Int. J. Communications, Network and System Sciences, 2010, 3, 943-944
doi:10.4236/ijcns.2010.312128 Published Online December 2010 (http://www.SciRP.org/journal/ijcns)

On Lucas Sequences Computation

Aleksey Koval
Computer Science Department, New Jersey Institute of Technology, Newark, USA
E-mail: ak77@njit.edu
Received September 28, 2010; *revised October* 29, 2010; *accepted November* 21, 2010

Abstract

This paper introduces an improvement to a currently published algorithm to compute both Lucas "sister" sequences V_k and U_k. The proposed algorithm uses Lucas sequence properties to improve the running time by about 20% over the algorithm published in [1].

Keywords: Lucas Sequences, Cryptography

1. Introduction

Lucas sequences have been used extensively in the field of cryptography. Two cryptosystems based on DLP for Lucas sequences LUCDIF and LUCELG were introduced in [2,3]. These are Diffie-Hellman and ElGamal algorithms formulated with Lucas sequences $V_k(P,Q)$ where $Q \equiv 1 \bmod p$. Several method for efficient computation of such sequences were subsequently published in [4,5] and [6]. The computation of Lucas sequences with any Q is also valuable. In [1] the authors generalize the algorithm published in [4] for any type of Lucas sequences. In this paper we propose a change for this algorithm that significantly improves its average running time.

Such an algorithm could be useful for various purposes. As an example the authors suggest using it to compute the order of an elliptic curve. We can suggest using it for cryptosystems based on exponentiation of Gaussian integers (for example [7,8]). Gaussian integer exponentiation can be expressed in terms of Lucas sequences and vise versa ([9]). In fact, efficient algorithm to compute Lucas sequences could have many possible applications that we can't anticipate at this time.

2. Overview of Lucas Sequences

Lucas sequences are defined as sequences $U_k(P,Q)$ and $V_k(P,Q)$ ($P^2 - 4Q \neq 0$) by recurrence relations:

$$U_0 = 0;\ U_1 = 1;\ U_k(P,Q) = PU_{k-1} - QU_{k-2} \tag{1}$$

$$V_0 = 2;\ V_1 = P;\ V_k(P,Q) = PV_{k-1} - QV_{k-2} \tag{2}$$

Lucas sequences have many interesting properties and relations ([10] can be used for a reference). For the purposes of this paper we are interested in the following relation:

$$U_k = \frac{2V_{k+1} - PV_k}{P^2 - 4Q} \tag{3}$$

3. Algorithm to Compute V_k and U_k

Algorithm 3.1. Algorithm to compute V_k and U_k

Inputs: $k = \sum_{i=0}^{n-1} k_i 2^i$, where $n = \lceil \log_2 k \rceil$

(P, Q)–Lucas sequence parameters

Outputs: (V_k, U_k)

1) $V_l := 2$; $V_h := P$;
2) $Q_l := 1$; $Q_h := 1$;
3) **for** $j = n - 1$ **downto** 0
4) $\quad Q_l := Q_l * Q_h$;
5) $\quad$ **if** $(k[j] = 1)$
6) $\quad\quad Q_h := Q_l * Q$;
7) $\quad\quad V_l := V_h * V_l - P * Q_l$;
8) $\quad\quad V_h := V_h * V_h - 2 * Q_h$;
9) $\quad$ **else**
10) $\quad\quad Q_h := Q_l$;
11) $\quad\quad V_h := V_h * V_l - P * Q_l$;
12) $\quad\quad V_l := V_l * V_l - 2 * Q_h$;
13) $\quad$ **endif**
14) **endfor**
15) $U_k := (2 * V_h - P * V_l)/(P * P - 4 * Q)$;
16) **return** (V_l, U_k)

The Algorithm 3.1 looks very much like the algorithm in [1]. The difference is that we do not compute U_h as it is done in [1]. Instead we use relation (3) to compute U_k

on line 15. This allows us to significantly cut the number of multiplications. For a random k the number of multiplications in the algorithm presented in [1] is $\frac{11\log_2 k}{2}$. The number of multiplications in the Algorithm 3.1 is

$$\frac{9\log_2 k}{2}+2. \tag{4}$$

Note: the multiplication by a small constants (2 or 4) on lines 8, 12 and 14 have the running time of additions and, therefore, are not included in Equation (4).

4. Conclusions

In this paper we presented an improved algorithm to compute Lucas sequences V_k and U_k. The proposed algorithm allows for approximately 20% improvement in running time, which could be significant, especially if it is used in real time cryptographic systems. Moreover, the improved algorithm does not require any special precalculations and there are no tradeoffs or restrictions.

5. References

[1] M. Joye and J. J. Quisquater, "Efficient Computation of Full Lucas Sequences," *Electronics Letters*, Vol. 32, No. 6, 1996, pp. 537-538.

[2] P. Smith, "Cryptography without Exponentiation," *Dr. Dobb's Journal*, Vol. 19, No. 4, April 1994, pp. 26-30.

[3] P. Smith, "Luc Public Key Encryption: A Secure Alternative to RSA," *Dr. Dobb's Journal*, Vol. 18, No. 1, 1993, pp. 44-49.

[4] S. M. Yen and C. S. Laih, "Fast Algorithms for Luc Digital Signature Computation," *IEE Proceedings*: *Computers and Digital Techniques*, Vol. 142, No. 2, 1995, pp. 165-169.

[5] C.-T. Wang, C.-C. Chang and C.-H. Lin, "A Method for Computing Lucas Sequences," *Computers* & *Mathematics with Applications*, Vol. 38, No. 11-12, 1999, pp. 187-196.

[6] M. Othman, E. M. Abulhirat, Z. M. Ali, M. R. M. Said and R. Johari, "A New Computation Algorithm for a Cryptosystem Based on Lucas Functions," *Journal of Computer Science*, Vol. 4, No. 12, 2008, pp. 1056-1060.

[7] A. El-Kassar, M. Rizk, N. Mirza and Y. Awad, "El-Gamal Public-Key Cryptosystem in the Domain of Gaussian Integers," *International Journal of Applied Mathematics*, Vol. 7, No. 4, 2001, pp. 405-412.

[8] H. Elkamchouchi, K. Elshenawy and H. Shaban, "Extended RSA Cryptosystem and Digital Signature Schemes in the Domain of Gaussian Integers," *Proceedings of the* 8*th IEEE International Conference on Communication Systems*, Singapore, Vol. 1, 25-28 November 2002, pp. 91-95.

[9] A. Koval and B. S. Verkhovsky, "On Discrete Logarithm Problem for Gaussian Integers," Proceedings of International Conference on Information Security and Privacy, Orlando, 13-16 July 2009, pp. 79-84.

[10] L. E. Dickson, "Recurring Series; Lucas' U_n, V_n," *History of the Theory of Numbers*: *Divisibility and Primality*, Dover Publications, New York, Vol. 1, 2005, pp. 393-411.