An Approximate Distribution for the Maximum Order Complexity

Diane Erdmann

Applied Decision Analysis Inc., 2710 Sand Hill Road,
Menlo Park, CA 94025, U.S.A.

Sean Murphy*

Information Security Group, Royal Holloway,
University of London, Egham, Surrey TW20 0EX, U.K.

September 18, 1995

Abstract

In this paper we give an approximate probability distribution for the maximum order complexity of a random binary sequence. This enables the development of statistical tests based on maximum order complexity for the testing of a binary sequence generator. These tests are analogous to those based on linear complexity.

Key Words. Binary Sequence, Stream Cipher, Feddback Shift Register, Maximum Order Complexity

*This author acknowledges the support of the Nuffield Foundation

1 Introduction

The *linear complexity* is a well–known tool for assessing the cryptographic strength of a binary sequence. For a given sequence, it measures the length of the shortest *linear feedback shift register* (LFSR) that can generate the sequence. The linear complexity is easily calculated using the Berlekamp–Massey algorithm [5], which also gives a corresponding LFSR. A sequence with a low linear complexity can therefore easily be simulated, and so a sequence with a large linear complexity is clearly necessary condition for a secure keystream. If we let S_n is the first n bits of the sequence S and $C(\cdot)$ be the linear complexity, then Rueppel [8] has shown that for a random sequence of n bits S_n,

$$E(C(S_n)) \approx \frac{n}{2} \text{ and } Var(C(S_n)) \approx 1.$$

This result can be used as the basis for statistically testing one aspect of a sequence's randomness. Rueppel also introduced the idea of a *linear complexity profile*, which is $C(S_n)$ considered as a function of n. He gave some properties of a "typical" linear complexity profile for a random sequence. Thus the linear complexity profile can be used to construct statistical tests for the randomness of a sequence as discussed by Wang [9], Carter [1] and Niederreiter [7].

In this paper, we consider general (nonlinear) *feedback shift registers* (FSR). Accordingly we define the *maximum order complexity* of a sequence S_n to be the shortest feedback register that can generate the sequence S_n. The maximum order complexity can be calculated efficiently using, for example, a *directed acyclic word graph* or a *suffix tree*. These two methods are similar, and further details are given in Jansen [4] and Erdmann [2] respectively. Therefore just as for low linear complexity, a sequence with low maximum order complexity can easily be simulated, and so a sequence with a large maximum order complexity is clearly necessary condition for a secure keystream. A detailed account of the cryptographic aspects of the maximum order complexity is given by Jansen [4]. To find the maximum order complexity of a sequence we need to find the largest value k such that there exists a subsequence of length k that appears more than once in the sequence and does not have the same element following it each time it appears. The maximum order complexity of that sequence is then $k+1$. For example, in the binary sequence

$$0110010101101,$$

k must be at least 4 since the 4-bit subsequence 0110 is followed by a 0 the first time it appears in the sequence and by a 1 the second time. On the other hand, k cannot be 5 since every 5-bit subsequence appears only once and so no two 5-bit subsequences have the same successor. Therefore $k = 4$, and the maximum order complexity of the sequence 0110010101101 is $k + 1 = 5$. The obvious brute force method of finding such a k for a general sequence, by checking all 1-bit subsequences and then all 2-bit subsequences and so on until the successor condition given above is no longer satisfied, is highly inefficient.

The maximum order complexity has many similarities with the linear complexity, and it would be useful for cryptographic purposes if we could construct analogous tests for maximum order complexity to those which have been constructed for linear complexity. In order to do this, it is necessary to know about the distribution of the maximum order complexity for random sequences. Unfortunately the distribution of the maximum order complexity of a random sequence is difficult to calculate exactly. In this paper we present a way to approximate the distribution of the maximum order complexity. From this approximation we can construct statistical tests in order to test the randomness of a sequence. Our approximation of the distribution of the maximum order complexity depends on a function that approximates $P(c, k)$, the probability that the first k c-tuples in a sequence are all different. The approximation to this function will be described in Section 2. In Section 3 we use this approximation when considering purely periodically repeating sequences, and in Section 4 when considering all sequences of a given length. In Section 5, we construct some statistical tests based on the maximum order complexity, and finally we give some conclusions in Section 6.

2 An Approximation for $P(c, k)$

In order to approximate the distribution of the maximum order complexity of a random sequence we first need to find a function that approximates $P(c, k)$, the probability that the first k c-tuples in a sequence are all different or *unique* We can give the following recursive formula for $P(c, k)$.

Lemma: $P(c,k) = \prod_{i=1}^{k-1} R(c,i)$, where $R(c,j-1)$ is the conditional probability of the j^{th} c-tuple being unique given that the first $(j-1)$ c-tuples are unique.

Proof: $P(c,k) = P(\text{first } k \text{ } c\text{-tuples are unique})$, so

$$\begin{aligned} P(c,k) &= P(k^{th} \text{ } c\text{-tuple is unique} \mid \text{first } k-1 \text{ } c\text{-tuples are unique}) \\ &\quad \times P(\text{first } k-1 \text{ } c\text{-tuples are unique}) \\ &= R(c,k-1)P(c,k-1), \end{aligned}$$

where $R(c,k-1)$ is the conditional probability of the k^{th} c-tuple being unique given that the first $k-1$ c-tuples are unique. Thus we have

$$P(c,k) = R(c,k-1)R(c,k-2)\cdots R(c,1)P(c,1),$$

and since $P(c,1) = P(\text{first } c\text{-tuple is unique}) = 1$, we have

$$P(c,k) = \prod_{i=1}^{k-1} R(c,i). \blacksquare$$

To approximate the conditional probability $R(c,i)$, we consider a (binary) *deBruijn–Good* graph of order $(c-1)$. A deBruijn–Good graph of order a consists of 2^a nodes each with a unique a-bit label. A directed edge exists from one node to another if the label of the latter begins with the final $(a-1)$ bits of the former. Note that every node has two in–edges and two out–edges, and there is a natural bijection between the 2^{a+1} edges and the set of $(a+1)$-tuples. Any sequence of length greater than a can now be represented as a path in the deBruijn–Good graph of order a consisting of the successive a-tuples of the sequence. For any node Z of this graph, we can define its *conjugate* node $\tilde{Z}$. If Z has label $z_1 z_2 \cdots z_a$ then its conjugate node $\tilde{Z}$ is the node with label $\bar{z_1} z_2 \cdots z_a$, where $\bar{z_1} = z_1 \oplus 1$. Nodes Z and $\tilde{Z}$ have their last $(a-1)$ bits in common, so there is a node that has a directed edge from both Z and $\tilde{Z}$.

$R(c,k-1)$ is the probability that the k^{th} c-tuple is unique given that the previous $(k-1)$ c-tuples were unique. Equivalently $R(c,k-1)$ is the probability that the next edge traversed in the path in the deBruijn–Good graph of order $(c-1)$ is one not traversed before, given that $(k-1)$ unique edges have been traversed so far. It is important to note here that there are exactly two edges leaving each node, and exactly two edges leading to each node, so while each edge must only be traversed once, it is possible for a node to be visited twice. Suppose we now call the last node in this sequence B, and the one preceding it A, so B is the k^{th} node and A is the $(k-1)^{th}$ node. If B was not seen before in the sequence then neither of the edges leaving B has been traversed and so taking either edge guarantees us that the k^{th} c-tuple is unique. However, if B was seen once before then one of its out–edges has been traversed, and so the next edge will be unique with probability $\frac{1}{2}$. Similarly if B was seen twice before, then both out–edges of B must have been traversed.

If we let β_i be the event that B occurs exactly i times in the path prior to the k^{th} node, and let α be the event that all the $k-1$ edges in the path are unique then

$$\begin{aligned} R(c,k-1) &= P(k^{th} c\text{-tuple is unique} \mid \text{first } k-1 \; c\text{-tuples are unique}) \\ &= P(\beta_0|\alpha) + \tfrac{1}{2}P(\beta_1|\alpha). \end{aligned}$$

The probability that B has already occurred in the sequence depends on the conjugate node of A, $\tilde{A}$. We introduce $\tilde{A}$ in our argument as it is easier to estimate the probability that $\tilde{A}$ occurs in our path than it is to estimate the same probability for B. This difference is a result of the fact that we know B has already occurred once in the path as the k^{th} node, so we know that one of its predecessors has also occurred at least once; whereas we know nothing about $\tilde{A}$ and its predecessors. Since our path has unique edges, B can only occur as one of the first $k-1$ nodes if either it is the first node or if it follows $\tilde{A}$, whereas $\tilde{A}$ can only occur as one of the first $k-1$ nodes if it is either the first node or if it follows either of its two predecessors. The following asymptotic result concerning the number of times the node $\tilde{A}$ occurs will enable us to calculate $R(c,k-1)$.

Lemma[2]: For large c and moderate h, the number of times the conjugate vertex $\tilde{A}$ of the penultimate vertex A appears in the first h nodes of a path is approximately binomially distributed with parameters h and $2^{-(c-1)}$ for almost all vertices A.

Corollary: If N_i denotes the i^{th} node in the path then

$$\begin{aligned} P(\tilde{A} \neq N_i \text{ for all } i \text{ in } 1,\ldots,h) &\approx \binom{h}{0}(1-\tfrac{1}{2^{c-1}})^h, \\ P(\tilde{A} = N_i \text{ for exactly one } i \text{ in } 1,\ldots,h) &\approx \binom{h}{1}(\tfrac{1}{2^{c-1}})(1-\tfrac{1}{2^{c-1}})^{h-1}, \\ P(\tilde{A} = N_i \text{ for exactly two } i\text{'s in } 1,\ldots,h) &\approx \binom{h}{2}(\tfrac{1}{2^{c-1}})^2(1-\tfrac{1}{2^{c-1}})^{h-2}. \end{aligned}$$

If $\tilde{A}$ appears in a path with random edges, then B follows $\tilde{A}$ with probability $\frac{1}{2}$. For the case of large c and moderate path lengths, the deBruijn–Good graph has many edges and many possible paths so there is little difference between the probabilities that B follows $\tilde{A}$ in a path with random edges and and in a path with unique edges. Thus we take the probability that B follows $\tilde{A}$ to be $\frac{1}{2}$ when we consider paths with unique edges. This gives the following result.

Theorem: $R(c,k-1) = 1 - \frac{k}{2^{c+1}} + O\left(\frac{k^3}{2^{3c}}\right).$

Proof: Let γ be the event that B is the first node, and γ' the event that B is not the first node, then we have

$$\begin{aligned} P(\beta_0|\alpha) &= P(\beta_0|\alpha,\gamma)P(\gamma) + P(\beta_0|\alpha,\gamma')P(\gamma') \\ \text{and } P(\beta_1|\alpha) &= P(\beta_1|\alpha,\gamma)P(\gamma) + P(\beta_1|\alpha,\gamma')P(\gamma'). \end{aligned}$$

Now, $P(\beta_0|\alpha,\gamma) = 0$, and, if N_i is the i^{th} node in the path then

$$\begin{aligned} P(\beta_0|\alpha,\gamma') &\approx \tfrac{1}{2}P(\tilde{A} = N_i \text{ for exactly one } i \text{ in } 1,\ldots,k-2 \;|\alpha) \\ &\quad + P(\tilde{A} \neq N_i \text{ for all } i \text{ in } 1,\ldots,k-2 \;|\alpha), \end{aligned}$$

$$\begin{aligned} P(\beta_1|\alpha,\gamma) &= P(B \neq N_i \text{ for all } i = 2,\ldots,k-1 \;|\alpha) \\ &\approx \tfrac{1}{2}P(\tilde{A} = N_i \text{ for exactly one } i \text{ in } 1,\ldots,k-2 \;|\alpha) \\ &\quad + P(\tilde{A} \neq N_i \text{ for all } i \text{ in } 1,\ldots,k-2 \;|\alpha), \end{aligned}$$

$$\begin{aligned} \text{and } P(\beta_1|\alpha,\gamma') &= P(B = N_i \text{ for exactly one } i \text{ in } 2,\ldots,k-1 \;|\alpha) \\ &\approx \tfrac{1}{2}P(\tilde{A} = N_i \text{ for exactly one } i \text{ in } 1,\ldots,k-2 \;|\alpha) \\ &\quad + P(\tilde{A} = N_i \text{ for exactly two } i\text{'s in } 1,\ldots,k-2 \;|\alpha). \end{aligned}$$

For reasons given above, we assume that these probabilities are approximately correct even when conditional on α, the event that all the previous edges are unique. Thus we have

$$\begin{aligned} P(\beta_0|\alpha) &= P(\beta_0|\alpha,\gamma)P(\gamma) + P(\beta_0|\alpha,\gamma')P(\gamma') \\ &\approx \tfrac{(k-2)}{2^c}(1-\tfrac{1}{2^{c-1}})^{k-2} + (1-\tfrac{1}{2^{c-1}})^{k-1} \end{aligned}$$

$$\begin{aligned} \text{and } P(\beta_1|\alpha) &= P(\beta_1|\alpha,\gamma)P(\gamma) + P(\beta_1|\alpha,\gamma')P(\gamma') \\ &\approx \tfrac{(k-2)^2}{2^{2c-1}}(1-\tfrac{1}{2^{c-1}})^{k-3} + (\tfrac{k}{2^c})(1-\tfrac{1}{2^{c-1}})^{k-2}, \end{aligned}$$

and since $R(c,k-1) = P(\beta_0|\alpha) + \frac{1}{2}P(\beta_1|\alpha)$, we have

$$\begin{aligned} R(c,k-1) &\approx \left(1-\tfrac{1}{2^{c-1}}\right)^{k-3}\tfrac{(k-2)^2}{2^{2c}} + \left(1-\tfrac{1}{2^{c-1}}\right)^{k-2}\left(\tfrac{k-2}{2^c}+\tfrac{k}{2^{c+1}}\right) \\ &\quad + \left(1-\tfrac{1}{2^{c-1}}\right)^{k-1} \\ &= \left(1-\tfrac{3(k-4)}{2^{c+1}}+\tfrac{(k-3)(k-4)}{2^{2c}}\right)\left(1-\tfrac{1}{2^{c-1}}\right)^{(k-3)} \\ &= 1-\tfrac{k}{2^{c+1}}+O\left(\tfrac{k^3}{2^{3c}}\right). \qquad \blacksquare \end{aligned}$$

Thus if we define

$$r(c,k-1) = 1-\frac{k}{2^{c+1}},$$

then $R(c,k-1)$ is well-approximated by $r(c,k-1)$, and so $P(c,k)$ is well-approximated by

$$p(c,k) = \prod_{i=1}^{k-1} r(c,i) = \prod_{i=1}^{k-1}\left(1-\frac{i+1}{2^{c+1}}\right).$$

To test this approximation, we compared the values of $p(c,k)$ with estimated values $\tilde{P}(c,k)$ for the probability $P(c,k)$ obtained by simulation. These are based on 10,000 simulations and are calculated by counting the number of sequences of length $k+c-1$ that have no repeated c-tuple. The results for $k=16$ are given in Table 1, and the results for $k=32$ in Table 2.

3 Pure Periodically Repeating Sequences

In this section we consider pure periodically repeating sequences, that is sequences that consist of k bits that form one period of the sequence and are then repeated. We give an approximation for the probability that such a sequence has complexity c. A pure periodically repeating sequence has complexity c if the first k c-tuples are unique but at least one of the first k $(c-1)$-tuples is repeated. Thus to determine the complexity of such a sequence, we need only to look at the first $(k+c-1)$ bits to see if the k c-tuples are unique and the first $(k+c-2)$ bits to see if the k $(c-1)$-tuples are not unique. If $Q(c,k)$ denotes the probability that the first k c-tuples are unique while the k $(c-1)$-tuples are not unique, then $Q(c,k)$ is well-approximated by $q(c,k)$, where

$$q(c,k) = \begin{cases} p(c,k) - p(c-1,k) & k \leq 2^{c-1} \\ p(c,k) & \text{otherwise.} \end{cases}$$

Note that $Q(c,k)$ and $q(c,k)$ are only defined for sequences of length at least $(k+c-1)$ and for $k \leq 2^c$.

We can compare the values we obtain using $q(c,k)$ with the true probabilities found by Jansen [4]. Table 3.2 of Jansen's thesis contains the results of computing the complexity of all pure periodically repeating sequences with period length $k \leq 24$. From this table we can calculate the true probability $Q(c,k)$ and compare it with our approximation $q(c,k)$. Figure 1 gives this comparison graphically for period length $k = 24$. We can see that the approximations look very similar to the true values.

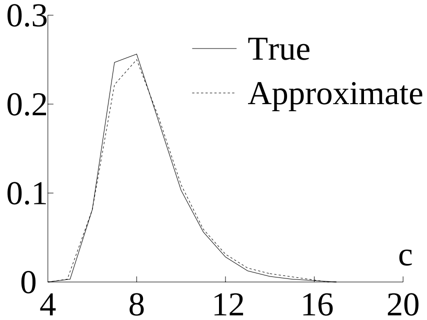

Figure 1:
True $(Q(c,24))$ versus Approximated $(q(c,24))$ Probability
that the first k c-tuples are unique
while the k $(c-1)$-tuples are not unique.

Having computed the approximate distribution of the maximum order complexity for random pure periodically repeating sequences, we can compute the approximate mean and variance of the complexity. Let Q_k be a random variable that denotes the maximum order complexity of a pure periodically repeating sequence of length k, and let $\hat{Q}_k$ denote our approximation

to Q_k. We now have the following result.
Lemma[2]: The approximate mean and variance of $\hat{Q}_k$ are given by:

$$E(\hat{Q}_k) \approx \textstyle\sum_{c=\lceil \log_2 k \rceil}^{k-1} c\; q(c,k) = (k-1) - \sum_{c=\lceil \log_2 k \rceil}^{k-2} p(c,k),$$
$$\mathrm{Var}(\hat{Q}_k) \approx (k-1)^2 - \textstyle\sum_{c=\lceil \log_2 k \rceil}^{k-2} (2c+1) p(c,k) - E(\hat{Q}_k)^2.$$

The results of these computations are given in Table 3, where they are compared with the true values. The approximation is becoming more accurate as k increases. As it is easy to compute the mean and variance of the maximum order complexity exactly for small k, it is not a problem that the results are not as accurate in that region.

The problem of determining the complexity of a periodically repeating sequence can be looked at in another way. Namely, we observe that calculating the probability that a periodically repeating sequence with period length k has complexity at most c is the same as calculating the probability that a random cycle of length k occurs in a deBruijn–Good graph of order at most c. Maurer [6] has obtained tight bounds asymptotically on the number of cycles of length k in deBruijn–Good graphs of order at most c. He has shown for every positive real number x that the $\limsup$ as $k \to \infty$ of the probability that a cycle of length k occurs in a deBruijn–Good graph of order at most $\lceil 2 \log_2 k - 2 \log_2 \log_2 k - x \rceil$ is less than or equal to $e^{-2^{x-5}}$. He also has shown that the $\liminf$ as $k \to \infty$ of the probability that a cycle of length k occurs in a deBruijn–Good graph of order at most $\lceil 2 \log_2 k + x \rceil$ is at least $1 - 2^{-(x+1)}$. Erdmann [2] has shown that the bounds obtained using the approximate probabilities are consistent with Maurer's results and that they are also consistent with the results of Zubkov and Mikhailov [10].

4 Random Sequences

In Section 3 we obtained an approximation for the distribution of the maximum order complexity for pure periodically repeating sequences. In this section we will approximate the distribution for all sequences of a given length. The sequences of length n that have maximum order complexity c can be separated into three distinct categories. The first category consists of sequences of length n that have no repeated c-tuples but have at least one repeated $(c-1)$-tuple in the first $n-c+1$ positions. The second category consists of the pure periodically repeating sequences of maximum order complexity c considered in Section 3. The third category consists of sequences that have some c-tuples that appear only once followed by a collection of c-tuples that are repeated, that is sequences which are ultimately periodic. The approximate number of sequences in the first category is

$$2^n q(c, n-c+1),$$

and in the second category is

$$\sum_{k=c+1}^{\min(2^c, n-c)} k\Psi_k q(c,k),$$

where Ψ_k is the number of distinct k-bit cycles. Ψ_k are computed using the following formula given by Golomb [3]:

$$\Psi_k = \frac{1}{k}\sum_{j|k} \mu(j) 2^{k/j}$$

where $\mu(j)$ is the Möbius μ-function, which is defined in the following manner. If we express j as a unique product of the form

$$j = \prod_{m=1}^{q} p_m^{\alpha_m},$$

where p_m are prime divisors of j and each α_m is an integer, then

$$\mu(j) = \begin{cases} 1 & j = 1 \\ 0 & (\prod_{m=1}^{q} \alpha_m) > 1 \\ (-1)^q & \text{otherwise.} \end{cases}$$

The number of sequences in the third category is harder to compute. Erdmann [2] has shown that there are approximately

$$\sum_{d=0}^{c} \sum_{k=d+1}^{\min(2^d,n)} \sum_{i=\max(c-d,1)}^{\min(2^c,n-c)-k} 2^i k\Psi_k a(c,k,i) q(d,k)$$

such sequences, where for $k + i \leq 2^c$

$$a(c,k,i) = \begin{cases} b(c,k,i) - b(c-1,k,i) & k+i \leq 2^{c-1} \\ b(c,k,i) & \text{otherwise} \end{cases},$$

and for sequences of length $(k+i+c-1)$,

$$b(c,k,i) = \prod_{j=k}^{i+k-1} r(c,j) = \prod_{j=1}^{i} \left(1 - \frac{k+j}{2^{c+1}}\right).$$

We can now calculate $N(c,n)$ the approximate number of sequences of length n with complexity c as

$$\begin{aligned} N(c,n) &= 2^n q(c, n-c+1) + \textstyle\sum_{k=c+1}^{\min(2^c,n-c)} k\Psi_k q(c,k) \\ &\quad + \textstyle\sum_{d=0}^{c} \sum_{k=d+1}^{\min(2^d,n)} \sum_{i=\max(c-d,1)}^{\min(2^c,n-c)-k} 2^i k \Psi_k a(c,k,i) q(d,k) \end{aligned}$$

Note that asymptotically almost all sequences are in the first category, so for large n, $N(c,n) = 2^n q(c, n-c+1)$. Therefore

$$m(c,n) = \frac{N(c,n)}{\sum_{i=0}^{n-1} N(i,n)}.$$

is the approximate probability that a sequence of length n has complexity c. We can compare these approximate probabilities with the true values obtained from Jansen's Table 3.1 [4]. This comparison is plotted graphically in Figure 2 for sequences of length $n = 24$.

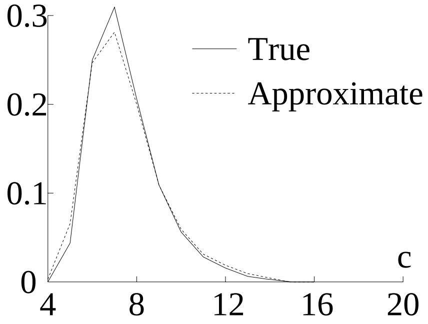

Figure 2:
True versus Approximated $(m(c, 24)$ Probability
that a random sequence of length n
has Maximum Order Complexity of c

We can now calculate the approximate expected value and variance of the maximum order complexity. If we let M_n be a random variable that denotes the maximum order complexity of a random sequence of length n, and $\hat{M}_n$ our approximation to M_n, then we have for an approximate mean

$$E(\hat{M}_n) = \sum_{c=0}^{n-1} cm(c,n),$$

and an approximate variance

$$\mathrm{Var}(\hat{M}_n) = \sum_{c=0}^{n-1} c^2 m(c,n) - \left(\sum_{c=0}^{n-1} c\, m(c,n) \right)^2 .$$

The results are shown for comparison with the true values in Table 4. We also note that $E(\hat{M}_n) \sim 2\log_2 n$ for large n, the theoretical asymptotic mean of the maximum order complexity.

5 Statistical Tests

Having calculated an approximate distribution, it is now a simple matter to construct statistical tests for randomness. For example, if we had a number of different sequences from a sequence generator, we could calculate the maximum order complexities of each sequence and so obtain an empirical distribution of maximum order complexities. This empirical distribution could then be compared with the theoretical approximate distribution by applying a goodness–of–fit technique. In this section, we show how the approximation can be used to develop a *maximum order complexity jumps test* which is directly analogous to the jumps test based on linear complexity [1] [9]. We now quote some results without proof that enable us to calculate an approximate jumps distribution.

Lemma[4]: Let $r^n_{c,d}$ denote the number of sequences of length $n+1$ with maximum order complexity d, that have maximum order complexity c when

the final bit is deleted, then:
(a) If $d \neq c$ and $d \leq n - 2^c$ then $r^n_{c,d} = 0$;
(b) If $d \geq n - c + 1$ when $n \geq 2c$ then $r^n_{c,d} = 0$.

Theorem[2]: Let $r^n_{c,d}$ denote the number of sequences of length $n+1$ with maximum order complexity d, that have maximum order complexity c when the final bit is deleted, and let N^n_c denote the number of sequences of length n with complexity c, then we have the following results:
(a) $\sum_{c=0}^{n-1} r^n_{c,d} = N^{n+1}_d$;
(b) $\sum_{d=c}^{n} r^n_{c,d} = 2N^n_c$;
(c) $\sum_{c=0}^{n-1} \sum_{d=c}^{n} r^n_{c,d} = 2^{n+1}$;
(d) $r^n_{c,d} = r^{n+1}_{c,d+1}$ for all $d > c$;
(e) $r^n_{c,d} = r^{n-d+c+1}_{c,c+1}$;
(f) $r^n_{c,c} \approx 2^{n-c-1}(n-c+1)q(c, n-c)$.

Suppose we now let J_n denote the total number of jumps that occur in all sequences of length n and J^k_n denote the number of jumps of size k that occur in all sequences of length n. Therefore J^k_n is the number of sequences that have a jump of size k when the $(m+1)^{th}$ bit is added, summed over $k \leq m \leq n-1$, and clearly $\sum_{k=1}^{n-1} J^k_n = J_n$. It can be seen that

$$J^k_n = \sum_{m=0}^{n-k-1} 2^{n-m-k-1} \sum_{c=0}^{m} r^{m+1}_{c,c+1} = \sum_{m=0}^{n-k-1} 2^{n-m-k-1} S_m,$$

where $S_m = \sum_{c=0}^{m} r^{m+1}_{c,c+1}$, so S_m is the total number of sequences of length $m+2$ that have a jump of size one when the $(m+2)^{nd}$ is added. We note that J^k_n depends only on the difference $(n-k)$, and so $J^{k+1}_{n+1} = J^k_n$ and $J^{k-1}_n = 2J^k_n + S_{n-k}$. Therefore J^k_n can be calculated recursively as

$$J^k_n = \frac{1}{2}(J^{k-1}_n - S_{n-k}).$$

J_n can now be calculated recursively as

$$J_{n+1} = 2J_n + \sum_{m=0}^{n-1} S_m.$$

We can compute an approximation $\hat{S}_m$ for S_m by using our approximation for $r^{m+1}_{c,c+1}$. The results for $0 \leq m \leq 23$ are given in Table 5.

J_n is the number of jumps in all sequences of length n, so the expected number of jumps in any particular sequence of length n is $\frac{J_n}{2^n}$. If we let $\mathcal{J}_n$ be a random variable given by the number of jumps in the maximum order complexity profile of a sequence of length n, then

$$E(\mathcal{J}_n) = \frac{J_n}{2^n} = \sum_{m=0}^{n-2} S_m(2^{-(m+1)} - 2^{-n}).$$

If $\mathcal{J}_n^k$ is the random variable of the number of jumps of size k in the maximum order complexity profile of a sequence of length n then

$$E(\mathcal{J}_n^k) = \frac{J_n^k}{2^n} = \sum_{m=0}^{n-k-1} \frac{S_m}{2^{m+k+1}}.$$

We can use our approximation to calculate approximations $\hat{\mathcal{J}}_n$ and $\hat{\mathcal{J}}_n^k$ to $\mathcal{J}_n$ and $\mathcal{J}_n^k$ respectively. Table 6 gives the expected number of jumps of size k in a sequence of length 24 as calculated by our approximation as well as the true value, and Table 7 gives the expected number of jumps in a sequence of length n $(1 \leq n \leq 24)$ as calculated by our approximation as well as the true value.

6 Conclusions

In this paper we have derived an approximate distribution for the maximum order complexity of random binary sequences, and we have used this approximation to show how to construct statistical tests to identify keystreams that can be simulated by short feedback shift registers. Two interesting areas for future research suggest themselves. Firstly, a theoretical study of how the maximum order complexity relates to other complexity measures, and secondly an extensive study of how accurate our approximations for long sequences when compared with results derived by simulation.

Acknowledgements

The authors would like to thank the referees for their helpful comments about this paper.

References

[1] G.D. Carter. *Aspects of Local Linear Complexity.* PhD thesis, Royal Holloway and Bedford New College, University of London, 1989.

[2] E.D. Erdmann. *Complexity Measures for testing Binary Keystreams.* PhD thesis, Stanford University, 1993.

[3] S.W. Golomb. *Shift Register Sequences.* Holden-Day, 1967.

[4] C.J.A. Jansen. *Investigations on Nonlinear Streamcipher Systems: Construction and Evaluation Methods.* PhD thesis, Technical University of Delft, 1989.

[5] J.L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inform. Theory*, IT-15:122–127, 1969.

[6] U.M. Maurer. Asymptotically–tight bounds on the number of cycles in generalized deBruijn–Good graphs. *Discrete Applied Mathematics*, 37/38:421–436, 1992.

[7] H. Niederreiter. The Linear Complexity Profile and the Jump Complexity of Keystream Sequences. In *Advances in Cryptology, Proceedings of EUROCRYPT 90*, pages 174–188. Springer–Verlag, 1991.

[8] R.A. Rueppel. *Analysis and Design of Stream Ciphers.* Springer-Verlag, 1986.

[9] M. Wang. *Cryptographic Aspects of Sequence Complexity Measures.* PhD thesis, Swiss Federal Institute of Technology, 1988.

[10] A.M. Zubkov and V.G. Mikhailov. Limit distributions of random variables associated with long duplications in a sequence of independent trials. *Theory of Probability and its Applications*, 19:172–179, 1974.

c	$\tilde{P}(c,16)$ Simulated	$p(c,16)$ Approximated
5	0.0741	0.0983
6	0.3259	0.3317
7	0.5970	0.5833
8	0.7811	0.7660
9	0.8877	0.8759
10	0.9419	0.9360
11	0.9707	0.9675
12	0.9859	0.9836
13	0.9922	0.9918
14	0.9964	0.9959
15	0.9981	0.9979

Table 1:
Simulated ($\tilde{P}$) and Approximated (p) Probability
that the first 16 c-tuples are unique.
(10000 Simulations.)

	$\tilde{P}(c, 32)$	$p(c, 32)$
c	Simulated	Approximated
6	0.0051	0.0109
7	0.1070	0.1163
8	0.3529	0.3493
9	0.6054	0.5944
10	0.7840	0.7721
11	0.8852	0.8790
12	0.9415	0.9376
13	0.9692	0.9683
14	0.9839	0.9840
15	0.9911	0.9920
16	0.9955	0.9960
17	0.9980	0.9980
18	0.9991	0.9990
19	0.9996	0.9995
20	0.9999	0.9998
21	0.9999	0.9999
22	0.9999	0.9999
23	0.9999	1.0000
24	1.0000	1.0000

Table 2:
Simulated ($\tilde{P}$) and Approximated (p) Probability
that the first 32 c-tuples are unique.
(10000 Simulations.)

	$E(\hat{Q}_k)$	$E(Q_k)$	$\mathrm{Var}(\hat{Q}_k)$	$\mathrm{Var}(Q_k)$
k	Approximate	True	Approximate	True
1	0.000	0.000	0.000	0.000
2	1.000	1.000	0.000	0.000
3	2.000	2.000	0.000	0.000
4	2.766	2.667	0.179	0.222
5	3.633	3.667	0.232	0.222
6	4.261	3.889	0.651	0.543
7	4.829	4.667	1.111	1.333
8	5.317	5.000	1.560	1.333
9	5.758	5.464	1.846	1.534
10	6.089	5.697	2.277	1.888
11	6.382	6.161	2.605	2.329
12	6.642	6.287	2.844	2.001
13	6.876	6.660	3.012	2.605
14	7.088	6.812	3.129	2.537
15	7.283	7.040	3.210	2.684
16	7.465	7.225	3.265	2.701
17	7.637	7.445	3.288	2.897
18	7.795	7.583	3.323	2.792
19	7.945	7.769	3.346	2.971
20	8.087	7.904	3.362	2.926
21	8.223	8.052	3.373	3.018
22	8.353	8.189	3.381	3.010
23	8.477	8.323	3.388	3.079
24	8.596	8.443	3.393	3.058

Table 3:
Mean and Variance of the
Approximated ($\hat{Q}$) and True (Q)
Maximum Order Complexity Distribution
for Pure Periodically Repeating Sequences

n	$E(\hat{M}_n)$ Approximate	$E(M_n)$ True	$\mathrm{Var}(\hat{M}_n)$ Approximate	$\mathrm{Var}(M_n)$ True
1	0.000	0.000	0.000	0.000
2	0.500	0.500	0.250	0.250
3	1.613	1.000	0.444	0.500
4	2.114	1.625	0.753	0.734
5	2.529	2.125	0.833	0.859
6	3.083	2.656	1.004	0.976
7	3.496	3.125	1.115	1.016
8	3.853	3.549	1.196	1.092
9	4.177	3.922	1.322	1.174
10	4.503	4.287	1.497	1.252
11	4.863	4.626	1.655	1.308
12	5.146	4.934	1.695	1.381
13	5.399	5.216	1.761	1.461
14	5.643	5.485	1.869	1.521
15	5.877	5.728	1.984	1.618
16	6.103	5.961	2.087	1.694
17	6.317	6.178	2.172	1.769
18	6.521	6.384	2.241	1.843
19	6.712	6.581	2.298	1.901
20	6.897	6.763	2.347	1.974
21	7.057	6.939	2.377	2.033
22	7.215	7.106	2.429	2.088
23	7.369	7.266	2.480	2.140
24	7.329	7.418	2.527	2.189

Table 4:
Mean and Variance of the
Approximated ($\hat{M}$) and True (M)
Maximum Order Complexity Distribution
for Truly Random Sequences

	$\hat{S}_m$	S_m		$\hat{S}_m$	S_m
m	Approximate	True	m	Approximate	True
0	2	2	12	938	972
1	0	0	13	1743	1768
2	3	4	14	3352	3340
3	0	0	15	6279	6156
4	8	12	16	11984	11988
5	9	8	17	22691	23006
6	12	22	18	43365	43478
7	35	36	19	82691	82846
8	76	98	20	155345	158968
9	129	138	21	303204	303290
10	260	248	22	581427	580180
11	396	480	23	1115503	1113224

Table 5:
Approximated ($\hat{S}_m$) and True Values (S_m)
of the total number of sequences of length $(m+2)$
with a Maximum Order Complexity jump of size One
when the $(m+2)^{nd}$ is added.

	$E(\hat{\mathcal{J}}_{24}^k)$	$E(\mathcal{J}_{24}^k)$		$E(\hat{\mathcal{J}}_{24}^k)$	$E(\mathcal{J}_{24}^k)$
k	Approximate	True	k	Approximate	True
1	1.729670	2.273267	13	0.000292	0.000419
2	0.847509	1.119361	14	0.000138	0.000200
3	0.414718	0.550669	15	0.000065	0.000095
4	0.202729	0.270709	16	0.000030	0.000044
5	0.098900	0.132890	17	0.000014	0.000021
6	0.048158	0.065148	18	0.000007	0.000010
7	0.023403	0.031892	19	0.000003	0.000004
8	0.011344	0.015583	20	0.000001	0.000002
9	0.005485	0.007599	21	0.000001	0.000001
10	0.002643	0.003695	22	0.000000	0.000000
11	0.001269	0.001792	23	0.000000	0.000000
12	0.000607	0.000865			

Table 6:
Approximated ($E(\hat{\mathcal{J}}_{24}^k)$) and True Values ($E(\mathcal{J}_{24}^k)$)
for the Expected Number of Jumps of size k
in Maximum Order Complexity
in a sequence of length 24

	$E(\hat{\mathcal{J}}_n)$	$E(\mathcal{J}_n)$		$E(\hat{\mathcal{J}}_n)$	$E(\mathcal{J}_n)$
n	Approximate	True	n	Approximate	True
1	0.000	0.000	13	2.372	2.749
2	0.500	0.500	14	2.486	2.872
3	0.750	0.750	15	2.596	2.988
4	1.063	1.125	16	2.702	3.097
5	1.219	1.313	17	2.803	3.198
6	1.414	1.594	18	2.899	3.294
7	1.585	1.797	19	2.991	3.387
8	1.717	1.984	20	3.078	3.474
9	1.852	2.148	21	3.161	3.557
10	1.994	2.326	22	3.239	3.637
11	2.128	2.482	23	3.315	3.713
12	2.258	2.621	24	3.387	3.785

Table 7:
Approximated ($E(\hat{\mathcal{J}}_{24})$) and True Values ($E(\mathcal{J}_{24})$)
for the Expected Number of Jumps
in Maximum Order Complexity
in a sequence of length 24