Control
and Cybernetics

VOL. 9 (1980) No. 1–2

Pattern recognition properties of multilinear response functions. Part I

by

R. E. KALMAN*

Stanford University
Stanford, CA, USA
and
Centre d'Automatique, Ecole National Supérieure des Mines, Paris, France

The theory of linear (discrete-time) dynamical systems has been definitively formulated from the algebraic viewpoint in the author's fundamental contribution [3]. The results obtained make it possible to identify the "state" of a dynamical system with the "memory" which the system possesses about a given input. Hence the "state" is a special feature of an input "pattern" which may be identified by the system.

This paper is concerned with extending these results to the theory of nonlinear systems.

Basic elements of the algebraic theory of multilinear systems are given. They indicate a great complexity of such systems. Some consequences concerning nonlinear filtering, pattern recognition, and nonlinear Fourier analysis are briefly discussed.

Preface

This paper was written in May–August 1968. It was first presented publicly in September, 1968 at the IFAC International Symposium on Technical and Biological Problems of Control, held at Erevan, Armenian SSR, USSR, and was subsequently published in Russian in the proceedings of this symposium. The precise reference is:

"Raspoznavanie obrazov polilineinymi mashinami", in *Trudy Mezhdunarodnogo Simposiuma po Tekhnicheskim i Biologicheskim Problemam Upravleniya,* vol. B, pages 7–29, Izdatel'stvo "Nauka", Moskva, 1971.

For several years, the problem of realization of multilinear systems stood essentially as outlined in this paper. In the last 5–8 years, however, attempts have

---

*) Now (1979) at Center for Mathematical System Theory, University of Florida, Gainesville, Fl. 32611 and Mathematische Systemtheorie, Eidgenössiche Technische Hochschule, CH-8092 Zürich, Switzerland.

been made to improve the theory. Unfortunately, some of these attempts have smuggled certain misconceptions into the literature. The situation is at present confused. It was therefore thought useful to publish here, for the first time in a good English translation, the original 1968 paper.

Aside from correcting minor misunderstandings and printing errors, the present version faithfully renders the Russian text into modern idiomatic English. For the benefit of the Western reader, we have added some comments and references, indicated by [ ] or footnotes 1, 2, ... Footnotes referenced as I, II, ... are from the original version.

The difficulties raised by the examples in this paper are not yet fully resolved. The reader may find the elementary exposition contained here easier to follow than some of the more advanced material which is now being published. In the perspective of ten years many of the difficulties, so mysterious in 1968, now appear in much sharper focus.

In possession of the original manuscript of this paper and after its public presentation, in December 1968 Arbib [13] has given an interesting reexposition of some of the results of this paper, without however introducing essentially new ideas. The reader may find that his account throws further light on the some of the problems discussed below.

1. Introduction

We give here a progress report about recent research in algebraic system theory. This field (which is quite new) emphasizes the application of modern mathematics and the tools of abstract algebra to the formulation, study, and solution of specific problems arising in dynamical systems [written in 1968]. It would not be wrong to call our field also *applied algebra*, emphasizing thereby the fact that it deals with a new aspect (application) of an old body of knowledge (algebra). For general background information, see [1; especially Parts III and IV] and [2].

The theory of linear (discrete-time) dynamical systems has been completely settled from the algebraic point of view by the writer's fundamental theorem [3] that the state space of such a system admits the structure of a finitely generated module over the ring $K[z]$ of polynomials in one variable z with coefficients in a fixed number field K. This result allows us, in particular, to characterize the states of a linear system in an intrinsic way using the language of polynomials. Further interesting implications arise in model building, even in theoretical biology [4–5], which are recalled in the next section. Briefly, the picture which has emerged is the following. The "state" of a dynamical system may be identified with the "memory" that the system retains of a particular input; in other words, the "state" is that special feature of an input "pattern" which the system is capable of "recognizing". While the intuitive description of a state in terms of an input may be very complicated (this should serve as a warning to modelmakers!), in the *linear* (and cyclic) case we have a very simple characterization: the state is equivalent to the remainder resulting from dividing the input polynomial by the characteristic polynomial of the system.

It is natural to attempt to extend this investigation to the theory of *multilinear systems*. These systems have been studied quite extensively already, as part of Norbert Wiener's attempts [6] to analyze nonlinear systems by generalized polynomial (Volterra–Lichtenstein) expansions. See especially [7–10]. One may state categorically, however, that neither Wiener nor his followers have been able to obtain any result of basic mathematical, system-theoretic, engineering, or practical significance. Multilinear systems appear as a subproblem of this area, since each term in a Volterra expansion of a nonlinear response function induces the response function of a multilinear system.

In 1966 the writer (in collaboration with U. Passy) began a comprehensive attack on this subproblem of Wiener's general nonlinear problem. The first results, reported here, give some insight into the extreme complexity of the situation. This complexity is undoubtedly at least in part responsible for the failure of earlier investigations.

The main features of the algebraic theory of multilinear systems are the following:

(i) The state space of a multilinear system is most appropriately characterized as a variety (=algebraic manifold) embedded in a linear space of rather high dimension.

(ii) This variety turns out to be abelian and admits the structure of a $K[z]$-module, thereby generalizing the fundamental theorem of linear system theory.

(iii) The ideal of the state variety represents intrinsic interconnections within the system in a way which is vaguely analogous to the role of the minimal polynomial in representing the dynamics of the system.

(iv) It is possible to give an explicit characterization of the states as equivalence classes of inputs, but this characterization, even in its abstract algebraic form, is far more complicated than in the linear case.

(v) The state variety may be explicitly computed.

Other questions, such as the theory of minimal realizations, are yet to be studied in depth. But even the preceding results are sufficient to give a sharp picture of the complexity of multilinear systems. Some consequences in regard to nonlinear filtering, pattern recognition, and nonlinear Fourier analysis are briefly mentioned in Sec. 6.

Many of the topics studied here first arose in conversations with Dr. U. Passy (now at Technion, ISRAEL) and Dr. W. R. Nico (Tulane, USA). While all ideas presented here are the original work of the author, these early conversations in the first half of 1967 had a stimulating effect on the final formalism. The essential results were obtained in May 1968, in Paris. The support of Ecole des Mines de Paris during this period is gratefully acknowledged.

2. Review of the Algebraic Theory of Linear Systems

The following discussion unavoidably requires a fairly advanced level of mathematical development. We assume, therefore, that the reader is familiar with the algebraic theory of linear systems via modules (see [1, Chapter 10]) and that he is also acquainted with the rudiments of abstract algebra and algebraic geometry. On the other hand, we shall try to emphasize intuitive and conceptual aspects of our topic so that he should not be unduly concerned about the precise understanding of mathematical technicalities. [The treatment of these "technicalities" is certainly incomplete in this paper.]

A *dynamical system* may be described in (at least) two fundamentally different ways.

We could specify its complete internal structure, saying exactly how all its components are interconnected, how they interact, and how they behave individually. We call this the *internal* or *state-variable* or *axiomatic* definition of a system and denote it usually by Σ. A system defined in this way is sometimes also called a *machine*, to emphasize the analogy with computers, complex control systems, etc.

We could also describe a system externally, emphasizing the aggregate behavior of all of its components and specifying exactly what the response of the system is to each of a large class of stimuli, but without saying anything about its internal properties. We call this the *external* or *input/output* or *empirical* definition of a system and usually denote it by f. The problems of experimental science, especially those of biology (whence the terminology of stimulus/response relationships) are congenial to this point of view. In many cases we are interested in constructing Σ from knowledge of a given f. We call this the *problem of realization.*

To specify a system, we must agree on the kind of numbers with which we shall write equations. In applied physics, it is customary to use real numbers R. For us, this would be an unnecessary (and perhaps misleading) restriction; it is best to assume that the numbers belong to an *arbitrary* (number) *field* K.

We must also specify a time set (values of time at which the behavior of the system is defined). In this paper the time set will always be the integers; we shall deal only with *discrete-time* systems.

The *input/output map f of a constant linear system* (external description!) is a K-linear map

$$f: K^m[z] \to zK^p[[z^{-1}]]: \omega \mapsto \sum_{j=1}^{\infty} f(z^{j-1}\omega))_j z^{-j}$$

such that $f(z\omega) = zf(\omega)$.

Notation: K^m is the m-dimensional vector space over K (that is, the vector space of all m-tuples with coefficients from K), $K^m[z]$ = polynominals in z with coefficients in K^m, $K^p[[z^{-1}]]$ = formal power series in z^{-1} with coefficients in K^p.

Interpretation: an element ω of $K^m[z]$ (sometimes called *input space* and denoted by Ω) is a *finite* sequence of vectors in K^m which arrive at the m input terminals of our system before or at $t=0$; the coefficient of the term z^k corresponds to the m

input symbols ($\in K$) applied at $t=-k$ (note minus sign); $f(\omega)=\gamma$ is an element of $K^p\,[[z^{-1}]]$, that is, a formal power series, whose coefficients correspond to the output of the system after the application of the input. (There is an implicit assumption in this setup that the input is identically zero for $t>0$.) In particular, the coefficient of the j-th term in $f(\omega)$, denoted as $(f(\omega))_j$, corresponds to the p symbols ($\in K$) appearing the p output terminals of the system at time $t=j$. (Again, z is a time-marker in that the coefficient of z^{-j} gives the output at $t=j$.) *As a special rule in defining f, we assume that $(f(\omega))_0$ is identically zero*[1]. Thus $(f(\omega))_1$ is the first observed output, appearing at $t=1$, in response to all the inputs fed into the system at all $t\leqslant 0$. The use of z as a time-marker and the division of the time set into disjoint subsets, one ($t\leqslant 0$) allotted to the inputs and the other ($t>0$) to the outputs, is a very convenient metamathematical convention. It does not restrict the generality of our setup and corresponds to the usual classical framework of linear system theory. Notice that f does not give us *directly* the output values of the system for $t\leqslant 0$, but these values can be compuıed indirectly whenever needed.

The role of algebra is a consequence of a number of fundamental theorems (see [1, Chapter 10] for more detail).

Let E be the equivalence relation defined on $K^m\,[z]$ by

$$\omega E\hat{\omega} \quad \text{iff} \quad f(\omega)=f(\hat{\omega}).$$

This means that two inputs ω, $\hat{\omega}$ are *E-equivalent* if and only if (they both vanish for $t>0$ and then) they produce the same output sequence after $t=0$.

$$(f(\omega))_1=(f(\hat{\omega}))_1, \quad (f(\omega))_2=(f(\hat{\omega}))_2, \ldots .$$

Let us recall that that two inputs ω, $\hat{\omega}$ are *Nerode equivalent* (see [1]) if and only if (they may differ up to $t=t_1$ and are arbitrary but equal for all $t>t_1$) they produce exactly the same output sequence after $t=t_1$. The Nerode-equivalence classes in Ω associated with a given input/output map f (which may be even nonlinear, in general) are the precise notion of state when the system is defined in external form.

It is clear that Nerode equivalence implies E equivalence. In the linear case, the converse is also true and we get the fundamental

REPRESENTATION THEOREM. *Two inputs ω and $\hat{\omega}$ to a linear system are Nerode equivalent if and only if they are E equivalent.*

We denote the E-equivalence class of ω by $[\omega]$. As a result of the representation theorem, we can call the set of E-equivalence classes

$$X_f=\{[\omega]:\ \omega\in K^m\,[z]\}$$

the *state set* of f. Moreover, X_f is also a K-vector space. Since knowing the state of our system is equivalent to specifying a point in the K-vector space X_f, it follows

[1]) Some authors do not require this. In any case, the value of $(f(\omega))_0$ is of no interest as far as the construction of the theory is concerned.

that the state can be represented by finitely many numbers (each in K) if and only if X_f is finite dimensional. Since each number in X_f must be stored in the machine Σ_f which realizes f, we are led to the second fundamental result:

FINITE REALIZATION CRITERION. *A linear input/output map f has a finite-dimensional realization if and only if the K-vector space X_f is finite dimensional.*

Necessity is clear. To prove sufficiency, one makes use of the third result, which we call

FUNDAMENTAL THEOREM OF LINEAR SYSTEM THEORY. *The natural state space X_f of a linear input/output map f admits the structure of a $K[z]$-module.*

To define a $K[z]$-module over a set X, it suffices to (i) make X into a K-vector space (here this is trivial) and (ii) define a scalar product $z \cdot x$. Step (ii) is possible because of the "shift" property $f(z\omega) = zf(\omega)$. The only correct definition of the scalar product is the obvious one:

$$z \cdot [\omega] := [z\omega].$$

Because of the possibility of multiplying by z, we can express the internal dynamics of the system very succinctly. Let $\omega \in K^m[z]$ be the input to the system initially and $p \in K^m$ the next input symbol. Then the state transformation induced by p is clearly given by

$$[\omega] \mapsto [z\omega + p] = z \cdot [\omega] + [p].$$

We call $p \mapsto [p]$ the read-in map and $z\colon [\omega] \mapsto z \cdot [\omega]$ the *free state transition* map of the realization Σ_f of f. Read: The state at $t = 0$ is the equivalence class $[\omega]$ of the input ω. The state at $t = 1$ is the equivalence class of the same input followed by the symbol p but thought to have been applied 1 unit of time earlier (stationarity!), that is, the equivalence class $[z\omega + p]$. To put it differently, viewing z as a time marker means that z defines a map

$$z\colon t \mapsto t + 1;$$

while viewing z as an operator acting via scalar multiplication means that z defines a map

$$z\colon \text{present state} \xmapsto{(0 \text{ input})} \text{next state}.$$

So the second operation, essential for dynamics, is a direct *algebraic* consequence of the first operation, the shift in time. With the aid of the module structure on X_f, the proof of the sufficiency of the finiteness criterion becomes an abstract triviality. We shall give the essential details of the construction of a realization of a multilinear input/output map in Section 4.

To give an intuitive description of the importance of the module concept for linear systems, it is best to consider the special case $m = p = 1$ (the general case turns out to be only slightly more general!). Then, if X_f is finite-dimensional, the homomorphism theorem shows that

$$X_f \approx \{\text{equivalence class of polynomials in } K[z] \text{ modulo a fixed polynomial } \chi_f\}.$$

The polynomial χ_f is a well-known and classical object: it is called the *characteristic polynomial* of f. Let π (ordinary polynomial in z) be any input signal in $K[z]=\Omega$. Our system Σ_f "recognizes" π by "storing" it as the state $[\pi]$. This means, according to the above characterization of X_f, that we may think of the stored data as any polynomial $\tilde{\pi}$ such that $\pi-\tilde{\pi}=0$ modulo χ_f (that is, $\pi-\tilde{\pi}$ is divisible by χ_f). The simplest choice for $\tilde{\pi}$ is to divide π by χ_f and then designate the remainder as $\tilde{\pi}$. Since the degree of the remainder is less than n, the degree of χ_f, we see that

If $m=p=1$, the elements of the state space X_f of f are polynomials in $K[z]$ of degree less than $n=\dim X_f=\deg \chi_f$; each $[\pi]$ is represented by the remainder $\tilde{\pi}$.

In other words, a linear system "remembers" by dividing by χ_f and keeping the remainder. If π and χ_f are polynomials of high degree, the division process does not have any obvious intuitive significance, even though it is a very simple operation from the algebraic point of view. This means that even the simplest class of dynamical systems, the linear ones, cannot be analyzed properly without making some concessions in the direction of abstract algebra. Some specific examples of this "pattern recognition" property of linear systems may be found in [1].

The possibility of representing states as polynomials suggests certain specific ways of setting up bases for a coordinate system in the state space. For instance, an obvious basis is $[1], [z], \dots, [z]^{n-1}$, because every polynomial π satisfies

$$\pi(z)=\xi_1+\xi_2 z+ \dots +\xi_n z^{n-1} \bmod \chi_f$$

(remember: $n=\deg \chi_f$). In particular, it is interesting to compute the *coordinate maps*

$$\pi \mapsto \xi_i(\pi)$$

which assign numerical values to certain features of the input pattern (=polynomial) π. (Of course, these maps depend on the choice of the basis. In other words, while the statement "Σ_f remembers π as $\tilde{\pi}$" is coordinate-free (or basis-free), the interpretation of the *numbers* $\xi_i(\pi)$ as input properties depends on the choice of the basis.)

To see what actually happens, consider the following case.

Suppose that $\chi(z)=\prod_{i=1}^{n}(z-a_i)$, $a_i \in K$, no two a_i equal. A convenient basis for the n-dimensional vector space of all polynomials modulo χ is given by the set of polynomials $\{e_i(z)=\chi(z)/(z-a_i),\ i=1, \dots, n\}$. Any polynomial π may be written *uniquely* modulo χ as

$$\pi(z)=\sum_{i=1}^{n}\xi_i(\pi)\, e_i(z) \bmod \chi,$$

where the coordinate functions are given by

$$\xi_i(\pi)=\pi(a_i)/e_i(a_i).$$

This formula is proved as follows [1]. Observe first that

(§) $$(z-a_i)\, e_j=(a_j-a_i)\, e_j \bmod \chi.$$

[1] Here the original derivation has been somewhat simplified. Note that the problem is a variant of the Lagrange interpolation formula.

Hence, for any π and any j, k the fact $\xi_j(\pi)=\xi_j(\tilde{\pi})$ implies

$$\xi_j((z-a_k)\,\pi)=\xi_j((a_j-a_k)\,\pi)=(a_j-a_k)\,\xi_j(\pi).$$

Since ξ_j is a linear function on polynomials, it follows that

$$\xi_j(\pi)=\xi_j(\pi(a_j)+(z-a_j)\,\sigma)=\xi_j(\pi(a_j))=\pi(a_j)\,\xi_j(1).$$

To determine $\xi_i(1)\in K$, observe that

$$e_i\cdot 1=e_i\cdot\sum_{j=1}^{n}\xi_j(1)\,e_j=\sum_{j=1}^{n}\xi_j(1)\prod_{k\neq i}(z-a_k)\,e_j.$$

Using (§) gives

$$=\sum_{j=1}^{n}\xi_j(1)\prod_{k\neq i}(a_j-a_k)\,e_j \mod \chi,$$

$$=\xi_i(1)\prod_{k\neq i}(a_i-a_k)\,e_i \mod \chi.$$

This shows that $\xi_i^{-1}(1)=e_i(a_i)\neq 0$, as claimed above.

In short: *relative to the basis* $\{e_i(z)/e_i(a_i)\}$ *the coordinate maps* $\pi\mapsto\xi_i(\pi)$ *are given by the evaluation of the value of the input polynomial at roots of the characteristic polynomial, normalized by the factor* $e_i^{-1}(a_i)$.

In the linear case, all this may appear to be just a modernized statement of the well-known method of partial-fraction expansion used in linear system theory. We emphasize, however:

In any dynamical system, linear or not, the coordinate maps correspond to abstracting certain characteristic features of the input relative to the basis chosen for the state space. The whole problem of realization may be viewed as the problem of effective computation of coordinate maps. Even if the internal structure of a system is known, its operation in response to inputs (for instance, its pattern recognition properties) cannot be fully understood until it is possible to say exactly what the value of each state variable says about the corresponding equivalence class $[\omega]$ *of inputs.*

The importance of this general principle will be clear when we attack the problem of computing coordinate maps for a new class of systems, namely those specified externally by a multilinear input/output map.

3. Multilinear Input/Output Maps

In analogy with the linear case, we now define a *multilinear input/output map* f *of degree* r *over a field* K as an r-linear map[11] (over K)

$$f\colon K^{m_1}[z_1]\times\ldots\times K^{m_r}[z_r]\to z^{-1}K^p[[z^{-1}]]$$

which commutes with the shift operator.

[11]) An *r-linear* map is linear in each of its r arguments separately when the remaining $r-1$ arguments are held fixed.

INTERPRETATION: This is a system with r input channels. Each input channel receives a vector polynomial $\omega_j(z_j)$. The output $f(\omega_1, ..., \omega_r)$ is an infinite sequence of vectors $(f(\omega_1, ..., \omega_r))_1$, $(f(\omega_1, ..., \omega_2))_2, ...$ in K^p, occurring at $t=1, 2, ...$

Intuitively, "r-linear" means that the various scalar signals entering at the r input channels are eventually all multiplied together, so that the output is a sum of products each of which contains one and only one factor from each channel. The mathematical problem is to explicate exactly how these multiplications take place within the system, using only general facts (such as "multilinear"). We shall be concerned mostly with bilinear (2-linear) maps.

The simplest example of a bilinear map is the product of two linear input/output maps. If g_1 and g_2 are linear input/output maps, the product is defined whenever $p_1=p_2=p$. Then $f=g_1\times g_2$ is given by

$$(g_1\times g_2)(\omega_1, \omega_2) := \sum_{j=1}^{\infty} (g_1(\omega_1))_j (g_2(\omega_2))_j z^{-j}$$

where $(g_i(\omega_i))_j$ is the j-th term in the output sequence of $g_i(\omega_i)$ and the multiplication is the componentwise product of two vectors in K^p. (Obviously this definition extends to the product of any finite number of linear maps, etc.)

In accordance with a tradition in mathematics which has become very strong since 1950, it is interesting to associate with any r-linear map f a linear map $f_\otimes$ [2] said to be induced by f. The map $f_\otimes$ may be regarded as a "universal linearization" of f: the map $f_\otimes$ can be constructed for any r-linear f, it is linear, and its properties in some sense mirror properties of f. We set [3]

$$f_\otimes(\omega_1 \cdots \omega_r) := f(\omega_1, \cdots, \omega_r).$$

This defines $f_\otimes: K^{m_1}[z_1] \otimes \cdots \otimes K^{m_r}[z_r] \to z^{-1} K^p[z]$ on polynomial vectors expressible as a tensor product $\omega_1(z_1) \otimes \cdots \otimes \omega_r(z_r)$ [III]. The definition is then extended by linearity to sums of such products, i.e., to the tensor space.

Useful as this procedure is in pure algebra, it fails to work in many cases in system theory. The reason is simple: although $f_\otimes$ is a linear map, it is *not* the input/output map of a linear *system* in the usual sense. In other words, $\Omega_\otimes = K^{m_1}[z_1] \otimes \cdots \otimes K^{m_r}[z_r]$, although a linear space, is not naturally isomorphic with the usual input space $\Omega = K^m[z]$; even though $\omega_1(z_1) \otimes \cdots \otimes \omega_r(z_r)$ may be viewed as an input, there is no natural way of viewing a *sum* of such terms as an input (to the multilinear system defined by f).

In spite of these serious conceptual difficulties, we might still define (in a rather abstract sense) a "tensor machine" associated with $f_\otimes$. The procedure is as follows. We consider the equivalence relation $E_{1\ldots r}$ (analogous to E in Sect. 2); we say that

$$(\omega_1 \cdots \omega_r)\, E_{1\ldots r}\, (\hat{\omega}_1 \cdots \hat{\omega}_r)$$

if and only if

$$f_\otimes(\omega_1 \cdots \omega_r) = f_\otimes(\hat{\omega}_1 \cdots \hat{\omega}_r).$$

---

[2]) $f_\otimes$ is called the *tensor map*.

[3]) The following paragraphs have been revised from the original for greater clarity.

[III]) This means the list (vector) of all products containing exactly one entry from each of the vectors $\omega_i(z_r)$.

The equivalence classes $[\cdots]_{1\ldots r}$ of $E_{1\ldots r}$ form a module $X_{f\otimes}$ with respect to the ring $K[z_1 \cdots z_r]$ isomorphic with $K[z]$.

The module $X_{f\otimes}$ has a natural and well-defined *read-out map*

$$h: X_{f\otimes} \to K^p: [\omega_1 \ldots \omega_r]_{1\ldots r} \mapsto (f_\otimes(\omega_1 \cdots \omega_r))_1 .$$

On the other hand, as already noted, $X_{f\otimes}$ has no natural read-in map; it is not clear how a sum of products $\omega_1 \cdots \omega_r$ can be interpreted as an input which, according to our basic convention, must be applied *serially in time*.

[We *could* adopt an arbitrary convention that the time of occurrence of inputs is given by the function t defined by the relations

$$-t(0, \ldots, 0) = 0,$$

$$-t(i_1, \ldots, i_r) = \Big(\sum_{j=0}^{|i|-1} r^j\Big) + \rho_{|i|}(i_1, \ldots, i_r), \qquad |i| \geqslant 1, \text{[4]}$$

where $|i| = i_1 + \cdots + i_r$ and the value of $\rho_q(i_1, \ldots, i_r)$ is k if $(i_1, \ldots, i_r)$ is the k-th member of the lexicographically ordered list of nonnegative integer r-tuples with $|i| = q$. Thus t is an isomorphism between the nonnegative integers and r-tuples of such numbers. (Of course, there are many such isomorphisms and there is no basic reason for preferring any one over the other.) With this convention, we have that

$$K^m[z_1] \otimes \cdots \otimes K^m[z_r] \approx K^m[z]$$

via the rule

$$\Sigma a_{i_1, \ldots, i_r} z_1^{i_1} \cdots z_r^{i_r} \approx \Sigma \alpha_{i_1, \ldots, i_r} z^{-t(i_1, \ldots, i_r)} .$$

One of the guiding principles of the theory to be developed in the next section is to systematically utilize tensor machines *without* having worry about such an arbitrary definition of the read-in map.]

Let us now mention a useful connection between "product" and "tensor product", noted informally as

$$X_{(g_1 \times g_2)\otimes} \subset X_{g_1} \otimes X_{g_2} .$$

In other words, we can obtain a realization of the bilinear map $g_1 \times g_2$ by taking (minimal) realizations Σ_{g_1} and Σ_{g_2} of g_1 and g_2 separately, forming the tensor product $\Sigma_{g_1} \otimes \Sigma_{g_2}$ (whose state space is of course $X_{g_1} \otimes X_{g_2}$), and then reducing the system to a minimal one. Example 2 of Section 5 displays a case where such a reduction actually occurs.

For many purposes, it is desirable to define an r-linear input/output map via formal power series. This gives a concrete rule for computing values of f. We associate to any r-linear input/output map f the power series

$$Z_f = \sum_{|i|>0}' \alpha_{i_1, \ldots, i_r} z_1^{-i_1} \cdots z_{i_r}^{-i_r} .$$

---

[4]) In these formulas, some minor printing errors of the original have been corrected

(More concretely: this power series is a formal r-dimensional discrete Fourier transform of the impulse-response function generated by f.) Then the output is computed by the formula

$$(f(\omega_1 \cdots \omega_r))_j = \beta_{j,\ldots,j}, \quad j>0,$$

where $\beta_{j,\ldots,j}$ is determined from

$$Z_f \omega_1 \cdots \omega_r = \Sigma \beta_{i_1,\ldots,i_r} z_1^{-i_1} \cdots z_r^{-i_r}.$$

The last requirement implies that the coefficients of Z_f are to be defined as[5]

$$\alpha_{1,\ldots,1} = f(1)_1,$$
$$\alpha_{i_1,\ldots,i_r} = f(z_1^{i_1-1} \cdots z_r^{i_r-1})_1.$$

The rule for computing the output can be interpreted as follows. Multiply Z_f by the (tensor) product of input polynomials, recompute the resulting power series, and retain only the "diagonal" terms, that is, the coefficients of $(z_1 \cdots z_r)^{-j}$.

Instead of dealing with power series, it is in many cases more convenient [10] to represent Z_f as a rational function involving ratios of polynomials in $z_1, z_2, \ldots, z_1 z_2, \ldots, z_1 z_2 \cdots z_r$. Such a transfer function is of course equivalent to a formal power series. See the examples in Section 5.

4. Construction of Multilinear Machines

To keep our notations manageable, we shall assume from now on that

$$m=p=1 \quad \text{and} \quad r=1. \tag{4.1}$$

This is clearly the simplest nontrivial case. Since it is in principle very easy to extend our constructions to the most general case, we shall announce our main results without special reference to (4.1). This section should be read in parallel with Section 5 which gives several concrete examples of the abstract computations defined below.

Given a bilinear response function f, the construction of a system (machine) realizing it proceeds as follows.

Step 1. We determine the linear system $f_\otimes$. More explicitly,

(i) We determine the equivalence classes $[\pi_1 \pi_2]_{12}$ of the equivalence relation E_{12} on $K[z_1] \otimes K[z_2] \approx K[z_1, z_2]$.

(ii) The set X_{12} of equivalence classes is made into a K-vector space (by the usual procedure in the linear case); then we choose a convenient basis (of equivalence classes) in X_{12}.

(iii) We determine the coordinates of the image of each element of $K[z_1] \otimes K[z_2]$ under the natural projection $K[z_1] \otimes K[z_2] \to X_{12}$.

[5] Theses formulas were not explicitly mentioned in the original version of the paper.

Step 2. We introduce two new equivalence relations on the subspaces $K[z_1] \otimes \{1\}$ and $\{1\} \otimes K[z_2]$ of $K[z_1] \otimes K[z_2]$, as follows:

$$\pi_i E_i \hat{\pi}_i \text{ if and only if } f_{\otimes}(z_i^l \pi_i(z_i)) = f_{\otimes}(z_i^l \hat{\pi}_i(z_i))$$

for all $l \geqslant 0$[6] and $i = 1, 2$. It is clear that *E_1 or E_2 equivalence implies E_{12} equivalence*! (In fact, this is the principal creative element involved in these definitions, which appear here for the first time.) It follows that we have the natural projections

$$[\pi_1]_1 \mapsto [\pi_1 \cdot 1]_{12} \text{ for all } \pi_1 \in K[z_1],$$
$$[\pi_2]_2 \mapsto [1 \cdot \pi_2]_{12} \text{ for all } \pi_2 \in K[z_2].$$

The equivalence classes under E_1 and E_2 are denoted by X_1 and X_2 respectively. Next, we determine module structures on X_1 and X_2 with respect to the rings $K[z_1]$ and $K[z_2]$ and then compute explicit formulas for the coordinates of the image under the maps $K[z_1] \to X_1$ and $K[z_2] \to X_2$.

Step 3. To be specific, we denote the coordinates of $[\pi_1 \pi_2]_{12}$ by $\xi_1, \ldots, \xi_{n_{12}}$, those of $[\pi_1]_1$ by $\eta_1, \ldots, \eta_{n_1}$, and those of $[\pi_2]_2$ by $\zeta_1, \ldots, \zeta_{n_2}$. (The case where at least one of the vector spaces X_{12}, X_1, X_2 is infinite-dimensional could occur, but we shall not be interested in it here, in view of the finiteness criterion stated below.) Let $N = n_{12} + n_1 + n_2$. Then $X_{12} \oplus X_1 \oplus X_2 \approx K^N$ and we have the collection of coordinate functions

$$\rho_f : K[z_1] \times K[z_2] \to K^N$$

given by

$$\rho_f : (\pi_1, \pi_2) \mapsto (\xi_1, \ldots, \xi_{n_{12}};\ \eta_1, \ldots, \eta_{n_1};\ \zeta_1, \ldots, \zeta_{n_2}).$$

We shall call $X_{12} \oplus X_1 \oplus X_2$ the *parametrized state space* of the basic bilinear response function f and denote it by X_f.

We observe that the last two groups of coordinates in K^N are linear functions of (π_1, π_2), while the first group is a bilinear function of (π_1, π_2). The last statement follows from the fact that the first group of coordinates is obtained by composing the coordinate map $K[z_1] \otimes K[z_2] \to K^{n_{12}}$ (which is linear, as are all coordinate maps) with the linearization map $\otimes : K[z_1] \times K[z_2] \to K[z_1] \otimes K[z_2]$ (which is bilinear). These statements prove: *The image V_f of ρ_f in K^N is an (affine) variety*[7] *(=algebraic manifold, not necessarily irreducible)* in K^N. Clearly V_f is intrinsically

---

[6]) The correct condition is actually $l > 0$. This error (which was not a printing error in the original version) has serious consequences in that the construction in the case $l \geqslant 0$ would yield a realization which is not necessarily observable. Observability questions were not studied in the present paper; see Kalman [14].

[7]) The intuitive idea which is supposed to justify this statement is that the image of *every* map $K^n \to K^N$ given by polynomial functions is an affine variety. Unfortunately, this is not precisely true as shown by the counterexample $K^2 \to K^2 : (x_1, x_2) \mapsto (x_1 x_2, x_2)$. In modern terminology: the image of a morphism of an algebraic set is not necessarily an algebraic set. Instead of considering the set V_f, which may not be a variety, we must study its closure $\overline{V}_f$ in the Zariski topology, which is (by definition) the smallest closed (affine) algebraic set containing V_f. See Dieudonne [15]. What remains true is that at this point algebraic geometry enters the picture in an essential way.

defined by f (independent of the chosen coordinates, hence of the specific form of the coordinate map). We shall call V_f the *variety* of f [8]. Note that V_f corresponds to the natural state space X_f in the linear case.

The main task in this step, then, is to compute V_f explicitly, from the knowledge of ρ_f. This may be done by establishing certain relationships of constraint between the coordinates. See the Examples of Section 5 for illustration of the details.

Step 4. The basic problem of realization theory is to write down the state-transition equations (equations of motion). It will be shown below that a point in $V_f \subset X_f$ uniquely specifies the state. The problem therefore amounts to the following: Suppose at time 0 the input sequence received by Channel 1 is a polynomial π_1, as in the linear case, and that the input sequence received by Channel 2 is a polynomial π_2. After one step in time, a new symbol $p_1 \in K$ is received at Channel 1 and p_2 is received at Channel 2. Hence the *new* polynomial seen by Channel 1 is $z_1 \pi_1 + p_1$ and that seen by Channel 2 is $z_2 \pi_2 + p_2$. At the next step, the new symbols received are (p_1', p_2'), then (p_1'', p_2''), and so on. To write down the state-transition equations for the machine Σ_f which realizes f (thereby specifying Σ_f) is equivalent to filling in the dotted arrows in the commutative diagram

$$
\begin{tikzcd}
(\pi_1, \pi_2) \arrow[r, "\rho_f"] \arrow[d, "\begin{array}{l}\text{receive}\\ (p_1, p_2)\end{array}"] & \text{coordinates} \arrow[d, dotted, "\begin{array}{l}\text{equations}\\ \text{of motion}\end{array}"] \\
(z_1 \pi_1 + p_1, z_2 \pi_2 + p_2) \arrow[r, "\rho_f"] \arrow[d, "\begin{array}{l}\text{receive}\\ (p_1', p_2')\end{array}"] & \text{coordinates} \arrow[d, dotted, "\begin{array}{l}\text{equations}\\ \text{of motion}\end{array}"] \\
(z_1^2 \pi_1 + z_1 p_1 + p_1', z_2^2 \pi_2 + z_1 p_2 + p_2') \arrow[r, "\rho_f"] \arrow[d, "\begin{array}{l}\text{receive}\\ (p_1'', p_2'')\end{array}"] & \text{coordinates} \arrow[d, dotted, "\begin{array}{l}\text{equations}\\ \text{of motion}\end{array}"] \\
\, & \,
\end{tikzcd}
$$

The transition equations are written down separately for each group of coordinates. In the spaces X_1 and X_2 they are computed in the usual way using the module structure; they are given by the linear equations

$$[\pi_1]_1 \mapsto [z_1 \pi_1 + p_1]_1 = z_1 [\pi_1]_1 + p_1 [1]_1 ,$$

$$[\pi_2]_2 \mapsto [z_2 \pi_2 + p_2]_2 = z_2 [\pi_2]_2 + p_2 [1]_2 .$$

---

[8]) It is an extremely surprising fact (unknown in 1968 but nevertheless explicitly verified by the two examples of Sec. 5) that $\bar{V}_f$, the closure of V_f, is *always* isomorphic to a linear space K^n, where n is the minimal number of state variables necessary to realize the system using only linear subsystems and real-time multiplication (see Kalman [16]).

In X_{12}, the computation is more complicated:

$$[\pi_1\,\pi_2]_{12} \mapsto [(z_1\,\pi_1+p_1)\,(z_2\,\pi_2+p_2)]_{12} = z_1\,z_2\,[\pi_1\,\pi_2]_{12} + [p_1 p_2]_{12} + \\ + [p_2\,z_1\,\pi_1]_{12} + [p_1 z_2\,\pi_2]_{12}\,^{9)}.$$

The first two terms are computed in the usual way using the module structure in X_{12}; the last two terms are first computed in X_1 and X_2 respectively and then *projected* into X_{12}.

Intuitively speaking, the purpose of keeping track of information stored in X_1 and X_2 is solely for the purpose of evaluating the transition map in X_{12}, because *this transition map cannot be evaluated in general using only the knowledge of the equivalence class* $[\pi_1\,\pi_2]_{12}$ *of* (π_1, π_2).

Step 5. The explicit specification of the state transition map in X_f already incorporates the input information (p_1, p_2), (p_1', p_2'), ... To complete the specification of Σ_f, it is necessary to specify what the read-out function h is. Since f is bilinear, it suffices to define h on X_{12} (that is, h is identically zero on the subspaces X_1 and X_2 of X_f). The obvious definition is

$$h\colon [\pi_1\,\pi_{12}] \mapsto (f_\otimes(\pi_1\,\pi_2))_1 = (f(\pi_1, \pi_2))_1.$$

The definition of E_{12} shows that h is a well-defined function.

Step 6. As a result of the previous steps, the function $(\pi_1, \pi_2) \to (f(\pi_1, \pi_2))_1$ is now explicitly given as a composition of two types of functions: a polynomial function of coordinates and the coordinate functions. Interpreting the coordinate functions as corresponding to bilinear or linear input/output maps, *it remains necessary to check whether or not these maps can be realized using a still smaller state space.* If not, then the state space V_f as originally defined is a minimal one. If so, some of the state transition equations found in Step 4 can be simplified and the dimension number N can be replaced by a smaller one [10]. The full theory of minimal realization is not yet available.

The mathematical content of the preceding prescriptions can be appreciated better by stating now the three main theorems of the paper.

---

[9]) These equations are essentially the same as those published later by Arbib [13, p. 700–710], using a different notation. He obtained our result by embedding the Nerode equivalence classes into linear spaces (which we got automatically via the module constructions), without studying the important question of whether or not the Nerode classes *coincide* with this linear embedding. The dimension of this embedding is *not* minimized by Arbib (see next several footnotes).

[10]) In the case of the functions η_j or ζ_k linear dependence is avoided by the standard procedures for the realization of a single input/single output linear system. In the case of the functions ξ_i, however, the situation is much more subtle. This point is overlooked by Arbib [13], whose universal realization of a bilinear input/output function (see [13, Fig. 1]), may not have the minimal number of state variables. In our Example 1 of Sec. 5 the realization built according the preceding equations has four state variables, but these can be reduced to three because the functions ξ_1 and ξ_2 are algebraically dependent on η_1 and ζ_1 via the identity $\xi_1 = \eta_1\zeta_1$. The realization equations given above always work but have no canonical (intrinsic) significance.

REPRESENTATION THEOREM. *To know the Nerode equivalence class of* $(\pi_1\,\pi_2)$ *it is necessary and sufficient to know*

$$[\pi_1\,\pi_2]_{12}, \quad [z_1\,\pi_1]_1, \quad [z_2\,\pi_2]_2.$$

This theorem is the appropriate generalization of the theorem called by the same name in Section 2 [11]. It extends to arbitrary r, m, and p.

Strictly speaking, $[z_i\,\pi_i]_i=[z_i\hat{\pi}_i]_i$ does not imply $[\pi_i]_i=[\hat{\pi}_i]_i$, but nevertheless this implication is true (for various reasons) in most practical cases. Instead of the state parametrization provided by the Representation Theorem, it is therefore often more convenient to use the equivalence classes $[\pi_1\,\pi_2]_{12}$, $[\pi_1]_1$, and $[\pi_2]_2$ [12]. (This is actually how we have defined above, in Step 3, the parametrized state space X_f.) In any case, we have the following result, corresponding exactly to the same-name theorem in Section 2:

FINITE REALIZATION CRITERION. *A bilinear input/output function f has a finite-dimensional realization if and only if each of the K-vector spaces of equivalence classes*

$$X_{12}=\{[\omega]_{12}:\, \omega\in K[z_1]\otimes K[z_2]\},$$
$$X_1=\{[\pi_i]_i:\, \pi_i\in K[z_i]\}, \qquad i=1,2,$$

is finite dimensional.

The sufficiency of this criterion is proved by our construction procedure sketched above. Its necessity is a rewording of the representation theorem [13]. It should be pointed out that $\dim X_{12}<\infty$ by no means implies that either $\dim X_1<\infty$ or $\dim X_2<$ $<\infty$. Again, this theorem extends to arbitrary r, m, and p.

The most important single result of the theory can now be stated in a way that does not even require specific reference to r, m, or p:

FUNDAMENTAL THEOREM OF MULTILINEAR SYSTEMS. *The state set of a multilinear input/output function f is an affine variety V_f which is intrinsically defined by f. Moreover, V_f is abelian and admits the structure of a $K[z]$-module.*

---

[11]) The reader who has difficulty in performing the easy verifications required to prove this theorem may refer to Arbib [13] where the complete argument is given.

[12]) There is a very subtle point involved here. The determination of the equivalence classes according to the Representation Theorem implies that all these equivalence classes are observable — an important remark which should have been included in the 1968 version. If we replace $[z_i\,\pi_i]_i$ by $[\pi_i]_i$, then observability may be lost (see Kalman [14]).

[13]) This is a seriously incorrect statement. Since the equivalence relations E_1, E_2, and E_{12} were introduced *arbitrarily* (not canonically), it is conceivable that the desired finiteness property of the Nerode equivalence is not inherited by the E_i. For example, a *finitely realizable* (bilinear) system may have a Hankel matrix of infinite rank. Nevertheless, the Finite Realizability Criterion is still a necessary conditon. A through discussion of the subtleties involved here will be found in Kalman [14].

The abelian structure on V_f is obvious: it is induced directly from the abelian structure on the input space $K^{m_1}[z_1] \times \ldots \times K^{m_r}[z_r]$. The fact that V_f admits a $K[z]$-module structure on V_f is basically a consequence of the fact that the passage of time $t \mapsto t+1$ induces a one-step state transition which sends V_f into itself. Intuitively, this result means that algebra can decompose the structure of multilinear systems *abstractly* into two parts: the dynamical part is always linear and is given by the $K[z]$-module structure on V_f; the nonlinear part is "orthogonal" to dynamics and is given by the ideal of the variety V_f. In other words, the module gives at once all the dynamical "modules" (boxes) embedded in the system, while the variety tells us how they are interconnected [14].

---

[14]) The statement of this theorem and the discussion of the last paragraph are much too imprecise. However, the basic idea is correct (see Kalman [16]).