. Ali, "Zigzag map: a variability-aware discrete-time chaotic-map truly random number generator," *Electronic Letters*, vol. 48, no. 24, pp. 654–656, 2012.

[12] J. A. P. Artiles, D. P. B. Chaves, J. V. C. Evangelista, and C. Pimentel, "Uma metodologia para geração de sequências aleatórias usando mapas caóticos," *in XXXIII Simpósio Brasileiro de Telecomunicações (SBrT 2015), Juiz de Fora*, pp. 1–5, Setembro 2015.

[13] L. D. Micco, C. González, H. Larrondo, M. Martin, A. Plastino, and O. Rosso, "Randomizing nonlinear maps via symbolic dynamics," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 14, pp. 3373 – 3383, 2008.

[14] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "Statistical test suite for random and pseudo random number generators for cryptographic applications.," *Special Publication 800-22 Revision 1a, National Institute of Standards and Technology*, April 2010.

[15] D. Chaves, C. Souza, and C. Pimentel, "A new map for chaotic communication," in *International Telecommunications Symposium (ITS 2014)*, pp. 1–5, Aug. 2014.

[16] J. Massey, *Cryptography: Fundamentals and Applications. Copies of transparencies*. Advances Technology Seminars, 1997.

[17] C. Paar and J. Pelzl, *Understanding Cryptography, A Textbook for Students and Practitioners*. Springer, 2010.

[18] W. Solomon and G. Guang, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar*. Cambridge University Press, July 2004.