Draft Article (For Engineering Journals)

A Finite Field Framework for Modeling, Analysis and Control of Finite State Automata

JOHANN REGER[†,1] AND KLAUS SCHMIDT[‡]

SUMMARY

In this paper we address the modeling, analysis and control of finite state automata, which represent a standard class of discrete event systems. As opposed to graph theoretical methods we consider an algebraic framework that resides on the finite field $\mathbb{F}_2$, which is defined on a set of two elements with the operations addition and multiplication, both carried out modulo 2. The key characteristic of the model is its functional completeness in the sense that it is capable of describing most of the finite state automata in use, including non-deterministic and partially defined automata. Starting from a graphical representation of an automaton and applying techniques from boolean algebra we derive the transition relation of our finite field model. For cases, in which the transition relation is linear, we develop means for treating the main issues in the analysis of the cyclic behavior of automata. This involves the computation of the elementary divisor polynomials of the system dynamics, and the periods of these polynomials, which are shown to completely determine the cyclic structure of the state space of the underlying linear system. Dealing with non-autonomous linear systems with inputs, we use the notion of feedback in order to specify a desired cyclic behavior of the automaton in the closed loop. The computation of an appropriate state feedback is achieved by introducing an image domain and adopting the well-established polynomial matrix method to linear discrete systems over the finite field $\mathbb{F}_2$. Examples illustrate the main steps of our method.

Keywords: Finite State Automata, Linear Modular Systems, Finite Fields.

1. INTRODUCTION

In the world of continuous dynamic systems, state space models are the dominant paradigm of representing a system algebraically. It is due to this algebraic setting that most of the real world system properties can be retrieved in the algebraic model, since the algebra entails a certain structural framework. In contrast to continuous systems, discrete event systems characteristically show a lack of structure, that is, these systems

[1] Address correspondance to: Johann Reger, Lehrstuhl für Regelungstechnik, Friedrich-Alexander-Universität Erlangen-Nürnberg, Cauerstraße 7, 91058 Erlangen, Germany.

[†] Erlangen-Nuremberg University, Institute of Automatic Control, Department of Electrical, Electronic and Communication Engineering, Erlangen, Germany.

[‡] Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, USA.

do not fit naturally into an algebraic frame since many of the real world properties can hardly be captured in some nice algebraic structures, as do eigenvalues, for example, when dealing with stability issues in linear continuous systems theory. However, a lot of effort has been made to setup a link between both classes of systems.

In search of analogies to linear continuous systems, state space models using so-called *arithmetical polynomials* have been introduced for representing finite state automata [4, 3]. A second method employs *Walsh functions* to model deterministic finite state automata as autonomous linear systems [10]. As pointed out in [13], the crucial drawbacks, which are common in both approaches, are the absence of sufficient and efficient criteria for an algebraic locating of the inherent cyclic properties of automata. Even for linear systems the former approach only provides necessary criteria, whereas the latter runs into numerical difficulties by enumerating the whole state space. Another severe problem is the computational complexity, as within these approaches solving for certain cyclic states is of non-polynomial complexity (NP-complete). This is due to the fact that the associated algebraic operations do not constitute a group because they are not closed under the operation sets. Unfortunately, there is no polynomial time algorithm that solves a linear system of equations over the rational numbers for boolean vectors. As a consequence, NP-completeness implies that any problem in practice is rendered intractable.

In contrast to the approaches from above, the model to be developed in this paper is capable of overcoming these obstacles. To this end, we consider an algebraic state space description that is formulated strictly in (modulo 2-) operations on the set of boolean numbers, mathematically speaking, we operate on a finite field $\mathbb{F}_2$.[1] Thus, contrary to the afore-mentioned approaches, we can benefit from the field property; at least in the linear case, it is possible to solve for cyclic states in polynomial time. Despite some peculiarities of finite fields, it will turn out that if one is concerned with state space modeling of automata, finite fields provide those algebraic concepts that are necessary for relating automata properties to structural, algebraic properties. This leads to a sufficient and efficient analysis of the cyclic behavior of deterministic and non-deterministic finite state automata, especially in the linear case. In this case, the key concepts are given by the notions of invariant polynomials and the period of a polynomial, which will be shown to grant the statement of sufficient criteria for determining all cycles of a deterministic linear automaton, in multiplicity and length.

Based on this knowledge the synthesis of linear state feedback for imposing properties on the controlled linear system with respect to cyclic states is carried out. We demonstrate that this amounts to setting the (invariant) elementary divisor polyno-

[1] Finite field models have already been under consideration in the control community [5]. Still, neither were finite fields utilized for determining the cyclic structure of automata nor were any analogies drawn to continuous time systems. On the other hand much of the theory was already developed as early as the sixties — for instance the design of linear feedback shift registers [7, 6] — but has not been adapted for control purposes yet.

mials of the closed loop dynamics. In the scope of continuous systems, resorting to standard methods as for example employing the parametric approach [15], this is a difficult task to perform — we comment on that in detail. Instead, we propose an image domain method for linear discrete systems [14], an adaption of the polynomial matrix method [11], in which our synthesis goal of setting the invariant polynomials apparently proofs to be more practicable. This results in an algorithm for generating a linear state feedback which fits a given linear automaton with specified cyclic properties. Additionally, the algorithm meets the requirements burdened by structural constraints, as stated in Rosenbrock's structure theorem [11].

The outline of the paper is as follows: Section 2 introduces the minimum necessary algebraic terminology. Section 3 exposes, in an exemplary fashion, two methods for obtaining a multilinear automaton model over the finite field $\mathbb{F}_2$ by referring to elementary boolean algebra. In Section 4 we are concerned with the analysis of linear discrete models over $\mathbb{F}_2$. Taking advantage of the notion of feedback we show how to impose structural properties on closed loop systems in Section 6. Finally, in Section 7 we recall the main ideas and give some hints in view of extending the setting.

2. ALGEBRAIC PRELIMINARIES

Linear algebra over the fields of real and complex numbers is a widely spread tool all over the engineering sciences. On the contrary, except from signal processing and coding theory, discrete mathematics and finite fields are encountered quite rarely in the academic education of engineers. On this account, the most important conceptual terms from algebra which represent the bases for our automaton model in view (Section 3) are recapitulated in the sequel. Some remarks spot the differences between finite and infinite fields. We refer to the comprehensive and thorough introduction to finite fields by Lidl and Niederreiter [12].

2.1. Finite Fields

Definition 1 (Group)
A group is a set $\mathcal{G}$ together with a binary operation $*$ such that

1. For all $a, b \in \mathcal{G}$, $a * b \in \mathcal{G}$.
2. The operation $*$ is associative, i. e. $a * (b * c) = (a * b) * c$ for any $a, b, c \in \mathcal{G}$.
3. An identity element, $e \in \mathcal{G}$, exists such that for all $a \in \mathcal{G}$, $a * e = e * a = a$.
4. For any $a \in \mathcal{G}$ an inverse element $a^{-1} \in \mathcal{G}$ exists such that $a * a^{-1} = a^{-1} * a = e$.

Moreover, a group is commutative (or abelian) if for all $a, b \in \mathcal{G}$, $a * b = b * a$. A group is called finite if the set $\mathcal{G}$ contains finitely many elements.

As we will make abundant use of polynomials in the subsequent sections we need to define the notion of a ring.

Definition 2 (Ring)
A ring is a set $\mathcal{R}$ together with two binary operations, addition + and multiplication $\cdot$, such that

1. $\mathcal{R}$ is a commutative group with respect to addition.
2. Multiplication is associative , i. e. $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for any $a, b, c \in \mathcal{R}$.
3. $\mathcal{R}$ is distributive with respect to addition and multiplication, that is $a \cdot (b + c) = a \cdot b + a \cdot c$ and $(b + c) \cdot a = b \cdot a + c \cdot a$ for all $a, b, c \in \mathcal{R}$.

A ring is called commutative if its multiplication is commutative.

The set $\mathcal{R}$ of polynomials in the independent variable λ with the usual addition and multiplication of polynomials forms a ring, the ring of polynomials denoted by $\mathcal{R}[\lambda]$.

It is essential for a ring that a multiplicative inverse need not exist in general. In order to be able to solve for multiplicatively bound indeterminates it is helpful to increase the requirements by excluding the critical element from the set.

Definition 3 (Field)
A ring on a set $\mathbb{F}$ with the operations addition an multiplication, + and $\cdot$, is a field if the subset $\mathbb{F} \setminus \{0\}$ is a commutative group with respect to multiplication. A field $\mathbb{F}$ with q elements, denoted by $\mathbb{F}_q$, is called finite if q is finite.

In further sections of the paper a special type of field is utilized that is based on the division remainder operation *modulo*.

Definition 4 (Galois-Field)
The set of integers $\{0, 1, \ldots, q-1\}$, where q is a prime number, with the operations addition and multiplication modulo q, is a finite field, called Galois-Field $\mathbb{F}_q$.

The primality of q is decisive for the existence of a multiplicative inverse element. Otherwise zero divisors would occur (for instance $2 \cdot 3$ modulo $6 = 0$, hence $\mathbb{F}_6$ is not a field). In the sequel, $\mathbb{F}_q$ will always denote a Galois Field and beginning with Section 3 we will concentrate on Galois-Fields fields $\mathbb{F}_2$ only. Consequently, for $a, b \in \mathbb{F}_2$

$$\begin{aligned} a + b &:= a + b \bmod 2\,, \\ a \cdot b &:= a \cdot b \bmod 2\,. \end{aligned}$$

Note that in this case subtraction modulo 2 coincides with addition modulo 2.

Theorem 1 (Fermat’s Little Theorem)
Let $q \in \mathbb{Z}$ be a prime number. Then for all integers λ, which are not divisible by q, q divides $\lambda^{q-1} - 1$.

Consequently, every $\lambda \in \mathbb{F}_q$ satisfies $\lambda^q = \lambda$. Hence, a polynomial $p \in \mathbb{F}_q[\lambda]$ can be identical to zero for arbitrary $\lambda \in \mathbb{F}_q$, since p may contain polynomials $\lambda^q - \lambda$, which are identical to zero. In contrast to finite fields, a polynomial $p \in \mathbb{R}[\lambda]$ over the infinite field of real numbers $\mathbb{R}$ is identical to zero if and only if all coefficients are zero.

2.2. Polynomials over Finite Fields

According to Gauß' fundamental theorem of algebra, a fundamental property of polynomials is that all polynomials over the field of real numbers $\mathbb{R}$ can be factorized (reduced) in quadratic factors in $\mathbb{R}[\lambda]$, or over the extension field $\mathbb{C}$ even more in linear factors in $\mathbb{C}[\lambda]$. Naturally one would expect that this holds true for finite fields as well. We will see that for finite fields $\mathbb{F}_q$, in general, this is not the case.

Factorization of Polynomials

Definition 5 (Monic Polynomial)
A polynomial $p(\lambda) = \sum_{i=0}^{d} a_i \lambda^i$ with degree d is called monic if $a_d = 1$.

Definition 6 (Irreducible Polynomial)
A non-constant polynomial $p \in \mathbb{F}[\lambda]$ is called irreducible over $\mathbb{F}$ if whenever $p(\lambda) = g(\lambda)h(\lambda)$ in $\mathbb{F}[\lambda]$ then either $g(\lambda)$ or $h(\lambda)$ is a constant.

In view of irreducibility we can rephrase Gauß' fundamental theorem of algebra.

Theorem 2 (Unique Factorization Theorem)
Any polynomial $p \in \mathbb{F}[\lambda]$ can be written in the form

$$p = a\, {p_1}^{e_1} \cdots {p_k}^{e_k}\,, \tag{1}$$

where $a \in \mathbb{F}$, $p_1, \ldots, p_k \in \mathbb{F}[\lambda]$ are distinct polynomials that are irreducible over $\mathbb{F}$, and $e_1, \ldots, e_k \in \mathbb{N}$. This factorization is unique apart from the sequence of the factors.

For the field $\mathbb{R}$ all polynomials p_i in Theorem 2 are of degree $e_i \leq 2$. This does not apply for finite fields, for example: $p(\lambda) = \lambda^5 + \lambda^2 + \lambda + 1 = (\lambda^3 + \lambda + 1)(\lambda + 1)^2$, $p \in \mathbb{F}_2[\lambda]$, because $\lambda^3 + \lambda + 1$ and $\lambda + 1$ are irreducible over $\mathbb{F}_2$. Another property which is peculiar to finite fields is the periodicity property.

Period of Polynomials

Definition 7 (Period of a Polynomial)
Let $p \in \mathbb{F}_q[\lambda]$ be a non-zero polynomial. If $p(0) \neq 0$, then the least positive integer τ for which $p(\lambda)$ divides $\lambda^\tau - 1$ is called the period (order) of the polynomial p. If $p(0) = 0$, then $p(\lambda) = \lambda^h g(\lambda)$, where $h \in \mathbb{N}$ and $g \in \mathbb{F}_q[\lambda]$ with $g(0) \neq 0$, and τ is defined as the period of g.

For polynomials which are powers of irreducible polynomials, so-called powered polynomials, we have the following theorem.

Theorem 3 (Period of a Powered Polynomial)
Let $p \in \mathbb{F}_q[\lambda]$ be irreducible over $\mathbb{F}_q$ with $p(0) \neq 0$ and period τ. Let $f = p^e \in \mathbb{F}_q[\lambda]$ with $e \in \mathbb{N}$. Let l be the least $l \in \mathbb{Z}$ such that $q^l \geq e$. Then the period of f is $q^l \tau$.

Example 1
We calculate the period of $f(\lambda) = \lambda^4 + \lambda^2 + 1 \in \mathbb{F}_2[\lambda]$. From

$$\lambda^2 f(\lambda) + f(\lambda) = (\lambda^2+1) f(\lambda) = \lambda^6 + 1 \quad \Rightarrow \quad f(\lambda) | \lambda^6 + 1$$

it follows that $\tau_f = 6$. If we use the factorization $f = p^2$ with $p = \lambda^2 + \lambda + 1$ then

$$\lambda p(\lambda) + p(\lambda) = (\lambda+1) p(\lambda) = \lambda^3 + 1 \quad \Rightarrow \quad p(\lambda) | \lambda^3 + 1$$

implies that $\tau_p = 3$. Thus, observing $e = 2$ and considering Theorem 3, we obtain $l = 1$ and therefore we get $\tau_f = 2^1 \cdot 3 = 6$ with $\tau_p = 3$.

Remark 1
Nilpotent polynomials $p \in \mathbb{F}_q[\lambda]$ with $p = \lambda^k$ for some $k \in \mathbb{N}$ are not periodic by definition. Hence, polynomials over finite fields are either periodic or nilpotent.

Remark 2
In practice, periods of polynomials can be found in tables like in [12], or are internally tabulated in computer algebra software like Maple® or Mathematica®.

2.3. Similarity and Invariants of Linear Systems over the finite field $\mathbb{F}_q$

Many major properties of a matrix are invariant by its structure and are preserved under elementary row and column operations, so-called similarity transformations.[2]

Similarity of Matrices

Definition 8 (Similarity of a Matrix)
Matrices $\mathbf{A}_1$, $\mathbf{A}_2 \in \mathbb{F}^{n \times n}$ are similar if for some invertible constant matrix $\mathbf{T} \in \mathbb{F}^{n \times n}$

$$\mathbf{A}_1 = \mathbf{T}^{-1} \mathbf{A}_2 \mathbf{T}. \tag{2}$$

When properties which are invariant under similarity transforms are concerned, polynomial matrices, matrices the elements of which are polynomials, are a tool of practical relevance.

[2] For brevity, we refrain from defining vector spaces and linear transformations since the well-known conventional definitions can be extended right away to the finite field case.

Theorem 4 (Smith Form of the Characteristic Matrix)
For any $\mathbf{A} \in \mathbb{F}^{n\times n}$, polynomial matrices $\mathbf{U}(\lambda), \mathbf{V}(\lambda)$ with non-zero determinant independent from λ exist such that

$$\mathbf{U}(\lambda)(\lambda\mathbf{I}-\mathbf{A})\mathbf{V}(\lambda) = \mathbf{S}(\lambda), \quad \mathbf{S}(\lambda) = \operatorname{diag}\big(c_1(\lambda),\ldots,c_n(\lambda)\big) \tag{3}$$

in which the monic polynomials $c_{i+1}|c_i,\ i = 1,\ldots,n-1$. The polynomial matrix $\mathbf{S}(\lambda)$ is called the Smith (normal) form of (the characteristic matrix wrt.) $\mathbf{A}$.

Matrices are similar iff they have the same Smith form. As the polynomials c_i, $i = 1,\ldots,n$ are preserved under similarity transforms this gives rise to define invariants.

Invariant Polynomials

Definition 9 (Invariant polynomials)
The unique non-constant monic polynomials $c_i(\lambda)$, $i = 1,\ldots,n$, referring to the Smith form $\mathbf{S}(\lambda)$ of a matrix $\mathbf{A}$, are the invariant polynomials (similarity invariants) of $\mathbf{A}$.

The uppermost polynomial $c_1(\lambda)$ is the minimal polynomial of the matrix $\mathbf{A}$. The product of all invariant polynomials is its characteristic polynomial $\det(\lambda\mathbf{I}-\mathbf{A})$.

Definition 10 (Elementary Divisor Polynomials)
Let $c_i \in \mathbb{F}[\lambda]$, $i = 1,\ldots,n$, be the invariant polynomials of a matrix $\mathbf{A}$ and let $c_i = p_{i,1}^{e_{i,1}} \cdots p_{i,N_i}^{e_{i,N_i}}$ with $N_i \in \mathbb{N}$ be the unique factorization of c_i due to Theorem 2. Then, the $N = \sum_{i=1}^{n} N_i$ non-constant factor polynomials $p_{i,j}^{e_{i,j}}$, $i = 1,\ldots,n$ and $j = 1,\ldots,N_i$, are termed elementary divisor polynomials of $\mathbf{A}$.

The Rational Canonical Form

In addition to the Smith form (3), we will use a normal form referring to the elementary divisor polynomials. This will involve the notion of a companion matrix.

Definition 11 (Companion Matrix)
Let $p(\lambda) = \sum_{i=0}^{d} a_i\lambda^i \in \mathbb{F}[\lambda]$ be a monic polynomial of degree d. The matrix $\mathbf{C} \in \mathbb{F}^{d\times d}$

$$\mathbf{C} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{d-2} & -a_{d-1} \end{pmatrix} \tag{4}$$

is called the companion matrix with respect to the polynomial $p(\lambda)$.

Companion matrices have some useful properties, e. g. its characteristic polynomial coincides with its minimal polynomial, which is just the defining polynomial $p(\lambda)$.

Theorem 5 (Rational Canonical Form)
For any $\mathbf{A} \in \mathbb{F}^{n\times n}$ there exists a similarity transform $\mathbf{T}$ by virtue of which

$$\mathbf{A}_{\text{rat}} = \mathbf{T}\mathbf{A}\mathbf{T}^{-1}, \quad \mathbf{A}_{\text{rat}} = \operatorname{diag}(\mathbf{C}_1, \ldots, \mathbf{C}_N) \tag{5}$$

with $j = 1, \ldots, N$ companion matrices $\mathbf{C}_j$ defined by the N elementary divisor polynomials of $\mathbf{A}$. The matrix $\mathbf{A}_{\text{rat}}$ is unique up to block ordering and is called (elementary divisor form of the) rational canonical form or classical canonical form of $\mathbf{A}$.

Example 2
For the following Smith form of a matrix $\mathbf{A} \in \mathbb{F}_2^{6\times 6}$,

$$\mathbf{S}(\lambda) = \operatorname{diag}\big((\lambda^2+\lambda+1)(\lambda+1)^2\lambda, \lambda+1, 1, 1, 1, 1\big)$$

we have the elementary divisor polynomials

$$p_1(\lambda) = \lambda^2+\lambda+1, \quad p_2(\lambda) = (\lambda+1)^2, \quad p_3(\lambda) = \lambda, \quad p_4(\lambda) = \lambda+1\,.$$

defining the companion matrices in the rational canonical form of $\mathbf{A}$, that is

$$\mathbf{C}_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \ \mathbf{C}_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \ \mathbf{C}_3 = \begin{pmatrix} 0 \end{pmatrix}, \ \mathbf{C}_4 = \begin{pmatrix} 1 \end{pmatrix}, \quad \mathbf{A}_{\text{rat}} = \operatorname{diag}(\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3, \mathbf{C}_4)\,.$$

Remark 3
We did not introduce the Jordan normal form of a matrix, which would follow from diagonalizing the rational canonical form. The reason is that the Jordan normal form is accompanied by the notion of an extension field $\mathbb{F}_{q^k}, k = 1, 2, \ldots$ associated to $\mathbb{F}_q$. In case of a finite field, the calculation of roots in this extension field $\mathbb{F}_{q^k}$ is much more cumbersome than it is in the extension field associated to the real numbers $\mathbb{R}$, which is $\mathbb{C}$, the field of complex numbers.

3. MULTILINEAR AUTOMATON MODEL OVER THE FINITE FIELD $\mathbb{F}_2$

In this section we develop an algebraic model for a non-deterministic finite state automaton with multiple inputs. It takes the form

$$f(\mathbf{x}[k+1], \mathbf{x}[k], \mathbf{u}[k]) = 0, \quad \mathbf{x} \in \mathbb{F}_2^n, \mathbf{u} \in \mathbb{F}_2^m, \tag{6}$$

where f marks an implicit scalar transition function $f : \mathbb{F}_2^n \times \mathbb{F}_2^n \times \mathbb{F}_2^m \ \rightarrow \ \mathbb{F}_2$, which relates the n states $\mathbf{x}[k]$ and the m inputs $\mathbf{u}[k]$ in an instant k with the possibly multiple successor states $\mathbf{x}[k+1]$ in the instant $k+1$, indicating possible behavior by mapping onto 0. The transition function is multilinear in the vector elements of $\mathbf{x}[k+1]$, $\mathbf{x}[k]$ and $\mathbf{u}[k]$ except for a constant. This will be shown subsequently.

3.1. The Relation of Boolean Algebra and the Finite Field $\mathbb{F}_2$

Some boolean algebra is required for calculating the automaton model over finite fields. Therefore the necessary basics of boolean algebra are recalled for convenience; concise introductions are given by [16, 1].

Definition 12 (Boolean Operations)
Given the set $\mathbb{B} = \{0,1\}$. The operations $\wedge$, $\vee$, $\oplus$ and $\bar{\ }$ defined on$\mathbb{B}$ as per

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

x_1	x_2	$x_1 \vee x_2$
0	0	0
0	1	1
1	0	1
1	1	1

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

x	$\bar{x}$
0	1
1	0

are termed boolean operations.

Typically, boolean operations are used for constructing boolean functions, usually expressed in normal forms. Problem oriented normal forms help reducing complexity in logical devices and admit an easier decomposition of logical functions into subfunctions in order to improve modularity. One standard normal form is the following.

Definition 13 (Disjunctive Normal Form)
Let f be a boolean function of indeterminates $x_1,\ldots,x_n \in \mathbb{B}$ and $\mathbf{c}$ be a vector $\in \mathbb{B}^n$. Then the disjunctive normal form of f is given by

$$f(x_1,\ldots,x_n) = \bigvee_{\mathbf{c}\in\mathbb{B}^n} f(\mathbf{c}^{\mathrm{T}}) \wedge \bigwedge_{i=1}^{n} (x_i \oplus c_i)\ . \tag{7}$$

Example 3
The disjunctive normal form of $f(x_1,x_2) = x_1 \oplus x_2$ is

$$\begin{aligned} f(x_1,x_2) \ =\ & \big(f(0,0)\wedge(x_1\oplus 0)\wedge(x_2\oplus 0)\big) \vee \big(f(0,1)\wedge(x_1\oplus 0)\wedge(x_2\oplus 1)\big) \vee \\ & \big(f(1,0)\wedge(x_1\oplus 1)\wedge(x_2\oplus 0)\big) \vee \big(f(1,1)\wedge(x_1\oplus 1)\wedge(x_2\oplus 1)\big) = \\ & \big((x_1\wedge(x_2\oplus 1)\big) \vee \big((x_1\oplus 1)\wedge x_2\big) = (x_1\wedge\bar{x}_2)\vee(\bar{x}_1\wedge x_2)\ . \end{aligned} \tag{8}$$

Instead of introducing all boolean operations from Definition 12 it is sufficient to confine oneself to the operations $\oplus$ and $\wedge$. By using DeMorgan's Law and observing $\bar{x} = 1\oplus x$, the disjunction $x_1 \vee x_2$ can be evaluated to

$$x_1 \vee x_2 = \overline{\bar{x}_1 \wedge \bar{x}_2} = 1 \oplus ((1\oplus x_1)\wedge(1\oplus x_2)) = x_1 \oplus x_2 \oplus x_1x_2, \quad x_1,x_2 \in \mathbb{B}\,.$$

Thus, if the operations $\oplus$ and $\wedge$ on the set $\mathbb{B} = \{0,1\}$ are identified with addition and multiplication, both modulo 2, then the following important theorem can be stated.

Theorem 6 (Isomorphism of $\mathbb{F}_2$ and $\mathbb{B}$)
The set $\mathbb{B} = \{0,1\}$ together with the operations $+ := \oplus$ and $\cdot := \wedge$ is a finite field. The finite field $\mathbb{B}$ is isomorphic to the Galois-Field $\mathbb{F}_2$.

Since any boolean function f can be manipulated so as to obtain an expression in $\oplus$ and $\wedge$ only, the calculation of the finite field representation of f over $\mathbb{F}_2$ amounts to simply interchange $\oplus$ by $+$ and $\wedge$ by $\cdot$, respectively (from now on all additions and multiplications taken modulo 2). Then for $x_1, x_2 \in \{0,1\}$ the following applies:

$$x_1 \wedge x_2 \iff x_1 x_2 \tag{9}$$

$$x_1 \vee x_2 \iff x_1 + x_2 + x_1 x_2 \tag{10}$$

$$x_1 \oplus x_2 \iff x_1 + x_2 \tag{11}$$

$$\bar{x} \iff 1 + x \tag{12}$$

These equivalences are of particular interest in the next sections.

3.2. Deriving the Algebraic Model by Use of the Disjunctive Normal Form

In the following, a single input example is elaborated to introduce the main steps for obtaining the transition function for a non-deterministic automaton over the finite field $\mathbb{F}_2$. The underlying algorithm can be generalized easily and is left out for clearness.

Consider the automaton depicted in Figure 1. The nodes are coded by binary vectors, which represent values for the states $\mathbf{x}^{\mathrm{T}} = (x_1, x_2)$. Arcs connect the states and indicate possible transitions between the states. Marked arcs denote that the transition depends on a certain condition on the input variables u. If no marking is specified on an arc a transition is possible for any choice of inputs. In general, leaving arcs do not determine unique successor states, i. e. the automaton is non-deterministic.

We will omit the symbol k in the denotation of $x_i[k]$ and $u[k]$ and abbreviate $x_i[k+1]$ by x'_i. Also, the logical interconnection of states x_1, x_2, input u and successor states x'_1, x'_2 is represented by a state table (see Figure 1). Regarding each row in the state table, a transition is possible if and only if its function value is $f_c = 1$.

Therefore, in view of Definition 13 and along the lines of Example 3, the disjunctive normal form of the function f with respect to the state table of Figure 1 reads

$$\begin{aligned} f(x'_1, x'_2, x_1, x_2, u) &= \overline{ux'_2}\,\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee \overline{ux'_2}\,\overline{x'_1}x_2\overline{x_1} \vee \overline{ux'_2}x'_1\overline{x_2}\,\overline{x_1} \vee \overline{ux'_2}x'_1\overline{x_2}x_1 \vee \bar{u}x'_2\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee \\ &\quad \bar{u}x'_2x'_1\overline{x_2}x_1 \vee \bar{u}x'_2x'_1x_2x_1 \vee u\overline{x'_2}\,\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee u\overline{x'_2}\,\overline{x'_1}x_2\overline{x_1} \vee u\overline{x'_2}x'_1\overline{x_2}x_1 \vee \\ &\quad ux'_2\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee ux'_2x'_1\overline{x_2}\,\overline{x_1} \vee ux'_2x'_1\overline{x_2}x_1 \vee ux'_2x'_1x_2x_1 \\ &= \overline{x'_2}\,\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee \overline{x'_2}\,\overline{x'_1}x_2\overline{x_1} \vee \overline{ux'_2}x'_1\overline{x_2}\,\overline{x_1} \vee \overline{x'_2}x'_1\overline{x_2}x_1 \vee x'_2\overline{x'_1}\,\overline{x_2}\,\overline{x_1} \vee \\ &\quad x'_2x'_1\overline{x_2}x_1 \vee x'_2x'_1x_2x_1 \vee ux'_2x'_1\overline{x_2}\,\overline{x_1} = 1\,, \end{aligned} \tag{13}$$

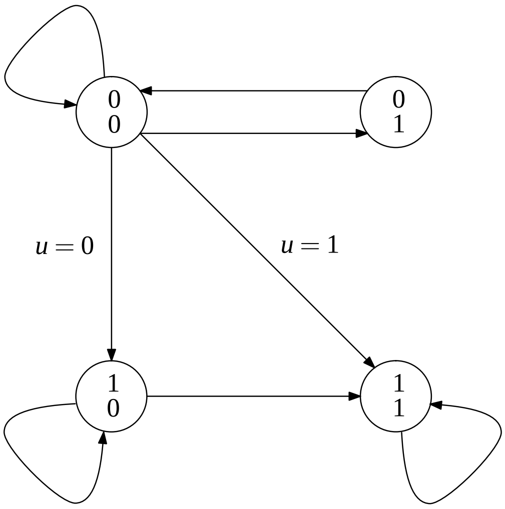

Fig. 1. Graph of an example automaton (above) and its state table (right hand side). The column marked f_c signifies whether a transition from $(x_1,x_2)^{\mathrm{T}}$ to $(x_1',x_2')^{\mathrm{T}}$ under input u is possible and vice versa.

u	x_2'	x_1'	x_2	x_1	f_c
0	0	0	0	0	1
0	0	0	0	1	0
0	0	0	1	0	1
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	1
0	0	1	1	0	0
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	0	1	0
0	1	0	1	0	0
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	0	1	1
0	1	1	1	0	0
0	1	1	1	1	1
1	0	0	0	0	1
1	0	0	0	1	0
1	0	0	1	0	1
1	0	0	1	1	0
1	0	1	0	0	0
1	0	1	0	1	1
1	0	1	1	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	0	1	0
1	1	0	1	0	0
1	1	0	1	1	0
1	1	1	0	0	1
1	1	1	0	1	1
1	1	1	1	0	0
1	1	1	1	1	1

where we used the abbreviation $a \wedge b = a\,b$. Observe that all disjunctions $\vee$ can be eliminated according to

$$a_1 \vee a_2 \vee \ldots \vee a_k = 1 \quad \Longleftrightarrow \quad (1 \oplus a_1) \wedge (1 \oplus a_2) \wedge \ldots \wedge (1 \oplus a_k) = 0\,, \tag{14}$$

which amounts to applying DeMorgan's law. The remaining negations in (13) vanish by setting $\bar{a} = a \oplus 1$. As a result, we get a function consisting of the operations $\wedge$ and $\oplus$ only. Therefore, via (9) and (11) we finally obtain the representation of the transition function in the finite field $\mathbb{F}_2$

$$\begin{aligned} f(x_1',x_2',x_1,x_2,u) \,=\, & \big(1+(1+x_2')(1+x_1')(1+x_2)(1+x_1)\big)\cdots \\ & \big(1+x_2'x_1'x_2x_1\big)\big(1+ux_2'x_1'(1+x_2)(1+x_1)\big) = 0\,, \end{aligned} \tag{15}$$

$$\begin{aligned} \Leftrightarrow \;\; f(x_1',x_2',x_1,x_2,u) \,=\, & x_1 + x_1x_1' + x_2x_1' + x_2x_2' + x_1x_2x_2' + x_1'x_2' + x_1x_1'x_2' + \\ & x_1x_2x_1'x_2' + x_1'u + x_1x_1'u + x_2x_1'u + x_1x_2x_1'u = 0\,. \end{aligned} \tag{16}$$

3.3. Simplifications Using Reed-Muller Generator Matrices

The regular tabulation of the state table in Figure 1 — which is binary counting, row by row — allows an efficient calculation of the transition function f. So-called Reed-Muller codes, well-known from linear coding theory, are based on this property [8].

Consider the recursively defined Reed-Muller generator matrices

$$\mathbf{G}_i := \begin{pmatrix} \mathbf{G}_{i-1} & \mathbf{0} \\ \mathbf{G}_{i-1} & \mathbf{G}_{i-1} \end{pmatrix}, \quad \mathbf{G}_0 := 1\ . \tag{17}$$

Then following [8] we have the simple matrix–vector product over $\mathbb{F}_2$

$$\mathbf{c}_{2n+m} = \mathbf{G}_{2n+m}\mathbf{f}_c\ , \tag{18}$$

in which $\mathbf{c}_{2n+m}$ is the $(2^{2n+m},1)$-vector of coefficients associated to a particular tabulation of monomials in a $(2^{2n+m},1)$-vector $\boldsymbol{\varphi}_{2n+m}$. In general, $\boldsymbol{\varphi}_{2n+m}$ contains all monomials of the n states x_i, of the m inputs u_i and of the next states x_i'. The vector $\mathbf{f}_c$ is the $(2^{2n+m},1)$-vector with respect to the rightmost column in the state table of Figure 1. Using equation (18) the demanded transition function

$$f(x_1',x_2',\ldots,x_n',x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_m) = \mathbf{c}_{2n+m}^{\mathrm{T}}\boldsymbol{\varphi}_{2n+m} + 1 = 0 \tag{19}$$

follows. It remains to explain how to tabulate the monomials in $\boldsymbol{\varphi}_{2n+m}$. We return to the example of Figure 1 with $n=2$ states and $m=1$ inputs. In this case we have

$$\mathbf{G}_5 = \left(\begin{array}{cccccccccccccccccccccccccccccccc}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&1&1&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0&1&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&1&1&0&0&0&0&0&0&1&1&0&0&0&0&0&0&1&1&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&1&0&1&0&0&0&0&0&1&0&1&0&0&0&0&0&1&0&1&0&0&0&0&0\\
1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0\\
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\
1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0\\
1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1
\end{array}\right),\ \boldsymbol{\varphi}_5 = \begin{pmatrix}
1\\ x_1\\ x_2\\ x_1x_2\\ x_1'\\ x_1x_1'\\ x_2x_1'\\ x_1x_2x_1'\\ x_2'\\ x_1x_2'\\ x_2x_2'\\ x_1x_2x_2'\\ x_1'x_2'\\ x_1x_1'x_2'\\ x_2x_1'x_2'\\ x_1x_2x_1'x_2'\\ u\\ x_1u\\ x_2u\\ x_1x_2u\\ x_1'u\\ x_1x_1'u\\ x_2x_1'u\\ x_1x_2x_1'u\\ x_2'u\\ x_1x_2'u\\ x_2x_2'u\\ x_1x_2x_2'u\\ x_1'x_2'u\\ x_1x_1'x_2'u\\ x_2x_1'x_2'u\\ x_1x_2x_1'x_2'u
\end{pmatrix} \tag{20}$$

with the Reed-Muller generator matrix $\mathbf{G}_5$ and the vector of monomials $\boldsymbol{\varphi}_5$. The tabulation regarding the elements of $\boldsymbol{\varphi}_5$ is carried out recursively: if we start the inspection with the vector entry 1 from the top, then for every new variable, the vector is extended by a copy of the former part of the vector multiplied with the new variable, and so on. Substituting (20) in (18) and (19) verifies the result from (16).

3.4. Enhancements and Generalizations

Abstracting from the latter exemplary viewpoint, in general we obtain a multilinear transition function $f : \mathbb{F}_2^n \times \mathbb{F}_2^n \times \mathbb{F}_2^m \to \mathbb{F}_2$

$$f(\mathbf{x}[k+1], \mathbf{x}[k], \mathbf{u}[k]) = 0 = \sum_{\mathcal{S}_1 \in 2^{\mathcal{I}_n}} \sum_{\mathcal{S}_2 \in 2^{\mathcal{I}_n}} \sum_{\mathcal{S}_3 \in 2^{\mathcal{I}_m}} \delta^{\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3} \Big(\prod_{j \in \mathcal{S}_1} x_j[k+1] \Big) \Big(\prod_{l \in \mathcal{S}_2} x_l[k] \Big) \Big(\prod_{m \in \mathcal{S}_3} u_m[k] \Big), \tag{21}$$

with $\mathbf{x} \in \mathbb{F}_2^n$ and $\mathbf{u} \in \mathbb{F}_2^m$, where the sets $\mathcal{I}_n = \{1, 2, \ldots, n\}$, $\mathcal{I}_m = \{1, 2, \ldots, m\}$ are index sets, $2^{\mathcal{I}_n}$ denotes the (possibly empty) power set of $\mathcal{I}_n$ and $\delta^{\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3}$ are constants in $\mathbb{F}_2$. If for fixed $\mathbf{x}[k]$ and $\mathbf{u}[k]$ we focus on $\mathbf{x}[k+1]$, we might observe multiple successors $\mathbf{x}[k+1]$. Strictly speaking, we then should call (21) not a function, but a relation. As a consequence, the finite field representation (21) is capable of modeling non-deterministic finite state automata as well.

Leaving aside the details in the following paragraphs, for brevity, we will only bring to light the ideas of some straight forward extensions of the setting.

Additional States

Considering more detailed and refined process models often is the remedy against the lack of information in coarse automaton models. In this process, usually further states and inputs have to be added to the automaton model. These further states and inputs may be integrated by concatenating the state table on the left with the respective columns of the new states and inputs (see Figure 1). Using the associated, bigger Reed-Muller generator matrices, the calculation of the monomial coefficients in the state transition function still amounts to the same procedure. However, one special feature of the Reed-Muller approach comes to the fore: only the coefficients referring to the new variables need to be calculated, the coefficients of the former representation are left unchanged. This is one major advantage of the Reed-Muller approach.

Partially Defined Transition Functions

A partially defined transition function is a function $f : \mathcal{X} \times \mathcal{X} \times \mathcal{U} \to \mathbb{F}_2$ which is defined on proper subsets $\mathcal{X} \subset \mathbb{F}_2^n$ and $\mathcal{U} \subset \mathbb{F}_2^m$, respectively. This means that only some states and inputs may be defined, for example not the whole number of 2^n states. Accordingly, only a few rows in the entire state table may be defined. To this account a check function $r(x_1,\dots,x_n,u_1,\dots,u_m)$ can be introduced in the same manner as f_c. The value of r equals 0 if the respective state and/or input is defined, and 1 elsewhere. Finally, the check function r and the transition function f can be combined in one single equation. Thus, for a state $\mathbf{x}[k]$ and an input $\mathbf{u}[k]$ the extended transition function is 0 if and only if $\mathbf{x}[k] \in \mathcal{X}$ and $\mathbf{u}[k] \in \mathcal{U}$.

Determinism

In case of deterministic automata, the state tables can be reshaped as illustrated in Figure 2. Thus, by employing the methods of Section 3.2 and 3.3 an explicit transition function, a so-called state equation, can be determined. This results in

$$\mathbf{x}[k+1] = \mathbf{f}(\mathbf{x}[k], \mathbf{u}[k]), \quad \mathbf{x} \in \mathbb{F}_2^n, \mathbf{u} \in \mathbb{F}_2^m \tag{22}$$

and reminds of a discrete time system in the continuous world. In the next sections we will restrict the further examinations to the deterministic linear case.

u_m	$\cdots$	u_2	u_1	x_n	$\cdots$	x_2	x_1	x'_n	$\cdots$	x'_2	x'_1
0	$\cdots$	0	0	0	$\cdots$	0	0	$f_{n,1}$	$\cdots$	$f_{2,1}$	$f_{1,1}$
0	$\cdots$	0	0	0	$\cdots$	0	1	$f_{n,2}$	$\cdots$	$f_{2,2}$	$f_{1,2}$
0	$\cdots$	0	0	0	$\cdots$	1	0	$f_{n,3}$	$\cdots$	$f_{2,3}$	$f_{1,3}$
$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
1	$\cdots$	1	1	1	$\cdots$	1	1	$f_{n,2^{n+m}}$	$\cdots$	$f_{2,2^{n+m}}$	$f_{1,2^{n+m}}$

Fig. 2. Scheme of a state table for a deterministic automaton

4. ANALYSIS OF AUTONOMOUS LINEAR MODULAR SYSTEMS

The modeling power of the finite field framework shall be examined. To this end, we consider a deterministic, linear system of the form

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k], \quad \mathbf{x} \in \mathbb{F}_2^n, \mathbf{u} \in \mathbb{F}_2^m, \tag{23}$$

a so-called *linear modular system* (LMS).[3] The matrix $\mathbf{A} \in \mathbb{F}_2^{n\times n}$ is the dynamics of the system, the matrix $\mathbf{B} \in \mathbb{F}_2^{n\times m}$ is the input matrix. At first, we examine linear

[3] Usually LMS are defined over $\mathbb{F}_q$ for some prime number q. Here, we will loosely speak of an LMS when we assume an LMS with characteristic $q = 2$, unless it is specified differently.

systems with $\mathbf{B}=\mathbf{0}$, according to the simplest automaton description in (22), that is

$$\mathbf{x}[k{+}1]=\mathbf{A}\,\mathbf{x}[k], \quad \mathbf{x}\in\mathbb{F}_2^n. \tag{24}$$

With regard to these linear systems the analysis for cyclic (periodic) states is carried out, recalling some results from [6, 13]. The properties of finite fields and polynomials over finite fields, which have been presented in Section 2, will provide the concepts necessary for solving the analysis problem.

4.1. Periodic Nullspace Decomposition of a Companion Matrix

Since a graph of an automaton typically shows cyclic and acyclic behavior the state space of the respective LMS decomposes into aperiodic and periodic subspaces. It is clear that in the autonomous case any information must be included in the dynamics $\mathbf{A}$. Thus, we ought to investigate $\mathbf{A}$ for information about periodic states which are characterized by the following definition.

Definition 14 (Period of a State)
A state $\mathbf{x}_\tau$ of an LMS is called τ-periodic if

$$\mathbf{x}_\tau\in\mathcal{X}_\tau, \quad \mathcal{X}_\tau:=\left\{\boldsymbol{\xi}\in\mathbb{F}_2^n\,|\,\exists\tau\in\mathbb{N},\boldsymbol{\xi}=\mathbf{A}^\tau\boldsymbol{\xi}\wedge\forall i\in\mathbb{N},i<\tau,\boldsymbol{\xi}\neq\mathbf{A}^i\boldsymbol{\xi}\right\},$$

in which $\mathcal{X}_\tau$ is denoting the set of τ-periodic states.

Regarding the linear system (24) we immediately obtain the relation

$$(\mathbf{A}^\tau+\mathbf{I})\,\mathbf{x}_\tau=\mathbf{0} \tag{25}$$

for determining the τ-periodic states $\mathbf{x}_\tau\in\mathbb{F}_2^n$. An obvious brute force procedure for calculating the τ-periodic states would be to solve the linear n-th order system (25) for all $\mathbf{x}_\tau,\tau=1,2,\ldots,2^n$. However, this quickly becomes numerically intractable, even for small orders n. Instead, we propose to benefit from a similarity transform of $\mathbf{A}$ into its rational canonical form $\mathbf{A}_{\mathrm{rat}}$. This transform only renumbers the state vectors and retains the elementary divisor polynomials unchanged, thus, periodicity properties are preserved. As a consequence, it is possible to examine the periodic subspaces by the rational canonical form of the dynamics, introduced in equation (5), which is structurally simpler. Accordingly, by transforming $\tilde{\mathbf{x}}=\mathbf{T}\mathbf{x}$ we rephrase (25) as per

$$(\mathbf{A}^\tau+\mathbf{I})\,\mathbf{x}_\tau=\mathbf{0} \quad\Longleftrightarrow\quad (\mathrm{diag}(\mathbf{C}_1^\tau,\mathbf{C}_2^\tau,\ldots,\mathbf{C}_N^\tau)+\mathbf{I})\,\tilde{\mathbf{x}}_\tau=\mathbf{0}, \tag{26}$$

which equivalently can be expressed as

$$(\mathbf{C}_i^{\tau_i}+\mathbf{I})\,\tilde{\mathbf{x}}_{\tau_i}=\mathbf{0}, \quad i=1,\ldots,N \qquad \tau=\mathrm{lcm}(\tau_1,\tau_2,\ldots,\tau_N), \tag{27}$$

in which each $\tau_i\in\mathbb{N}$ is minimal, $\tilde{\mathbf{x}}_\tau^{\mathrm{T}}=(\tilde{\mathbf{x}}_{\tau_1},\tilde{\mathbf{x}}_{\tau_2},\ldots,\tilde{\mathbf{x}}_{\tau_N})$ with $\tilde{\mathbf{x}}_\tau\in\mathbb{F}_2^{d_1}\times\ldots\times\mathbb{F}_2^{d_N}$, d_i is the dimension of C_i, $n=\sum_{i=1}^N d_i$ and lcm$(.)$ denotes the least common multiple of its arguments. Equation (26) indicates that the state space decomposes into N subspaces which can be examined separately. Note that for nilpotent companion matrices $\mathbf{C}_i$ the matrix $\mathbf{C}_i^{\tau_i}+\mathbf{I}$ is non-singular. In this case (27) can hold for the zero vector only, and in consequence it is sufficient to confine the examination on cyclic companion matrices $\mathbf{C}_i$. Moreover, as $\mathbf{A}_{\text{rat}}$ is a rational canonical form it contains companion matrices with respect to powers of individual irreducible polynomials only, hence, it remains to consider relation (27) in view of a companion matrix $\mathbf{C}$ whose defining (characteristic, minimal) polynomial $p_{\mathbf{C}}(\lambda)=\text{cp}_{\mathbf{C}}(\lambda)=\text{mp}_{\mathbf{C}}(\lambda)$ can be written as

$$p_{\mathbf{C}}(\lambda)=(p_{\text{irr},\mathbf{C}}(\lambda))^e\,. \tag{28}$$

The examination will be organized in four steps:[4] firstly, it will be recalled that the kernel of a matrix $p_{\mathbf{C}}(\mathbf{C})$ can be decomposed into e nested linear subspaces the dimensions of which shall be determined in a second step. With the knowledge about these dimensions, the number of states in the respective subspace is clear and the period of its states can be be derived. Finally, the superposition of the results for all $i=1,\dots,N$ subsystems leads to the main theorem of this section.

The following property follows from the fact that the rank deficiency of singular matrices strictly increases by its exponent.

Lemma 1 (Nesting Property of Nullspaces)
Let $p_{\mathbf{C}}(\lambda)=(p_{\text{irr},\mathbf{C}}(\lambda))^e\in\mathbb{F}_2[\lambda]$, $e\in\mathbb{N}$, be the d-th degree defining polynomial of a companion matrix $\mathbf{C}\in\mathbb{F}_2^{d\times d}$. Assume the basis polynomial $p_{\mathbf{C},\text{irr}}(\lambda)$ to be irreducible over $\mathbb{F}_2$. Then the strict inclusion property (nesting) applies

$$\mathcal{N}_0\subset\mathcal{N}_1\subset\mathcal{N}_2\subset\ldots\subset\mathcal{N}_e=\mathbb{F}_2^d\,, \tag{29}$$

where the $\mathcal{N}_j$ are nullspaces, $\mathcal{N}_0:=\{\mathbf{0}\}$ and $\mathcal{N}_j:=\text{Ker}\big((p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\big)$, $j=1,\dots,e$.

By virtue of the non-singularity of cyclic companion matrices $\mathbf{C}$ it follows

$$(p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\,\mathbf{x}_j=\mathbf{0}\quad\Longleftrightarrow\quad\mathbf{C}\,(p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\,\mathbf{x}_j=\mathbf{0}\quad\Longleftrightarrow\quad(p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\,\mathbf{C}\,\mathbf{x}_j=\mathbf{0}$$

for any $\mathbf{x}_j\in\text{Ker}\big((p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\big)$, $j=1,\dots,e$, which implies

Lemma 2 (Invariance of the Nullspaces)
Let $(p_{\text{irr},\mathbf{C}})^e\in\mathbb{F}_2[\lambda]$ be the defining polynomial of the cyclic companion matrix $\mathbf{C}\in\mathbb{F}_2^{d\times d}$ with $p_{\text{irr},\mathbf{C}}$ irreducible. Then the $j=1,\dots,e$ nullspaces of the matrices $(p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j$ are invariant under the transformation $\mathbf{C}$ on any $\mathbf{x}_j\in\text{Ker}\big((p_{\text{irr},\mathbf{C}}(\mathbf{C}))^j\big)$.

In light of Lemma 2, for $j=1,\dots,e$ define the map $\mathbf{C}|_{\mathcal{N}_j}:\mathcal{N}_j\rightarrow\mathcal{N}_j$ such that

$$\mathbf{C}|_{\mathcal{N}_j}\,\mathbf{x}_j=\mathbf{C}\,\mathbf{x}_j,\quad\forall\mathbf{x}_j\in\mathcal{N}_j \tag{30}$$

[4] For brevity, only outlines of the proofs are presented.

describes the action of the linear transform $\mathbf{C}$ on the subspace $\mathcal{N}_j$ only. Consequently,

$$(p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}|_{\mathcal{N}_j}))^j\,\mathbf{x}_j = (p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}))^j\,\mathbf{x}_j, \quad \forall \mathbf{x}_j \in \mathcal{N}_j \tag{31}$$

and as $\mathbf{x}_j$ lies in the kernel of $\mathcal{N}_j$ we have

$$(p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}|_{\mathcal{N}_j}))^j\,\mathbf{x}_j = \mathbf{0}, \quad \forall \mathbf{x}_j \in \mathcal{N}_j\,. \tag{32}$$

Based on the nesting property, stated in Lemma 1, it can be shown that $(p_{\mathrm{irr},\mathbf{C}}(\lambda))^j$ is not only an annihilating but the minimal polynomial of the matrix $\mathbf{C}|_{\mathcal{N}_j}$, i. e.

$$\mathrm{mp}_{\mathbf{C}|_{\mathcal{N}_j}}(\lambda) = (p_{\mathrm{irr},\mathbf{C}}(\lambda))^j\,. \tag{33}$$

Since to any minimal polynomial corresponds a companion matrix the dimension of which is the degree of its minimal polynomial, we conclude

Lemma 3 (Dimension of the Nullspaces)
Let $(p_{\mathrm{irr},\mathbf{C}})^e \in \mathbb{F}_2[\lambda]$ be the defining polynomial of the cyclic companion matrix $\mathbf{C} \in \mathbb{F}_2^{d\times d}$ with $p_{\mathrm{irr},\mathbf{C}}$ irreducible and $\delta = \deg(p_{\mathrm{irr},\mathbf{C}})$. Let the nullspaces $\mathcal{N}_j := \mathrm{Ker}\big((p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}))^j\big)$, $j = 1,\ldots,e$. Then the dimension of each nullspace $\mathcal{N}_j$ is

$$\dim(\mathcal{N}_j) = \deg\big((p_{\mathrm{irr},\mathbf{C}})^j\big) = \delta\, j\,. \tag{34}$$

It remains to investigate the periods of the subspace states. To this end, let t_κ, $\kappa \in \mathbb{N}$, denote the period of the polynomial $(p_{\mathrm{irr},\mathbf{C}}(\lambda))^\kappa$, hence

$$g(\lambda)(p_{\mathrm{irr},\mathbf{C}}(\lambda))^\kappa = \lambda^{t_\kappa} - 1 \tag{35}$$

for some polynomial $g(\lambda)$ and therefore

$$g(\mathbf{C}|_{\mathcal{N}_j})(p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}|_{\mathcal{N}_j}))^\kappa = (\mathbf{C}|_{\mathcal{N}_j})^{t_\kappa} - \mathbf{I}\,. \tag{36}$$

Right-multiplication by an arbitrary state $\mathbf{x}_j \in \mathcal{N}_j$ yields

$$g(\mathbf{C}|_{\mathcal{N}_j})(p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}|_{\mathcal{N}_j}))^\kappa \mathbf{x}_j = ((\mathbf{C}|_{\mathcal{N}_j})^{t_\kappa} - \mathbf{I})\,\mathbf{x}_j\,. \tag{37}$$

Thus $\mathbf{x}_j$ is t_κ-periodic if

$$(p_{\mathrm{irr},\mathbf{C}}(\mathbf{C}|_{\mathcal{N}_j}))^\kappa \mathbf{x}_j = \mathbf{0}\,, \tag{38}$$

which is the case if $\mathbf{x}_j \in \mathcal{N}_\kappa$. Defining $\mathcal{D}_j := \mathcal{N}_j \backslash \mathcal{N}_{j-1}$, $j = 1,\ldots,e$, the states in $\mathcal{D}_\kappa$ turn out to be exactly those which are τ_κ-periodic, and we obtain

Lemma 4 (Period of the States in $\mathcal{D}_j$)
Let the dynamics matrix of an LMS be given by a cyclic companion matrix $\mathbf{C} \in \mathbb{F}_2^{d\times d}$ the defining polynomial $(p_{\mathrm{irr},\mathbf{C}})^e \in \mathbb{F}_2[\lambda]$ of which is the power of an irreducible polynomial $p_{\mathrm{irr},\mathbf{C}}$. Furthermore, let $\mathcal{D}_j := \mathcal{N}_j \backslash \mathcal{N}_{j-1}$, $j = 1,\ldots,e$. Then any state vector in the set $\mathcal{D}_j$ is τ_j-periodic, where τ_j is the period of the polynomial $(p_{\mathrm{irr},\mathbf{C}}(\lambda))^j$.

Recalling $\delta = \deg(p_{\mathrm{irr},\mathbf{C}})$, all $q^{j\delta} - q^{(j-1)\delta}$ states in $\mathcal{D}_j$ have period τ_j such that $\nu_j = \big(q^{j\delta} - q^{(j-1)\delta}\big)/\tau_j$ cycles of τ_j-periodic states lie in the space $\mathcal{D}_j$. Adding up the number of states in $\mathcal{D}_j$ from $j = 1, \ldots, e$ plus the remaining zero state results in

$$1 + \sum_{j=1}^{e} q^{j\delta} - q^{(j-1)\delta} = q^{e\delta} = q^d \tag{39}$$

which shows that the entire space $\mathbb{F}_2^d$ is composed of these cycles. Collecting all lemmas and referring to Theorem 3 for the period of powered polynomials implies

Theorem 7 (Periodic Nullspace Decomposition of a Companion Matrix)
Given a cyclic companion matrix $\mathbf{C} \in \mathbb{F}_2^{d \times d}$ with respect to the d-th degree polynomial $p_{\mathbf{C}} = (p_{\mathrm{irr},\mathbf{C}})^e$, where $p_{\mathrm{irr},\mathbf{C}} \in \mathbb{F}_2[\lambda]$ is irreducible with degree δ such that $d = e\delta$. Then the associated state space $\mathbb{F}_2^d$ is entirely composed of periodic states as per

$$\begin{array}{lcl}
\nu_0 = 1 & \text{cycles of length} & \tau_0 = 1 \\
\nu_1 = \big(2^{\delta} - 1\big)/\tau_1 & \text{''} & \tau_1 = \tau \\
\nu_2 = \big(2^{2\delta} - 2^{\delta}\big)/\tau_2 & \text{''} & \tau_2 = 2\tau \\
\cdots\cdots\cdots\cdots & \cdots\cdots\cdots & \cdots\cdots \\
\nu_j = \big(2^{j\delta} - 2^{(j-1)\delta}\big)/\tau_j & \text{''} & \tau_j = 2^{l_j}\tau \\
\cdots\cdots\cdots\cdots & \cdots\cdots\cdots & \cdots\cdots \\
\nu_e = \big(2^{e\delta} - 2^{(e-1)\delta}\big)/\tau_e & \text{''} & \tau_e = 2^{l_e}\tau
\end{array}$$

where each l_j, $j = 1, \ldots, e$, is the least integer such that $2^{l_j} \geq j$.

The periodic decomposition can be written in a more convenient form by applying

Definition 15 (Cycle Sum)
The cycle sum Σ is the formal sum of cycle terms

$$\Sigma = \nu_1[\tau_1] \dotplus \nu_2[\tau_2] \dotplus \ldots \dotplus \nu_\kappa[\tau_{N_\Sigma}]\,, \tag{40}$$

in which the cycle term $\nu_i[\tau_i]$ denotes ν_i cycles of length τ_i and the binary operation $\dotplus$ satisfies $\nu_i[\tau] \dotplus \nu_j[\tau] = (\nu_i + \nu_j)[\tau]$. The number of cycles in Σ is denoted by N_Σ.

Using this definition the result of Theorem 7 can be rewritten as

$$\Sigma = 1[1] \dotplus \frac{2^{\delta} - 1}{\tau_1}[\tau_1] \dotplus \frac{2^{2\delta} - 2^{\delta}}{\tau_2}[\tau_2] \dotplus \ldots \dotplus \frac{2^{e\delta} - 2^{(e-1)\delta}}{\tau_e}[\tau_e]\,, \tag{41}$$

in which τ_j, $j = 1, \ldots, e$, marks the periods of the polynomial $\big(p_{\mathrm{irr},\mathbf{C}}(\lambda)\big)^j$ which can be computed via Theorem 3.

4.2. The Cycle Sum of an Autonomous LMS

As we have developed the cycle set theory for one single companion matrix we just have to superpose the results for all N companion matrices in $\mathbf{A}_{\text{rat}}$. To this end, we consider the following $(d_1+d_2)\times(d_1+d_2)$ block-diagonal matrix

$$\mathbf{C}=\begin{pmatrix}\mathbf{C}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2\end{pmatrix}, \quad \mathbf{C}_1\in\mathbb{F}_2^{d_1\times d_1}, \mathbf{C}_2\in\mathbb{F}_2^{d_2\times d_2}$$

for which we may assume the corresponding cycle sums

$$\Sigma_i = 1[1]+\nu_i[\tau_i], \quad i=1,2 \tag{42}$$

Therefore, we see that the subspaces $\mathcal{X}_1$ and $\mathcal{X}_2$ associated to $\mathbf{C}_1$ and $\mathbf{C}_2$ consist of $d_1=1+\nu_1\tau_1$ and $d_2=1+\nu_2\tau_2$ elements, respectively. In view of the Cartesian product $\mathcal{X}=\mathcal{X}_1\times\mathcal{X}_2$, the number d of elements in the space $\mathcal{X}$ associated to $\mathbf{C}$ is

$$d=1+\nu_1\tau_1+\nu_2\tau_2+\nu_1\nu_2\tau_1\tau_2$$

There are only 4 possible combinations between the periodic subspaces $\mathcal{X}_1$ and $\mathcal{X}_2$

1. Combination of the 0-state in $\mathcal{X}_1$ and the 0-state in $\mathcal{X}_2$
   $\Rightarrow$ the number of 1-periodic 0-states in $\mathcal{X}$ is 1
2. Combination of the 0-state in $\mathcal{X}_1$ and the ν_2 cycles of τ_2-periodic states in $\mathcal{X}_2$
   $\Rightarrow$ the number of τ_2-periodic states in $\mathcal{X}$ is $\nu_2\tau_2$
3. Combination of the ν_1 cycles of τ_1-periodic states in $\mathcal{X}_1$ and the 0-state in $\mathcal{X}_2$
   $\Rightarrow$ the number of τ_1-periodic states in $\mathcal{X}$ is $\nu_1\tau_1$
4. Combination of the ν_1 cycles of τ_1-periodic states in $\mathcal{X}_1$ and the ν_2 cycles of τ_2-periodic states in $\mathcal{X}_2$
   $\Rightarrow$ the number of $\operatorname{lcm}(\tau_1\tau_2)$-periodic states in $\mathcal{X}$ is $\nu_1\nu_2\tau_1\tau_2$

As a consequence of point 4, the number of cycles

$$\nu_{12}=\frac{\nu_1\nu_2\tau_1\tau_2}{\operatorname{lcm}(\tau_1,\tau_2)}=\nu_1\nu_2\gcd(\tau_1,\tau_2),$$

which comprise the $\operatorname{lcm}(\tau_1,\tau_2)$-periodic states can be determined by calculating the greatest common divisor of τ_1 and τ_2. Hence, we may define a product of cycle terms.

Definition 16 (Product of Cycle Terms)
The product

$$\nu_1[\tau_1]\nu_2[\tau_2]=\nu_1\nu_2\gcd(\tau_1,\tau_2)[\operatorname{lcm}(\tau_1,\tau_2)] \tag{43}$$

is called cycle term product. The expressions $\gcd(\tau_1,\tau_2)$ and $\operatorname{lcm}(\tau_1,\tau_2)$ are greatest common divisor and least common multiple of τ_1, τ_2 respectively.

By means of the denotation of sum and product of cycle terms, from (42) we extend the notion of product to the superposition Σ of the cycle sums Σ_1 and Σ_2

$$\begin{aligned}\Sigma = \Sigma_1\Sigma_2 = (1[1] \dot{+} \nu_1[1])(1[1] \dot{+} \nu_2[2]) = \\ 1[1] \dot{+} \nu_1[\tau_1] \dot{+} \nu_2[\tau_2] \dot{+} \nu_1\nu_2 \mathrm{gcd}(\tau_1,\tau_2)[\mathrm{lcm}(\tau_1,\tau_2)]\end{aligned} \quad (44)$$

The next theorem is an obvious generalization of the latter.

Theorem 8 (Superposition)
The cycle sum Σ superposing N cycle sums $\Sigma_i, i = 1,\ldots,N$ can be calculated distributively by the product

$$\Sigma = \Sigma_1\Sigma_2\cdots\Sigma_N \ . \quad (45)$$

All together we finally have shown

Theorem 9 (Cycle Sum of an Autonomous LMS)
Let the dynamics $\mathbf{A} = \mathrm{diag}(\mathbf{C}_1,\ldots,\mathbf{C}_N) \in \mathbb{F}_2^{n\times n}$ of an autonomous LMS(q) be block diagonally composed of $i = 1,\ldots,N$ cyclic companion matrices $\mathbf{C}_i$, each with respect to one of the $i = 1,\ldots,N$ elementary divisor polynomials $p_{\mathbf{C}_i} \in \mathbb{F}_2[\lambda]$ of degree d_i. Let each elementary divisor polynomial $p_{\mathbf{C}_i}$ be given in fully factorized form $p_{\mathbf{C}_i} = (p_{\mathrm{irr},\mathbf{C}_i})^{e_i}$ subject to its irreducible factor polynomial $p_{\mathrm{irr},\mathbf{C}_i}$ of degree δ_i such that $d_i = e_i\,\delta_i$. Then each elementary divisor polynomial $p_{\mathbf{C}_i}$ contributes the cycle sum

$$\Sigma_i = 1[1] \dot{+} \frac{2^{\delta_i}-1}{\tau_1^{(i)}}\left[\tau_1^{(i)}\right] \dot{+} \frac{2^{2\delta_i}-2^{\delta_i}}{\tau_2^{(i)}}\left[\tau_2^{(i)}\right] \dot{+} \ldots \dot{+} \frac{2^{e_i\delta_i}-2^{(e_i-1)\delta_i}}{\tau_{e_i}^{(i)}}\left[\tau_{e_i}^{(i)}\right] , \quad (46)$$

where $\tau_j^{(i)}$ denotes the period[5] of the polynomial $(p_{\mathrm{irr},\mathbf{C}_i})^j$. The cycle sum Σ of the autonomous LMS follows from Superposition of all cycle sums Σ_i as per $\Sigma = \Sigma_1\Sigma_2\cdots\Sigma_N$.

Remark 4
As already pointed out in Remark 1, a simple consequence of Theorem 9 is that nilpotent elementary divisor polynomials are not related to periodic subspaces.

Subsumingly, the whole cycle sum of a linear modular system over $\mathbb{F}_2$ can be calculated along the following algorithm:

1. Calculate the Smith normal form $\mathbf{S}(\lambda)$ of $\mathbf{A}$ by unimodular left and right transforms on $\lambda\mathbf{I}+\mathbf{A}$ via polynomial matrices $\mathbf{U}(\lambda)$ and $\mathbf{V}(\lambda)$ (alternatively calculate the rational canonical form $\mathbf{A}_{\mathrm{rat}}$ of $\mathbf{A}$).

[5] At least here we are justified to have introduced the same symbol τ for the period of a state although firstly τ was introduced for the period of polynomials in Definition 7.

2. Determine the N elementary divisor polynomials $p_i(\lambda), i = 1, \dots, N$ of $\mathbf{A}$ by factorizing the system invariants in $\mathbf{S}(\lambda)$.
3. Assign the periods $\tau_j^{(i)}$ to each polynomial $p_{i,\mathrm{irr}}^j(\lambda),\ j = 1, \dots, e_i$ with $p_i(\lambda) = p_{i,\mathrm{irr}}^{e_i}(\lambda)$ and $p_{i,\mathrm{irr}}(0) \neq 0$ (due to Remark 4 we omit polynomials $p_i(\lambda) = \lambda^k, k \in \mathbb{N}$).
4. Compute the cycle sum Σ_i for each elementary divisor polynomial $p_i(\lambda)$.
5. The cycle sum Σ of the entire automaton then follows by distributively superposing all cycle sets $\Sigma_i, i = 1, \dots, N$.

4.3. Example

Consider the dynamics matrix $\mathbf{A} \in \mathbb{F}_2^{5\times 5}$ of an LMS with its respective Smith form $\mathbf{S}(\lambda) = \mathbf{U}(\lambda)(\lambda\mathbf{I} + \mathbf{A})\mathbf{V}(\lambda)$ according to

$$\mathbf{A} = \begin{pmatrix} 1\,0\,0\,1\,1 \\ 1\,1\,0\,0\,1 \\ 0\,0\,1\,0\,1 \\ 0\,0\,0\,0\,1 \\ 1\,0\,0\,0\,1 \end{pmatrix}, \quad \mathbf{S}(\lambda) = \begin{pmatrix} (\lambda^2+\lambda+1)(\lambda+1)^2 & 0 & 0\,0\,0 \\ 0 & \lambda+1 & 0\,0\,0 \\ 0 & 0 & 1\,0\,0 \\ 0 & 0 & 0\,1\,0 \\ 0 & 0 & 0\,0\,1 \end{pmatrix}.$$

Then the only invariant polynomials of the matrix $\mathbf{A}$ which are different from 1 are

$$c_1(\lambda) = (\lambda^2+\lambda+1)(\lambda+1)^2, \quad c_2(\lambda) = \lambda+1 .$$

Hence, $\mathbf{A}$ has the elementary divisor polynomials

$$p_1(\lambda) = \lambda^2+\lambda+1, \quad p_2(\lambda) = (\lambda+1)^2, \quad p_3(\lambda) = \lambda+1 ,$$

the irreducible basis polynomial degrees of which are $\delta_1 = 2$, $\delta_2 = 1$ and $\delta_3 = 1$, respectively. In view of Definition 7 and Theorem 3 we calculate the associated periods:

$$\begin{aligned} p_{1,\mathrm{irr}}(\lambda) = p_1(\lambda) | \lambda^3+1 \quad &\Longrightarrow \quad \tau_1^{(1)}{=}3 \\ p_{2,\mathrm{irr}}(\lambda) = \lambda+1 \quad &\Longrightarrow \quad \tau_1^{(2)}{=}1 \\ \big(p_{2,\mathrm{irr}}(\lambda)\big)^2 = (\lambda+1)^2 = \lambda^2+1 \quad &\Longrightarrow \quad \tau_2^{(2)}{=}2 \\ p_{3,\mathrm{irr}}(\lambda) = \lambda+1 \quad &\Longrightarrow \quad \tau_1^{(3)}{=}1 \end{aligned}$$

Theorem 9 yields

$$\Sigma_1 = 1[1] \dotplus 1[3], \quad \Sigma_2 = 2[1] \dotplus 1[2], \quad \Sigma_3 = 2[1]$$

and by superposition according to Theorem 8 and using (15) and (43), we get

$$\begin{aligned} \Sigma = \Sigma_1\Sigma_2\Sigma_3 = (1[1] \dotplus 1[3])(2[1] \dotplus 1[2])(2[1]) = \\ (2[1] \dotplus 1[2] \dotplus 2[3] \dotplus 1[6])(2[1]) = 4[1] \dotplus 2[2] \dotplus 4[3] \dotplus 2[6] . \end{aligned}$$

Therefore, the considered linear automaton given by the dynamics $\mathbf{A}$ comprises 4 cycles of length 1, 2 cycles of length 2, 4 cycles of length 3 and 2 cycles of length 6.

5. PROPERTIES OF LINEAR MODULAR SYSTEMS WITH INPUTS

As the main goal of this paper involves the synthesis of a control for linear discrete systems over $\mathbb{F}_2$ the notion of controllability of an LMS has to be taken into account. For this purpose the well-known solution

$$\mathbf{x}[k] = \mathbf{A}^k \mathbf{x}[0] + \sum_{i=0}^{k-1} \mathbf{A}^{k-1-i} \mathbf{B}\mathbf{u}[i]. \tag{47}$$

of the state equation (23) of an LMS is recalled from [2]. Owing to this result, controllability can be defined for an LMS and a controllability criterion can be specified.

Definition 17 (Controllability)
An n-th order LMS is l-controllable iff for all ordered pairs of states $(\mathbf{x}_1, \mathbf{x}_2)$ the system can be driven from state $\mathbf{x}_1$ to state $\mathbf{x}_2$ in exactly l steps. An LMS is controllable iff it is l-controllable for some l.

Theorem 10 (Controllability Criterion)
An n-th order LMS is l-controllable iff the matrix $(\mathbf{B}\ \mathbf{AB}\ \ldots\ \mathbf{A}^{l-1}\mathbf{B})$ has full rank n.

This theorem will be used for establishing the controllability companion form, which can be determined by applying linear transforms on the state equation.

Using Theorem 10 the reduced controllability matrix $\mathbf{L} \in \mathbb{F}_2^{n\times n}$ of an LMS can be determined by choosing n linearly independent columns from $(\mathbf{B}\ \mathbf{AB}\ \ldots\ \mathbf{A}^{l-1}\mathbf{B})$ with regard to minimal multiples of $\mathbf{A}$, see [17]. This procedure yields the matrix

$$\mathbf{L} = \left(\mathbf{b}_1 \ldots \mathbf{A}^{c_1-1}\mathbf{b}_1\ \mathbf{b}_2 \ldots \mathbf{A}^{c_2-1}\mathbf{b}_2\ \ldots\ \mathbf{b}_m\ \ldots\ \mathbf{A}^{c_m-1}\mathbf{b}_m\right), \tag{48}$$

where the vectors $\mathbf{b}_i$, $i = 1, \ldots, m$, are the respective column vectors of the input matrix $\mathbf{B}$ and the numbers $c_i \in \mathbb{N}$ are the controllability indices with the properties:

- the set of c_i is unique,
- the set of c_i is invariant with respect to similarity transformations,
- $\sum_{i=1}^m c_i = n$,
- the list $\sigma_i := \sum_{j=1}^i c_j$, $i = 1, \ldots, m$, implies a structural system decomposition.

Given a controllable LMS, a characteristic companion form of the state equations (23) can be found using a similarity transformation which employs (48) and the set of controllability indices c_i [17]. It is called the controllability companion form (CCF), marked by the superscript c in the subsequent sections. The system representation in CCF reads

$$\mathbf{x}^c[k+1] = \underbrace{\begin{pmatrix} \mathbf{A}^c_{11} & \cdots & \mathbf{A}^c_{1m} \\ \mathbf{A}^c_{21} & \cdots & \mathbf{A}^c_{2m} \\ \vdots & \ddots & \vdots \\ \mathbf{A}^c_{m1} & \cdots & \mathbf{A}^c_{mm} \end{pmatrix}}_{=:\,\mathbf{A}^c} \mathbf{x}^c[k] + \underbrace{\begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x & x & \cdots & x & x \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & x & \cdots & x & x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix}}_{=:\,\mathbf{B}^c} \mathbf{u}[k], \tag{49}$$

$$\mathbf{A}^c_{ii} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ x & x & x & x & x \end{pmatrix}, \quad \mathbf{A}^c_{ij,i\neq j} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \\ x & x & x & x & x \end{pmatrix},$$

with $\mathbf{A}^c_{ij} \in \mathbb{F}_2^{c_i \times c_j}$. For separating structural and informal properties of the system in CCF the rows with undetermined entries x are collected in the matrices

$$\mathbf{A}^c_\sigma = \begin{pmatrix} \text{row } \sigma_1 \text{ of } \mathbf{A}^c \\ \text{row } \sigma_2 \text{ of } \mathbf{A}^c \\ \vdots \\ \text{row } \sigma_m \text{ of } \mathbf{A}^c \end{pmatrix}, \quad \mathbf{B}^c_\sigma = \begin{pmatrix} \text{row } \sigma_1 \text{ of } \mathbf{B}^c \\ \text{row } \sigma_2 \text{ of } \mathbf{B}^c \\ \vdots \\ \text{row } \sigma_m \text{ of } \mathbf{B}^c \end{pmatrix} = \begin{pmatrix} 1 & x & x & \cdots & x \\ 0 & 1 & x & \cdots & x \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}. \tag{50}$$

These matrices will be needed in Section 6.3. In light of the above-stated definitions we can now describe the objectives of our approach and cite a fundamental theorem which will provide a solution to the synthesis problem.

6. CONTROL SYNTHESIS FOR LINEAR MODULAR SYSTEMS

In the previous sections characteristic structural properties of subspaces of the state space associated to an LMS have been exposed. It was pointed out, particularly, that the analysis is based on the properties of elementary divisors of the system dynamics **A**. In the sequel we propose a synthesis procedure which allows of fitting the given system with desired elementary divisor polynomials and thus imposing a specific cycle sum on an LMS. To this end we will pursue the idea of state feedback.

6.1. State Feedback

Changing the elementary divisor polynomials, which is equivalent to changing the system invariants of an LMS, is closely related to changing the eigenvalues of the system dynamics **A**. From the theory of linear discrete time systems over the field of real numbers $\mathbb{R}$ it is well-known that changing the eigenvalues of $\mathbf{A}$ can be done by introducing a (static) linear state feedback of the form $\mathbf{u}[k] = -\mathbf{K}\mathbf{x}[k] + \mathbf{w}[k]$ with the constant feedback matrix $\mathbf{K} \in \mathbb{F}_2^{m\times n}$ and the reference input vector $\mathbf{w}[k] \in \mathbb{F}_2^m$. Referring to this concept it is intuitive to introduce a linear state feedback

$$\mathbf{u}[k] = \mathbf{K}\mathbf{x}[k] + \mathbf{w}[k] \tag{51}$$

for the control of an LMS as well. This leeds to the closed-loop state representation

$$\mathbf{x}[k{+}1] = (\mathbf{A} + \mathbf{B}\mathbf{K})\,\mathbf{x}[k] + \mathbf{B}\mathbf{w}[k]. \tag{52}$$

The invariants of the dynamics $\mathbf{A} + \mathbf{B}\mathbf{K}$ can be specified by the state feedback $\mathbf{K}$.

6.2. Structural Constraints

The closed-loop system (52) complies with the structure theorem, recalled from [9].

Theorem 11 (Rosenbrock Structure Theorem)
Given an n-th order controllable LMS with controllability indices $c_1 \geq \ldots \geq c_m$ and desired monic invariant polynomials $c_{i,\mathbf{K}} \in \mathbb{F}_2[\lambda]$ with $\deg(c_{1,\mathbf{K}}) \geq \ldots \geq \deg(c_{m,\mathbf{K}})$, $c_{i+1,\mathbf{K}} | c_{i,\mathbf{K}}$, $i = 1, \ldots, m-1$, and $\sum_{i=1}^{m} \deg(c_{i,\mathbf{K}}) = n$. Then a constant matrix $\mathbf{K}$ such that $\mathbf{A} + \mathbf{B}\mathbf{K}$ has the invariant polynomials $c_{i,\mathbf{K}}$ exists iff for $k = 1, 2, \ldots, m$

$$\sum_{i=1}^{k} \deg(c_{i,\mathbf{K}}) \geq \sum_{i=1}^{k} c_i\,. \tag{53}$$

Rosenbrock's structure theorem entails a limit when striving for maximal liberality in specifying invariant polynomials. There are many methods for computing a linear state feedback, mainly by specifying desired eigenvalues in some “time domain”.

Intuitively, pole placing methods seem to be applicable. But specifying invariant polynomials is a stronger requirement than specifying eigenvalues. Consequently, standard pole placing methods can be ruled out. An approach which enables the modification of the eigenstructure of a system is the parametric approach [15]. However, this approach is not capable of serving the requirements for the subsequent reasons.

Remark 5
Synthesis of state feedback via the parametric approach has considerable drawbacks:

- An assumption in this approach is that the open-loop and the closed-loop eigenvalues are distinct, which is a very restrictive assumption in the framework of LMS.
- Assigning multiple eigenvalues, which is indispensable for the realization of rather standard cycle sums (e. g. cycles of even length), turns out to be very cumbersome as the computation of generalized eigenvectors is required.

- As eigenvalues of matrices over a finite field $\mathbb{F}_q$ are roots of a polynomial over a finite field the notion of zeroes is important. These zeroes typically lie in some extension field $\mathbb{F}_q$ that deeply depends on the factors and the degree of the polynomial itself. Moreover, these extension fields have no unique defining element [12]. This is a severe difference to the field of real numbers in which any polynomial in $\mathbb{R}[\lambda]$ can be factorized into quadratic irreducible polynomials over $\mathbb{R}$ (see Definition 6). Hence, any zero of a polynomial in $\mathbb{R}[\lambda]$ lies in the corresponding extension field, which is the field of complex numbers $\mathbb{C}$ with unique defining element $i = \sqrt{-1}$. In general, such a factorization is not possible for polynomials in $\mathbb{F}_q[\lambda]$. Consequently, the computation of eigenvalues in the extension field of $\mathbb{F}_2$ entails enormous symbolical computation effort.
- The structural theorem imposes constraints on realizable invariant polynomials in the closed-loop system. Thus, if the task is to assign invariant polynomials this is much more straight-forward in the frequency domain even for continuous systems.

In view of these issues we will define an image domain for LMS in the next section.

6.3. An Image Domain for LMS

Similar to discrete continuous time systems an image domain can be defined [14].

The $\mathcal{A}$-Transform

Definition 18 ($\mathcal{A}$-Transform)
The $\mathcal{A}$-transform for a causal, discrete function $f : \mathbb{N} \to \mathbb{F}_2$ is

$$F(a) := \mathcal{A}(f[k]) := \sum_{k=0}^{\infty} f[k]\, a^{-k} . \tag{54}$$

Some relevant relations are shown in Figure 3. Applying (54) the state equation (23) can be transformed into the $\mathcal{A}$-domain and as a first outcome the solution of the state equation can be verified.

original domain (function of k)	image domain (function of a)
$\sum_\nu \alpha_\nu f_\nu[k]$	$\sum_\nu \alpha_\nu F_\nu(a)$
$f[k+1]$	$aF(a) + a\, f[0]$

Fig. 3. $\mathcal{A}$-transform for causal functions f[k]

Solution of the State Equation

Referring to Figure 3 the $\mathcal{A}$-transform of (23) is

$$a\,\mathbf{X}(a) = \mathbf{A}\,\mathbf{X}(a) + \mathbf{B}\,\mathbf{U}(a) + a\,\mathbf{x}[0]\,. \tag{55}$$

Parameters in capital letters with argument denote functions in the $\mathcal{A}$-Domain. This representation directly leads to the $\mathcal{A}$-transform of the system state

$$\mathbf{X}(a) = (a\,\mathbf{I} + \mathbf{A})^{-1}(\mathbf{B}\,\mathbf{U}(a) + a\,\mathbf{x}[0])\,, \tag{56}$$

which can readily be used to determine the well-known solution of the difference equation. Therefore, we need the inverse transform of the $\mathcal{A}$-transform.

Definition 19 (Inverse of the $\mathcal{A}$-Transform)
The inverse transform of the $\mathcal{A}$-transform is given by

$$\mathcal{A}^{-1}(\mathbf{F}(a)) := \mathbf{f}[0], \mathbf{f}[1], \mathbf{f}[2], \ldots \tag{57}$$

$$\mathbf{f}[k] := [a^k \mathbf{F}(a)]_{\text{ind}}\,, \tag{58}$$

where the operator $[a^k \mathbf{F}(a)]_{\text{ind}}$ provides the addend of the rational function $a^k \mathbf{F}(a)$ that is independent of a.

Using this definition, the state vector $\mathbf{x}[k]$ in the original domain can be computed. For this purpose, we apply the formula for the geometrical series on the expression $(a\,\mathbf{I} + \mathbf{A})^{-1} = \frac{1}{a}(\mathbf{I} + \frac{\mathbf{A}}{a})^{-1} = \frac{1}{a}\sum_{i=0}^{\infty}(\frac{\mathbf{A}}{a})^i$ and obtain

$$\begin{aligned}
\mathbf{x}[k] &= \Big[a^k\,\mathbf{X}(a)\Big]_{\text{ind}} = \Big[a^k\,(a\,\mathbf{I} + \mathbf{A})^{-1}(\mathbf{B}\,\mathbf{U}(a) + a\,\mathbf{x}[0])\Big]_{\text{ind}} \\
&= \Big[a^k \frac{1}{a}\sum_{i=0}^{\infty}\Big(\frac{\mathbf{A}}{a}\Big)^i\Big(\mathbf{B}\sum_{j=0}^{\infty}\mathbf{u}[j]\,a^{-j} + a\,\mathbf{x}[0]\Big)\Big]_{\text{ind}} \\
&= \Big[\Big(\sum_{i=0}^{\infty}\mathbf{A}^i\,a^{k-i}\Big)\,\mathbf{x}[0]\Big]_{\text{ind}} + \Big[\Big(\sum_{i=0}^{\infty}\mathbf{A}^i\,a^{k-i-1}\Big)\,\mathbf{B}\,\Big(\sum_{j=0}^{\infty}\mathbf{u}[j]\,a^{-j}\Big)\Big]_{\text{ind}} \\
&= \Big[\sum_{i=0}^{\infty}\mathbf{A}^i\,a^{k-i}\,\mathbf{x}[0]\Big]_{\text{ind}} + \sum_{i=0}^{\infty}\mathbf{A}^i\,\mathbf{B}\,\mathbf{u}[k-i-1] \\
&= \mathbf{A}^k\,\mathbf{x}[0] + \mathbf{B}\,\mathbf{u}[k-1] + \mathbf{A}\,\mathbf{B}\,\mathbf{u}[k-2] + \ldots + \mathbf{A}^{k-1}\,\mathbf{B}\,\mathbf{u}[0],
\end{aligned}$$

finally having used causality. The last expression equals (47). However, the system representation in the $\mathcal{A}$-domain cannot only be used to solve the state equation. Its most important feature is reflected in the context of assigning the cyclic properties of the system.

Transfer Matrix

In view of (56) we can define $\mathbf{F}(a)$ in

$$\mathbf{X}(a)\Big|_{\mathbf{x}[0]=0} = \mathbf{F}(a)\,\mathbf{U}(a) = (a\,\mathbf{I}+\mathbf{A})^{-1}\mathbf{B}\,\mathbf{U}(a)\,. \tag{59}$$

as the system transfer matrix. It is obvious that the cyclic properties of the system are contained in $\mathbf{F}(a)$ as the properties of a periodic system state are described by the expression $(a\,\mathbf{I}+\mathbf{A})$, or by $(a\,\mathbf{I}+\mathbf{A})^{-1}$, alternatively. In the next sections we will compute a linear state feedback using (59) by employing the polynomial matrix approach.

Polynomial Matrix Fraction of the Transfer Matrix

For a better understanding, the most important notions and concepts which evolve in the polynomial matrix approach have to be recalled [17, 11].

Definition 20 (Polynomial Matrix Fraction)
A right (left) polynomial matrix fraction RPMF (LPMF) of a rational matrix $\mathbf{R}(a)$ is an expression of the following form

$$\mathbf{R}(a) = \mathbf{N}(a)\,\mathbf{D}^{-1}(a) \quad \big(\mathbf{R}(a) = \mathbf{D}^{-1}(a)\,\mathbf{N}(a)\big) \tag{60}$$

with the polynomial matrices denominator matrix $\mathbf{D}(a)$ and numerator matrix $\mathbf{N}(a)$.

By means of this definition the following theorem can be stated.

Theorem 12 (Conservation)
The product of an arbitrary polynomial matrix $\mathbf{R}(a)$ and an arbitrary unimodular polynomial matrix $\mathbf{U}(a)$ has the same invariant polynomials as $\mathbf{R}(a)$.

Due to the fact that the transfer matrix in (59) is a rational matrix, known results on rational matrices can be utilized.

Theorem 13 (Existence)
For any rational matrix there $\mathbf{R}(a)$ exists a right (left)-prime polynomial matrix fraction representation.

Theorem 14 (Invariant Polynomials)
Let $\mathbf{R}(a)$ be a rational matrix. Then

- the numerator matrices of arbitrary right- or left-prime polynomial matrix fractions of $\mathbf{R}(a)$ have the same invariant polynomials and
- the denominator matrices of arbitrary right- or left-prime polynomial matrix fractions of $\mathbf{R}(a)$ have the same invariant polynomials.

As the transfer matrix representation in (59) is a left-prime polynomial matrix fraction, Theorem 14 reveals that the invariant polynomials of the denominator matrix of each polynomial matrix fraction of $\mathbf{F}(a)$ are equal to the invariant polynomials of the system dynamics $\mathbf{A}$. For a system in CCF (see equations (49) and (50)) an analytic expression for a right-prime polynomial matrix fraction can be determined as [17]

$$\mathbf{F}(a) = \mathbf{Q}(a)\left((\mathbf{B}^c_\sigma)^{-1}(\boldsymbol{\gamma}(a) + \mathbf{A}^c_\sigma \mathbf{Q}(a))\right)^{-1}, \tag{61}$$

where the matrices $\mathbf{Q}(a)$ and $\boldsymbol{\gamma}(a)$ show the structure

$$\mathbf{Q}(a) = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ a & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a^{c_1-1} & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a^{c_2-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a^{c_m-1} \end{pmatrix}, \quad \boldsymbol{\gamma}(a) = \begin{pmatrix} a^{c_1} & 0 & \cdots & 0 \\ 0 & a^{c_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a^{c_m} \end{pmatrix}. \tag{62}$$

This means that if the LMS is given in CCF, it is straight-forward to find an expression for the polynomial matrix fraction of the system transfer matrix (59). Similarly, for the closed-system (52) we have the RPMF

$$\mathbf{F}(a) = \mathbf{Q}(a)\left(\underbrace{(\mathbf{B}^c_\sigma)^{-1}(\boldsymbol{\gamma}(a) + \overbrace{(\mathbf{A}^c_\sigma + \mathbf{B}^c_\sigma \mathbf{K}^c)}^{\mathbf{A}^c_{\sigma,\mathbf{K}}}\,\mathbf{Q}(a))}_{\mathbf{D}_\mathbf{K}(a)}\right)^{-1} \tag{63}$$

with the following properties:

- The numerator matrix $\mathbf{Q}(a)$ of the RPMF is left unchanged by linear state feedback.
- The denominator matrix $\mathbf{D}_\mathbf{K}(a)$ and the corresponding closed-loop system dynamics $\mathbf{A}+\mathbf{BK}$ have the same invariant polynomials.
- The controllability indices equal the column degrees[6] of the denominator matrix.

Since the feedback matrix $\mathbf{K}$ can be uniquely determined if $\mathbf{D}_\mathbf{K}(a)$ in (63) is known, finding an adequate state feedback for fitting a closed-loop LMS with desired invariant polynomials amounts to determine a denominator matrix $\mathbf{D}_\mathbf{K}(a)$ with the properties:

1. The invariant polynomials of $\mathbf{D}_\mathbf{K}(a)$ coincide with the desired polynomials $c_{i,\mathbf{K}}(a)$.

[6] This is the highest polynomial degree in the corresponding column.

2. The column degrees of $\mathbf{D_K}(a)$ equal the controllability indices c_i of the LMS.[7]

With $\mathbf{D_K}(a) = (\mathbf{B}^c_\sigma)^{-1}\mathbf{D}^*_\mathbf{K}(a)$ it suffices to consider $\mathbf{D}^*_\mathbf{K}(a)$ as $(\mathbf{B}^c_\sigma)^{-1}$ is unimodular and, thus by Theorem 12, $\mathbf{D}^*_\mathbf{K}(a)$ has the same invariant polynomials as $\mathbf{D_K}(a)$.

6.4. Main Theorem

With the results from the previous sections the main theorem for the synthesis of a linear state feedback can be stated.

Theorem 15 (Synthesis Algorithm)
Let a controllable LMS be given in CCF, let c_i, $i = 1,\dots,m$ be its controllability indices, let $c_{i,\mathbf{K}} \in \mathbb{F}_2[a]$, $i = 1,\dots,m$, be desired invariant polynomials and let $\mathbf{D}^*(a) = \mathrm{diag}(c_{i,\mathbf{K}}(a))$, $i = 1,\dots,m$ with $\deg(c_{1,\mathbf{K}}) \geq \dots \geq \deg(c_{m,\mathbf{K}})$ and $\sum_{i=1}^m \deg(c_{i,\mathbf{K}}) = \sum_{i=1}^m c_i = n$. The following algorithm is given:[8]

1. Check the structural theorem for c_i and $c_{i,\mathbf{K}}(a)$. If (53) is fulfilled **go to** step 2, else such a state feedback matrix **K** does not exist.
2. Examine $\mathbf{D}^*(a)$.
   - **if** the column degrees of $\mathbf{D}^*(a)$ equal the ordered list of controllability indices **go to** step 5.
   - **else** detect the first column of $\mathbf{D}^*(a)$ which differs from the ordered list of controllability indices, starting with column 1. Denote this column col_u. ($\deg(col_u) > c_u$).
   - Do the same beginning with column m. Denote the specified column col_d. ($\deg(col_d) < c_d$).
3. Adapt the column degrees of $\mathbf{D}^*(a)$ by unimodular transformations.
   - Multiply row_d with a and add the result to $row_u \Rightarrow \mathbf{D}^*(a) \rightarrow \mathbf{D}^+(a)$.
   - **if** $\deg(col_u^+) = \deg(col_u) - 1$
     - $\mathbf{D}^+(a) \rightarrow \mathbf{D}^{++}(a)$ and **go to** step 4.
   - **else**
     - define: $r := \deg(col_u) - \deg(col_d) - 1$
     - multiply col_u^+ with a^r and subtract the result from col_d^+. $\Rightarrow \mathbf{D}^+(a) \rightarrow \mathbf{D}^{++}(a)$

[7] Controllability indices c_i do not change by linear state feedback.

[8] For abbreviation, the i-th matrix columns and rows are denoted by col_i and row_i, $i = 1,\dots,m$, respectively.

4. Generate the column pointer matrix[9] $\boldsymbol{\Gamma}^{++}$ of $\mathbf{D}^{++}(a) \Rightarrow \mathbf{D}^{*}(a) = (\boldsymbol{\Gamma}^{++})^{-1} \cdot \mathbf{D}^{++}(a)$ and **go to** step 2
5. $\mathbf{D}^{*}_{\mathbf{K}}(a) := \mathbf{D}^{*}(a)$ and **return** $\mathbf{D}^{*}_{\mathbf{K}}(a)$

If the conditions from above are fulfilled, and $\mathbf{D}^{*}_{\mathbf{K}}(a)$ is returned by the algorithm, then $\mathbf{D}^{*}_{\mathbf{K}}(a)$ can be generated by linear state feedback $\mathbf{K}$.

In Section 6.3 we stated that if $\mathbf{D}_{\mathbf{K}}(a)$ is known then it is straightforward to compute the state feedback matrix $\mathbf{K}$. To illustrate this, consider

$$\begin{aligned}\mathbf{D}_{\mathbf{K}}(a) &= (\mathbf{B}^{c}_{\sigma})^{-1}\mathbf{D}^{*}_{\mathbf{K}}(a)\\ &= (\mathbf{B}^{c}_{\sigma})^{-1}(\boldsymbol{\gamma}(a) + \mathbf{A}^{c}_{\sigma,\mathbf{K}}\mathbf{Q}(a))\end{aligned}$$

which leads to

$$\mathbf{A}^{c}_{\sigma,\mathbf{K}}\mathbf{Q}(a) = \boldsymbol{\gamma}(a) + \mathbf{B}^{c}_{\sigma}\mathbf{D}_{\mathbf{K}}(a) \tag{64}$$

and by comparison of coefficients the matrix

$$\mathbf{A}^{c}_{\sigma,\mathbf{K}} = \mathbf{A}^{c}_{\sigma} + \mathbf{B}^{c}_{\sigma}\mathbf{K}^{c} \tag{65}$$

can be determined, which directly provides $\mathbf{K}^{c} = (\mathbf{B}^{c}_{\sigma})^{-1}(\mathbf{A}^{c}_{\sigma,\mathbf{K}} + \mathbf{A}^{c}_{\sigma})$ and thus a feedback matrix $\mathbf{K}^{c}$ has been derived which fits the given LMS with the desired closed-loop invariant polynomials $c_{i,\mathbf{K}}$, $i = 1,\ldots,m$. Note that, in general, the solution for $\mathbf{D}^{*}_{\mathbf{K}}(a)$ is not unique.

6.5. Example

In this section we want to illustrate the latter notions by the following state equations

$$\mathbf{x}[k+1] = \begin{pmatrix} 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1\\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \mathbf{x}[k] + \begin{pmatrix} 0 & 0\\ 0 & 0\\ 1 & 0\\ 0 & 0\\ 0 & 1 \end{pmatrix} \mathbf{u}[k]\,.$$

[9] The column pointer matrix is a matrix with elements in $\mathbb{F}_2$ consisting of the coefficients of the greatest degree monomials in each column of $\mathbf{D}^{++}(a)$.

Obviously, this LMS over $\mathbb{F}_2$ is already represented in controllability companion form and we can determine the characteristic matrices $\mathbf{A}^c$, $\mathbf{B}^c$, $\mathbf{A}^c_\sigma$ and $\mathbf{B}^c_\sigma$, which are

$$\mathbf{A}^c = \begin{pmatrix} 0\,1\,0\,0\,0 \\ 0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1 \\ 1\,0\,0\,1\,0 \end{pmatrix}, \; \mathbf{B}^c = \begin{pmatrix} 0\,0 \\ 0\,0 \\ 1\,0 \\ 0\,0 \\ 0\,1 \end{pmatrix} \quad \Longrightarrow \quad \begin{aligned} \mathbf{A}^c_\sigma &= \begin{pmatrix} 0\,0\,0\,0\,0 \\ 1\,0\,0\,1\,0 \end{pmatrix} \\ \mathbf{B}^c_\sigma &= \begin{pmatrix} 1\,0 \\ 0\,1 \end{pmatrix} \end{aligned}$$

The controllability indices of the system from above are $c_1 = 3$ and $c_2 = 2$. For synthesis we want the controlled system to have the invariant polynomials which have been determined in the example of Section 4.3, being $c_{1,\mathbf{K}}(a) = (a^2+a+1)(a+1)^2$ and $c_{2,\mathbf{K}}(a) = a+1$. So the controlled system will have 4 cycles of length 1, 2 cycles of length 2, 4 cycles of length 3 and 2 cycles of length 6.

For computing an appropriate state feedback we now use the algorithm proposed in Theorem 15:

$$\xrightarrow{1} \quad \begin{aligned} &\sum_{i=1}^{1} \deg(c_{i,\mathbf{K}}(a)) = 4 \geq \sum_{i=1}^{1} c_i = 3 \quad \surd \\ &\sum_{i=1}^{2} \deg(c_{i,\mathbf{K}}(a)) = 5 \geq \sum_{i=1}^{2} c_i = 5 \quad \surd \end{aligned}$$

$$\xrightarrow{2} \quad \mathbf{D}^*(a) = \begin{pmatrix} a^4+a^3+a+1 & 0 \\ 0 & a+1 \end{pmatrix}$$

$$\xrightarrow{3} \quad \mathbf{D}^+(a) = \begin{pmatrix} a^4+a^3+a+1 & a^2+a \\ 0 & a+1 \end{pmatrix} \rightarrow \mathbf{D}^{++}(a) = \begin{pmatrix} a+1 & a^2+a \\ a^3+a^2 & a+1 \end{pmatrix}$$

$$\xrightarrow{4} \quad \mathbf{\Gamma}^{++}(a) = \begin{pmatrix} 0\,1 \\ 1\,0 \end{pmatrix} \rightarrow \mathbf{D}^*(a) = \begin{pmatrix} a^3+a^2 & a+1 \\ a+1 & a^2+a \end{pmatrix}$$

$$\xrightarrow{2,5} \quad \mathbf{D}^*_{\mathbf{K}}(a) = \begin{pmatrix} a^3+a^2 & a+1 \\ a+1 & a^2+a \end{pmatrix}$$

Now $\mathbf{K}^c$ can be computed. With (64) we have

$$\mathbf{A}^c_{\sigma,\mathbf{K}} \begin{pmatrix} 1 & 0 \\ a & 0 \\ a^2 & 0 \\ 0 & 1 \\ 0 & a \end{pmatrix} = \underbrace{\begin{pmatrix} a^3 & 0 \\ 0 & a^2 \end{pmatrix}}_{\boldsymbol{\gamma}(a)} + \underbrace{\begin{pmatrix} 1\,0 \\ 0\,1 \end{pmatrix}}_{\mathbf{B}^c_\sigma} \underbrace{\begin{pmatrix} a^3+a^2 & a+1 \\ a+1 & a^2+a \end{pmatrix}}_{\mathbf{D}^*_{\mathbf{K}}(a)} = \begin{pmatrix} a^2 & a+1 \\ a+1 & a \end{pmatrix}$$

and with (65) the feedback matrix $\mathbf{K}^c$, which fits the given system with the desired invariant polynomials reads

$$\mathbf{K}^c = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{-1}}_{\mathbf{B}^c_\sigma} \left(\underbrace{\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}_{\sigma,\mathbf{K}^c}} + \underbrace{\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix}}_{\mathbf{A}^c_\sigma} \right) = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix} .$$

7. CONCLUSIONS

An algebraic model over the Galois-Field $\mathbb{F}_2$ has been derived for finite state automata. Starting from an exemplary prospect by referring to a graphical representation of a non-deterministic example automaton we have presented a coding scheme for computing a transition relation, similar to a state space model in the continuous world. Two ways for constructing this model have been pointed out: the first method invokes the calculation of the disjunctive normal form, elimination of the negations and using the law of DeMorgan. The second method is based on Reed-Muller generator matrices, which proof to be tailored for the problem, implying much less elaborate computations for determining the coefficients of the transition function in view. In the general prospect, both methods yield a scalar implicit polynomial transition relation over the finite field $\mathbb{F}_2$. In order to examine the power of the model, linear modular systems have been concerned. For these systems we have deduced a necessary and sufficient criterion for determining all automaton cycles in length and number. For the application of this criterion, periods of particular invariant polynomials, i. e. the elementary divisor polynomials of the system dynamics, have to be calculated. The latter is, using computer algebra systems like Maple or Mathematica, a rather easy task to perform. Since these invariant polynomials of the system dynamics fully determine the cyclic properties of an LMS we have referred to the notion of feedback, which is known to be an adequate means for specifying the invariants in the closed-loop system. To this end, we have obtained a structured representation of the given system by first introducing the controllability companion form and then deriving the polynomial matrix fraction of the system transfer function after defining an image domain for finite fields. Based on the Rosenbrock structure theorem we have presented an algorithm which decides if a linear feedback exists that fits the system with desired invariant polynomials and, if the decision is positive, computes an appropriate feedback matrix. Further research will keep track of the computation of the cyclic state vectors and of the nonlinear case, since almost all practically important cases are multilinear. For these systems, exact methods for solving nonlinear systems of equations, for instance employing Gröbner-bases [5], have to be taken into account.

ACKNOWLEDGEMENTS

The afore-presented research was partially supported under scholarship granted by the "Studienstiftung des deutschen Volkes" and by the German Research Council "Deutsche Forschungsgemeinschaft (DFG)" under Grant No. RO 2262/3-1.

REFERENCES

1. D. Bochmann and C. Posthoff, Binäre Dynamische Systeme, Oldenbourg, Munich, 1981.
2. M. Cohn, Controllability in Linear Sequential Networks, IEEE Transactions on Circuit Theory 9, (1962) 74–78
3. D. Franke, Modelling Nondeterministic Discrete-Event Behaviour by Descriptor Systems, in: Proc. 3rd MATHMOD, Vienna, 2000.
4. D. Franke, Sequentielle Systeme, Binäre und Fuzzy Automatisierung mit arithmetischen Polynomen, Vieweg, Braunschweig, 1994.
5. R. Germundsson, Symbolic Systems — Theory, Computation and Applications, Linköping, 1995.
6. A. Gill, Graphs of Affine Transformations, with Applications to Sequential Circuits, in: Proc. 7th IEEE International Symposium on Switching and Automata Theory, Berkeley (1966) 127–135.
7. A. Gill, Linear modular systems, in: L. A. Zadeh, E. Polak, System Theory, McGraw-Hill, New York, 1969.
8. D. Hankerson et al., Coding Theory and Cryptography — The Essentials, Marcel Dekker Inc., New York, 2000.
9. T. Kailath, Linear Systems, Prentice-Hall, Englewood Cliffs, 1980
10. U. Konigorski, Modeling of Linear Systems and Finite Deterministic Automata by means of Walsh Functions, in: Proc. 3rd MATHMOD, Vienna, 2000.
11. V. Kučera, Analysis and Design of Discrete Linear Control Systems, Prentice-Hall, Cambridge, 1991.
12. R. Lidl and H. Niederreiter, Introduction to Finite Fields and their Application, Cambridge Univ. Press, New York, 1994.
13. J. Reger, Cycle Analysis for Deterministic Finite State Automata, in: Proc. 15th IFAC World Congress, Barcelona, 2002.
14. J. Richalet, Operational Calculus for Finite Rings, IEEE Transactions on Circuits and Systems, 12, (1965) 558–570.
15. G. Roppenecker, On parametric state feedback design, International Journal of Control, 43, (1986) 793–804.
16. A. Thayse, Boolean Calculus of Differences, Lecture Notes in Computer Science, Vol. 101, Springer, 1981.
17. W. A. Wolovich, Linear Multivariable Systems, Springer, New York, 1974.