EFFECTIVE COMPUTATION OF CRYPTANALYTIC MEASURES FOR STREAM CIPHER DATA BY THE RISSANEN ALGORITHMUS

Franz Pichler

Prof. Emeritus (Systems Theory)
Johannes Kepler University Linz
A-4040 Linz, Austria
E-mail: franz.pichler@jku.at

Abstract: The paper presents an application of the algebraic theory of linear systems realization as originally established in mathematical systems theory by Rudolf Kalman to the problem of the determination of the linear complexity profile of pseudo-random sequences as they appear in the cryptanalyis of stream cipher systems. For the necessary effectiveness of the realization computation the PQ- decomposition of Hankel matrices according to the method of Rissanen is used. The proposed new method of cryptanalysis generalizes the Massey-Berlekamp algorithmus to the case of multi-variable sequences over GF(q).

Resumen: El trabajo presenta una aplicación de la teoría algebraica de la realización de sistemas lineales, tal como se estableció originalmente en la teoría matemática de sistemas por Rudolf Kalman, al problema de la determinación del perfil de complejidad lineal de secuencias seudoaleatorias como aparecen en el criptoanálisis de los sistemas cifrados en cadena. Para la necesaria efectividad de la realización se usa la computación de la descomposición PQ de matrices de Hankel, de acuerdo con el método de Rissanen. El nuevo método de criptoanálisis aquí propuesto generaliza el algoritmo de Massey-Berlekamp para el caso de secuencias multivariable sobre GF(q).

1 INTRODUCTION

An important problem in cryptanalysis is to determine for a given stream of data $S=S(0),S(1),S(2),\ldots$an associated autonomous linear finite state machine ALFSM$=(F,H)$ and an initial state $x(0)$ of it, such that ALFSM generates from $x(0)$ the data stream S. A similar problem is to determine from a given observed impulse response $A= A_0,A_1,A_2,\ldots$. the corresponding linear system $\Sigma=(F,G,H)$. In mathematical systems theory this problem is known as the “realization problem”. The mathematical basis for its solution is provided by the important work of Rudolf Kalman by his “Algebraic Theory of Linear Systems” ([Kalman 1963], Kalman-Falb-Arbib [1969]). A computational effective method for the computation of a minimal realization using the concept of Hankel matrices has been developed by Kalman and Ho ([Ho-Kalman 1966]. In practical cases of cryptanalytic investigations the observed stream cipher data have a finite length $S(M)= S(0),S(1),S(2),\ldots,S(M-1)$. In systems theory this compares to a impulse response $A(M)= A_0,A_1,A_2,\ldots,A_{M-1}$ of finite length. In this case we have the systems-theoretical problem of "partial realization". A computational effective recursive method to solve the partial realization problem, which applies even for very long data streams $A(M)$ has been originally developed by Jorma Rissanen ([Rissanen 1971]). The method works for all linear systems $\Sigma=(F,G,H)$ over a field $\boldsymbol{K}$. For the desired applications in the field of cryptology it is sufficient to assume that we deal with linear systems over a finite field $\boldsymbol{K}=GF(q)$.

In this paper we give first an introduction to the algebraic theory of realization and to the Rissanen method of partial realization. Then we show how the Rissanen method can be applied to measure the cryptanalytic quality of sequences $S(M)$ as they appear in stream cipher applications. This is possible, since the solution of the partial realization problem gives also a solution of the associated state identification problem. For scalar data streams $S(M)$ such measures are provided by the Massey-Berlekamp algorithm, which is well known in the field of cryptanalysis. Our method by applying the "Rissanen algorithm" extends this results to multi-valued data streams $S(M)$.

2 ALGEBRAIC THEORY OF LINEAR SYSTEMS REALIZATION

2.1 Linear automata fundamentals

Let a linear finite state machine LFSM over the finite field $\boldsymbol{K}=GF(q)$ be given by LFSM=(F,G,H) with the state transition given by $x(t+1) = F\,x(t) + G\,u(t)$ and associated output $y(t)$ by $y(t) = H\,x(t)$. F,G and H denote here matrices of size $n\times n$, $n\times m$ and $p\times n$, respectively. $x(t)$ denotes the state at time t, $u(t)$ and $y(t)$ are the values of the input word u and the output word y at time t, respectively. A LFSM is a linear time-invariant discrete time system with the time set T given by the set $\boldsymbol{N_0}$ of natural numbers $\boldsymbol{N_0}= 0,1,2,\ldots$. It is convenient to write the input words and the output words of a LFSM as time functions of the kind $u\colon \boldsymbol{N_0} \to \boldsymbol{K}^m$ and y: $\boldsymbol{N_0} \to \boldsymbol{K}^p$ with finite support such that there exists for each input function u and each output function y numbers $/u/$ and $/v/$,such that for all $t>/u/$ and $t>/y/$ we have $u(t)=0$ and $y(t)=0$, respectively. For a given initial state $x(0)$ and input function u the linear finite state machine LFSM computes a related state trajectory x and output function y as follows:

$$x(t) = F^{t}x(0) + \sum_{i=0}^{t-1} F^{t\text{-}i\,-1}Gu(i) \qquad \text{for } t=1,2,\ldots/u/ \qquad (2.1)$$

and

$$y(t) = H\,x(t) \quad \text{for } t=0,1,2,\ldots/\text{u}/\text{-}1 \qquad (2.2)$$

Let U and Y denote the set of all input functions of LFSM and let Q denote its state set. For our purpose it is convenient to define for a LFSM the function $M{:}Q{\times}U{\rightarrow}Y$ which assigns for the initial state $q=x(0)$ and input function u the related output function $y=M(x(0),u)$. By linearity of LFSM we have $M(x(0),u) = M(x(0),0) + M(0,u)$. From the expression (2.1) and (2.2) we see that

$$M(x(0),0)(t) = F^{t}x(0)\,,\;\; M(0,t)(0)=0 \quad \text{and} \quad M(0,u)(t) = \sum_{i=0}^{t-1} HF^{t\text{-}i\,-1}Gu(i) \;\; \text{for } t=1,2,\ldots/\text{u}/ \qquad (2.3)$$

$M(x(0),0)$ is called the *zero-input response function* of LFSM and $M(0,u)$ is the *zero- state response function*. The specific output function a_i of LFSM which is the zero- state response function $M(0,e_i)$, where e_i denotes for $i=1,2,\ldots,m$ the unit impulse input function, is called the *e_i-impulse response function*. The matrix-valued function $a=(a_1,a_2,\ldots,a_m)$ which is generated as output of LFSM if we concatenate the different I/O-experiments (e_i,a_i) is called the *impulse response function* of the LFSM.*)

2.2 FSM machine identification and FSM state identification

For our further discussion of the realization problem we repeat the following definitions from the theory of finite state machines. A single I/O pair (u,y) of a finite state machine FSM is called a *single experiment*, a set of single experiments is called a *multiple experiment*. The problem to determine for a given experiment the associated finite state machine FSM is called a *machine identification problem*. The problem to determine from a given experiment the initial state of the FSM which generates the experiment is called a *state identification experiment*.

The theory of finite state machines offers a number of approaches to solve problems of machine identification and state identification.

*) observe that we distinguish between the impulse response $A=A_0,A_1,A_2,\ldots$ as usually defined in systems theory and the impulse response function $a=a(0),a(1),a(2),\ldots$ as response of a linear system when started in the zero state by performing a multiple experiment with the unit impulses. For the expression of a in terms of A we have $a=0,A_0,A_1,A_2,\ldots$.

It is known that for the general case of finite state machines the computation of a solution in machine identification or in state identification requires algorithm which a complexity depending exponentially on the size of the state set. For specific finite state machines, however, such algorithms can have a feasible complexity. This is also the case for a linear finite state machines LFSM.

2.3 The linear system realization problem

The realization problem in linear system theory is defined by the problem to determine for a given impulse response function $a=a(0),a(1),a(2),\ldots$ with $a(t)=(a_1(t),a_2(t),\ldots,a_m(t))$ an associated minimal linear system (F,G,H). In our case the linear system is given by a LFSM with a state space $Q=\boldsymbol{K}^n$ of minimal dimension n. and each value $a(t)$ is a matrix over $K=GF(q)$ of size p×m. In LFSM theory the realization problem of linear system can be considered as a machine identification problem with a multiple experiment which is given by the set $\{(e_1,a_1),(e_2,a_2),\ldots(e_m,a_m)\}$. Let us now determine the values $a(t)$ of the impulse response function a in terms of LFSM=(F,G,H).

Since $a_i=M(0,e_i)$ we get from (2.3) $a_i(0)=0$ for $i=1,2,\ldots,m$ and therefore $a(0)=0$.
If $t>0$ we have $a_i(t) = HF^{t-1}Ge_i$ for i=1,2,…,m. For the matrix $A(t)$ for $t>0$ we get

$$a(t) = [HF^{t-1}Ge_1 \mid HF^{t-1}Ge_2 \mid \ldots \mid HF^{t-1}Ge_m]$$

If we represent G by its column vectors as $G= [g_1 \mid g_2 \mid \ldots \mid g_m]$ we have
$a(t) = [HF^{t-1}g_1 \mid HF^{t-1}g_2 \mid \ldots \mid HF^{t-1}g_m]$ or also
$a(t) = HF^{t-1}G$. We see that the impulse response function a of a LFSM=(F,G,H) can be written as

$$a = (0, HG, HFG, HF^2G, HF^3G, \ldots.) \qquad (2.4)$$

With this result the linear system realization problem can now be defined in pure algebraic terms as the problem to determine for an observed impulse response function a (or equivalently for a given impulse response A) matrices H,F,G ,where F is minimal of the size $n{\times}n$, which fulfil the equation (2.4).

2.4 LFSM state identification

As pointed out in the introduction, state identification by experiments is an important problem in cryptanalysis. In the following we will show, how the solution of the linear system realization problem leads also to the solution of the associated state identification problem. Since we are interested to solve the identification problem as part of the cryptanalyis of stream ciphers, we can assume that m=1 such that the impulse response function *a* has values in $\boldsymbol{K}^p$ .Let us assume that LFSM=*(F,G,H)* is a linear realization of the impulse response function $a=(0,a(1),a(2),\dots)$. Then the autonomous linear finite state machine ALFSM=*(F,H)* generates from state $x(0) = Ge$, where *e* is the unit impulse input function $e{:}\boldsymbol{N}_0 \rightarrow \boldsymbol{K}$ which is given by $e(0)=1$ and $e(t)=0$ for $t>0$ the output function *S* with $S(t)=a(t+1)$. This follows directly from the fact that a LFSM is time-invariant.

This result gives us the following procedure for the determination of a ALFSM and the initial state *x(0)* which generates as output a given vector-valued output function $S=(S(0),S(1),S(2),\dots..$:

2.5 define *a* by $a=(0,S(0),S(1),S(2),\dots)$ (2.5)

2.6 determine the linear realization LFSM=*(F,G,H)* of *a*

2.7 compute *x(0)* by $x(0)=Ge$

2.8 the ALFSM given by *(F,H)* generates from *x(0)* the output function *S*.

The theory of linear realization is an important part of linear system theory. In university courses, however, the presentation of its algebraic foundation, as developed mainly by the work of Rudolf Kalman, is often neglected. For the convenience of the reader we give in the following a short introduction to the algebraic theory of linear systems realization. We specialize the theory, however, to the case of LFSM realization.

2.5 Abstract approach to dynamical system realization

Let $f\colon U\rightarrow Y$ denote a function which assigns to each input function $u{:}\boldsymbol{Z}\rightarrow\boldsymbol{K}^m$ with finite support a output function $y\colon \boldsymbol{Z}\rightarrow\boldsymbol{K}^p$ such that /u/=/y/. As an example the function f can be considered as an I/O function of a finite state machine FSM with a fixed initial state. We assume that f is time-invariant in the following sense: Let $u\rightarrow t$ and $y\rightarrow t$ denote the *t*-shifted function of *u* which are given by $u\rightarrow t(\tau){:}= u(t+\tau)$ and $y\rightarrow t(\tau){:}=y(t+\tau)$. Then if $f(u) = y$ we have for all $t\in\boldsymbol{Z}$ $f(u\rightarrow t) = y\rightarrow t$.

Furthermore we assume that f is non-anticipatory, that means that the values f(u)(t) are independent from the values u(t´) for t´>t.

In the theory of finite state machines two input functions *u* and $u*$ are called equivalent to each other ($u \sim u*$) if for all $v \in U$ we have $f(uv) = f(u*v)$. Here *uv* and $u*v$ denotes the concatenation of u and $u*$ with *v* which is given by $uv(t) = u(t)$ for all *t* with $0 \leq t < /u/$ and $uv(t) = v(t-/u/)$ for all *t* with $/u/ \leq t$. The relation ~ on *U* is a equivalence relation. The quotient set $U/\sim$, the set of equivalence classes [*w*] , consisting of all input functions $w' \in U$ which are equivalent to $w \in U$, can serve as the state set $Q(f)$ of a discrete time dynamical system $\Sigma(f) =(U,Y,Q(f),\varphi(f),\beta(f))$ which is able to realize *f*. $\Sigma(f)$ can be defined as follows: Because of the time-invariance of f we can assume that a input function *w* which generates the state [*w*] has for all $t \geq 0$ the value *0*. The (global) state transition function $\varphi(f)$ of $\Sigma(f)$ can be defined by the function $\varphi(f):\boldsymbol{N}_0 \times Q(f) \times U \rightarrow Q(f)$ with $\varphi(f)(t,[w],u):=[wu|[0,t]]$ where *u* is a input function $u:\boldsymbol{N}_0 \rightarrow \boldsymbol{K}^m$ and $u|[0,t]$ denotes the restriction of *u* to the interval $[0,t]$. The output function $\beta(f):Q(f) \rightarrow \boldsymbol{K}^p$ of $\Sigma(f)$ *is defined by* $\beta(f)([w]):=f(w)(0)$. For *u* with $/u/=1$ (*u* is then basically a input letter $u(0)$ of $\Sigma(f)$, we get for $\varphi(f)$ the (local) state transition function $\delta(f):Q(f) \times \boldsymbol{K}^m \rightarrow Q(f)$ which is given by $\delta(f)([w],u(0)):=[wu(0)]$.

2.6 Linear state machine realization

In the case that the I/O function $f:U \rightarrow Y$ is linear, we are able to construct the state set $Q(f)$ of $\Sigma(f)$ as the quotient space $U/ker(f)$ of the linear space *U* of input functions modulo the kernel $ker(f)$ of the function *f*. Since *U* is a linear space, we can represent any input function *wu* by $wu = w0(u) + 0(w)u$, where $0(u)$ and $w(0)$ denotes a zero- input function of length $/0(u)/$ and $/w(0)/$, respectively. The equality $[w0(u) + w(0)u] = [w0(u)] + [w(0)u]$ allows us to represent the (local) state transition function $\delta(f)$ by $\delta(f)([w],u(0)) = [w0] + [u(0)]$. If we denote by $F(f),G(f)$ and $H(f)$ the linear operators which are given by $F(f):Q(f) \rightarrow Q(f)$ with $F(f)([w]):=[w0]$, by $G(f):\boldsymbol{K}^m \rightarrow Q(f)$ with $G(f)(a):=[u(0)]$ and by $H(f):Q(f) \rightarrow \boldsymbol{K}^p$ with $H(f)([w]:=f(w)(0)$ we have for $\Sigma(f)$ the representation as a linear state machine by $\Sigma(f) = (F(f),G(f),H(f))$

2.7 Linear finite state machine realization

The next step is to make the necessary assumptions to assure that the state space $Q(f) = U/ker(f)$ is finite dimensional. This is the case if any state *q* of $Q(f)$ can be represented by a linear combination of finite many states $q_1,q_2,\ldots,q_n$ which form a basis for $Q(f)$. Let $u_1,u_2,\ldots,u_n$ denote a set of input functions which generate this basis; $q_1 = [u_1]$, $q_2 = [u_2]$, … ,$q_n = [u_n]$. Then any state q which is given by $q = [u]$ can be represented by $q = x_1q_1 + x_2q_2 + \ldots + x_nq_n = x_1[u_1] + x_2[u_2] + \ldots + x_n[u_n] = [\, x_1u_1 + x_2u_2 + \ldots + x_nu_n]$. We see that any u with $q = [u]$ can be represented by $u = x_1u_1 + x_2u_2 + \ldots + x_nu_n$ where $u_1,u_2, \ldots ,u_n$ is a set of input functions which generate a basis for $Q(f)=U/ker(f)$.

For the determination of input functions $u_1,u_2, \ldots ,u_n$ which have this property we proceed as follows: Let $e_1,e_2, \ldots ,e_m$ denote again the unit impulse input functions and let for $k=0,1,2,\ldots$ and $i=1,2, \ldots,m$ .With e_{ik} we denote the k-shifted version of e_i which is defined by $e_{ik}(t):=e_i(t+k)$ for $k=0,1,2,..$. Since the functions e_{ik} form a basis for the input function u which generate the states $[u]$ of $Q(f)$ it is evident that each state-generating input function u can be represented by a linear combination of the functions e_{ik}. Therefor, to investigate $ker(f)$ it is of interest to investigate the values $f(e_{ik})$ restricted as functions defined on $\boldsymbol{N}_0$. For $k=0$ we have $e_{ik}=e_i$ and $f(e_i) = a_i$ or for $(e_1,e_2,\ldots e_m)$ we have $(f(e_1),f(e_2),\ldots,f(e_m)) = (a_1,a_2,\ldots,a_m) = a$ where a is the impulse response function $a =(a(0),a(1),a(2),\ldots)$. which was defined in section (2.1). For $f(e_{ik})$ we get $(f(e_{1k}),f(e_{2k}),\ldots,f(e_{mk})) =a \rightarrow k=(a(k),a(k+1),a(k+2),\ldots$). For our purpose it is sufficient to deal with a finite length impulse response function which is given by $a(M)=(a(0),a(1),a(2),\ldots \ldots,a(M-1)$.The matrix $h(N)$ over $\boldsymbol{K}$ which is defined by the block matrix

$$h(N) = \begin{bmatrix} a(0)a(1)a(2)a(3).........................a(N-1) \\ a(1)a(2)a(3)a(4).........................a(N) \\ .... \\ .... \\ .... \\ a(N-1)a(N)a(N+1).................a(M-1) \end{bmatrix}, \qquad (2.6)$$

is called a Hankel matrix of order N. The (finite) Hankel matrix $h(N)$ as given by (2.6) provides the data for the determination of the dimension n of the state space $Q(f)$ of the realization LFSM which generates as impulse response from its zero state the output sequence $a=(a(0),a(1),a(2),\ldots \ldots,a(M-1))$.If $h(N)$ has a finite rank then $n = rank\ h(N)$ is valid. This can be seen as follows: If we restrict the dimension of the input space U to $dim\ U=M$ then any input function u has length M and can be represented by a linear combination of the set of M unit impulse input functions e_{ik}, with $i=1,2,\ldots,m$ and $k=0,1,2,\ldots,N-1$ where $N= (M+1)/2$. Since for e_{ik} the values $f(e_{ik})$ are given by the rows of $h(N)$ the dimension K of the kernel space $ker(f)$ is given by M minus the rank of $h(N);$ $K= M - rank\ h(N)$. For the state space $Q(f) =U/ker(f)$ we have $dim\ Q(f) = dim\ U – dim\ ker(f)$ or $dim\ Q(f) = M – K$ or for rank $h(N)=n$ we get $dim\ Q(f)= M-(M-n) = n$.

This result offers also the computational means for the construction of a basis $\Xi=\{\xi_1,\xi_2,\xi_3,\dots,\xi_n\}$ of the state space $Q(f)=\boldsymbol{K}^n$ of $\Sigma(f)$ as follows: As basis vectors ξ_i we choose the states $\xi_i=[\varepsilon_i]$ where ε_i are chosen from the set of unit impulse functions e_{ik} such that the set of rows $f(e_{ik})$ of $h(N)$ consists of linear independent elements. In general there are several ways to choose the basis Ξ.

With the results of section 2.6 and the construction of a basis for the state space by means of the function f and the Hankel matrix $h(N)$ we are able to determine the linear operators $F(f),G(f)$ and $H(f)$ of $\Sigma(f)$ in matrix form. From section (2.6) we have the results

$$F(f)([w])=[w0]\ ,\ G(f)(u(0))=[u(0)]\ ,\ H(f)([w])=f(w)(0) \tag{2.7}$$

where $w\in U$, $u(0)\in\boldsymbol{K}^m$ and $e=(e_1,e_2,\dots,e_m)$. Since we interpret $[w]$ here as the state of $\Sigma(f)$ at time 0 the input value $u(0)$ appears also at time 0 such that $0(w)$ has to be considered as the empty word Λ of length 0, so that $G(f)(u(0))=[u(0)]$. As usual we use the notation Σ, F,G,H for $\Sigma(f),F(f),G(f)$ and $H(f)$, respectively.

(1) Determination of F:

For the $n\times n$ matrix $E_n=[[\varepsilon_1]\,|\,[\varepsilon_2]\,|\dots|\,[\varepsilon_n]]$ constructed by the basis vectors of the state space of Σ we have $F[[\varepsilon_1]\,|\,[\varepsilon_2]\,|\dots|\,[\varepsilon_n]]=[[\varepsilon_1 0]\,|\,[\varepsilon_2 0]\,|\dots|\,[\varepsilon_n 0]]$. With basis Ξ the matrix E_n is the unit matrix and we have the result

$$F=[[\varepsilon_1 0]\,|\,[\varepsilon_2 0]\,|\dots|\,[\varepsilon_n 0]] \tag{2.8}$$

(2) Determination of G:

We take as input values to be processed by G the values $e_1(0),e_2(0),\dots,e_m(0)$ and form the $m\times m$ matrix $E_m=[\,e_1(0)\,|\,e_2(0)\,|\dots|\,e_m(0)]$. By (2.7) we have $G[\,e_1(0)\,|\,e_2(0)\,|\dots|\,e_m(0)]=[\,[e_1]\,|\,[e_2]\,|\dots|\,[e_m]]$. Since E_m is a unit matrix we have as result

$$G=[\,[e_1]\,|\,[e_2]\,|\dots|\,[e_m]]. \tag{2.9}$$

(3) Determination of *H:*

For *H* we have from (2.7) $H([w]=f(w)(0)$. For the states $[\varepsilon_1],[\varepsilon_2], \ldots,[\varepsilon_n]$ which form the basis of the state space $Q=\boldsymbol{K}^m$ have $H[[\varepsilon_1]|[\varepsilon_2]| \ldots|[\varepsilon_n]] =[f(\varepsilon_1)(0)|f(\varepsilon_2)(0)| \ldots |f(\varepsilon_n)(0)]$. Since $[[\varepsilon_1]|[\varepsilon_2]| \ldots|[\varepsilon_n]]$ is the unit matrix we have the result

$$H=[f(\varepsilon_1)(0)|f(\varepsilon_2)(0)|\ldots|f(\varepsilon_n)(0)]. \qquad (2.10)$$

With the result as shown in (2.8)-(2.10) we are able to determine for a given impulse response function $a=(0,a(1),a(2),\ldots a(M-1)$ (or equivalently for a given impulse response $A=(A_0,A_1,A_2, \ldots,A_{M-1})$) of length *M* of a multi-variable "blackbox" over a finite field ***K*** with *m* "input ports" and *p* "output ports" by means of the Hankel-(block)matrix *h(N)* of size *N* a linear finite state machine LFSM given by $\Sigma=(F,G,H)$ which generates *A* as its impulse response. The realization Σ is minimal, the dimension n of the state space is minimal and its states are controllable and observable. *)

The approach of this chapter follows mainly the book on "System Theory" of Louis Padulo and Michael A. Arbib [Padulo-Arbib 1974, chapter 8-2]. Related work, with emphasis on general systems aspects, has been published by the author [Pichler 1974],[Pichler 1976]. The fundamental theory for this approach of solving the realization problem is due to the research results in mathematical systems theory by Rudolf E. Kalman [Kalman1963],[Kalman-Falb-Arbib 1969]. Computational aspects have been considered by Ho and Kalman [Ho-Kalman 1966] and J. Rissanen [Rissanen 1971],[Rissanen-Kailath 1972]. In the next chapter we discuss the contributions of J. Rissanen for the effective computation of realizations.

*) the usual approach in systems theory is to make here use of the Hankel matrix *H(N)* of order *N* which is constructed by the values $A_0,A_1,A_2,\ldots,A_{M-1}$ of the impulse response *A*. This is possible since by the time invariance of LFSM we have always *rank h(N) = rank H(N).* In the following we will also make use of *H(N).*

3. RISSANEN PQ-DECOMPOSITION OF HANKEL MATRICES

Jorma Rissanen introduced in his important papers [Rissanen 1971] and [Rissanen –Kailath 1972] an effective method to decompose recursively step by step a sequence of Hankel matrices $H(1),H(2),H(3),\ldots$which are generated by a associated sequence of impulse responses $A(1),A(2),A(3,\ldots$ of increasing length $M(1),M(2),M(3),\ldots$ If at a step k the Hankel matrix $H(K)$ is reached and the associated realization $\Sigma_k =(F_k,G_k,H_k)$ is determined,the method of Rissanen allows the computation of $H(k+1)$ and $\Sigma_{k+1} =(F_{k+1}),G_{k+1}),H_{k+1}))$ at step $k+1$ by keeping the results of step k. At each step the Rissanen method provides a decomposition of the form $H(n,N)=P(n)Q(n,m)$ where $H(n,N)$ consists of n linear independent scalar rows of the Hankel matrix $H(N)$ and $P(n)$ is a lower triangular scalar matrix of dimension $n\times n$. If rank $H(N)=rank\ H(N+1)$ the columns of $P(n)$ can be taken as basis vectors for the state space $Q=\boldsymbol{K}^n$ of the associated realization. The matrices F,G, and H of the realization can be effectively derived from $P(n)$ and $Q(n,m)$. The effectiveness of the method of Rissanen for the computation of realizations is crucial for our further discussion, it is outside of the scope of this paper to discuss it in greater detail . The reader is at advised to consult the existing original publications. For the future a complete algorithmic description and a software implementation of the Rissanen algorithmus for applications in the field of cryptanalysis is in preparation.

4 CRYPTANALYTIC MEASURES FOR STREAM CIPHER DATA

Stream ciphers are based on pseudo random generators PRG which generate from a initial state $x(0)$ (which contains information on the cryptographic key in use) a key sequence $S=S(0),S(1),S(2),\ldots$.. In many practical cases of PRG´s the value set of the key sequence S is given by the finite field $GF(2)$. However today in modern PRG´s for fast secure data transmission also $(GF(2))^m$ and more general $K^m=(GF(q))^m$ is of interest. In practical applications only a finite part $S(M)= S(0),S(1),S(2),\ldots,S(M-1)$ of length M is available for cryptanalysis. The finite string $S(M)$ can have been derived from a successful “known plain text attack” or by a “chosen plain text attack” by inversion of the mixing operation of the stream cipher system. Also the direct observation of the key stream of the PRG is a possibility. For a derived sequence S there is the cryptanalytic problem to determine a possible structure of the PRG and (or) the determination of the used key. By technological reasons it can be assumed that the PRG of the stream cipher system can be modelled by a autonomous finite state machine AFSM. By assuming a finite state machine model the problem of finding the structure of the PRG and (or) the key in use is seen as the problem of machine identification and (or) state identification as discussed in section (2.2).

The solution of both problems together with the problem of the determination of the cardinality of the state set Q of the ALFSM are seen in cryptanalysis as complexity measures for the key sequence S. The measure which computes for a given sequence S the cardinality of the state set Q is known as the Chaitin-Kolmogoroff complexity of S. For the general case of autonomous finite state machines the implementations of such measures are computational difficult and practical infeasible. However, for specific classes of AFSM´s effective methods for the implementations of the measures might be derived. One such class is given by the class of autonomous linear finite state machines ALFSM. For the case of scalar-valued sequences $S=s(0),s(1),s(2),s(3),\ldots$ with $s(i)\in GF(2)$ or $s(i)\in GF(q)$ the Massey- Berlekamp algorithmus provides an effective method to derive from a finite string $S(M)$ of S a autonomous linear feedback shift register ALFSM of minimal length $n(M)$ and an associated initial state $x(0)$ which generates $S(M)$ form state $x(0)$. The function $L:\boldsymbol{N}_0\rightarrow\boldsymbol{N}_0$ which computes for a given scalar-valued sequence S for any string $S(M)$ of it the minimal length $n(M)$ of the realizing ALFSM has been called in cryptanalysis the linear complexity profile of S. We see that the Massey-Berlekamp algorithmus solves for the analysis of scalar stream cipher data the machine identification problem, the state identification problem and also the linear complexity problem.

Our goal is, to solve such problems also for the case of vector-valued stream cipher data S. The principal approach which we take has been already outlined in chapter 1 of this paper. The technical details for it are provided by the content of chapter 2 and chapter 3 of this paper. Therefor we are in the position to describe the method, which we will call the “Rissanen algorithmus” by its main steps

Rissanen algorithmus for the computation of cryptanalytic linear measures

Assume that a key sequence $S=S(0),S(1),S(2),\ldots$ has been observed. Then we are able to perform for each finite string $S(M)$ of length M the following two steps:

Step 1: we consider the string $S(M)=S(0),S(1),S(2),\ldots,S(M-1)$ as the impulse response $A(M)$ of a LFSM. The associated impulse response function a as response of a LFSM from the zero state is then given by $a=0,S(0),S(1),S(2),\ldots,S(M-1)$. Application of the realization method of chapter 2 gives by means of the Hankel matrix $H(N)$ of rank n the associated linear finite state machine Σ by $\Sigma=(F,G,H)$ with state space $Q=\boldsymbol{K}^n$. The computation of Σ requires that $H(N+1)$ has also rank n.

Step 2: we consider now the given string $S(M)= S(0),S(1),S(2),\ldots,S(M\text{-}1)$ of the key sequence S as the output word of the autonomous linear finite state machine ALFSM=(F,H) with initial state $x(0)$ which is given by $x(0)=Ge$ where e is the (scalar) unit impulse input function given by $e(0)=1$ and $e(t)=0$ for $t>0$.

Increasing recursively the length M of the key sequence S for $M=1,2,3,4,\ldots$we reach by *step 1* the associated Hankel matrices $H(1),H(2),H(3),\ldots$ and the associated LFSM´s $\Sigma(1), \Sigma(2), \Sigma(3),\ldots$and the associated sequence $n(1), n(2), n(3),\ldots$indicating the dimension of the associated state spaces $Q(1), Q(2), Q(3),\ldots$.. By increasing the length M of the key string $S(M)$ and having the results of *step 1* we determine by *step 2* the associated autonomous linear finite state machines $(F(1),G(1)), (F(2),G(2)), (F(3),G(3)), \ldots$ and the associated initial states $x(1)(0), x(2)(0), x(3)(0),\ldots$. To have an effective method for the stepwise computation of the realizations $\Sigma(1), \Sigma(2), \Sigma(3),\ldots$the Rissanen method for establishing a PQ-decomposition of Hankel matrices of chapter 3 has to be applied.

The Rissanen algorithmus, which is described above, allows for even very large observed key streams the effective computation of linear cryptanalytic measures, such as given by linear state machine identification, initial state identification and determination of the linear complexity. Since the key streams are allowed to have values in the linear space $\boldsymbol{K}^m$ with $m\geq 1$ the Rissanen algorithmus generalizes the cryptanalytic methods provided by the Massey- Berlekamp algorithmus.

5 CONCLUDING REMARKS

The paper presents with the Rissanen algorithmus a new method for taking linear measures of vector-valued data as they appear in stream ciphering. The theoretical basis for this method is provided by the theory of linear system realization as developed as part of mathematical systems theory by Rudolf Kalman. For the practical application in the context of cryptanalyis the Rissanen method for recursive identification of linear systems has been proven as essential.

The result of this paper generalizes the method of linearization as developed by Massey and Berlekamp which was originally developed in the theory of BCH-codes [Massey 1967],[Berlekamp 1968]. Jonckheere and Ma investigated the Massey-Berlekamp algorithmus from the point of view of linear systems realization but did not deal with the possible more general case of our paper [Jonckheere-Ma 1989].Our presentation made use of the existing classical theory of linear finite state machine . A deeper structural knowledge about realizations could be achieved by taking the R-module approach in the theory of algebraic linear systems as developed by Kalman . A further goal of research could be the investigation of a possible direct effective computation of realizations such that the structure of $\Sigma=(F,G,H)$ becomes a parallel coupling of linear feedback shift registers. The matrix F is then of rational canonical form. We hope that our paper can show that the results of mathematical systems theory, as developed mainly in the 60´s of the last century, have until today some importance for practical applications in modern topics of engineering.

REFERENCES

[Jonckheere-Ma 1989] Jonckheere,E. and Chingwo Ma: A Simple Hankel Interpretation of the Massey- Berlekamp Algorithm. Linear Algebra and its Applications, 125, Elsevier Science Publ. Co. 1989, 65-76.

[Massey 1967] Massey J. : Shift register synthesis and BCH decoding. IEEE Trans. on Information Theory IT-15, 1967, 122-127.

[Berlekamp 1968] Berlekamp E.R. : Algebraic Coding Theory, McGraw Hill, New York 1968.

[Kalman 1963] Kalman, R.E.: Mathematical description of linear dynamical systems. SIAM Journal on Control, (1963), 152-192

[Kalman 1969] Kalman, R.E. , P.L. Falb, M.A. Arbib : Topics in Mathematical Systems Theory , chapter 10 Algebraic theory of linear systems, McGraw Hill, New York 1969.

[Kalman-Ho 1966] Ho, B.L. and R.E. Kalman: Effective construction of linear state-variable models from input/output functions. Regelungstechnik, Oldenbourg 1966, 545-548.

[Padulo-Arbib 1974] Padulo,L. and M. Arbib: System Theory: A Unified Approach to Continuous and Discrete Systems. Hemisphere Publishing Corporation, Washington D.C. 1974.

[Pichler 1974] Pichler,F. : Realisierung linearer Input-Output Prozesse I: Diskrete Prozesse.
Technischer Bericht SYS-PED 1, Lehrkanzel für Systemtheorie,
Universität Linz,Dezember 1974, (35 pages).

[Pichler 1976] Pichler,F. : General Dynamical Systems: Construction and Realization.
In “Mathematical Systems Theory-Udine 1975”
Lecture Notes in Economics and Mathematical Systems,
Springer-Verlag Berlin 1976, 393- 408.

[Rissanen 1971] Rissanen J.: Recursive Identification of Linear Systems
SIAM Journal on Control, Vol 9,No 3 August 1971, 420-430.

[Rissanen –Kailath 1972] Rissanen,J, and T. Kailath : Partial Realization of Random Systems
Automatica, Vol. 8, Pergamon Press 1972, 389-396.

.