IMPROVING EXISTING PRSG USING QSP

J. Markovski, V. Dimitrova

Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University
Arhimedova b.b., PO Box 162, 1000 Skopje, Macedonia
{jasen, vesnap}@ii.edu.mk

Abstract: Pseudo random sequence generators (PRSG) are deterministic algorithms that produce sequences of elements that imitate natural random behavior. Random sequences have extensive use in scientific experiments as input sequences for different kinds of simulators, in cryptography for preparation of keys and establishing communication, in authentication for preparation of identification numbers, smart cards, serial numbers, etc. For this reason the field of pseudo random generators is widely exploited. However, widely available PRSGs have limited periods (for example, 264), which mean that the pseudo random sequences start repeating the same elements (after at most 264 elements). This makes them inappropriate for large-scale scientific experiments, cryptography and authentication, because they produce predictable pseudo random output sequences. In this paper we will try to improve some of existing linear PRSGs using quasigroup sequence processing (QSP).

Keywords: pseudo random sequence generator, quasigroup sequence processing, period of PRSG

1 Introduction

Pseudo random sequences are widely used in many fields like cryptography, authentication, cryptanalysis etc. They are produced by Pseudo Random Sequence Generators (PRSG), which present deterministic algorithms that produce seemingly random sequences of elements. Input of the PRSG is some truly random sequence of elements called a seed (Knuth, 1974) (Schneier, 1996) (Simmons, 1992).

Since PRSG has a deterministic nature, every pseudo random sequence has a period. It represents the distance between two sequential appearances of the same pseudo random sequence. This means that the sequence of pseudo random elements will eventually start to repeat. Obviously, the ideal random generator has a period of infinite length (Knuth, 1974)(Schneier, 1996).

Every PRSG has its deficiencies and there are not many all purpose PRSGs. For example, lots of widely available PRSGs are inappropriate for cryptography and authentication, because they produce predictable pseudo random output sequence that start repeating after relatively short time of usage (Knuth, 1974) (Schneier, 1996) (Simmons, 1992).

Secure PRSGs produce unpredictable sequences of elements and have seemingly infinite period (Menezes, 1996). For example, it is possible to build next bit predictors for the linear congruence generators using several schemes. These predictors provide the next pseudo random bit if they have sufficiently many consecutive bits of the pseudo random sequence (Schneier, 1996).

There are two big families of PRSGs: (1) linear PRSGs, which rely on linear congruence functions or linear shift feedback registers and (2) nonlinear PRSGs, which are built using some other method. The best-known and most widely available implementations of non-linear PRSGs are based on some combination, filtering or summation of several linear PRSG (Menezes, 1996)(Scheiner, 1996).

In our previous research we gave several possible implementations of a new type of PRSG called Quasigroup PRSG (QPRSG) (Dimitrova, 2003a)(Dimitrova, 2003b). We have shown that QPRSG presents highly flexible PRSG which can produce pseudo random sequences with arbitrary periods.

In this paper we will show how to use the method of Quasigroup Sequence Processing (QSP) (Markovski, 1999)(Markovski 2000) to improve existing PRSG. In the beginning we give a brief introduction to quasigroup sequence processing and QPRSG. Afterwards, we review the property of the quasigroups called coefficient of period growth introduced in (Dimitrova, 2003b). Next, we extend the implementation of QPRSG introduced in (Dimitrova, 2003a) and present statistical results to examine the properties of the extended QPRSG. We finish with appropriate conclusion and directions for future work.

2 Properties of QSP and QPRSG

A quasigroup (Q, $*$) is a groupoid (i.e. algebra with one binary operation $*$ on the set Q) satisfying the law:

$$(\forall u, v \in Q)(\exists! x, y \in Q)\ (x * u = v \ \&\ u * y = v) \qquad (1)$$

In other words, the equations $x*u=v$, $u*y=v$ for each given $u, v \in Q$ have unique solutions x, y (Denes, 1974).

Let Q be a set of elements ($|Q|\geq 2$). We denote by $Q^{+}=\{x_1x_2...x_k | x_i \in Q, k\geq 2\}$ the set of all finite sequences with elements of Q. Assuming that (Q,$*$) is a given quasigroup, for a fixed element a$\in Q$, we define transformation $E_a^{(1)}: Q^{+} \rightarrow Q^{+}$ on the quasigroup as follows (Markovski, 1999):

$$E_a^{(1)}(x_1x_2...x_k) = y_1y_2...y_k \Leftrightarrow \begin{cases} y_1 = a * x_1 \\ y_{i+1} = y_i * x_{i+1} \end{cases} \qquad (2)$$

Also, we define $E_a^{(s)} = E_a^{(1)} \circ E_a^{(1)} \circ ... \circ E_a^{(1)}$ (s times) (Markovski, 2003) to be:

$$E_a^{(s)}\ (a_1a_2...a_k) = a_1^{(s)}\ a_2^{(s)} ...\ a_k^{(s)} \qquad (3)$$

Based on the transformation E, one possible implementation of the QPRSG is shown on Fig. 1 (Dimitrova, 2003a).

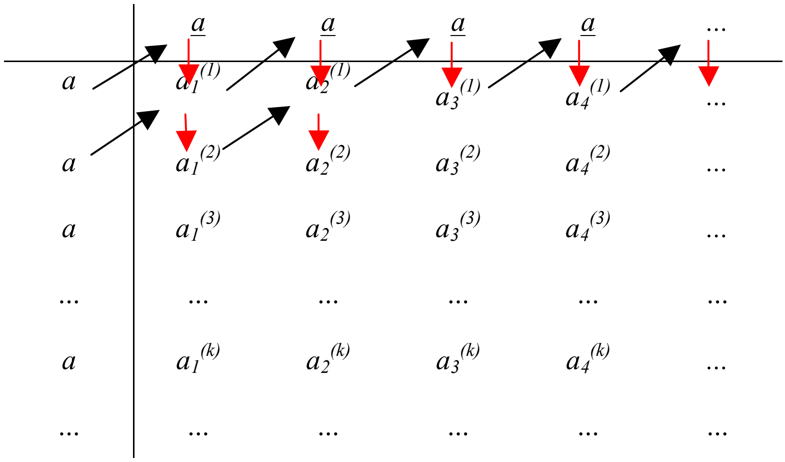
a a a a ...
a $a_1^{(1)}$ $a_2^{(1)}$ $a_3^{(1)}$ $a_4^{(1)}$ ...
a $a_1^{(2)}$ $a_2^{(2)}$ $a_3^{(2)}$ $a_4^{(2)}$ ...
a $a_1^{(3)}$ $a_2^{(3)}$ $a_3^{(3)}$ $a_4^{(3)}$ ...
... ...
a $a_1^{(k)}$ $a_2^{(k)}$ $a_3^{(k)}$ $a_4^{(k)}$ ...
... ...

Fig. 10: Implementation of a PRSG using quasigroup processing

Element a is an arbitrary element of the quasigroup Q such that $a*a \neq a$. Sequence $a_1^{(1)}\ a_2^{(1)}\ a_3^{(1)}\ldots$ is obtained as $E_a\,(aaa\ldots)$. Sequence $a_1^{(k)}\ a_2^{(k)}\ a_3^{(k)}\ldots$ is obtained as

$$E_a^{(k)}\ (aaa\ldots)= E_a\ (a_1^{(k-1)}\ a_2^{(k-1)}\ a_3^{(k-1)}\ldots) \qquad (4)$$

The pseudo random sequence is $a_1^{(k)}a_2^{(k)}a_3^{(k)}\ldots$, where k is large enough. This sequence presents a periodical sequence. How fast the period grows depends solely on the quasigroup used to produce the sequence (Dimitrova, 2003b). The backbone for the QPRSG provides the Theorem 1 proved in (Markovski, 1999).

Theorem 1: Let $1 \leq l \leq n$, $\alpha = a_1a_2\ldots a_k \in Q^+$ and $\beta = E^{(n)}(\alpha)$. Then the distribution of subsequences of β of length l is uniform. ■

This provides a natural behaviour of the pseudo random subsequences of length not greater than n.

Experimental results presented in (Dimitrova, 2003b) show that the growth of the period of the QPRSG depends not only of quasigroup, but also of the times of application of transformation E.

The relation between the times of application of the transformation E and the period growth of the QPRSG shows the Theorem 2 proved in (Markovski, 2003).

Theorem 2: Let α be a sequence of k elements. If the period of $E_a(\alpha)$ is p_0, then the sequences $E_a^{(s)}(\alpha)$ are periodical with periods p_{s-1} correspondingly, all of which are multiples of p_0. The periods satisfy the law $p_{p_{s-1}} > p_{s-1}$ for each $s \geq 1$. ■

We have in (Dimitrova, 2003b) that not all quasigroups provide the same period of the QPRSG. Experimental results show that, there exist quasigroups which increase the period of the QPRSG with each application of the transformation E.

Every quasigroup has a coefficient of period growth, which represents how many times the period has grown after one application of the transformation E. Fig. 2 presents the coefficient of the period growth for quasigroups of order 10 (Dimitrova, 2003b).

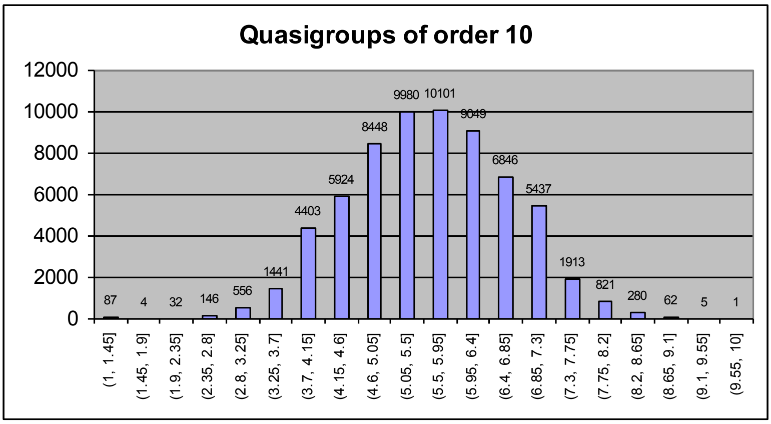

Fig. 2: Distribution of the coefficient of growth for quasigroups of order 10

This experiment was made on 2^{16} randomly chosen quasigroups. The quasigroups we subjected to seven consecutive applications of the transformation E. Afterwards, the average of the obtained coefficients of the period growth was counted for 20 intervals from 1 to 10.

3 Extending QPRSG

Using the method Quasigroup Sequence Processing (QSP) (Markovski, 1999) (Markovski 2000) we try to improve existing PRSG. We extend the implementation of QPRSG taking the starting array $b_1b_2b_3$...to be a pseudo random sequence.

On Fig. 3 is shown implementation of the extending PRSG.

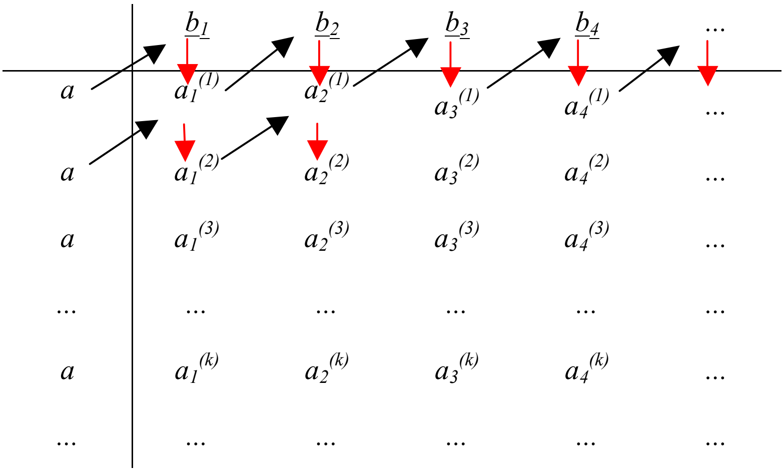

Fig 3: Implementation of the extending PRSG

The element a is an arbitrary element of the quasigroup Q such that $a*a \neq a$. The elements b_i , $i=1, 2, 3, \ldots$ are also arbitrary elements of the quasigroup Q.

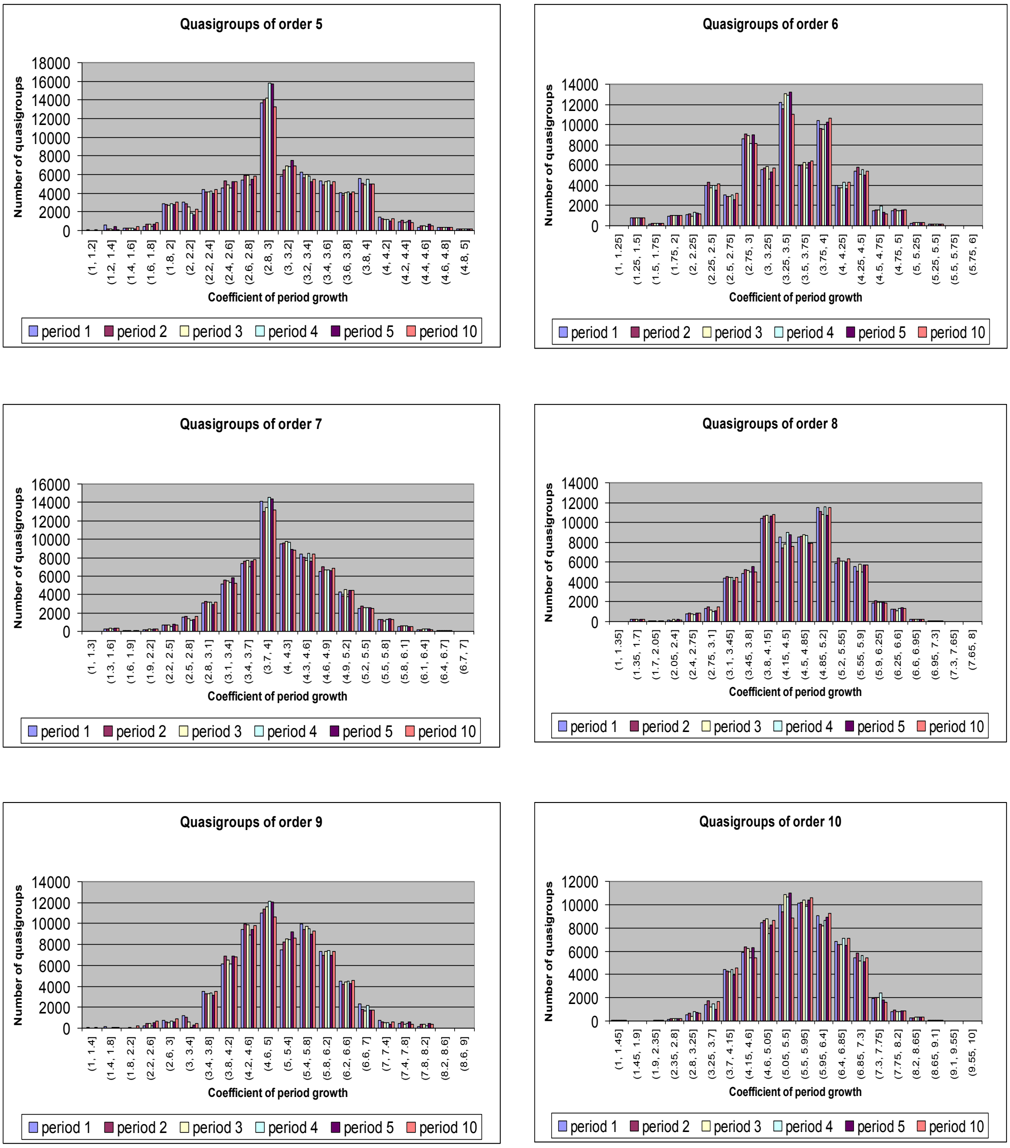

Fig. 4: Comparison of coefficient of period growth for quasigroups of various orders

The sequence $a_1^{(1)}\ a_2^{(1)}\ a_3^{(1)} \ldots$ is obtained from the transformation E as $E_a\ (b_1 b_2 b_3 \ldots)$. The sequence $a_1^{(k)}\ a_2^{(k)}\ a_3^{(k)} \ldots$ is obtained as $E_a(a_1^{(k-1)}\ a_2^{(k-1)}\ a_3^{(k-1)} \ldots)$, which means that the transformation E is applied k times on the starting sequence $b_1 b_2 b_3 \ldots$.

The pseudo random sequence produced from the extending QPRSG is $a_1^{(k)}a_2^{(k)}a_3^{(k)}\ldots$, where k is large enough.

Fig. 4 presents comparison of coefficients of period growth for quasigroups of orders 5, 6, 7, 8, 9 and 10. The values of the coefficient of period growth are grouped into 20 equal intervals.

Fig. 5 presents an experimental result of the period after first transformation for quasigroups of order 5, 6, 7, 8, 9 and 10. The period grows proportionally to the period of the starting array. Thus, the choice of the starting array may improve the slow start of the QPRSG.

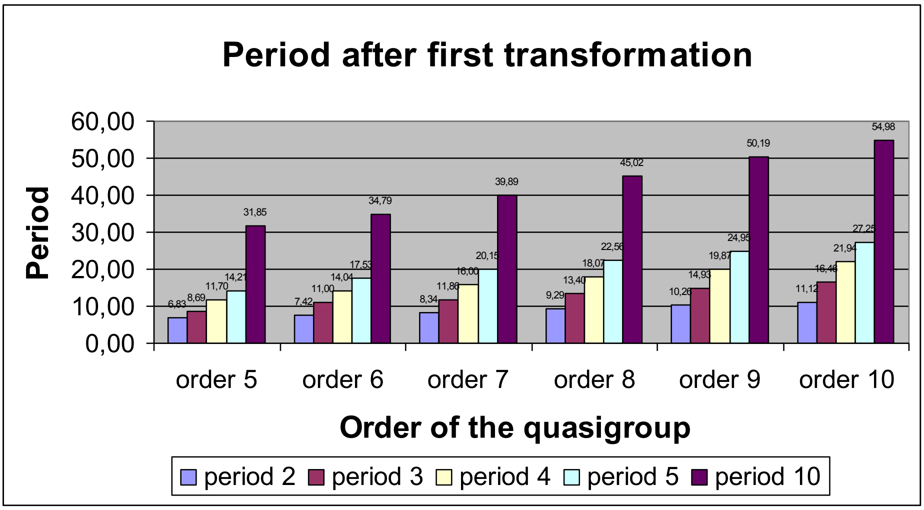

Fig. 5: Period after first transformation

4 Conclusion

In this paper we presented a way to improve existing PRSG by increasing their period using QSP. Additionally, we can improve the slow start time of the QPRSG introduced in (Dimitrova, 2003b), which makes it even more applicable in real-world applications.

Future work should be done in improving the choice of the quasigroups for the basis of the QPRSG. Additionally, some quasigroups have low coefficients of period growth for monotonous sequences, on which they improve for higher number of applications of the transformation E.

5 Acknowledgements

We would like to thank Prof. Smile Markovski for his useful comments and guidelines.

6 References

1. Dènes, J., Keedwell, D., *Latin Squares and their Applications*, English Univer. Press. Ltd., 1974
2. Dimitrova, V., Markovski, J. "Implementation of Pseudo Random Sequence Generator using Quasigroup Processing", *ETAI' 2003*, Ohrid, Macedonia, 2003
3. Dimitrova, V., Markovski, J., "On Quasigroup Pseudo Random Sequence Generators", *First Balkan Conference in Informatics, BCI1*, Thessaloniki, Greece, 2003
4. Knuth, D., The *Art of Computer Programming, Volume 2*, Addison-Wesley, 1977
5. Markovski, S., "Quasigroup String Processing and Applications in Cryptography", *1st Conference of Mathematics and Informatics for Industry*, 2003, Thessaloniki, Greece
6. Markovski, S., Gligoroski, D., Bakeva, V., "Quasigroup and Hash Functions", *Disc. Math. and Appl., Sl. Shtrakov and K. Denecke ed., Proceedings of the 6th ICGMA*, Bansko, 2001
7. Markovski, S., Gligoroski, D., Bakeva, V., "Quasigroup String Processing – Part 1", *ISSN 0351-3246*, Macedonian Academy of Sciences and Arts, Skopje, 1999
8. Markovski, S., Kusakatov, V., " Quasigroup String Processing – Part 2", *Contributions, Sec. math. Tech. Sci.*, MANU, Skopje, 2000
9. Menezes, A., van Oorschot, P., Vanstone, S., *Handbook of Applied Cryptography*, CRC Press, 1996
10. Schneier, B., *Applied Cryptography*, John Wiley & Sons, 1996
11. Simmons, G., *Contemporary Cryptography*, IEEE press, 1992
12. Stinson, D., *Cryptography -Theory and Practice*, CRC press, 1995