Time-Variant and Quasi-separable Systems*
– Supplementary Reading –
.
Impulse Response and Transfer Function

Klaus Diepold, Patrick Dewilde

Spring 2025

1 State-Space Representation for LTV Systems

1.1 Properties of LTV Systems

The major properties of matrices to correspond to with a time-invariant systems are

1. The matrix T needs to be infinite dimensional.
2. The matrix T needs to have Toeplitz structure.
3. If the matrix T is finite dimensional than it has to be a circulant matrix (periodicity due to sampling).

Note that a circulant matrix represents one period of a periodic Toeplitz matrix, which appears to be finite dimensional, but which is conceptually infinite dimensional. If one of these conditions is not satisfied the theory of time-invariant systems can no longer applied. In particular this implies that Fourier techniques or other transformation tools (frequency domain, z-transforms) are no longer applicable.

Dealing with time-variant systems we need an extension of the existing theory for time-invariant systems. In particular, we need this extension to handle

- finite-dimensional matrices without circulant structure and
- matrices without Toeplitz structure.

The Toeplitz structure associated with time-invariant systems was due to the invariance of the impulse response under temporal shifts. Each impulse response, which can be measured at different time instances is recorded as a column of T. This results in T to consist of appropriately shifted versions of the first column. Obviously, if the impulse response changes from time instance to time instance so do the columns of T and hence the Toeplitz structure is no longer created. While Fourier-Theory and spectral

*P. Dewilde, K. Diepold, A.-J. v.d. Veen. Time-Variant ans Quasi-separable Systems, Cambridge University Press, 2024 representations no longer apply for time-variant systems, the system description based on state-space representation prevails.

1.2 Elements that Change with Time

Once we are leaving the domain of time-invariant systems we may ask which elements of the systems may change with time. Here is a list of such changing elements.

- We use state equations to represent linear systems. The matrices involved, i.e. the matrices A, B, C, D do not change with time for an LTI system. We can now allow the entries in these matrices to change from time instant to time instant. We notify this by adding a time index k to now have $\{A_k, B_k, C_k, D_k\}$.
- Once we allow the matrix entries to change with time, we can also allow the dimension of the state-vector x to change with time. So, as time progresses the number of entries in the state-vector x_k may increase or shrink, to denote the time-variant character of the system dynamics. We can use the sequence d_k to denote the dimension of the state-space at time index k, i.e. we have $x_k \in \mathcal{R}^{d_k}$.
- Once the dimension of the state-vector may change with time, this also induces that the dimension of the matrices $\{A_k, B_k, C_k, D_k\}$ may change accordingly. If the dimension of the state-vector increases with time, then obviously the matrix A_k will be tall, i.e. it will comprise more rows than columns.

  If the dimension of the state-vector stays constant, then A_k will be square and if he length of x_k is shrinking then the matrix A_k will be wide, i.e. it has more columns than rows. As a consequence, the matrix A_k will have the dimension $d_{k+1} \times d_k$. Similarly, the matrix B_k will have d_{k+1} rows and the matrix C_k will d_k columns.
- We can also allow the dimensions of the input vectors and the output vectors to change with time, i.e. the individual samples of $u_k \in \mathcal{R}^{m_k}$ or $y_k \in \mathcal{R}^{n_k}$ may be scalar valued or vector valued with dimensions changing from time to time (MIMO- systems). Hence, the number of columns n_k in B_k and the number of rows m_k in C_k may change synchronously to the changing dimensions of u_k or y_k.

2 LTV State-Equations

We move from time-invariant state-space equations to time-variant state-space equations, taking into account the items discussed in the previous section.

Based on the algebraic approach to describe linear systems we identify the conditions that need to be satisfied to use the theory for time-invariant systems. We now look at

$$\begin{aligned} x_{k+1} &= A_k \cdot x_k + B_k \cdot u_k \\ y_k &= C_k \cdot x_k + D_k \cdot u_k, \end{aligned} \tag{1}$$

or, written in a slightly more compactly matrix form, we have

$$\begin{bmatrix} x_{k+1} \\ y_k \end{bmatrix} = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right] \cdot \begin{bmatrix} x_k \\ u_k \end{bmatrix}. \tag{2}$$

These state-space equations are similar to the equations for the time-invariant case except that now the entries $\{A_k, B_k, C_k, D_k\}$ of the realization matrix are depending on the time index k, which means that these matrices can change from time step to time step, incorporating the time-variation that we are looking for.

Based on these time-variant state-equations we can derive a generic computational model for a causal time-variant system as shown in Figure 1.

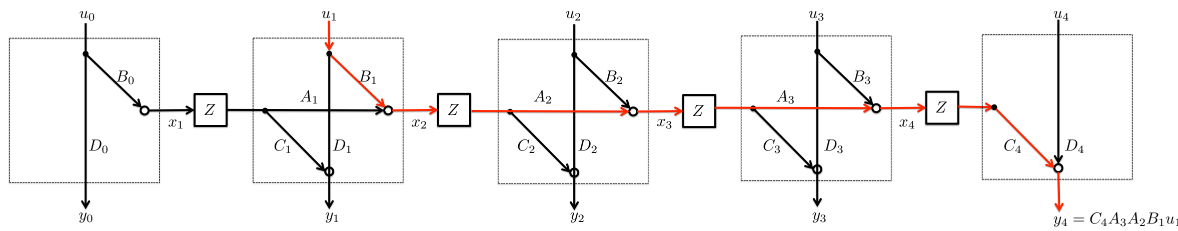

Figure 1: A time-variant system

We recall that besides the time-variation of the system parameters $\{A_k, B_k, C_k, D_k\}$ we can see that the dimension of the state-space may change from time step to time step and that the dimension of the input signal vector u_k and the dimension of the output signal vector y_k may also change with time. So, in the computational model shown in Figure 1 the individual lines carrying signals may just as well represent vector valued signals, or busses, if you prefer a more technical term. As long as the dimensions of matrices and vectors match to allow for well-defined matrix-vector multiplications we are save.

Take a short moment to observe in Figure 1 that the first and the last box does not have a signal path denoted with an A-matrix (A_1 and A_4 are missing). Also, the first box lacks a signal path labeled with a C-matrix (no C_1 and the last box is lack a B-matrix (no B_4). Looking at the computational model it becomes obvious why these matrices are missing, that this is a result of the model to begin and to end, respectively.

3 Time-Variant Impulse Response

Using the computational model of a time-variant system as shown in Figure 1 we seek to determine the I/O operator, i.e. the Toeplitz operator for this time-variant causal system.

We proceed in a similar fashion as with the time-invariant case, by applying an impulse at time instant $k = 0$ and recording the output of the system. In the time-variant case, the impulse response t_k depends on the time k when we injected the impulse. We start recording the impulse response to the input impulse at time index k as the column k in the matrix T. Of course, we assume that the system is initialized with all state-vectors set to 0. By progressing like this, which means that we are following the input signal as it progress through the computation model and writing down all the operations that apply along the way. We arrive at

$$T = \begin{bmatrix} D_0 & \cdot & \cdot & \cdot & \cdot \\ C_1B_0 & D_1 & \cdot & \cdot & \cdot \\ C_2A_1B_0 & C_2B_1 & D_2 & \cdot & \cdot \\ C_3A_2A_1B_0 & C_3A_2B_1 & C_3B_2 & D_3 & \cdot \\ C_4A_3A_2A_1B_0 & \boxed{C_4A_3A_2B_1} & C_4A_3B_2 & C_4B_3 & D_4 \end{bmatrix} \tag{3}$$

The boxed matrix entry $C_4A_3A_2B_1$ shown in Equation 3 corresponds to following the red path drawn in Figure 1 determining the value of the response y_4 to an impulse u_1 injected at time index $k = 1$.

We can note from the above that

- i^{th} column represents the impulse response of the system for an impulse input at $k = i$, i.e.

$$u_i = \begin{cases} 1 & i = k, \\ 0 & else \end{cases}$$

- When collecting the impulse response we assume the states to be initialized to 0.
- T has the same functionality as Toeplitz matrix in the time-invariant case. In fact, if the system is time-invariant, all the matrices A_k, B_k, C_k, D_k will not change with time, hence T will exhibit Toeplitz structure.
- We can identify the system to be causal by lower-triangular structure of T.
- The clear difference is that T has no obvious, i.e. no clearly visible Toeplitz structure.
- Because of its time-variant nature, the dimension of the matrix T may be finite. The matrix has a beginning and an end, both points indicating a change in the temporal behaviour of the system.
- The matrices A_0, A_4, C_0 and B_4 do not exist in the Toeplitz operator (3). The may be denoted by empty matrices, i.e. matrices which have zero dimension, i.e. we look at 0×0-matrices or $m \times 0$ or $0 \times n$.

In the matrix T we see the entries containing the the products $A_{i+n-1} \cdot \ldots \cdot A_{i+1} \cdot A_i$. For the matrix T to be well defined even in very large to infinite dimensional cases, we define the spectral radius

$$\ell_A = \lim_{n\to\infty} \sup_i \|A_{i+n-1} \cdot \ldots \cdot A_{i+1} \cdot A_i\|^{1/n}. \tag{4}$$

This convention on the spectral radius is closely related to the notion of spectral radius for time-invariant systems . There, if the spectral radius fullfills $|A| < 1$ then the condition of stability is satisfied. For the time-variant case, the realization and hence T is said to be *exponentially stable* iff $\ell_A < 1$.

4 Time-Variant Transfer Function

4.1 The Causal Shift Operator

4.1.1 The Infinite Dimensional Case

We introduce the causal *Shift operator*, which we denote by the symbol Z. The Shift operator has the function of shifting all entries of a vector downward by one position. We can give this downward shift the interpretation of a temporal delay, that is,

$$x_k = Z \cdot x_{k+1}, \quad k = -\infty, \ldots, \infty.$$

For our matrix-oriented notation we will use the matrix representation for the shift operator, which acts from the left on a given vector u and pushes all its entries down by one notch

$$Z \cdot x = Z \cdot \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} \mapsto \begin{bmatrix} \vdots \\ x_{-2} \\ \boxed{x_{-1}} \\ x_0 \\ x_1 \\ \vdots \end{bmatrix}, \qquad Z^{-1} x = Z^{-1} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} \mapsto \begin{bmatrix} \vdots \\ x_0 \\ \boxed{x_1} \\ x_2 \\ x_3 \\ \vdots \end{bmatrix}. \tag{5}$$

The rectangular box indicates our reference of the time origin. Likewise, the inverse also holds, that is, Z^{-1} acting on the vector x pushes the entries of the vector upwards by one notch. It is worth noting that with *infinite dimensional* vectors and matrices, the shift operator Z is orthogonal, i.e. $Z'Z = ZZ' = I$, since for infinite dimensional vectors, shifting up and down does not truncate the vectors.

The matrix Z itself is actually an infinite dimensional lower (causal) matrix

$$Z = \begin{bmatrix} \ddots & & & & & \\ \ddots & 0 & & & & \\ & 1 & 0 & & & \\ & & 1 & 0 & & \\ & & & 1 & 0 & \\ & & & & \ddots & \ddots \end{bmatrix}, \qquad Z^{-1} = Z' = \begin{bmatrix} \ddots & \ddots & & & & \\ & 0 & 1 & & & \\ & & 0 & 1 & & \\ & & & 0 & 1 & \\ & & & & 0 & \ddots \\ & & & & & \ddots \end{bmatrix}.$$

Similarly, the shift operator and its inverse can also be applied to row vectors from the right $x' \cdot Z$

$$\begin{bmatrix} \cdots & x_{-1} & \boxed{x_0} & x_1 & x_2 & \cdots \end{bmatrix} \cdot Z = \begin{bmatrix} \cdots & x_{-1} & x_0 & \boxed{x_1} & x_2 & \cdots \end{bmatrix},$$

shifting all vector entries one position to the left. Accordingly, we achieve a shift by one position to the right by post-multiplication of the row vector x' with the inverse shift operator, that is, we then have

$$\begin{bmatrix} \cdots & x_{-1} & \boxed{x_0} & x_1 & x_2 & \cdots \end{bmatrix} \cdot Z' = \begin{bmatrix} \cdots & x_{-2} & \boxed{x_{-1}} & x_0 & x_1 & \cdots \end{bmatrix}.$$

Taking the k^{th} power of Z shifts the vector entries by k positions.

We can also apply the shift operator simultaneously from the left and from the right to a matrix A such as ZAZ', which has the effect of pushing the entries of A south-east along the diagonal, which looks like

$$Z \cdot \begin{bmatrix} \boldsymbol{A} \end{bmatrix} \cdot Z' = \begin{bmatrix} \ddots & \\ & \boldsymbol{A} \end{bmatrix}.$$

We can push down the matrix along its diagonal by k slots if we apply the shift operators from the left and from the right k times like $Z^k A (Z')^k$.

4.1.2 The Finite Dimensional Case

In the finite dimensional case we consider a finite version of the matrix Z of dimension $N \times N$. The finite dimensional causal shift operator of dimension N satisfies $Z^N = 0$, i.e. it is a nilpotent matrix. For N=5 we have the example

$$Z = \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & 1 & 0 & \\ & & & 1 & 0 \end{bmatrix}.$$

It is also obvious that this type of shift operator is no longer orthogonal, it is singular. Shifting upwards is still accomplished by the transpose Z'. Shifting a vector first up and then down will create a zero at the top of the vector, i.e. data gets lost.

4.1.3 The Cyclic Shift

In the finite dimensional case we can also consider a cyclic shift operator Z of dimension $N \times N$. The cyclic shift operator of dimension N satisfies $Z^N = I$. For N=5 we have the example

$$Z = \begin{bmatrix} 0 & & & & 1 \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & 1 & 0 & \\ & & & 1 & 0 \end{bmatrix}.$$

The cyclic shift operator is orthogonal. When shifting the entries of a vector down, the bottom element that is about to be dropped from the matrix will be re-injected at the top of the next column. The cyclic shift operator is of particular importance for the Discrete Fourier Transformation.

4.2 Block Diagonal Representation of State Space Equations

We aim to derive a purely algebraic representation of the input-output operator T. We will call this input-output operator the *Toeplitz Operator*, even though it may not exhibit the typical Toeplitz structure that we know from time-invariant systems.

We combine the set of all time-variant state-space realization matrices into block diagonal matrices

$$A = \begin{bmatrix} \ddots & & \\ & \boxed{A_k} & \\ & & \ddots \end{bmatrix} B = \begin{bmatrix} \ddots & & \\ & \boxed{B_k} & \\ & & \ddots \end{bmatrix} C = \begin{bmatrix} \ddots & & \\ & \boxed{C_k} & \\ & & \ddots \end{bmatrix} D = \begin{bmatrix} \ddots & & \\ & \boxed{D_k} & \\ & & \ddots \end{bmatrix}.$$

Note that the dimension of the individual block matrices may change from time index to time index $A_k \in \mathcal{R}^{d_{k+1} \times d_k}$, $B_k \in \mathcal{R}^{d_{k+1} \times m_k}$, $C_k \in \mathcal{R}^{n_k \times d_k}$, $D_k \in \mathcal{R}^{m_k \times n_k}$ Correspondingly, we combine all input-, output- and state-signals to form the vectors

$$x = \begin{bmatrix} \vdots \\ x_{k-1} \\ \boxed{x_k} \\ x_{k+1} \\ \vdots \end{bmatrix}, x_k \in \mathcal{R}^{d_k}, \qquad u = \begin{bmatrix} \vdots \\ u_{k-1} \\ \boxed{u_k} \\ u_{k+1} \\ \vdots \end{bmatrix}, u_k \in \mathcal{R}^{m_k}, \qquad y = \begin{bmatrix} \vdots \\ y_{k-1} \\ \boxed{y_k} \\ y_{k+1} \\ \vdots \end{bmatrix}, y_k \in \mathcal{R}^{n_k}$$

where the parameters d_k, m_k and n_k (state dimension, number of inputs, number of outputs, respectively) and hence the length of the corresponding vectors may change from time step to time step.

Using these block-diagonal matrices, the vectors and the causal shift-down operator Z we can combine the state space equations for all time indexes k as

$$\begin{bmatrix} \vdots \\ x_k \\ \boxed{x_{k+1}} \\ x_{k+2} \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & & & \\ & A_{k-1} & & & \\ & & \boxed{A_k} & & \\ & & & A_{k+1} & \\ & & & & \ddots \end{bmatrix} \cdot \begin{bmatrix} \vdots \\ x_{k-1} \\ \boxed{x_k} \\ x_{k+1} \\ \vdots \end{bmatrix} + \begin{bmatrix} \ddots & & & \\ & B_{k-1} & & & \\ & & \boxed{B_k} & & \\ & & & B_{k+1} & \\ & & & & \ddots \end{bmatrix} \cdot \begin{bmatrix} \vdots \\ u_{k-1} \\ \boxed{u_k} \\ u_{k+1} \\ \vdots \end{bmatrix},$$

or, even in a a more compact way as

$$Z^{-1}x = A \cdot x + B \cdot u. \tag{6}$$

Similarly, we can combine the output equations for all time indexes k as

$$\begin{bmatrix} \vdots \\ y_{k-1} \\ \boxed{y_k} \\ yx_k \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & & & \\ & C_{k-1} & & & \\ & & \boxed{C_k} & & \\ & & & C_{k+1} & \\ & & & & \ddots \end{bmatrix} \cdot \begin{bmatrix} \vdots \\ x_{k-1} \\ \boxed{x_k} \\ x_{k+1} \\ \vdots \end{bmatrix} + \begin{bmatrix} \ddots & & & \\ & D_{k-1} & & & \\ & & \boxed{D_k} & & \\ & & & D_{k+1} & \\ & & & & \ddots \end{bmatrix} \cdot \begin{bmatrix} \vdots \\ u_{k-1} \\ \boxed{u_k} \\ u_{k+1} \\ \vdots \end{bmatrix},$$

which we also can denote compactly as

$$y = C \cdot x + D \cdot u. \tag{7}$$

Our goal is to formulate the transfer function, i.e. a pure input-output description of a time-variant system. To this end we need to eliminate the internal state vectors x from the equations. Starting out from equation 6 we can separate the state vector x as

$$x = (I - ZA)^{-1} ZBu$$

and plug this expression into the output equation 7 for the output signal y. The resulting expression for the output only depends on the input signal u and on the parameters of the state-representation

$$y = \left[D + C\,(I - ZA)^{-1} ZB\right] u.$$

Looking at this result we can identify a representation of the input-output, or Toeplitz operator in terms of the linear fractional transformation (LFT)

$$T = D + C\,(I - ZA)^{-1} ZB, \tag{8}$$

which is given in terms of the state space realization matrices and the shift operator.

This representation of the Toeplitz operator (input-output operator) has an obvious structural resemblance with the standard transfer function representation for time-invariant state-space systems. However, note that Z is the matrix representation of the shift operator and not a complex variable; the state-space matrices A, B, C and D are block-diagonal matrices.

4.3 Neuman Expansion

A look at Equation (8) does not immediately reveal that this linear fractional transformation based on block-diagonal matrices does actually represent the Toeplitz operator

In this section we actually plug in the block-diagonal matrices in (8) to convince ourselfes of the correctness of this representation. We consider the the block-diagonal matrices for the case $N = 5$

$$A = \begin{bmatrix} A_0 & & & & \\ & A_1 & & & \\ & & A_2 & & \\ & & & A_3 & \\ & & & & A_4 \end{bmatrix}, \quad B = \begin{bmatrix} B_0 & & & & \\ & B_1 & & & \\ & & B_2 & & \\ & & & B_3 & \\ & & & & B_4 \end{bmatrix},$$

$$C = \begin{bmatrix} C_0 & & & & \\ & C_1 & & & \\ & & C_2 & & \\ & & & C_3 & \\ & & & & C_4 \end{bmatrix}, \quad D = \begin{bmatrix} D_0 & & & & \\ & D_1 & & & \\ & & D_2 & & \\ & & & D_3 & \\ & & & & D_4 \end{bmatrix},$$

and the shift operator

$$Z = \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & 1 & 0 & \\ & & & 1 & 0 \end{bmatrix}.$$

We can safely assume that the dimensions of the block-entries of all matrices are compatible to make the matrix multiplications meaningful. We first have a closer look at the middle part of Equation (8), which is the expression $(I - ZA)^{-1}$. This expression we can re-write as the Neuman expansion[1]

$$(I - ZA)^{-1} = \sum_{k=0}^{\infty} (ZA)^k = I + ZA + (ZA)^2 + (ZA)^3 + \ldots.$$

For this inifinite series expansion to converge the condition $|ZA| < 1$ (spectral radius) needs to be satisfied. In this expansion we see the powers of the term ZA, which is the lower bi-diagonal (block) matrix. One may be tempted to abbreviate the representation by writing $(ZA)^2 = ZAZA$ as $Z^2A^2 = ZZAA$. This is not permissible because in time-variant systems the Shift operator does not commute with the diagonal matrix A. This only works in time-invariant systems. There, this property is used as a defining

[1]Notice that this expansion is very simiilar to the well-known formula for the geometric series $\sum_{k=0}^{k} q^k = \frac{1}{1-q}, |q| < 1$.

characteristic for time-invariance, i.e. if the Shift operator commutes with A matrix, then the system is time-invariant. We can visualize this fact easily in the following way.

$$Z \cdot A = \begin{bmatrix} A & & & & \\ \downarrow & A & & & \\ & \downarrow & A & & \\ & & \downarrow & A & \\ & & & \downarrow & A \end{bmatrix} = \begin{bmatrix} 0 & & & & \\ A & 0 & & & \\ & A & 0 & & \\ & & A & 0 & \\ & & & A & 0 \end{bmatrix} = \begin{bmatrix} A & & & & \\ \leftarrow & A & & & \\ & \leftarrow & A & & \\ & & \leftarrow & A & \\ & & & \leftarrow & A \end{bmatrix} = A \cdot Z.$$

If the entries of the block diagonal matrix A are different, this commutative property is lost.

For the purpose of determining the transfer function for a time-variant systems we can direclty compute the individual terms of the Neuman expansion as

$$ZA = \begin{bmatrix} 0 & & & & \\ A_0 & 0 & & & \\ & A_1 & 0 & & \\ & & A_2 & 0 & \\ & & & A_3 & 0 \end{bmatrix}, \quad (ZA)^2 = ZAZA = \begin{bmatrix} 0 & & & & \\ 0 & 0 & & & \\ A_1A_0 & 0 & 0 & & \\ & A_2A_1 & 0 & 0 & \\ & & A_3A_2 & 0 & 0 \end{bmatrix}$$

$$(ZA)^3 = \begin{bmatrix} 0 & & & & \\ 0 & 0 & & & \\ 0 & 0 & 0 & & \\ A_2A_1A_0 & 0 & 0 & 0 & \\ & A_3A_2A_0 1 & 0 & 0 & 0 \end{bmatrix}, \quad (ZA)^4 = \begin{bmatrix} 0 & & & & \\ 0 & 0 & & & \\ 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & \\ A_3A_2A_1A_0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

With these individual expressions for the elements in the series expansion we can write

$$(I - ZA)^{-1} = \begin{bmatrix} 1 & & & & \\ A_0 & 1 & & & \\ A_1A_0 & A_1 & 1 & & \\ A_2A_1A_0 & A_2A_1 & A_2 & 1 & \\ A_3A_2A_1A_0 & A_3A_1A_1 & A_3A_2 & A_3 & 1 \end{bmatrix}.$$

In the finite dimensional case we can see that the series expansion will come to an end after N terms, as $Z^N = 0$ is producing only zeros. That implies that the convergence of the series expansion is a lesser problem compared to the iniinite dimensional case. In out current context, we are not considering the infinite dimensional case.

5 Putting it all together

In the previous section we observed how the term $(I - ZA)^{-1}$ can be evaluated in terms of a series expansion. On the way to the transfer function we also see the term ZB which amounts to

$$ZB = \begin{bmatrix} 0 & & & & \\ B_0 & 0 & & & \\ & B_1 & 0 & & \\ & & B_2 & 0 & \\ & & & B_3 & 0 \end{bmatrix}.$$

So we have the individual components of the transfer function $T = D + C(I - ZA)^{-1}ZB$ readily available to compute

$$
\begin{bmatrix} C_0 & & & & \\ & C_1 & & & \\ & & C_2 & & \\ & & & C_3 & \\ & & & & C_4 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & & \\ A_0 & 1 & & & \\ A_1A_0 & A_1 & 1 & & \\ A_2A_1A_0 & A_2A_1 & A_2 & 1 & \\ A_3A_2A_1A_0 & A_3A_2A_1 & A_3A_2 & A_3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & & & & \\ B_0 & 0 & & & \\ & B_1 & 0 & & \\ & & B_2 & 0 & \\ & & & B_3 & 0 \end{bmatrix} .
$$

$$
= \begin{bmatrix} 0 & & & & \\ C_1B_0 & 0 & & & \\ C_2A_1B_0 & C_2B_1 & 0 & & \\ C_3A_2A_1B_0 & C_3A_2B_1 & C_2B_1 & 0 & \\ C_4A_3A_2A_1B_0 & C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & 0 \end{bmatrix}
$$

Taking this intermediate result we can finally generate the Toeplitz operator according to Equation 6 by adding the block-diagonal D to conclude the construction for the causal transfer function T, i.e. we arrive at

$$
T = \begin{bmatrix} D_0 & & & & \\ C_1B_0 & D_1 & & & \\ C_2A_1B_0 & C_2B_1 & D_2 & & \\ C_3A_2A_1B_0 & C_3A_2B_1 & C_3B_2 & D_3 & \\ C_4A_3A_2A_1B_0 & C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & D_4 \end{bmatrix}
$$

Comparing this most recent result with the matrix we've constructed when collecting the impulse response measurements of an LTV systems (see Equation (3), we recognize that both matrices are identical! This identity should make clear that the characterization of a linear system in terms of impulse responses and in terms of a transfer function is not only equivalent, it is even identical. Both representations are different ways to represent one and the same thing. Of course, this statement is also true for time-invariant systems.

We can recognize that we have successfully represented the causal time-variant Toeplitz operator by means of the transfer function formula. This construction is entirely based on purely algebraic elements and no transformation into any type of frequency domain happens. The transfer function formula is nothing but a compact and convenient representation of the Toeplitz operator using just block-.diagonal matrices and the Shift operator.

Literatur

[1] G. Strang. *Computational Science and Engineering.* Wellesley-Cambridge Press, 2007.

[2] P. Dewilde, A.-J. van der Veen. *time-variant Systems and Computations.* Kluwer Academic Publishers, 1998.

[3] P. Dewilde, K. Diepold, A.-J. van der Veen. *Time-Variant and Quasi-separable Systems.* Cambridge University Press, 2024.