The second set of times above is identical to that of enabling times. Identification of the first set above is ensured by Conditions 2) and 4).

Finally, Conditions 2) and 4) ensure that we can extract all interrupted times, which are the time epochs when the state transits from S_e to S_e^c due to the occurrence of an event other than e.

Necessity: We use contraposition.

If Condition 1) is not satisfied, then we have an occurrence of e with an associated null transition observation. By Lemma 1, e is not invertible.

Suppose Condition 2) does not hold. Then, there exist two trajectories from $s(0)$ to s and s', respectively, that have the same observation. Since $s \in S_e$ and $s' \in S_e^c$, there is a transition either from S_e to S_e^c or from S_e^c to S_e that has a null observation. The former constitutes an occurrence or interruption time of e, while the latter an enabling time. Hence, by Lemma 1, e is not invertible.

If Condition 3) does not hold, then there exists an event $e' \neq e$ whose occurrence we cannot distinguish from that of e. If Condition 4) does not hold, a similar situation occurs, the difference being that in this case we cannot distinguish between the occurrence and interruption of e. In either case, by Lemma 1, e is not invertible.

Examples illustrating the conditions for d-invertibility and an algorithm to extract lifetimes of d-invertible event from observations can be found in [12].

IV. Conclusions

The framework that we have introduced opens up many possibilities for further work. A problem that arises naturally in our framework is that of designing an observation map Λ for a given system $\mathcal{G}$ to achieve invertibility while minimizing some cost function. Such a problem is practically relevant in the context of sensor configuration design. We envision that its solution paves a path toward a general systematic framework for the design of on-line monitoring systems. Some work along these lines is reported in [13].

References

[1] A. Bouloutas, G. W. Hart, and M. Schwartz, "On the design of observers for fault detection in communication networks," in *Network Management and Control*, A. Kershenbaum, M. Malek, and M. Wall, Eds. New York: Plenum, 1990, pp. 319–338.

[2] C. G. Cassandras, *Discrete Event Systems: Modeling and Performance Analysis*. Homewood, IL: Irwin and Aksen, 1993.

[3] C. G. Cassandras and J. Pan, "Parallel sample path generation for discrete event systems and the traffic smoothing problem," *Discrete Event Dynamic Systems: Theory Appl.*, vol. 5, no. 2/3, pp. 187–217, Apr./July 1995.

[4] R. Cieslak, C. Desclaux, A. Fawaz, and P. Varaiya, "Supervisory control of discrete-event processes with partial observations," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 249–260, Mar. 1988.

[5] P. Glasserman, *Gradient Estimation via Perturbation Analysis*. Norwell, MA: Kluwer, 1991.

[6] Y. C. Ho, "Dynamics of discrete event systems," *Proc. IEEE*, vol. 77, pp. 3–6, Jan. 1989.

[7] Y.-C. Ho and X.-R. Cao, *Perturbation Analysis of Discrete Event Dynamic Systems*. Norwell, MA: Kluwer, 1991.

[8] F. Lin, "Diagnosability of discrete event systems and its applications," *Discrete Event Dynamic Systems: Theory Appl.*, vol. 4, no. 2, pp. 197–212, May 1994.

[9] F. Lin and W. Wonham, "Decentralized control and coordination of discrete event systems with partial observation," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 1330–1337, Dec. 1990.

[10] C. M. Özveren and A. S. Willsky, "Invertibility of discrete-event dynamic systems," *Math. Contr., Signals, Syst.*, vol. 5, pp. 365–390, 1992.

[11] ——, "Observability of discrete event systems," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 797–806, July 1990.

[12] Y. Park, "Model-Based monitoring of discrete event systems," Ph.D. dissertation, School of Electrical and Computer Engineering, Purdue Univ., West Lafayette, IN, 1996.

[13] Y. Park and E. K. P. Chong, "Sensor assignment for invertibility in interruptive timed discrete event systems," in *Proc. 9th IEEE Int. Symp. Intelligent Contr.*, Columbus, OH, July 1994, pp. 207–212.

[14] ——, "Distributed inversion in timed discrete event systems," *Discrete Event Dynamic Systems: Theory Appl.*, vol. 5, no. 2/3, pp. 219–241, Apr./July 1995.

[15] A. A. Pritsker, *Introduction to SLAM II*. New York: Halsted, 1986.

[16] P. Ramadge, "Observability of discrete event systems," in *Proc. 25th Conf. Decision Contr.*, Athens, Greece, Dec. 1986, pp. 1108–1112.

[17] P. Ramadge and W. Wonham, "The control of discrete event systems," *Proc. IEEE*, vol. 77, pp. 81–98, Jan. 1989.

[18] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 40, pp. 1555–1575, Sept. 1995.

[19] B. A. Schroeder, "On-line monitoring: A tutorial," *IEEE Computer*, pp. 72–78, June 1995.

[20] K. Williams, M. Andersland, J. Gannon, J. Lumpp, and T. Casavant, "Perturbation tracking," in *Proc. 32nd Conf. Decision Contr.*, San Antonio, TX, 1993, pp. 674–679.

On Constructing a Shortest Linear Recurrence Relation

Margreet Kuijper and Jan C. Willems

***Abstract*— It has been shown in the literature that a formulation of the minimal partial realization problem in terms of exact modeling of a behavior lends itself to an iterative polynomial solution. For the scalar case, we explicitly present such a solution in full detail. Unlike classical solution methods based on Hankel matrices, the algorithm is constructive. It iteratively constructs a partial realization of minimal McMillan degree. The algorithm is known in information theory as the Berlekamp–Massey algorithm and is used for constructing a shortest linear recurrence relation for a finite sequence of numbers.**

***Index Terms*—Behaviors, Berlekamp–Massey algorithm, linear systems, minimal partial realizations, shortest linear recurrence relations.**

I. Introduction

In this paper we consider the minimal partial realization problem. In [14], a connection with a problem in coding theory, namely the decoding of certain types of error-correcting block codes (BCH codes), has first been mentioned. The essential step in the decoding of a BCH code is the construction of a shortest linear recurrence relation for a finite sequence of numbers $a_1 a_2, \cdots, a_N$; see, e.g., [8] and [5]. The length of the recurrence relation corresponds to the number of errors that have occurred in transmitting a message. It has to be minimized in maximum likelihood decoding when errors are assumed to be independent. In 1968, Berlekamp and Massey presented an algorithm to compute a shortest linear recurrence

Manuscript received March 3, 1995; revised May 21, 1996. M. Kuijper was supported in part by the Australian Research Council.

M. Kuijper is with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, Victoria 3052, Australia (e-mail: m.kuijper@ee.mu.oz.au).

J. C. Willems is with the Department of Mathematics, University of Groningen, 9700 AV Groningen, The Netherlands.

Publisher Item Identifier S 0018-9286(97)07626-5.

relation. Although it was originally designed for decoding purposes, it later became important for cryptographic applications, namely for calculating the complexity profile of a sequence of numbers; see [13]. It is explained below that the denominator of a minimal partial realization corresponds to a shortest linear recurrence relation.

Despite the observations in [14], [7], and [1], the Berlekamp–Massey algorithm has not been welcomed in system theory as a constructive solution method for minimal partial realization. Instead, most system-theoretic results on minimal partial realization are based on Hankel matrices (e.g., [7] and references therein) and do not produce explicit algorithms for calculating a minimal partial realization.

In recent literature [16], [17], a framework for modeling data into a "minimal" behavior has been presented. It includes an outline for an iterative procedure. In [3] it is shown how this framework can be applied to model a partially given impulse response in the general multivariable case. In this paper, our aim is to use this approach to derive a constructive and efficient iterative algorithm. The algorithm is based on the outline given in [17] and coincides with the Berlekamp–Massey algorithm. In fact, we find that the Berlekamp–Massey algorithm not only constructs the denominator but also the numerator of a minimal partial realization. We restrict ourselves to the scalar case because of its simplicity and elegance. However, the algorithm can be extended to the multivariable case; see [10] where a connection is made with the generalized Berlekamp–Massey algorithm of [6].

Let us now formulate the issues in a precise way. Assume that $a_1, a_2, \cdots, a_N$ is a finite sequence of numbers from a field $\mathbf{F}$. For the purpose of this paper, we do not need to make a distinction between finite and infinite fields. The problem of finding a shortest linear recurrence relation for $a_1, a_2, \cdots, a_N$ is the following: find real numbers $e_0, e_1, \cdots, e_{L-1}$, such that

$$\begin{aligned} &a_{j+L} + e_{L-1}a_{j+L-1} + \cdots + e_0 a_j = 0 \\ &\text{for} \quad j = 1, \cdots, N-L. \end{aligned} \tag{1}$$

Here L should be as small as possible in order to capture as much as possible of the structure underlying $a_1, a_2, \cdots, a_N$. If $a_1, a_2, \cdots, a_N$ does not allow any relation of the type (1) for $L < N$, then we set $L = N$ and consider (1) to be trivially satisfied for any $e_0, e_1, \cdots, e_{L-1}$. In the sequel, we denote a linear recurrence relation for $a_1 a_2, \cdots, a_N$ by the polynomial $e(s) := s^L + e_{L-1}s^{L-1} + \cdots + e_0$; a shortest linear recurrence relation has minimal degree. Note that the polynomial e together with the initial values $a_1, a_2, \cdots, a_L$ completely define the original finite sequence. One can therefore view the problem of finding a shortest linear recurrence relation as a problem of data reduction.

The scalar minimal partial realization problem for a finite sequence $a_1, a_2, \cdots, a_N$ is the following: find polynomials e and h for which

$$\frac{h(s)}{e(s)} = a_0 + a_1 s^{-1} + \cdots + a_N s^{-N} + \phi(s)s^{-(N+1)}$$

such that deg e is as small as possible and ϕ is proper rational. It is not difficult to see that the denominator e of a minimal partial realization is a shortest linear recurrence relation for $a_1, a_2, \cdots, a_N$. Vice versa, a shortest linear recurrence relation e for $a_1, a_2, \cdots, a_N$ gives rise to a minimal partial realization for $a_1, a_2, \cdots, a_N$ by defining $h(s) := h_L s^L + h_{L-1}s^{L-1} + \cdots + h_0$ where, for $j = 0, 1, \cdots, L$

$$h_j := a_j + e_{L-1}a_{j-1} + \cdots + e_1 a_1 + e_0 a_0. \tag{2}$$

In Section III, we reformulate the above problem in a behavioral setting, as in [3]. This reformulation then provides a natural basis for an iterative solution, the Berlekamp–Massey algorithm, which we present in full detail in Section IV. Here, we adhere to the version as presented in [5, p. 180], which is the original algorithm from [4, p. 184] with a modification from [12]. Thus, our approach has enabled a system-theoretic explanation of the algorithm.

In the sequel, we put ideas from [16] to work. We take a behavioral point of view and first need to introduce some basic ideas of the behavioral approach.

II. Preliminaries on Modeling of Behaviors

In the behavioral approach [15]–[17], a system is essentially defined as a behavior $\mathcal{B}$ which is a set of trajectories; in this paper we consider linear shift-invariant behaviors on the time-set $\mathbf{Z}_+$ of the form $\mathcal{B} = \ker R(\sigma)$, where R is a polynomial $g \times q$-matrix and σ is the backward shift operator

$$\sigma((w_0, w_1, w_2, \cdots)) := (w_1, w_2, \cdots).$$

In other words, $\mathcal{B}$ consists of trajectories $\boldsymbol{w}: \mathbf{Z}+ \mapsto \mathbf{R}^q$, for which

$$R(\sigma)\boldsymbol{w} = 0. \tag{3}$$

Representation (3) is called a *kernel representation* of $\mathcal{B}$.

Let us repeat some notions from [15] and start with the following lemma (see, e.g., [9, Th. 3.9] for a detailed proof).

Lemma 1: Let $R_1 \in \mathbf{R}^{g_1 \times q}$ and $R_2 \in \mathbf{R}^{g_2 \times q}$. Then

$$\ker R_1(\sigma) \subset \ker R_2(\sigma)$$

if and only if there exists a polynomial matrix $F \in \mathbf{R}^{g_2 \times g_1}$ such that

$$R_2 = FR_1.$$

It is a corollary of the above lemma that polynomial matrices R_1 and R_2 of full row rank represent the same behavior if and only if there exists a unimodular matrix U (i.e., a polynomial matrix with constant nonzero determinant) such that $R_1 = UR_2$.

The behavioral approach can be used for obtaining models from a set of observed time series. The general ideas stem from [16]. In this section, we restrict ourselves to exact modeling of discrete-time series, as presented in [17]. In the following, we briefly recall the basic concepts.

Let us assume that we have a *data set* $\mathbf{D} = \{\boldsymbol{b}_0, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_N\}$ where $\boldsymbol{b}_i \in (\mathbf{R}^q)^{\mathbf{Z}_+}$ are observed trajectories $(i = 0, 1, \cdots, N)$. A behavior $\mathcal{B}$ is called an *unfalsified model* for $\mathbf{D}$ if $\mathbf{D} \subseteq \mathcal{B}$. A model $\mathcal{B}_1$ is called *more powerful* than a model $\mathcal{B}_2$ if $\mathcal{B}_1 \subsetneq \mathcal{B}_2$. A model $\mathcal{B}^*$ is called the *most powerful unfalsified model (MPUM)* for $\mathbf{D}$, if $\mathcal{B}^*$ is unfalsified for $\mathbf{D}$ and $\mathbf{D} \subseteq \mathcal{B} \Rightarrow \mathcal{B}^* \subseteq \mathcal{B}$. It has been shown in [16] that a unique MPUM $\mathcal{B}^*$ exists for $\mathbf{D}$. Note, however, that a kernel representation (3) of $\mathcal{B}^*$ is far from unique. In fact, any other kernel representation can be obtained by left multiplication by a unimodular matrix.

Example 2 ($\mathbf{T} = \mathbf{Z}_+$ *and* $q = 2$)*:* Let $\mathbf{D} = \{\mathbf{b}\}$, with

$$\mathbf{b} = \left(\begin{bmatrix}3\\0\end{bmatrix}, \begin{bmatrix}2\\0\end{bmatrix}, \begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}0\\0\end{bmatrix}, \cdots\right).$$

The MPUM is a three-dimensional vector space: $\mathcal{B}^* = \text{span}\ \{\sigma^2\mathbf{b}, \sigma\mathbf{b}, \mathbf{b}\}$. A kernel representation of $\mathcal{B}^*$ is given by

$$\begin{bmatrix}1 & -1 - 2\sigma - 3\sigma^2 \\ 0 & \sigma^3\end{bmatrix}\begin{bmatrix}\boldsymbol{w}_1\\ \boldsymbol{w}_2\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}.$$

Alternative kernel representations of $\mathcal{B}^*$ are, e.g.,

$$\begin{bmatrix}1 - 2\sigma + \sigma^2 & -1 \\ \sigma^3 & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{w}_1\\ \boldsymbol{w}_2\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}$$

and

$$\begin{bmatrix}2 - 3\sigma & -2 - \sigma \\ -\sigma + 2\sigma^2 & \sigma\end{bmatrix}\begin{bmatrix}\boldsymbol{w}_1\\ \boldsymbol{w}_2\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}. \tag{4}$$

We are now ready to present the procedure of [17, p. 289], which underlies the iterative partial realization algorithm of Section IV. The procedure provides a framework for the iterative construction of a kernel representation of the MPUM for $\mathbf{D} = \{\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_N\}$. It can be easily understood from Lemma 1.

Procedure 3: Initially define

$$R_{-1} := I \text{ (where } I \text{ is the identity matrix).}$$

Proceed iteratively as follows for $k = 0, \cdots, N$. Define, after receiving $\{\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_k\}$, the kth *error trajectory* $\boldsymbol{e}_k$ as

$$\boldsymbol{e}_k := R_{k-1}(\sigma)\mathbf{b}_k.$$

Compute a kernel representation $V_k(\sigma)\boldsymbol{w} = 0$ of the MPUM for $\{\boldsymbol{e}_k\}$. Then, define

$$R_k := V_k R_{k-1}.$$

Theorem 4 [17]: For $k = 0, \cdots, N$, the kernel representation $R_k(\sigma)\boldsymbol{w} = 0$ of the above procedure represents the MPUM for $\{\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_k\}$.

Next, we recall the notion of *controllability* from [15]. For the purpose of this paper, we consider the following as a definition: a behavior $\mathcal{B} = \ker R(\sigma)$ on $\mathbf{Z}_+$ is controllable if $R(s)$ has constant rank for all $s \in \mathbf{C}$. For the specific case that $q = 2$, it follows from Lemma 1 that two distinct nontrivial controllable models $\mathcal{B}_1$ and $\mathcal{B}_2$ that are unfalsified for the same data set are "incomparable," i.e., neither $\mathcal{B}_1 \subseteq \mathcal{B}_2$ nor $\mathcal{B}_2 \subseteq \mathcal{B}_1$. However, an ordering can still be introduced on the basis of the "complexity" of a model. As a measure of complexity we introduce the *order* $n(\mathcal{B})$ of a behavior. Since this concept will also be important for noncontrollable behaviors, we give a general definition: the order $n(\mathcal{B})$ is defined as the minimum value of the sum of the row degrees of R, where the minimum is taken over all possible kernel representations (3) of $\mathcal{B}$. This minimum is attained exactly when R is "row reduced."

Definition 5: Let $R \in \mathbf{R}^{g\times q}$ have full row rank. Define $R_d \in \mathbf{R}^{g\times q}$ as the *leading row coefficient matrix* of R, i.e., the constant matrix that consists of the coefficients of the highest degree terms in each row of R. Define R to be *row reduced* if R_d has full row rank.

When a matrix R is not row reduced, a unimodular matrix U can be found such that UR is row reduced. A procedure is given in [18, p. 27] (see also [9, p. 24]) where it is shown that not only the sum of the minimal row degrees is an invariant of a behavior, but also the minimal row degrees themselves are invariants of a behavior. For example, the minimal row degrees of $\mathcal{B}$ in Example 2 are 1 and 2; this can be seen from the row-reduced representation (4).

In accordance with [2], we call a model $\mathcal{B}$ a *controllable-minimal complexity unfalsified model (C-MCUM)* for $\mathbf{D}$ if $\mathcal{B}$ is controllable, unfalsified for $\mathbf{D}$, and of least order among all controllable unfalsified models for $\mathbf{D}$.

III. Partial Realization as Exact Modeling

In this section, we put the minimal partial realization problem in a behavioral framework, as in [3]. However, we prefer to use $\mathbf{Z}_+$ instead of $\mathbf{Z}_-$ as the time-axis for reasons of exposition.

First, we transform the data $a_1, a_2, \cdots, a_N$ into a trajectory $\mathbf{b}$ that can be interpreted as a reversed partial impulse response. The trajectory $\mathbf{b}$ is defined from $\mathbf{Z}_+$ to $\mathbf{R}^2$ by

$$\mathbf{b} = \left(\begin{bmatrix} a_N \\ 0 \end{bmatrix}, \cdots, \begin{bmatrix} a_1 \\ 0 \end{bmatrix}, \begin{bmatrix} a_0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \cdots\right). \tag{5}$$

Not surprisingly, a model for $\{\mathbf{b}\}$ corresponds to a partial realization. More precisely, we have the following theorem which can be readily verified. (We define the *reciprocal* $P^r(s)$ of a polynomial matrix $P(s) = P_n s^n + P_{n-1}s^{n-1} + \cdots + P_1 s + P_0$ $(P_n \neq 0)$ by $P^r(s) := P_n + P_{n-1}s + \cdots + P_1 s^{n-1} + P_0 s^n$.)

Theorem 6: Let $\mathbf{b}$ be defined by (5). Let

$$[c(\sigma) \quad -p(\sigma)]\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = 0$$

be a kernel representation of row degree L of a C-MCUM for $\{\mathbf{b}\}$ with $c(0) = 1$. Define polynomials e and h by $[e \quad h] := [c \quad p]^r$. Then e is a shortest linear recurrence relation for $a_1, a_2, \cdots, a_N$ and h/e is a minimal partial realization for $a_0, a_1, \cdots, a_N$.

Because of the above theorem, the minimal partial realization problem can be reformulated as follows.

Problem Statement: Find a C-MCUM on $\mathbf{Z}_+$ for $\{\mathbf{b}\}$.

The MPUM for $\{\mathbf{b}\}$ simply equals

$$\operatorname{span}\{\sigma^N\mathbf{b}, \sigma^{N-1}\mathbf{b}, \cdots, \sigma\mathbf{b}, \mathbf{b}\}.$$

As noted in [3], a kernel representation for it is given by $A(\sigma)\boldsymbol{w} = 0$, where

$$A(s) = \begin{bmatrix} 1 & -(a_0 + a_1 s + \cdots + a_N s^N) \\ 0 & s^{N+1} \end{bmatrix}. \tag{6}$$

This is not a unique kernel representation. As noted in Section II, any other kernel representation can be obtained by left multiplication by a unimodular matrix. Here, we are specifically interested in row-reduced kernel representations; see also [3] and [2]. The reason for this is contained in the following theorem.

Theorem 7: Let $\mathbf{b}$ be defined by (5). Let the MPUM for $\{\mathbf{b}\}$ be given by

$$R(\sigma)w = 0 \quad \text{with} \quad R = \begin{bmatrix} c & -p \\ f & -g \end{bmatrix}. \tag{7}$$

Assume that $c(0) \neq 0$ and that R is row reduced with row degrees κ_1 and κ_2. If $\kappa_1 \leq \kappa_2$ and/or $f(0) = 0$, then

$$[c(\sigma) \quad -p(\sigma)]\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = 0 \tag{8}$$

represents a C-MCUM for $\{\mathbf{b}\}$.

Proof: First observe that, by Lemma 1, there exists a unimodular matrix U such that

$$R(s) = U(s)\begin{bmatrix} 1 & -(a_0 + a_1 s + \cdots + a_N s^N) \\ 0 & s^{N+1} \end{bmatrix}.$$

As a result, $R(s)$ can only lose rank at $s = 0$, so that f and g have no common zeros, except possibly at $s = 0$. We have to prove that a representation

$$[d(\sigma) \quad -n(\sigma)]\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = 0$$

that constitutes a controllable unfalsified model for $\{\mathbf{b}\}$, necessarily has row degree $\geq \kappa_1$. If

$$\det\begin{bmatrix} d & -n \\ f & -g \end{bmatrix} \neq 0 \tag{9}$$

then this would give rise to an unfalsified finite-dimensional model for $\{\mathbf{b}\}$, which necessarily has first row degree $\geq \kappa_1$, because of the assumption that R is row reduced. Now, if $f(0) = 0$, then necessarily (9) holds, since otherwise deg $[d \quad -n] < \kappa_2$, so that

$$\begin{bmatrix} c(\sigma) & -p(\sigma) \\ d(\sigma) & -n(\sigma) \end{bmatrix}\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

would be an unfalsified finite-dimensional model for $\{\mathbf{b}\}$ of order $<\kappa_1 + \kappa_2$, which contradicts the row reducedness of R. Next, if $f(0) \neq 0$, then (9) can only fail to hold if deg $[d \quad -n] = \kappa_2$. By assumption, we have that in this case $\kappa_2 \geq \kappa_1$, so that deg $[d \quad -n] \geq \kappa_1$. ■

By the above theorem, one of the rows of a row-reduced representation of the MPUM for $\{\mathbf{b}\}$ represents the C-MCUM for $\{\mathbf{b}\}$. Our approach is therefore to construct a row-reduced representation of the MPUM for $\{\mathbf{b}\}$. For this, one can proceed in two different ways.

1) Make the polynomial matrix A in (6) row reduced, i.e., construct a unimodular matrix U such that UA is row reduced.
2) Use the data $a_0, a_1, \cdots, a_N$ iteratively. At each step, construct a row-reduced representation of the MPUM corresponding to $a_1, a_2, \cdots, a_k$ $(k = 0, \cdots N)$. For this, use the iterative modeling Procedure 3, applied to $\{\sigma^N \mathbf{b}, \sigma^{N-1}\mathbf{b}, \cdots, \sigma\mathbf{b}, \mathbf{b}\}$.

For 1), the procedure of [18] can be used. The algorithm is then essentially the Euclidean algorithm, applied to the polynomials s^{N+1} and $a_0 + a_1 s + \cdots + a_N s^N$; see [11]. For 2), the row of R_k of smallest degree that does not lose rank at $s = 0$, represents the C-MCUM at step k. One can therefore think of an iterative procedure that requires a check on the value at $s = 0$ and the row degrees at each step, as in [2, p. 1795] and [3]. However, such a check is not needed if we choose the V_k's in such a way that the R_k's are not only row reduced, but also have a second row that loses rank at $s = 0$. Then, by Theorem 7, the C-MCUM at step k is *unambiguously* given by the first row of R_k. This is the clever idea behind the iterative algorithm of the next section, which is the Berlekamp–Massey algorithm.

Remark 8: If a representation (7) of the MPUM for $\{\mathbf{b}\}$ is row reduced and $f(0) = 0$, then a parameterization of all C-MCUM's, represented by

$$[d(\sigma) \quad -n(\sigma)] \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = 0$$

is given by

$$[d \quad -n] = [1 \quad \alpha] \begin{bmatrix} c & -p \\ f & -g \end{bmatrix}.$$

Here α is a polynomial with deg $\alpha = 2\kappa_1 - (N+1)$ for $\kappa_1 \geq (N+1)/2$ and $\alpha = 0$ for $\kappa_1 < (N+1)/2$. In particular, a C-MCUM is unique if and only if $\kappa_1 < (N+1)/2$. The above parameterization can also be found in [2] and [12].

IV. An Iterative Algorithm

In this section, we present our main result by working out the above-mentioned option 2) in detail.

For $k = 0, \cdots, N$, let $\mathbf{b}_k := \sigma^{N-k}\mathbf{b}$, where $\mathbf{b}$ is defined by (5). Applying Procedure 3 to the data set $\{\mathbf{b}_0, \mathbf{b}_1, \cdots, \mathbf{b}_N\}$, we get error trajectories $\boldsymbol{e}_k$, for which

$$\sigma \boldsymbol{e}_k = \sigma R_{k-1}(\sigma)\mathbf{b}_k = R_{k-1}(\sigma)\mathbf{b}_{k-1} = 0.$$

Consequently, $\boldsymbol{e}_k$ is of the simple form

$$\boldsymbol{e}_k = \left(\begin{bmatrix} \Delta_k \\ \tilde{\Delta}_k \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \cdots \right).$$

Define L_k and $\tilde{L}_k$ as the degree of the first and the second row of R_k, respectively. Define the update matrix V_k as

$$V_k(s) := \begin{bmatrix} \tilde{\Delta}_k & -\Delta_k \\ 0 & s \end{bmatrix}, \qquad \text{if} \quad \Delta_k = 0$$

$$\text{or} \quad L_{k-1} > \tilde{L}_{k-1},$$

$$V_k(s) := \begin{bmatrix} \tilde{\Delta}_k & -\Delta_k \\ s/\Delta_k & 0 \end{bmatrix}, \qquad \text{otherwise.}$$

Clearly, in both cases, $V_k(\sigma)\boldsymbol{w} = 0$ represents the MPUM for $\{\boldsymbol{e}_k\}$. Consequently, $R_k(\sigma)\boldsymbol{w} = 0$ represents the MPUM for $\{\mathbf{b}_k\}$ (Theorem 4). It can be proven by induction that $\tilde{\Delta}_k = 1$ for $k = 0, \cdots, N$. It also follows by induction that the above choice of the V_k's ensures that each R_k is row reduced $(k = -1, 0, \cdots, N)$. Indeed, R_{-1} is trivially row reduced and the assumption that R_{k-1} is row reduced, implies that R_k is row reduced. If $\Delta_k = 0$ or

$$L_{k-1} > \tilde{L}_{k-1}$$

then

$$R_k(s) = \begin{bmatrix} 1 & -\Delta_k \\ 0 & s \end{bmatrix} R_{k-1}(s) \tag{10}$$

is clearly row reduced again; note that then $L_k = L_{k-1}$. If $\Delta_k \neq 0$ and $L_{k-1} \leq \tilde{L}_{k-1}$, then

$$R_k(s) = \begin{bmatrix} 1 & -\Delta_k \\ s/\Delta_k & 0 \end{bmatrix} R_{k-1}(s)$$

is again row reduced; note that then

$$L_k = \tilde{L}_{k-1}. \tag{11}$$

Finally, it can be proven by induction that

$$R_k(0) = \begin{bmatrix} 1 & -a_0 \\ 0 & 0 \end{bmatrix}, \qquad \text{for} \quad k = 0, \cdots, N \tag{12}$$

so that, by Theorem 7, we may conclude that the first row of R_k gives rise to a C-MCUM for $\{\mathbf{b}_k\}$.

In order to be able to write the algorithm in compact form, we note that $\det R_k(s) = s^{k+1}$, so that

$$L_k + \tilde{L}_k = k + 1. \tag{13}$$

As a result, (10) coincides with the condition $L_{k-1} > k/2$, whereas (11) translates into $L_k = k - L_{k-1}$. Let us denote

$$R_k := \begin{bmatrix} c_k & -p_k \\ f_k & -g_k \end{bmatrix}.$$

By definition, the number Δ_k is the coefficient of s^k in $(a_0 + a_1 s + \cdots + a_k s^k)c_{k-1}(s) - p_{k-1}(s)$. This equals the coefficient of s^k in $(a_0 + a_1 s + \cdots + a_k s^k)c_{k-1}(s)$ because of the fact that deg $p_{k-1} < k$ [use (12) and (13)].

Below, we rewrite the above algorithm in compact form. The resulting algorithm is a slightly generalized version of the Berlekamp–Massey algorithm, in the sense that a_0 is not fixed (the Berlekamp–Massey algorithm sets $a_0 = 1$).

Algorithm 9: Denote $c_k := [1 \quad 0] R_k \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$. Initially define

$$R_{-1} := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{and} \quad L_{-1} := 0.$$

Proceed iteratively as follows for $k = 0, \cdots, N$. Define, after receiving $a_0, a_1, \cdots, a_k$, the number Δ_k as the coefficient of s^k in $(a_0 + a_1 s + \cdots + a_k s^k)c_{k-1}(s)$.

Compute the matrix R_k and the integer L_k as follows:

$$R_k := V_k R_{k-1}$$

where, if $\Delta_k = 0$ or $L_{k-1} > k/2$

$$V_k(s) := \begin{bmatrix} 1 & -\Delta_k \\ 0 & s \end{bmatrix}, \qquad L_k := L_{k-1}$$

and, if otherwise

$$V_k(s) := \begin{bmatrix} 1 & -\Delta_k \\ s/\Delta_k & 0 \end{bmatrix}, \qquad L_k := k - L_{k-1}.$$

Then the first row of R_k represents a C-MCUM for $\{\mathbf{b}_k\}$; the reciprocal of that row yields a minimal partial realization as in Theorem 6.

Example 10: Let $(a_0, a_1, a_2, a_3, a_4) = (0, 0, 1, 1, 0)$. Application of the above algorithm yields:

$$\Delta_0 = 0, \quad L_0 = 0, \quad R_0 = \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} R_{-1} = \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix}$$

$$\Delta_1 = 0, \quad L_1 = 0, \quad R_1 = \begin{bmatrix} 1 & 0 \\ 0 & s \end{bmatrix} R_0 = \begin{bmatrix} 1 & 0 \\ 0 & s^2 \end{bmatrix}$$

$$\Delta_2 = 1, \quad L_2 = 2, \quad R_2 = \begin{bmatrix} 1 & -1 \\ s & 0 \end{bmatrix} R_1 = \begin{bmatrix} 1 & -s^2 \\ s & 0 \end{bmatrix}$$

$$\Delta_3 = 1, \quad L_3 = 2$$

$$R_3 = \begin{bmatrix} 1 & -1 \\ 0 & s \end{bmatrix} R_2 = \begin{bmatrix} 1-s & -s^2 \\ s^2 & 0 \end{bmatrix}$$

$$\Delta_4 = -1, \quad L_4 = 2$$

$$R_4 = \begin{bmatrix} 1 & 1 \\ -s & 0 \end{bmatrix} R_3 = \begin{bmatrix} 1-s+s^2 & -s^2 \\ -s+s^2 & s^3 \end{bmatrix}.$$

As a result

$$[1 - \sigma + \sigma^2 \quad -\sigma^2] \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = 0$$

represents a C-MCUM for $\{\mathbf{b}\}$. Taking the reciprocal row vector, we get the (unique) minimal partial realization $1/(s^2 - s + 1)$.

V. Conclusions

The minimal partial realization problem has been considered as an instance of exact modeling of a behavior on a half-axis, as in [3]. Solutions within this framework are based on polynomials rather than Hankel matrices. A central role is played by behaviors that are the span of a finite number of trajectories and thus do not have a transfer function. It is for this reason that the notion of a behavior rather than a transfer function is essential to the approach. We put the theory to work in deriving an efficient and constructive iterative solution for the scalar case: the celebrated Berlekamp–Massey algorithm. An interesting feature of the algorithm is that its efficiency is enhanced by the update at each step of four polynomials rather than two. It is a topic of future research to put this idea to work for identification purposes, in the context of approximate modeling.

Acknowledgment

The authors would like to thank Prof. J. L. Massey for pointing out the relevance of the Berlekamp–Massey algorithm for cryptographic applications.

References

[1] A. C. Antoulas, “On recursiveness and related topics in linear systems,” *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 1121–1135, 1986.

[2] A. C. Antoulas and J. C. Willems, “A behavioral approach to linear exact modeling,” *IEEE Trans. Automat. Contr.*, vol. 38, pp. 1776–1802, 1993.

[3] A. C. Antoulas, “Recursive modeling of discrete-time time series,” in *Linear Algebra for Control Theory*, P. Van Dooren and B. Wyman, Eds., IMA vol. 62. New York: Springer-Verlag, 1994, pp. 1–20.

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

[5] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1983.

[6] G.-L. Feng and K. K. Tzeng, “A generalization of the Berlekamp-Massey algorithm for multisequence shift-register synthesis with applications to decoding cyclic codes,” *IEEE Trans. Inform. Theory*, vol. 37, pp. 1274–1287, 1991.

[7] W. B. Gragg and A. Lindquist, “On the partial realization problem,” *Lin. Alg. Appl.*, vol. 50, pp. 277–319, 1983.

[8] R. Hill, *A First Course in Coding Theory*, Oxford Appl. Math. Computing Sci. Series. Oxford: Clarendon, 1986.

[9] M. Kuijper, *First-Order Representations of Linear Systems*, Series on Syst. Contr.: Found. Appl. Boston, MA: Birkhäuser, 1994.

[10] ——, “An algorithm for constructing a minimal partial realization in the multivariable case,” *Syst. Contr. Lett.*, 1996, to be published.

[11] ——, “Partial realization and the Euclidean algorithm,” *IEEE Trans. Automat. Contr.*, to be published.

[12] J. L. Massey, “Shift-register synthesis and BCH decoding,” *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 122–127, 1969.

[13] R. A. Rueppel, *Analysis and Design of Stream Ciphers*. New York: Springer-Verlag, 1986.

[14] M. K. Sain, “Minimal torsion spaces and the partial input/output problem,” *Inform. Contr.*, vol. 29, pp. 103–124, 1975.

[15] J. C. Willems, “From time series to linear system—Part I: Finite-dimensional linear time invariant systems,” *Automatica*, vol. 22, pp. 561–580, 1986.

[16] ——, “From time series to linear system. Part II: Exact modeling,” *Automatica*, vol. 22, pp. 675–694, 1986.

[17] ——, “Paradigms and puzzles in the theory of dynamical systems,” *IEEE Trans. Automat. Contr.*, vol. 36, pp. 259–294, 1991.

[18] W. A. Wolovich, *Linear Multivariable Systems*. New York: Springer Verlag, 1974.

A Descriptor Solution to a Class of Discrete Distance Problems

M. M. M. Al-Husari, I. M. Jaimoukha, and D. J. N. Limebeer

***Abstract*—Hankel norm and Nehari-type approximation problems arise in model reduction and $\mathcal{H}_\infty$-control theory. Existing solutions to the discrete-time version of these problems may be derived using a standard state-space framework, but the resulting solution formulas require an invertible A-matrix. As a further complication, the D-matrix in the representation formula for all solutions becomes unbounded in the optimal case. The aim of this paper is to show that both these complications may be removed by analyzing these problems in a descriptor framework.**

***Index Terms*—Descriptor systems, discrete-time Nehari problem, H_∞ control, model reduction.**

I. Introduction

It is known that many model reduction and $\mathcal{H}_\infty$-control problems may be transformed into the following distance problem: let $R(z)$ be a stable real rational transfer matrix with McMillan degree n. Then for any $\gamma > 0$ and any integer $k < n$, find all transfer matrices $Q(z)$, with at most k poles inside the unit disc, that satisfy $\|R(z)+Q(z)\|_\infty \le \gamma$ [1], [6]. A necessary and sufficient condition for the existence of a solution requires $\gamma \ge (k+1)$st Hankel singular value of $R(z)$ [1], [4]. The discrete-time version of this problem has received less attention than its continuous-time counterpart. Although the discrete problem can be tackled using a standard state-space approach, this approach breaks down if $R(z)$ has poles at the origin [7], [8]. This difficulty may be traced to the fact that the conjugation operation cannot be carried out in a standard state-space framework because $R^\sim(z)$ is

Manuscript received January 15, 1993; revised July 29, 1996.

The authors are with the Department of Electrical and Electronic Engineering, Imperial College, London SW7-2BT, U.K.

Publisher Item Identifier S 0018-9286(97)07628-9.