AEÜ, Vol. 49 (1995), No. 4/5 © Hirzel-Verlag Stuttgart

Time-Varying Lossless Systems and the Inversion of Large Structured Matrices

Alle-Jan van der Veen

Time-Varying Lossless Systems and the Inversion of Large Structured Matrices

In the inversion of large matrices, direct methods might give undesired 'unstable' results. Valuable insight into the mechanism of this effect is obtained by viewing the matrix as the input-output operator of a time-varying system, which allows to translate 'unstable' into 'anti-causal' but bounded inverses. Inner-outer factorizations and other lossless factorizations from system theory play the role of QR factorizations. They are computed by state space techniques and lead to a sequence of QR factorizations on time-varying realization matrices. We show how several such results can be combined to solve the inversion problem.

Zeitvariante verlustlose Systeme und die Inversion großer strukturierter Matrizen

Direkte Methoden ergeben bei der Inversion großer Matrizen m"oglicherweises 'instabile' Ergebnisse. Wertvolle Einsichten in den Mechanismus dieses Effektes erhält man durch die Auffassung der Matrix als Eingangs-/Ausgangs-Operator eines zeitvarianten Systems. Hierdurch werden 'instabile' in 'antikausale', aber beschränkte Inverse umgesetzt. Inner/Outer-Zerlegungen und andere verlustlose Faktorisierungen der linearen Systemtheorie übernehmen hierbei die Rolle der QR-Zerlegung. Sie werden auf der Basis von Zustandsmodellen berechnet und führen auf eine Folge von QR-Zerlegungen zeitvarianter Realisierungsmatrizen. Wir zeigen, wie aus solchen Ergebnissen eine Lösung des Inversionsproblems konstruiert werden kann.

Keywords: Large matrix inversion, time-varying systems, inner-outer factorization.

1. Introduction

The inversion of large structured matrices is a delicate problem which often arises in finite element modeling applications, or (implicitly) in non-stationary inverse filtering problems in signal processing. To stress the fact that these matrices might be fairly large and even so large that ordinary linear algebra techniques might fail, we allow them to have infinite size, i.e., they are operators on the space of ℓ_2-sequences. We study some of the ways in which system theory and state space techniques can assist in the inversion problem. To set the scene, consider the infinite Toeplitz matrix

$$T = \begin{bmatrix} \ddots & \ddots & & & \\ & \boxed{1} & -1/2 & & \mathbf{0} \\ & & 1 & -1/2 & \\ & & & 1 & -1/2 \\ & \mathbf{0} & & & 1 & \ddots \\ & & & & & \ddots \end{bmatrix}. \qquad (1)$$

The position $(0,0)$ of T is indicated by a square. The inverse of T is given by

$$T^{-1} = \begin{bmatrix} \ddots & \vdots & & & \vdots & \\ & \boxed{1} & 1/2 & 1/4 & 1/8 & \cdots \\ & & 1 & 1/2 & 1/4 & \\ & & & 1 & 1/2 & \\ & \mathbf{0} & & & 1 & \cdots \\ & & & & & \ddots \end{bmatrix},$$

as is readily verified: $TT^{-1} = I$, $T^{-1}T = I$. One way to obtain T^{-1} in this case is to restrict T to a finite matrix and invert this matrix. For example,

$$\begin{bmatrix} 1 & -1/2 & 0 \\ 0 & 1 & -1/2 \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1 & 1/2 \\ 0 & 0 & 1 \end{bmatrix}.$$

In general, however, this does not always give correct results.

Another way to obtain T^{-1}, perhaps more appealing to engineers, goes via the z-transform:

$$\begin{aligned} T(z) &= 1 - \frac{1}{2}z \\ \Rightarrow \quad T^{-1}(z) &= \frac{1}{1-\frac{1}{2}z} = 1 + \frac{1}{2}z + \frac{1}{4}z^2 + \cdots. \end{aligned}$$

The expansion is valid at least for $|z| = 1$.

What happens if we now take

$$T = \begin{bmatrix} \ddots & \ddots & & & \\ & \boxed{1} & -2 & & \mathbf{0} \\ & & 1 & -2 & \\ & & & 1 & -2 \\ & \mathbf{0} & & & 1 & \ddots \\ & & & & & \ddots \end{bmatrix} \qquad (2)$$

Received February 1, 1995.

Dr. A. J. van der Veen, Delft University of Technlogy, Department Electrical Engineering, Mekelweg 4, NL-2628 CD Delft, The Netherlands, E-mail: allejan@cas.et.tudelft.nl.

and treat it in the same way? In that case, we obtain

$$T^{-1} = \begin{bmatrix} \ddots & \vdots & & & \vdots & \\ & \boxed{1} & 2 & 4 & 8 & \cdots \\ & & 1 & 2 & 4 & \\ & & & 1 & 2 & \\ & \mathbf{0} & & & 1 & \cdots \\ & & & & & \ddots \end{bmatrix},$$
$$T^{-1}(z) = 1 + 2z + 4z^2 + \cdots .$$

Thus, T^{-1} is *unbounded*, and the series expansion for $T^{-1}(z)$ is not valid for $|z| = 1$. The correct, bounded inverse is obtained via

$$T^{-1}(z) = \frac{1}{1-2z} = \frac{-\frac{1}{2}z^{-1}}{1-\frac{1}{2}z^{-1}} = $$
$$= -\frac{1}{2}z^{-1} - \frac{1}{4}z^{-2} - \cdots$$

$$\Rightarrow T^{-1} = \begin{bmatrix} \ddots & \ddots & & & & \\ \cdots -1/2 & \boxed{0} & & & \mathbf{0} & \\ -1/4 & -1/2 & 0 & & & \\ -1/8 & -1/4 & -1/2 & 0 & & \\ \cdots -1/16 & -1/8 & -1/4 & -1/2 & \ddots & \\ \vdots & & & \vdots & & \ddots \end{bmatrix}. \quad (3)$$

Again, it is readily verified that $TT^{-1} = I$, $T^{-1}T = I$. Moreover, this inverse is *bounded*. It is seen that, for infinite matrices (operators), the inverse of an upper operator need not be upper. In the light of finite dimensional linear algebra, this seems to be a strange result. An intuitive explanation is that, because the matrix is so large, the location of the main diagonal is not clear: a shift of the whole matrix over one (or a few) positions should be allowed and should only give a similar (reverse) shift in the inverse. For example, T^{-1} is obtained from finite matrices after shifting the origin over one position:

$$\begin{bmatrix} -2 & 0 & 0 \\ 1 & -2 & 0 \\ 0 & 1 & -2 \end{bmatrix}^{-1} = \begin{bmatrix} -1/2 & 0 & 0 \\ -1/4 & -1/2 & 0 \\ -1/8 & -1/4 & -1/2 \end{bmatrix}.$$

A better explanation is to say that $T(z)$ is not an *outer function*, that is to say non-minimum phase, and hence $T^{-1}(z)$ is not causal, which translates to a lower triangular matrix representation.

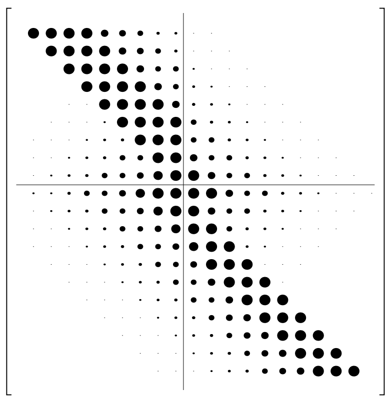
The above example gives a very elementary insight in how system theory (i.e. the z-transform) can help in the bounded inversion of large matrices. The examples so far were cast in a time-invariant framework: all matrices were Toeplitz. We now go beyond this and consider general matrices and their connection to *time-varying* systems. An illustrative example is provided by the combination of the above two cases:

$$T = \left[\begin{array}{cccc|cccc} \ddots & \ddots & & & & & & \\ & 1 & -1/2 & & & & \mathbf{0} & \\ & & 1 & & -1/2 & & & \\ \hline & & & & \boxed{1} & -2 & & \\ & & & & & 1 & -2 & \\ & & & & & & 1 & -2 \\ & \mathbf{0} & & & & & & 1 \;\ddots \\ & & & & & & & \ddots \end{array}\right]. \quad (4)$$

Is T^{-1} upper? But then it will be unbounded:

$$\left[\begin{array}{ccc|ccccc} \ddots & \vdots & & \vdots & & \vdots & & \\ & 1 & 1/2 & 1/4 & 1/2 & 1 & 2 & \cdots \\ & & 1 & 1/2 & 1 & 2 & 4 & \\ \hline & & & \boxed{1} & 2 & 4 & 8 & \cdots \\ & & & & 1 & 2 & 4 & \\ & \mathbf{0} & & & & 1 & 2 & \\ & & & & & & 1 & \cdots \\ & & & & & & & \ddots \end{array}\right].$$

Something similar happens if we opt for a lower triangular matrix representation. A bounded T^{-1} (if it exists!) will most likely be a combination of upper (the top-left corner) and lower (the bottom-right corner), and some unknown interaction in the center: something like

$T^{-1} =$  (5)

The purpose of this paper is to do away with speculations, and to give precise answers to such questions, using time-varying system theory.

There are several applications that could benefit from such a theory.

1) *Time-varying filter inversion*: e.g., T in (4) could represent an adaptive FIR filter, with a zero that moves from $z = 2$ to $z = 1/2$. Think e.g. of an adaptive channel estimate that has to be inverted to retrieve the input signal from an observed output signal [1]. As the example shows, a direct inversion might lead to unstable results.
2) *Finite element matrix inversion*: Finite element matrices are often very large, and hence the effects observed above might play a role. Presently, stability of the inverse is ensured by careful selection of boundary conditions: the borders of the matrix are chosen such that its inverse (as determined by finite linear algebra) is well behaved. Time-varying techniques might

give additional insight. It is even envisioned that one might do without explicit boundary conditions: extend T to an infinite matrix, which is constant (Toeplitz) towards $(-\infty, -\infty)$ and $(+\infty, +\infty)$. LTI systems theory gives explicit starting points for inversion recursions. It is even possible to "zoom in" on a selected part of T^{-1}, without computing all of it.

3) System inversion also plays a role in control, e.g., the manipulation of a flexible robot-arm [2].

The key idea which provides the connection of matrices T and the rich field of system theory is that we view T as the input-output matrix of a linear time-varying system, mapping input signals (vectors) to output signals (vectors). T is supposed to have a certain *structure*, which allows us to obtain a time-varying state space representation of the system with a low state dimension. This stucture is fairly general. For example, a banded matrix has a state representation where the number of states is equal to the width of the band. Moreover, even though the inverse of a band matrix is not sparse, it has the same number of states as the original matrix, and hence the state representation is a very efficient way to specify this inverse. Such results are already partly known: e.g. for a three-diagonal matrix, the inverse can be computed by a well-known three-term recursion. Our results generalize on this. All computations are performed in a state space context, and they are computationally efficient if the state dimension is low.

In an operator theoretic context, inversion is of course a solved problem. Time-varying systems have been formulated in terms of a nest algebra, for which factorization and inversion results have been presented among others by Arveson [3]. The key ingredient is an *inner-outer factorization*, which can be viewed as a QR factorization on operators. However, it is not clear from Arveson's paper how these abstract results translate to practical algorithms. This was the motivation for additional work. The time-varying inner-outer factorization provides a splitting into causal (upper) and anti-causal (lower) parts: a dichotomy. In the connection with time-varying state space theory, it has been investigated by Gohberg and coworkers [4], [5]. State space algorithms for inner-outer factorizations lead, not surprisingly, to time-varying Riccati recursions [6], and can be computed as well via a QR recursion on state space matrices. In this paper, we collect several of these results and apply them to the problem of matrix inversion.

2. Lossless Factorizations and Operator Inversion

2.1 Time-Varying Systems

Let $T = [T_{ij}]$ be a (finite) matrix or (infinite) operator, with entries T_{ij}. For additional generality, we allow T to be a block matrix so that its entries are matrices themselves: T_{ij} is an $m_i \times n_j$ matrix, where the dimensions m_i and n_j are finite but need not be constant over i and j. They may even be equal to zero at some points, so that finite matrices fit in the same (infinite) framework. A connection with system theory is obtained by viewing a row vector as a signal sequence in discrete time. The multiplication of such a sequence by this operator,

$$[\cdots \ \boxed{y_0} \ y_1 \ \cdots] = [\cdots \ \boxed{u_0} \ u_1 \ \cdots]T,$$

is the mathematical description of the application of a linear system to the signal represented by u: T is the input-output operator of the system. The ith row of the operator is then the impulse response of the system due to an impulse at time i, i.e., an input vector $u = [\cdots \ 0 \ \ 1_i \ \ 0 \ \cdots]$. The system is causal if the operator is block upper, and anti-causal if it is lower.

For mathematical convenience, only signals that have bounded energy are admitted: row vectors are in ℓ_2. Systems have to be bounded as $\ell_2 \to \ell_2$-operators. This puts our theory in a Hilbert space context. We define

$\mathcal{X}$: the space of bounded $\ell_2 \to \ell_2$ operators,
$\mathcal{U}$: the space of bounded upper operators

$$\mathcal{U} = \{T \in \mathcal{X} : T_{ij} = 0 \ (i > j)\},$$

$\mathcal{L}$: the space of bounded lower operators.

2.2 Inner-Coprime Factorization

Let $(\cdot)^*$ denote a complex conjugate transpose (Hermitian conjugate). An operator U is *left isometric* if $U^*U = I$, *right isometric* if $UU^* = I$, and *unitary* if both properties hold. U is *inner* if it is both upper and unitary. A prime example of an inner operator is the shift operator Z:

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

The inverse of a unitary operator U is U^*. This shows that the inverse of an inner operator is not upper, but lower. (In ordinary linear algebra, this would imply that U is diagonal, but not so for operators, and also not for block-upper finite matrices).

The equivalent of the familiar QR factorization from linear algebra is called the *inner-coprime factorization*, which is a factorization of $T \in \mathcal{X}$ as

$$T = Q^* R, \qquad Q \text{ inner}, R \in \mathcal{U} .$$

(Almost) every $T \in \mathcal{X}$ has such a factorization (there are some borderline exceptions having to do with marginally stable systems, but they are not of interest to us here). Note that Q^* is lower, so that this is a lower times upper factorization. This factorization can be used to map a general operator in $\mathcal{X}$ to an upper operator R, which reduces the problem to the inversion of upper operators: $T^{-1} = R^{-1}Q$. Hence, the inner-coprime factorization is a useful preprocessing step, but the delicate part still has to be done.

An LTI example of this factorization is

$$T(z) = \frac{1}{1-2z} = \left[\frac{2-z}{1-2z}\right]\left[\frac{1}{2-z}\right] =$$

$$= \left[\frac{\frac{1}{2} - z^{-1}}{1 - \frac{1}{2}z^{-1}}\right] \left[\frac{\frac{1}{2}}{1 - \frac{1}{2}z}\right].$$

In this case, T has an unstable pole: $T \in \mathcal{L}$ (its matrix representation is as in (3)). After factorization, Q^* contains the unstable pole, and $Q = (\frac{1}{2} - Z)(1 - \frac{1}{2}Z)^{-1} \in \mathcal{U}$ is inner. R contains the reflection of the pole and is stable (upper).

For a more complicated LTV example, consider

$$T = \left[\begin{array}{ccc|cccc} \ddots & & & & & & \\ & 1 & & & & \mathbf{0} & \\ & & 1 & & & & \\ \hline & & & \boxed{1} & & & \\ & & & 1/2 & 1 & & \\ & & & 1/4 & 1/2 & 1 & \\ & \mathbf{0} & & 1/8 & 1/4 & 1/2 & 1 \\ & & & \vdots & & & \vdots\ \ddots \end{array}\right].$$

The inner-coprime factorization $T = Q^* R$ is

$$Q^* = \left[\begin{array}{ccc|cccc} \ddots & & & & & & \\ & 1 & & & & \mathbf{0} & \\ & & 1 & & & & \\ \hline & & & \boxed{[1\ \ 0]} & & & \\ & & & [0\ \ \frac{1}{2}\sqrt{3}] & 1/2 & & \\ & & & [0\ \ \frac{1}{4}\sqrt{3}] & -3/4 & 1/2 & \\ & \mathbf{0} & & [0\ \ \frac{1}{8}\sqrt{3}] & -3/8 & -3/4 & 1/2 \\ & & & \vdots & & & \vdots\ \ddots \end{array}\right],$$

$$R = \left[\begin{array}{ccc|ccccc} \ddots & & & & & & & \\ & 1 & & & & \mathbf{0} & & \\ & & 1 & & & & & \\ \hline & & & \begin{bmatrix} 1 \\ \frac{1}{3}\sqrt{3} \end{bmatrix} & \begin{bmatrix} 0 \\ \frac{2}{3}\sqrt{3} \end{bmatrix} & \begin{bmatrix} 0 \\ \frac{1}{3}\sqrt{3} \end{bmatrix} & \begin{bmatrix} 0 \\ \frac{1}{6}\sqrt{3} \end{bmatrix} & \cdots \\ & & & & 0 & -1 & -\frac{1}{2} & -\frac{1}{4}\ \cdots \\ & & & & & 0 & -1 & -\frac{1}{2} \\ & \mathbf{0} & & & & & 0 & -1\ \cdots \\ & & & & & & & \ddots \end{array}\right].$$

It is not hard to verify this by direct multiplications: Q is unitary and $T = Q^* R$, but obviously, this factorization is not trivially obtained. It has been computed by Matlab using the state space methods of Section 3. Note that the number of inputs of Q and R is not constant: it is equal to 2 at time $k = 0$. This is a remarkable aspect of time-varying factorizations.

2.3 Inner-Outer Factorization

An operator $T_0 \in \mathcal{U}$ is *left outer* if it has a left inverse $T_{0,\ell}^{-1}$ which is upper, and *right outer* if it has a right inverse $T_{0,r}^{-1}$ which is upper. T_0 is *outer* if it is both left outer and right outer. The fact that the inverse of an outer operator is upper again is very helpful in computing this inverse. For LTI systems, outer means that $T(z)$ does not have 'unstable' zeros: T in (1) is outer, but T in (2) is not. Also, an inner operator is not outer (unless it is diagonal) because its inverse is equal to its conjugate transpose and always lower. In more abstract operator language, equivalent definitions are that T_0 is left outer if $\overline{T_0 \mathcal{U}_2} = \mathcal{U}_2$, and right outer if $\overline{\mathcal{U}_2 T_0} = \mathcal{U}_2$, where $\mathcal{U}_2$ is the space of all operators in $\mathcal{U}$ that are bounded in the Hilbert-Schmidt norm (the operator version of the Frobenius norm: root-sum-square of all entries). It might happen that ranges are not closed, in which case the inverse is only densely defined. We do not want to go to this level of detail in this paper.

The content of the inner-outer factorization theorem is [3], [6]:

Theorem 1. Every operator $T \in \mathcal{U}$ has factorizations

$$T = \begin{cases} U T_{o,r} & \text{[Inner-outer]} \\ T_{o,\ell} V & \text{[Outer-inner]} \end{cases}$$

where $U, V, T_{o,r}, T_{o,\ell} \in \mathcal{U}$ satisfy

$$\begin{array}{ll} U^* U = I, & T_{o,r}: \text{ right outer,} \\ V V^* = I, & T_{o,\ell}: \text{ left outer.} \end{array}$$

Again, these factorizations can be viewed as some form of QR (or RQ) factorization, although less obviously than for the inner-coprime factorization, because T is of course already upper triangular. The objective here is to obtain outer factors $T_{o,r}$ and $T_{o,\ell}$, centered on the main diagonal: to be outer, it is at least necessary that the main diagonal is (left or right) invertible. U and V are in general only isometries. They are unitary (inner) if the columns, resp. rows, of T span all of $\mathcal{X}_2$.

A simple LTV example which can be computed by hand is

$$T = \left[\begin{array}{ccc|cccc} \ddots & & & & & & \\ & 1 & & & \mathbf{0} & & \\ & & 1 & & & & \\ & & 1 & 1 & & & \\ \hline & & & \boxed{0} & 1 & & \\ & & & & 0 & 1 & \\ & \mathbf{0} & & & & 0 & 1 \\ & & & & & \ddots & \ddots \end{array}\right]. \qquad (6)$$

In this example, T does not have full row span: $[\cdots\ 0\ \ 0\ \ \boxed{1}\ \ 0\ \ 0\ \cdots]$ is not contained in it. The outer-inner factorization is

$$T = T_{0,\ell} V =$$

$$= \left[\begin{array}{ccc|cccc} \ddots & & & & & & \\ & 1 & & & \mathbf{0} & & \\ & & 1 & & & & \\ & & & \sqrt{2} & & & \\ \hline & & & & 1 & & \\ & & & & & 1 & \\ & \mathbf{0} & & & & & 1 \\ & & & & & & \ddots \end{array}\right] \left[\begin{array}{ccc|cccc} \ddots & & & & & & \\ & 1 & & & \mathbf{0} & & \\ & & 1 & & & & \\ & & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & & & \\ \hline & & & 0 & 1 & & \\ & & & & 0 & 1 & \\ & \mathbf{0} & & & & 0 & \ddots \\ & & & & & & \ddots \end{array}\right].$$

$T_{0,\ell}$ obviously has a left inverse $T_{0,\ell}^{-1}$ which is upper (it is even diagonal and a right inverse in this case). V is only an isometry: $VV^* = I$, but $V^*V \neq I$. The inner-outer factorization is

$$T = UT_{0,r} =$$

$$= \left[\begin{array}{cccc|cccc}
\ddots &&&& \cdot &&& \\
&1&&& \cdot && \mathbf{0} & \\
&&1&& \cdot &&& \\
&&&1& \cdot &&& \\ \hline
&&&& \boxed{\cdot} & 1 && \\
&&&& \cdot & 0 & 1 & \\
& \mathbf{0} &&& \cdot && 0 & \ddots \\
&&&& \cdot &&& \ddots
\end{array}\right]
\left[\begin{array}{cccc|cccc}
\ddots &&&&&&& \\
&1&&&&&& \\
&&1&&&&& \\
&&&1&1&&& \\ \hline
\cdot&\cdot&\cdot&\cdot&\boxed{\cdot}&\cdot&\cdot&\cdot \\
&&&&&1&& \\
&&&&&&1& \\
&&&&&&&\ddots
\end{array}\right] .$$

U has a column with zero horizontal dimension (signified by '$\cdot$'), but $U^*U = I$ nonetheless. $T_{0,r}$ has a right inverse $T_{0,r}^{-1}$ which is upper,

$$T_{0,r}^{-1} = \left[\begin{array}{cccc|ccc}
\ddots &&&& \cdot && \\
&1&&& \cdot && \\
&&1&& \cdot && \\
&&&1& \cdot && \\ \hline
&&&& \boxed{\cdot} && \\
&&&& \cdot & 1 & \\
&&&& \cdot && 1 \quad \ddots \\
&&&& \cdot &&
\end{array}\right]$$

but $T_{0,r}^{-1}T_{0,r} \neq I$: it is not a left inverse. If our purpose is the inversion of T, then it is clear in this case that T only has a right inverse. The outer-inner factorization is useful for computing this inverse: we directly have $T_r^{-1} = V^* T_{0,\ell}^{-1}$.

2.4 Operator Inversion

The strategy for the inversion of an operator $T \in \mathcal{X}$ is to determine the following factorizations:

$$\begin{array}{rcllll}
T &=& Q^*R & \text{[Inner-coprime]:} & Q \text{ inner} \\
R &=& UR_{0,r} & \text{[Inner-outer]:} & U^*U = I, \\
&&&& R_{0,r} \text{ right outer} \\
R_{0,r} &=& R_{00}V & \text{[Outer-inner]:} & VV^* = I, \\
&&&& R_{00} \text{ left outer}
\end{array}$$

$$\text{(all factors in } \mathcal{U}), \tag{7}$$

so that $T = Q^*UR_{00}V$. The final factor, R_{00}, is upper and both left and right outer, hence invertible in $\mathcal{U}$, and its inverse is easily obtained. T is not necessarily invertible: U and V are isometries, and might not be unitary. In any case, T has a Moore-Penrose (pseudo-)inverse

$$T^\dagger = V^* R_{00}^{-1} U^* Q ,$$

and T is invertible with $T^{-1} = T^\dagger$ if U and V are both unitary. The inverse is thus specified as a lower-upper-lower-upper factorization. The factors may be multiplied to obtain an explicit matrix representation of $T^\dagger$, but because each of them will be known by its state representation, it is computationally efficient to keep it in factored form. State representations are the topic of the next section.

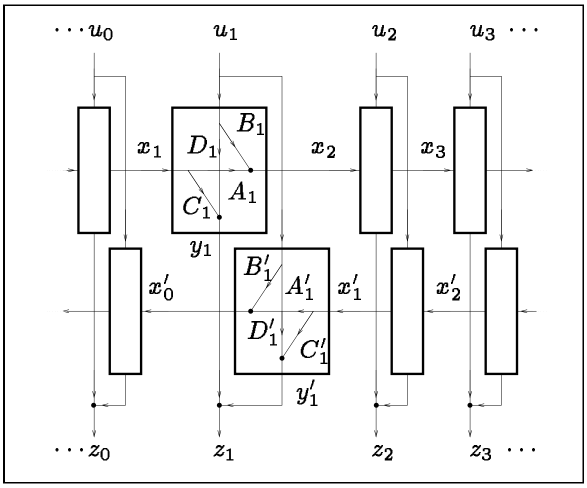


Fig. 1. State realization which models the multiplication $z = uT$.

3. State Representations and Recursions

In the previous section, we have considered bounded operators $T \in \mathcal{X}$, and looked at it as the input-output operator of a linear time-varying system. In this section, we go further and consider systems that have time-varying state realizations with a finite, and hopefully low, number of states at each point in time.

3.1 Time-Varying State Realizations

Let $\{\mathbf{T}_k\}$, $\{\mathbf{T}'_k\}$ be series of matrices with block entries

$$\mathbf{T}_k = \begin{bmatrix} A_k & C_k \\ B_k & D_k \end{bmatrix}, \qquad \mathbf{T}'_k = \begin{bmatrix} A'_k & C'_k \\ B'_k & 0 \end{bmatrix},$$

and consider the time-varying forward and backward *state recursions*,

$$\begin{array}{ll}
(\mathbf{T}) & \left\{ \begin{array}{rcl} x_{k+1} &=& x_k A_k + u_k B_k \\ y_k &=& x_k C_k + u_k D_k \end{array} \right. \\
(\mathbf{T}') & \left\{ \begin{array}{rcl} x'_{k-1} &=& x'_k A'_k + u_k B'_k \\ y'_k &=& x'_k C'_k \end{array} \right. \\
& \quad z_k = y_k + y'_k .
\end{array} \tag{8}$$

See Fig. 1. The recursion maps the input sequences $[u_k]$ to output sequences $[y_k]$, $[y'_k]$, and finally to $[z_k]$. The intermediate quantities in the recursion are x_k, the forward state, and x'_k, the backward state. The matrices $\{A_k, B_k, C_k, D_k, A'_k, B'_k, C'_k\}$ must have compatible dimensions in order for the multiplications to make sense, but they need not be square or have constant dimensions. Zero dimensions are also allowed. The relation between input $u = [\cdots u_1, u_2, \cdots]$ and output $z = [\cdots z_1, z_2, \cdots]$, as generated by the above state recursions, is

$$z = uT :$$

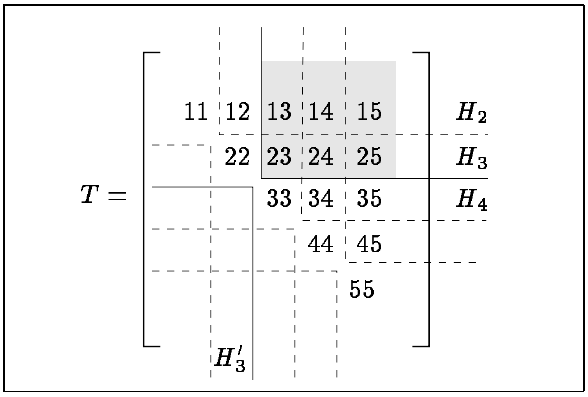


Fig. 2. Hankel matrices are submatrices of T. H_3 is shaded.

$$T = \begin{bmatrix} \ddots & \vdots & & & \vdots & \\ \cdots & D_1 & B_1C_2 & B_1A_2C_3 & B_1A_2A_3C_4 & \cdots \\ & B_2'C_1' & D_2 & B_2C_3 & B_2A_3C_4 & \\ & B_3'A_2'C_1' & B_3'C_2' & D_3 & B_3C_4 & \\ \cdots & & B_4'A_3'C_2' & B_4'C_3' & D_4 & \cdots \\ & \vdots & & & \vdots & \ddots \end{bmatrix}$$

so that the state recursions can be used to compute a vector-matrix multiplication $z = uT$, where the matrix T is of the above form. Accordingly, we will say that a matrix T has a (time-varying) *state realization* if there exist matrices $\{\mathbf{T}_k\}$, $\{\mathbf{T}_k'\}$ such that the block entries of $T = [T_{ij}]$ are given by

$$T_{ij} = \begin{cases} D_i\,, & i = j\,, \\ B_i A_{i+1} \cdots A_{j-1} C_j\,, & i < j\,, \\ B_i' A_{i-1}' \cdots A_{j+1}' C_j'\,, & i > j\,. \end{cases} \tag{9}$$

The upper triangular part of T is generated by the forward state recursions $\{\mathbf{T}_k\}$, the lower triangular part by the backward state recursions $\{\mathbf{T}_k'\}$. To have nicely converging expressions in (9), we always require realizations to be *exponentially stable*, in the sense that

$$\lim_{n\to\infty} \sup_i \,\|\, A_{i+1} \cdots A_{i+n} \,\|^{1/n} \; < 1\,,$$
$$\lim_{n\to\infty} \sup_i \,\|\, A_{i-1}' \cdots A_{i-n}' \,\|^{1/n} \; < 1\,.$$

The computation of a vector-matrix product using the state equations is more efficient than a direct multiplication if, for all k, the dimensions of x_k and x_k' are relatively small compared to the matrix size. If this dimension is, on average, equal to d, and T is an $n \times n$ matrix, then a vector-matrix multiplication has complexity $\mathcal{O}(d^2 n)$ (this can be reduced further to $\mathcal{O}(dn)$ by considering special types of realizations, viz. [7], [8]), and a matrix inversion has complexity $\mathcal{O}(d^2 n)$ rather than $\mathcal{O}(n^3)$.

3.2 Computation of a State Realization

At this point, a first question that emerges is whether, for any given matrix, a state realization exists. If so, then subsequent questions are (i) how to find it, and (ii) what will be its complexity. To answer these questions, define the submatrices

$$H_k = \begin{bmatrix} T_{k-1,k} & T_{k-1,k+1} & \cdots \\ T_{k-2,k} & T_{k-2,k+1} & \\ \vdots & & \ddots \end{bmatrix}, \tag{10}$$

$$H_k' = \begin{bmatrix} T_{k,k-1} & T_{k,k-2} & \cdots \\ T_{k+1,k-1} & T_{k+1,k-2} & \\ \vdots & & \ddots \end{bmatrix}. \tag{11}$$

See Fig. 2. The H_k are called (time-varying) Hankel matrices, but they have a Hankel *structure* only in the time-invariant context. Even without this structure, a number of important properties of LTI systems carry over. For example, when we substitute eq. (9) into (10), we obtain

$$\begin{aligned} H_k &= \begin{bmatrix} B_{k-1}C_k & B_{k-1}A_kC_{k+1} & \cdots \\ B_{k-2}A_{k-1}C_k & B_{k-2}A_{k-1}A_kC_{k+1} & \\ B_{k-3}A_{k-2}A_{k-1}C_k & & \ddots \\ \vdots & & \end{bmatrix} \\ &= \begin{bmatrix} B_{k-1} \\ B_{k-2}A_{k-1} \\ B_{k-3}A_{k-2}A_{k-1} \\ \vdots \end{bmatrix} \cdot \\ &\quad \cdot [C_k \;\; A_kC_{k+1} \;\; A_kA_{k+1}C_{k+2} \cdots] = \mathcal{C}_k \mathcal{O}_k\,. \end{aligned}$$

Just as in the LTI case, the Hankel matrices of an LTV system generated by state recursions (8) admit factorizations, and the rank of the factorization of H_k is (at most) equal to the state dimension at time k. Conversely, the structure of this factorization can be used to derive realizations from it. The ideas for this were already contained in the classical Kalman realization theory [9].

Theorem 2 ([10, 11]). Let $T \in \mathcal{X}$, and define $d_k = \mathtt{rank}(H_k)$, $d_k' = \mathtt{rank}(H_k')$. If all d_k, d_k' are finite, then there are (marginally) exponentially stable time-varying state realizations that realize T. The minimal dimension of x_k and x_k' of any state realization of T is equal to d_k and d_k', respectively.

Hence, the state dimensions of the realization (which determine the computational complexity of multiplications and inversions using state realizations) are equal to the ranks of the Hankel matrices. Note that these ranks are not necessarily the same for all k, so that the number of states may be time-varying.

Minimal state realizations are obtained from minimal factorizations of the H_k and H_k'. In principle, the following algorithm from [7] is suitable. Let $H_k = Q_k R_k$ be a QR factorization of H_k, where Q_k is an isometry ($Q_k^* Q_k = I_{d_k}$), and R_k has full row rank d_k. Likewise, let $H_k' = Q_k' R_k'$. Then a realization of T is given by

$$\begin{aligned} \mathbf{T} : A_k &= [0 \quad Q_k^*] Q_{k+1}\,, \\ B_k &= (Q_{k+1})(1,:)\,, \\ C_k &= R_k(:,1)\,, \\ D_k &= T_{k,k}\,, \\ \mathbf{T}' : A_k' &= [0 \quad Q_{k+1}'^{*}] Q_k'\,, \end{aligned}$$

$$\begin{aligned} B'_k &= Q'_k(1,:), \\ C'_k &= R'_{k+1}(:,1), \\ D'_k &= 0. \end{aligned}$$

(For a matrix X, the notation $X(1,:)$ denotes the first row of X, and $X(:,1)$ the first column.) Important refinements are possible. For example, it is not necessary to act on the infinite size matrix H_k: it is sufficient to consider a principal submatrix that has rank d_k [12]. Also note that H_k and H_{k+1} have many entries in common, which can be exploited by considering updating algorithms for the QR factorizations. It is also possible to compute optimal approximate realizations of lower system order [13], [14].

Band matrices are important examples of systems with a low state dimension: d_k is equal to the band width -1, and a realization can be written down directly by inspection:

$$\begin{bmatrix} A_k & C_k \\ B_k & D_k \end{bmatrix} = \left[\begin{array}{cccc|c} 0 & & & & T_{k-d_k,k} \\ 1 & 0 & & & T_{k-d_k+1,k} \\ & \ddots & \ddots & & \vdots \\ & & 1 & 0 & T_{k-1,k} \\ \hline 0 & \cdots & 0 & 1 & T_{k,k} \end{array}\right].$$

But also the inverse of a band matrix, although it is not sparse, has a low state dimension: d_k is at each point the same as that of the original band matrix. This is shown in Section 3.3. Examples are the matrices considered so far in this paper: they all have constant state dimensions equal to 1.

3.3 State Complexity of the Inverse

Suppose that T is an invertible matrix or operator with a state realization of low complexity. Under some regularity conditions, it is straightforward to prove that the inverse has a state realization of the same complexity.

Proposition 1. Let $T \in \mathcal{X}$ be an invertible operator with finite dimensional Hankel matrices $(H_T)_k$ and $(H'_T)_k$, defined by (10), (11). Put $d_k := \mathrm{rank}\,(H_T)_k$ and $d'_k := \mathrm{rank}(H'_T)_k$.

If, for each k, at least one of the submatrices $[T_{ij}]_{i,j=-\infty}^{k-1}$ or $[T_{ij}]_{i,j=k}^{\infty}$ is invertible, then $S = T^{-1}$ has Hankel matrices with the same ranks: $\mathrm{rank}\,(H_S)_k = d_k$ and $\mathrm{rank}(H'_S)_k = d'_k$.

Proof: We will use Schur's inversion lemma. In general, let A, B, C, D be matrices or operators such that A and D are square, and A is invertible, then

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & D - CA^{-1}B \end{bmatrix} \begin{bmatrix} I & A^{-1}B \\ 0 & I \end{bmatrix}.$$

If in addition the inverse of this block matrix exists, then $D^\times := D - CA^{-1}B$ is invertible and the inverse of the block matrix is given by

$$\begin{aligned} \begin{bmatrix} A' & B' \\ C' & D' \end{bmatrix} &= \\ &= \begin{bmatrix} I & -A^{-1}B \\ 0 & I \end{bmatrix} \begin{bmatrix} A^{-1} & 0 \\ 0 & (D^\times)^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix} = \\ &= \begin{bmatrix} (*) & -A^{-1}B(D^\times)^{-1} \\ -(D^\times)^{-1}CA^{-1} & (D^\times)^{-1} \end{bmatrix}. \end{aligned}$$

In particular, D' is invertible, $\mathrm{rank}\,B' = \mathrm{rank}\,B$, $\mathrm{rank}\,C' = \mathrm{rank}\,C$. The proposition follows if $\left[\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right]$ is taken to be a partitioning of T, such that $B = (H_T)_k$ and $C = (H'_T)_k$. □

3.4 Outer Inversion

If a matrix or operator is block upper and has an inverse which is again block upper (i.e., the corresponding time-varying system is outer), then it is straightforward to derive a state realization of the inverse.

Proposition 2 ([7]). Let $T \in \mathcal{U}$ be outer, so that $S := T^{-1} \in \mathcal{U}$. If T has a state realization $\mathbf{T} = \{A_k, B_k, C_k, D_k\}$, then a realization of S is given by

$$\mathbf{S}_k = \begin{bmatrix} A_k - C_k D_k^{-1} B_k & -C_k D_k^{-1} \\ D_k^{-1} B_k & D_k^{-1} \end{bmatrix}.$$

Proof: From $T^{-1}T = I$ and $TT^{-1} = I$, and the fact that T^{-1} is upper, we obtain that all $D_k = T_{k,k}$ must be invertible. Using this, we rewrite the state equations:

$$\begin{aligned} &\begin{cases} xZ^{-1} = xA + uB \\ y \quad\;\; = xC + uD \end{cases} \\ \Leftrightarrow \quad &\begin{cases} xZ^{-1} = x(A - CD^{-1}B) + yD^{-1}B \\ u \quad\;\; = -xCD^{-1} \qquad\quad + yD^{-1}. \end{cases} \end{aligned}$$

The second set of state equations generates the inverse mapping $y \to u$, so that it must be a realization of T^{-1}. The remaining part of the proof is to show that $\{A_k - C_k D_k^{-1} B_k\}$ is a *stable* state operator. The proof of this is omitted, but it is essentially a consequence of the fact that T is outer and hence has a bounded upper inverse [11]. □

Note that the realization of the inverse is obtained locally: it is, at point k, only dependent on the realization of the given matrix at point k. Hence, it is quite easy to compute the inverse of an operator once we know that it is outer.

3.5 Inner-Coprime Factorization

In order to use the above inversion proposition on a matrix T which is not block upper, we compute a kind of QR factorization of T as $T = Q\Delta$, where Q is block lower and unitary, and Δ is block upper. Since Q is unitary, its inverse is equal to its Hermitian transpose and can trivially be obtained. We first consider the special case where T is lower triangular. This case is related to the inner-coprime factorization in [13].

Proposition 3 ([13]).

(a) Suppose that $T \in \mathcal{L}$ has an exponentially stable finite dimensional state realization $\mathbf{T}' =$

$\{A'_k, B'_k, C'_k, D'_k\}$, with $A'_k : d'_k \times d'_{k-1}$. Then T has a factorization $T = Q^* R$, where $Q \in \mathcal{U}$ is inner and $R \in \mathcal{U}$.

(b) Denote realizations of Q and R by

$$\mathbf{Q}_k = \begin{bmatrix} (A_Q)_k & (C_Q)_k \\ (B_Q)_k & (D_Q)_k \end{bmatrix}, \quad \mathbf{R}_k = \begin{bmatrix} (A_R)_k & (C_R)_k \\ (B_R)_k & (D_R)_k \end{bmatrix}.$$

Then $\mathbf{Q}_k$ and $\mathbf{R}_k$ follow recursively from the QR factorization

$$\left[\begin{array}{c|cc} Y_k A'_k & I & Y_k C'_k \\ B'_k & 0 & D'_k \end{array}\right] = \mathbf{Q}_k^* \left[\begin{array}{c|c} Y_{k-1} & \mathbf{R}_k \\ 0 & \end{array}\right] \tag{12}$$

where $Y_k : d'_k \times d'_k$ is a square matrix.

The state operators of $\mathbf{Q}$ and $\mathbf{R}$ are the same: $(A_Q)_k = (A_R)_k$, and they are related to A'^*_k via a state transformation. The resulting number of inputs of Q and R may be time-varying. In particular, Q can be a block matrix whose entries are matrices, even if T itself has scalar entries.

Eq. (12) is a recursion: for a given initial matrix Y_{k_0}, we can compute $\mathbf{Q}_{k_0}, \mathbf{R}_{k_0}$, and Y_{k_0-1}. Hence we obtain the state realization matrices for Q and R in turn for $k = k_0 - 1,\ k_0 - 2,\ \cdots$. All we need is a correct initial value for the recursion. Exact initial values can be computed in the case of systems that are LTI for large k ($Y_{k_0}^* Y_{k_0}$ satisfies a Lyapunov equation), or periodically varying, or that have zero state dimensions for $k > k_0$. However, even if this is not the case, we can obtain Q and R to any precision we like by starting the recursion with any (invertible) initial value, such as $\tilde{Y}_{k_0} = I$. The assumption that T has an exponentially stable realization implies that $\tilde{Y}_k \to Y_k$ ($k \to -\infty$), the correct value for Y. Convergence is monotonic, and the speed of convergence is depending on the 'amount of stability' of the A'_k.

The more general case ($T \in \mathcal{X}$) is a corollary of the above proposition. Split $T = T_{\mathcal{L}} + T_{\mathcal{U}}$, with $T_{\mathcal{L}} \in \mathcal{L}$ and $T_{\mathcal{U}} \in Z\mathcal{U}$ (strictly upper). The above inner-coprime factorization, applied to $T_{\mathcal{L}}$, gives $T_{\mathcal{L}} = Q^* R$. Then T has a factorization $T = Q^*(R + QT_{\mathcal{U}}) =: Q^*\Delta$, where $\Delta \in \mathcal{U}$. The realization for Q is only dependent on $T_{\mathcal{L}}$, and follows from the recursion (12). A realization for Δ is obtained by multiplying Q with $T_{\mathcal{U}}$, and adding R. These operations can be done in state space. Using the fact that $A_Q = A_R$ and $B_Q = B_R$, we obtain

$$\mathbf{\Delta}_k = \left[\begin{array}{cc|c} (A_Q)_k & (C_Q)_k B_k & (C_R)_k \\ 0 & A_k & C_k \\ \hline (B_Q)_k & (D_Q)_k B_k & (D_R)_k \end{array}\right].$$

3.6 Inner-Outer Factorization

Let $T \in \mathcal{U}$, with exponentially stable finite dimensional realization $\mathbf{T} = \{A_k, B_k, C_k, D_k\}$, where $A_k : d_k \times d_{k+1}$, $A'_k : d'_k \times d'_{k-1}$. The inner-outer factorization $T = UT_{0,r}$, where $U^*U = I$ and $T_{0,r}$ is right outer, can be computed recursively, as follows. Suppose that, at point k, we know the matrix Y_k. Compute the following QR factorization:

$$\begin{array}{c} \\ m_k \\ (d_Y)_k \end{array} \begin{array}{c} \begin{array}{cc} n_k & d_{k+1} \end{array} \\ \begin{bmatrix} D_k & B_k \\ Y_k C_k & Y_k A_k \end{bmatrix} \end{array} =:$$

$$=: \mathbf{W}_k \cdot \begin{array}{c} \\ (m_0)_k \\ (d_Y)_{k+1} \\ \\ \end{array} \begin{array}{c} \begin{array}{cc} n_k & d_{k+1} \end{array} \\ \begin{bmatrix} (D_0)_k & (B_0)_k \\ 0 & Y_{k+1} \\ 0 & 0 \end{bmatrix} \end{array} \tag{13}$$

where $\mathbf{W}_k$ is unitary, and the partitioning of the factors at the right hand side of (13) is such that $(D_0)_k$ and Y_{k+1} both have full row rank. This also defines the dimensions $(m_0)_k$ and $(d_Y)_{k+1}$. Since the factorization produces Y_{k+1}, we can perform the QR factorization (13) in turn for $k+1, k+2, \cdots$.

A non-trivial result from [6], [11] claims that this recursion determines the inner-outer factorization. $\mathbf{W}_k$ has a partitioning as

$$\mathbf{W}_k = \begin{array}{c} \\ m_k \\ (d_Y)_k \end{array} \begin{array}{c} \begin{array}{ccc} (m_0)_k & (d_Y)_{k+1} & \end{array} \\ \begin{bmatrix} (D_U)_k & (B_U)_k & * \\ (C_U)_k & (A_U)_k & * \end{bmatrix} \end{array}.$$

It turns out that $\mathbf{U} = \{(A_U)_k, (B_U)_k, (C_U)_k, (D_U)_k\}$ is a realization of U, and $\mathbf{T}_0 = \{A_k, (B_0)_k, C_k, (D_0)_k\}$ is a realization of $T_{0,r}$.

In [6], the inner-outer factorization was solved using a time-varying Riccati equation (see also [15]). The above recursive QR factorization is a square-root variant of it. Correct initial points for the recursion can be obtained in a similar way as for the inner-coprime factorization. If T is Toeplitz for $k < k_0$, then Y_{k_0} can be computed from the underlying time-invariant Riccati equation (viz. [16]), which is retrieved upon squaring of (13), thus eliminating $\mathbf{W}_k$. As is well known, this calls for the solution of an eigenvalue problem. Similar results hold for the case where T is periodically varying before $k < k_0$, or has zero state dimensions ($d_k = 0, k < k_0$). But, as for the inner-coprime factorization, we can in fact take any invertible starting value, such as $\tilde{Y}_{k_0} = I$, and perform the recursion: because of the assumed stability of A, $\tilde{Y}_k \to Y_k$. In a sense, we are using the familiar QR-iteration [17] for computing eigenvalues! (Open question is how the shifted QR iteration fits in this framework.)

The outer-inner factorization $T = T_{0,\ell}V$ ($VV^* = I$, $T_{0,\ell}$ left outer) is computed similarly, now by the backward recursive LQ factorization

$$\begin{array}{c} \\ m_k \\ d_k \end{array} \begin{array}{c} \begin{array}{cc} n_k & (d_Y)_k \end{array} \\ \begin{bmatrix} D_k & B_k Y_k \\ C_k & A_k Y_k \end{bmatrix} \end{array} =:$$

$$=: \begin{array}{c} \\ m_k \\ d_k \end{array} \begin{array}{c} \begin{array}{ccc} (n_0)_k & (d_Y)_{k-1} & \end{array} \\ \begin{bmatrix} (D_0)_k & 0 & 0 \\ (C_0)_k & Y_{k-1} & 0 \end{bmatrix} \end{array} \mathbf{W}_k \tag{14}$$

The partitioning is such that $(D_0)_k$ and Y_{k-1} have full column rank. $\mathbf{W}_k$ is unitary and has a partitioning as

$$\mathbf{W}_k = \begin{array}{c} \\ (n_0)_k \\ (d_Y)_{k-1} \\ \\ \end{array} \begin{array}{c} \begin{array}{cc} n_k & (d_Y)_k \end{array} \\ \begin{bmatrix} (D_V)_k & (B_V)_k \\ (C_V)_k & (A_V)_k \\ * & * \end{bmatrix} \end{array}.$$

Realizations of the factors are given by $\mathbf{V} = \{(A_V)_k, (B_V)_k, (C_V)_k, (D_V)_k\}$ and $\mathbf{T}_0 = \{A_k, B_k, (C_0)_k, (D_0)_k\}$.

An example of the outer-inner factorization is given in Section 3.8.

3.7 Inversion

At this point, we have obtained state space versions of all operators in the factorization $T = Q^* U R_{00} V$ of eq. (7): Q is obtained by the (backward) inner-coprime factorization of Section 3.5, U by the (forward) inner-outer QR recursion in eq. (13), and V by the (backward) outer-inner LQ recursion in eq. (14). We also have obtained a state space expression for the inverse of the outer factor R_{00}, viz. Section 3.4. The realizations of the (pseudo-)inverses of the inner (isometric) factors are obtained simply via transposition: e.g., the realization for V^* is anti-causal and given by $\{(A_V)_k^*, (C_V)_k^*, (B_V)_k^*, (D_V)_k^*\}$. The (pseudo-)inverse of T is given by $T^\dagger = V^* R_{00}^{-1} U^* Q$.

It is possible to obtain a single set of state matrices for $T^\dagger$, by using formulas for the multiplication and addition of realizations. This is complicated to some extent because of the alternating upper-lower nature of the factors. Moreover, it is often not necessary to obtain a single realization: matrix-vector multiplication is carried out more efficiently on a factored representation than on a closed-form realization. This is because for a closed-form representation, the number of multiplications per point in time is roughly equal to the square of the sum of the state dimensions of all factors, whereas in the factored form it is equal to the sum of the squares of these dimensions. See also [7].

3.8 Example

We finish this section with an example. Consider again T from eq. (4). A realization for T is straightforward to obtain, since it is a banded matrix:

$$\mathbf{T}_k = \begin{cases} \left[\begin{array}{c|c} 0 & -1/2 \\ \hline 1 & 1 \end{array}\right], & k = -\infty, \cdots, 0 \\ \left[\begin{array}{c|c} 0 & -2 \\ \hline 1 & 1 \end{array}\right], & k = 1, \cdots, -\infty. \end{cases}$$

T is already upper, so an inner-coprime factorization is not necessary. It is also not hard to see that the inner-outer factorization of T is $T = I \cdot T$. This is because the initial point of the recursion (13), given by the LTI solution of the inner-outer factorization of the top-left block of T, produces $(d_Y)_0 = 0$, and hence all subsequent Y_k's have zero dimensions. Consequently, T is already right outer by itself.

Our purpose now is to compute the outer-inner factorization of T. An initial point for the recursion (14) is obtained as $Y_k = \sqrt{3}$, $k \geq 1$. It requires the solution of a Riccati equation to find it (this equation is the square of (14), which eliminates $\mathbf{W}_k$), but it is easy to verify that it is a stationary solution of (14) for $k \geq 1$: it satisfies the equation

$$\begin{array}{c} \begin{bmatrix} 1 & \sqrt{3} \\ -2 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 0 & \cdot \\ -1 & \sqrt{3} & \cdot \end{bmatrix} \begin{bmatrix} \frac{1}{2} & \frac{1}{2}\sqrt{3} \\ -\frac{1}{2}\sqrt{3} & \frac{1}{2} \\ \cdot & \cdot \end{bmatrix} \\ \begin{bmatrix} D & BY \\ C & AY \end{bmatrix} = \begin{bmatrix} D_0 & 0 & 0 \\ C_0 & Y & 0 \end{bmatrix} . \qquad \mathbf{W}_k \end{array} \tag{15}$$

(zero dimensions are denoted by '$\cdot$') and we'll leave it by that. Alternatively, we can start the recursion with $\tilde{Y}_{20} = 1$, say, and obtain $\tilde{Y}_0 = 1.7321\cdots \approx \sqrt{3}$. Eq. (15) also shows that the realization of the outer factor has $(D_0)_k = 2$ and $(C_0)_k = -1$, for $k \geq 0$. Continuing with the recursion gives us

$$\begin{aligned}
(\mathbf{T}_0)_1 &= \left[\begin{array}{c|c} 0 & -1 \\ \hline 1 & 2 \end{array}\right], \quad \mathbf{V}_1 = \left[\begin{array}{c|c} 0.5 & -0.866 \\ \hline 0.866 & 0.5 \end{array}\right], \\
Y_0 &= 1.732, \\
(\mathbf{T}_0)_0 &= \left[\begin{array}{c|c} 0 & -0.25 \\ \hline 1 & 2 \end{array}\right], \quad \mathbf{V}_0 = \left[\begin{array}{c|c} 0.5 & -0.866 \\ \hline 0.866 & 0.5 \end{array}\right], \\
Y_{-1} &= 0.433, \\
(\mathbf{T}_0)_{-1} &= \left[\begin{array}{c|c} 0 & -0.459 \\ \hline 1 & 1.090 \end{array}\right], \quad \mathbf{V}_{-1} = \left[\begin{array}{c|c} 0.918 & -0.397 \\ \hline 0.397 & 0.918 \end{array}\right], \\
Y_{-2} &= 0.199, \\
(\mathbf{T}_0)_{-2} &= \left[\begin{array}{c|c} 0 & -0.490 \\ \hline 1 & 1.020 \end{array}\right], \quad \mathbf{V}_{-2} = \left[\begin{array}{c|c} 0.981 & -0.195 \\ \hline 0.195 & 0.981 \end{array}\right], \\
Y_{-3} &= 0.097, \\
(\mathbf{T}_0)_{-3} &= \left[\begin{array}{c|c} 0 & -0.498 \\ \hline 1 & 1.005 \end{array}\right], \quad \mathbf{V}_{-3} = \left[\begin{array}{c|c} 0.995 & -0.097 \\ \hline 0.097 & 0.995 \end{array}\right], \\
Y_{-4} &= 0.049, \\
(\mathbf{T}_0)_{-4} &= \left[\begin{array}{c|c} 0 & -0.499 \\ \hline 1 & 1.001 \end{array}\right], \quad \mathbf{V}_{-4} = \left[\begin{array}{c|c} 0.999 & -0.048 \\ \hline 0.048 & 0.999 \end{array}\right], \\
Y_{-5} &= 0.024, \\
(\mathbf{T}_0)_{-5} &= \left[\begin{array}{c|c} 0 & -0.500 \\ \hline 1 & 1.000 \end{array}\right], \quad \mathbf{V}_{-5} = \left[\begin{array}{c|c} 1.000 & -0.024 \\ \hline 0.024 & 1.000 \end{array}\right], \\
Y_{-6} &= 0.012.
\end{aligned}$$

Thus, Y_k tends towards zero as $k \to -\infty$, and at the same time, $\mathbf{V}_k$ tends towards the identity matrix. At a certain point, (say around $k = -10$, but actually depending on the desired accuracy), we will decide on $d_{Y,k-1} = 0$, after which the number of states in $\mathbf{V}_k$ will be reduced to zero as well:

$$\begin{aligned}
\mathbf{V}_{-9} &= \left[\begin{array}{c|c} 1.000 & -0.000 \\ \hline 0.000 & 1.000 \end{array}\right], \\
\mathbf{V}_{-10} &= \left[\begin{array}{c|c} \cdot & \cdot \\ \hline 0.000 & 1.000 \end{array}\right],
\end{aligned}$$

$$\mathbf{V}_{-11} = \left[\begin{array}{c|c} \cdot & \cdot \\ \hline \cdot & 1.000 \end{array}\right].$$

This brings us back to the LTI solution for this part of T. It is seen from $\mathbf{V}_{-10}$ that it is not unitary at this point in time: only $\mathbf{V}_{-10}\mathbf{V}_{-10}^* = I$ holds, but $\mathbf{V}_{-10}^*\mathbf{V}_{-10} \neq I$. Consequently, $VV^* = I$ but $V^*V \neq I$, i.e., V is not unitary but isometric, and hence T is only left invertible. The situation is not unlike T in (6), but less pronounced.

The outer-inner factorization of T is thus

$$T_0 = \left[\begin{array}{cccccc|cccc}
\ddots & \ddots &&&&&&&& \\
& 1 & -0.5 &&&&&& \mathbf{0} & \\
&& 1 & -0.5 &&&&&& \\
&&& 1 & -0.49 &&&&& \\
&&&& 1 & -0.46 &&&& \\
&&&&& 1.09 & -0.25 &&& \\ \hline
&&&&&& 2 & -1 && \\
& \mathbf{0} &&&&&& 2 & -1 & \\
&&&&&&&& 2 & \ddots \\
&&&&&&&&& \ddots
\end{array}\right],$$

$$V = \left[\begin{array}{cccccc|cccc}
\ddots & \vdots &&&&&&& \vdots & \\
& 1 & -0.00 & -0.00 & -0.00 & -0.01 & -0.02 & -0.01 & -0.00 & \cdots \\
&& 1 & -0.00 & -0.01 & -0.02 & -0.04 & -0.02 & -0.01 & \\
&&& 1 & -0.02 & -0.04 & -0.08 & -0.04 & -0.02 & \\
&&&& 0.98 & -0.08 & -0.15 & -0.08 & -0.04 & \\
&&&&& 0.92 & -0.34 & -0.17 & -0.09 & \\ \hline
&&&&&& 0.5 & -0.75 & -0.37 & \\
&&&&&&& 0.5 & -0.75 & \\
& \mathbf{0} &&&&&&& 0.5 & \cdots \\
&&&&&&&&& \ddots
\end{array}\right].$$

The (left) inverse of T is
$T^\dagger = V^* T_{0,\ell} =$

$$\left[\begin{array}{ccccc|ccccc}
\ddots & \vdots &&&&&&& \vdots & \\
\cdots & 1.00 & 0.49 & 0.24 & 0.10 & 0.01 & 0.01 & 0.00 & 0.00 & \cdots \\
& -0.01 & 0.99 & 0.48 & 0.20 & 0.03 & 0.01 & 0.01 & 0.00 & \\
& -0.01 & -0.02 & 0.95 & 0.40 & 0.05 & 0.03 & 0.01 & 0.01 & \\
& -0.02 & -0.05 & -0.10 & 0.80 & 0.10 & 0.05 & 0.03 & 0.01 & \\ \hline
& -0.05 & -0.10 & -0.20 & -0.40 & 0.20 & 0.10 & 0.05 & 0.03 & \\
& -0.02 & -0.05 & -0.10 & -0.20 & -0.40 & 0.05 & 0.03 & 0.01 & \\
& -0.01 & -0.02 & -0.05 & -0.10 & -0.20 & -0.47 & 0.01 & 0.01 & \\
\cdots & -0.01 & -0.01 & -0.02 & -0.05 & -0.10 & -0.24 & -0.49 & 0.00 & \cdots \\
& \vdots &&&&&&& \vdots & \ddots
\end{array}\right].$$

It has indeed the structure which we announced in eq. (5): it is Toeplitz towards $(-\infty, -\infty)$ and $(+\infty, +\infty)$, and equal to the solution of the LTI subsystems of T in those regions. In addition, there is some limited interaction in the center which glues the two solutions together. All entries are nicely bounded.

4. Conclusion

In this paper, we have looked at what could be called the 'stable inversion' of large matrices. For the case of upper triangular matrices, this means that instead of insisting on an upper triangular but unstable inverse, we allow the inverse to have a lower triangular anti-causal part. In LTI systems theory, the relation between unstability and anti-causality is well-known (they are the same in the z-domain), but for the general time-varying framework and from the matrix point of view, essentially the same notions lead to perhaps surprising results. Also remarkable is the fact that global lossless factorizations (QR factorizations) can be computed by local QR factorizations on time-varying realization matrices. It should be noted that, although all our time-varying examples were intentionally of the most elementary form (a single step of only one parameter), the theory and algorithms really apply to time-varying systems in general, i.e., to any large matrix.

Acknowledgement

Many of the results and insights leading to this paper have been obtained during years of fruitful and pleasant collaboration with Professor P. M. Dewilde.

This research was supported in part by the EC under the ESPRIT BRA program 6632 (NANA2).

References

[1] Deprettere, E. F.; Theunis, R.: Piecewise stationary perfect reconstruction filter banks. Arch. f. Elektron. u. Übertr.tech. **49** (1995).

[2] Di Benedetto, M. D.; Lucibello, P.: Inversion of nonlinear time-varying systems. IEEE Trans. Automat. Control AC–**38** (1993), 1259–1264.

[3] Arveson, W.: Interpolation problems in nest algebras. J. Functional Analalysis **20** (1975), 208–233.

[4] Gohberg, I.; Kaashoek, M. A.; van Schagen, F.: Non-compact integral operators with semi-separable kernels and their discrete analogues: Inversion and Fredholm properties. Integral Equation Operator Theory **17** (1984), 642–703.

[5] Ben-Artzi, A.; Gohberg, I.; Kaashoek, M. A.: Exponentially dominated infinite block matrices of finite Kronecker rank. Integral Equation Operator Theory **17** (1994).

[6] van der Veen, A. J.: Computation of the inner-outer factorization for time-varying systems. Challenges of a Generalized System Theory, Amsterdam, The Netherlands, 1993. Dewilde, P. et al. (eds.). North-Holland, 99–117.

[7] van der Veen, A. J.; Dewilde, P. M.: Time-varying computational networks: Realization, orthogonal embedding and structural factorization. Integration, the VLSI journal **16** (1993), 267–291.

[8] Dewilde, P. M.: On the synthesis of lossless computational circuits. Arch. f. Elektron. u. Übertr.tech. **49** (1995).

[9] Kalman, R. E.; Falb, P. L.; Arbib, M. A.: Topics in mathematical system theory. New York: McGraw-Hill, 1970. – (Int. Series in Pure and Applied Math.).

[10] van der Veen, A. J.; Dewilde, P. M.: Time-varying system theory for computational networks. – In: Algorithms and Parallel VLSI Architectures, II. Quinton, P.; Robert, Y. (eds.). Amsterdam: Elsevier, 1991, 103–127.

[11] van der Veen, A. J.: Time-varying system theory and computational modeling: Realization, approximation, and factorization. Dissertation. Delft, The Netherlands: Delft University of Technology, 1993.

[12] Gohberg, I.; Kaashoek, M. A.; Lerer, L.: Minimality and realization of discrete time-varying systems. – In: Time Variant Systems and Interpolation. Gohberg, I. (ed.). Basel: Birkhäuser Verlag, 1992, 261–296.

[13] Dewilde, P. M.; van der Veen, A. J.: On the Hankel-norm approximation of upper-triangular operators and matrices. Integral Equation Operator Theory **17** (1993), 1–45.
[14] van der Veen, A. J.; Dewilde, P. M.: On low-complexity approximation of matrices. Linear Algebra and its Appl. **205/206** (1994), 1145–1201.
[15] De Nicolao, G.: On the time-varying Riccati difference equation of optimal filtering. SIAM J. Control a. Optimization **30** (1992), 1251–1269.
[16] Bittanti, S.; Laub, A. J.; Willems, J. C. (eds.): The Riccati equation. Berlin: Springer Verlag, 1991. – (Comm. Control Eng. Series).
[17] Golub, G.; van Loan, C. F.: Matrix computations. Baltimore: The Johns Hopkins University Press, 1989.

Alle-Jan van der Veen was born in The Netherlands in 1966. He graduated (cum laude) from Delft University of Technology, Department of Electrical Engineering, in 1988, and received the Ph.D. degree (cum laude) in 1993 with the Network Theory Section at the same university. In 1994, he was a postdoctoral scholar at Stanford University, in the Computational Mathematics department and the Information Systems Lab. His research interests are in the areas of signal processing, system theory (in particular system identification, model reduction and time-varying system theory), and in parallel algorithms for linear algebra.