2019年 NOLTAソサイエティ大会
IEICE, NOLTA
新潟, 2019年6月8日

拡張 Lorenz 写像に基づく疑似乱数生成ハードウェアにおける LUT の検討

Use of LUT in Pseudorandom Number Generator Hardware Based on Augmented Lorenz Map

宮内清孝 [1] 堀尾喜彦 [1] 宮野尚哉 [2] 長憲一郎 [2]
Kiyotaka Miyauchi Yoshihiko Horio Takaya Miyano Kenichiro Cho

[1] 東北大学　電気通信研究所
Research Institute of Electrical Communication, Tohoku University
[2] 立命館大学　理工学部機械工学科
Department of Mechanical Engineering, Ritsumeikan University

1．まえがき

拡張 Lorenz 写像[1] は，カオス時系列を生成することができ，この時系列から得られた 2 進乱数列は，標準的な乱数検定手法である NIST SP800-22 [2]，TestU01 BigCrush に合格する安全性を有している．しかし，現代のストリーム暗号と比較すると，その生成速度に課題があり，高速化が必要である．本稿では，ハードウェア実装による高速化，特に式中の sin 関数をルックアップテーブル(LUT)により実装する方法を検討する．

2．拡張 Lorenz 写像に基づくストリーム暗号

拡張 Lorenz 写像は，変数 Y_i を中心ノードとする $2N+1$ 次元の星型ネットワークを形成しており，

$$Y_{i+1} = \sum_{n=1}^{N} \frac{X_{n,i}}{M_n^2} \tag{1}$$

$$X_{n,i+1} = X_{n,i} Y_i - Z_{n,i} \tag{2}$$

$$Z_{n,i+1} = \sin(W_n Y_i) \tag{3}$$

と表される．ここで，

$$W_n = R \sin(M_n \phi) \tag{4}$$

$$M_n = n + \varepsilon Q_{n-1} \tag{5}$$

であり，$X_{n,i}$, Y_i, $Z_{n,i}$ は変数，R, ϕ は分岐パラメータ，i は離散時間ステップ，n は 1 から N までの整数，ε は正の微小な定数，Q_{n-1} は共有鍵を表す 2 進乱数列である．M_n は Q_{n-1} から得られるパラメータであり，$Q_0 = 0$, $M_1 = 1$ である．この際，鍵空間の大きさは 2^{N-1} となる．なお，N は任意の自然数に設定することができるので，鍵長は可変である．

2 進疑似乱数列 $P_{n,i}$ は，次式により生成される．

$$P_{n,i} = \left(10^{\alpha} X_{n,i}\right) \bmod 2 \tag{6}$$

式(6) は，$X_{n,i}$ の小数点第 α 位を参照し，その偶奇によって 2 進乱数列を生成する．この 2 進乱数列と 2 進数表示された平文との排他的論理和を計算することで暗号化を行う．

3．sin 関数の LUT を用いた実装

式(3)と(4)には sin 関数が用いられている．文献[1]では，式(3)の sin 関数は 5 次 Maclaurin 展開で，式(4)は C の math ライブラリで計算されている．このうち，式(4)の計算は最初に一度行われるのみであるが，式(3)では時間ステップ毎に N 回行われるため，計算速度の向上には式(3)の sin 関数の計算を高速化することが重要である．しかし，Maclaurin 展開による sin 関数の計算には複数回の演算が必要であり，一般に LUT を用いた計算の方が高速であると考えられる．そこで本稿では，sin 関数を LUT により実装する方法を検討する．

LUT を用いて sin 関数を実装する際には，真の sin 関数との間に誤差が生じるため，疑似乱数の性能を保つにはテーブル化に際して精度の確保が必要となる．これを検討するため，sin 関数の 1/4 波長に対する LUT を様々な分割数を用いて作成した．評価は，作成した LUT を用いて生成した疑似乱数列に対する NIST SP800-22 による検定により行った．検定には 10^6 ビットの 2 進乱数列を 1000 セット用いた．

表 1 に各 LUT を用いて生成した疑似乱数列の検定結果の一部を示す．表 1 の各列の左側は，それぞれの区間での分割数を示す．例えば，左列上段の場合，sin 関数の 1/4 波長を 2048 等分割している．表 1 より，2048，1024 等分割の場合は検定に合格するが，512 等分割の場合は合格しないことがわかる．さらに，LUT の規模を削減するため，2048 等分割の場合から部分的に分割数を減らした(表の中列・右列)．ここでの分割数は，式(3)の sin 関数に代入される変数 $W_n Y_i$ の分布を基に選択した．これらの結果より，変数値の分布を考慮することで，1/4 波長を単純に等分割するよりも少ない分割数で LUT を作成できることがわかる．

表 1　sin 関数の LUT 実装方法による乱数検定結果

分割数 [0, π/2]	検定結果	分割数 [0, π/4], [π/4, π/2]	検定結果	[0, π/8], [π/8, π/4], [π/4, 3π/8], [3π/8, π/2]	検定結果
2048	○	1024, 512	○	512, 256, 256, 128	○
1024	○	1024, 128	○		
512	×	512, 256	×	512, 256, 128, 64	○

さらに，Xilinx 社の FPGA 開発ツールである Vivado2018.3 を用いたシミュレーションにより，1/4 波長を 1024 等分割して作成した LUT を用いても，文献[1]の結果より約 70 倍高速に疑似乱数列を生成できることがわかった．また，本稿では，文献[1]と同様に 64 ビット浮動小数点演算を行ったが，演算方法を工夫することで更なる高速化が可能であると考えられる．

4．まとめ

本稿では，sin 関数を LUT により実装する方法を検討した．その結果，高速に計算が可能であることと，sin 関数に代入される変数値の分布を考慮することで，部分的に LUT の分割数を削減できることがわかった．この事は，チップ面積と消費電力の削減や，更なる高速化の可能性を示している．さらに，LUT 内のデータを飛び飛びに参照するなど，その参照方法を変更できる可能性を示している．すなわち，LUT の参照方法は，暗号のキーの 1 つとして使用できると考えられる．今後は，固定小数点演算を行うなど更なる高速化について検討する．

謝辞

本研究は JSPS 科研費 18H03307 の助成を受けたものである．

参考文献

[1] 長憲一郎，宮野尚哉，電子情報通信学会論文誌，vol. J101-A, no. 8, pp. 210-218, 2018.
[2] A. Rukhin *et al.*, NIST Special Publication 800-22 Revision 1a (Revised April 2010).