Time-Variant and Quasi-separable Systems*
– Supplementary Reading –

.

Semi-Separable Matrices

Klaus Diepold, Patrick Dewilde

Spring 2025

1 Introduction

This chapter provides a short introduction to the notion of Semi-separable matrices. This matrix structure occurs frequently with banded matrices or inverses thereof. While there exists a variety of definitions for semi-separability I will focus on the notion that is in alignment with the theory of time-varying state-space systems [2]. This approach allows for engineers to directly comprehend various concepts essential in this context. A more comprehensive exposition to this topic is available in [2].

2 Separable Matrix

A separable function, given as a multivariate function $T(x, y)$ is called separable, if it can be factorized in two independent factors, i.e. when we have

$$T_S(x, y) = P(x) \cdot Q(y).$$

In matrix terms, we express this separability as the matrix T being factorized into two rank-1 factors

$$T_S = p \cdot q^T = \begin{bmatrix} p_1q_1 & p_1q_2 & p_1q_3 & \cdots & p_1q_n \\ p_2q_1 & p_2q_2 & p_2q_3 & \cdots & p_2q_n \\ p_3q_1 & p_3q_2 & p_3q_3 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_mq_1 & p_mq_2 & p_mq_3 & \cdots & p_mq_n \end{bmatrix}. \tag{1}$$

Note that the matrix T_S is restricted to have rank-1. We encounter the concept of separability regularly in the context of image processing, where we exploit this separability to implement image filtering operations

*P.Dewilde, K.Diepold, A.-J. v.d. Veen. Time-Variant and Quasi-separable Systems, Cambridge University Press, 2024 more efficiently. A separable image filter allows us to first process all the rows of an image with a first 1-dimensional filter, say, $P(x)$ and then apply another 1-dimensional filter $Q(y)$ to the columns of row-wise filtered image.

3 Semi-separable Matrix

3.1 Representation

The concept of semi-separability [1] uses a restriction of the notion of separability (Equation 1) on the upper or lower triangular part of a matrix. Using Matlab notation to represent the lower triangular part of a separable matrix, we can denote the semi-separable (SS) matrix as

$$T_{SS} = \text{tril}(p \cdot q^T) = \begin{bmatrix} p_1q_1 & & & & \\ p_2q_1 & p_2q_2 & & & \\ p_3q_1 & p_3q_2 & p_3q_3 & & \\ \vdots & \vdots & \vdots & \ddots & \\ p_nq_1 & p_nq_2 & p_nq_3 & \cdots & p_nq_n, \end{bmatrix} \tag{2}$$

where p and q denote vectors of dimension n.

Only half the matrix is covered by the notion of separability, hence the name *semi-separability*. Observe, that this matrix T_{SS} may have full rank, in in contrast to the matrix T from the previous section. We can also observe that all the submatrices that we can build in the strictly lower triangular part of matrix T_{SS} have rank 1.

Alternatively, we can also denote an upper triangular structure based on the concept of semi-separability.

$$T_{SS} = \text{triu}(v \cdot w^T) = \begin{bmatrix} v_1w_1 & v_1w_2 & v_1w_3 & \cdots & v_1w_n \\ & v_2w_2 & v_2w_3 & \cdots & v_2w_n \\ & & v_3w_3 & \cdots & v_3w_n \\ & & & \ddots & \vdots \\ & & & & v_nw_n \end{bmatrix} \tag{3}$$

Again, this matrix may also exhibit full rank. Slightly more general, combining these two matrices for an upper- and a lower triangular part we talk about a semi-seperable matrix if we can represent it in the form

$$T_{SS} = tril(p \cdot q^T) + triu(v \cdot w^T, 1) = \begin{bmatrix} p_1q_1 & v_1w_2 & v_1w_3 & \cdots & v_1w_n \\ p_2q_1 & p_2q_2 & v_2w_3 & \cdots & v_2w_n \\ p_3q_1 & p_3q_2 & p_3q_3 & \cdots & v_3w_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_mq_1 & p_mq_2 & p_mq_3 & \cdots & p_mq_n \end{bmatrix}$$

There exist various different forms of definitions for semi-separable matrices [4]. For example. this form of semi-separability is often slightly modified to just apply to strictly upper and/or strictly lower triangular matrices, such that the diagonal is kept as a separate additive item , i.e. $T_{SS} = \text{tril}(p \cdot q^T, -1) + \text{triu}(v \cdot w^T, 1) + \text{diag}(T_{ss})$.

One particular form of semi-separability comes in the shape of sequentially semi-separability, to be covered next.

3.2 Matrix-Vector Multiplication wit Semi-separable Matrices

We now have a look at a matrix-vector multiplication, where the coefficient matrix is a lower triangular semi-separable matrix is. More specifically, we look at

$$y = T_{SS} \cdot u = \begin{bmatrix} p_1q_1 & 0 & \dots & \dots & 0 \\ p_2q_1 & p_2q_2 & 0 & \dots & 0 \\ p_3q_1 & p_3q_2 & p_3q_3 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ p_nq_1 & p_nq_2 & p_nq_3 & \dots & p_nq_n, \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}. \tag{4}$$

Expanding the rows leads to the following equations

$$\begin{aligned} y_1 &= p_1q_1u_1 \\ y_2 &= p_2q_1u_1 + p_2q_2u_2 = p_2(q_1u_1 + q_2u_2) \\ y_3 &= p_3q_1u_1 + p_3q_2u_2 + p_3q_3u_3 = p_3 \underbrace{(q_1u_1 + q_2u_2 + q_3u_3)}_{\Delta_3} \\ y_4 &= p_4q_1u_1 + p_4q_2u_2 + p_4q_3u_3 + p_4q_4u_4 = p_4 \underbrace{(\Delta_3 + q_4u_4)}_{\Delta_4} \\ \vdots & \quad \vdots \\ y_n &= p_n(q_1u_1 + q_2u_2 + \dots + q_nu_n) = p_n(\Delta_{n-1} + q_nu_n) \end{aligned}$$

Looking at this set of equations we can rewrite the formula for each component as

$$y_i = p_i \sum_{k=1}^{i} q_k u_k.$$

Computing each of the n component in y requires 3 operations (2 Multiplications, 1 Addition), except for the first component, which requires only 2 multiplications. This leads to a computational complexity $3n - 1$ operations for performing this matrix-vector multiplication. This is a significant reduction in computations as compared to a matrix-vector multiplication using a generic lower triangular matrix.

3.3 Inverse of Bi-Diagonal and Tri-Diagonal Matrices

3.3.1 Bi-diagonal Matrices

We can observe that the inverse of a lower bi-diagonal matrix B is lower, but no longer bi-diagonal. Check for example the inversion of a bi-diagonal matrix B given as

$$B = \begin{bmatrix} 1 & & & & \\ b_1 & 1 & & & \\ 0 & b_2 & 1 & & \\ \vdots & \ddots & \ddots & \ddots & \\ 0 & \dots & 0 & b_{n-1} & 1 \end{bmatrix} \quad \rightarrow \quad B^{-1} = \begin{bmatrix} 1 & & & & & \\ -b_1 & 1 & & & & \\ b_1b_2 & -b_2 & 1 & & & \\ -b_1b_2b_3 & -b_2b_3 & -b_3 & 1 & & \\ \vdots & \vdots & \vdots & & \ddots & \\ \pm b_1 \dots b_{n-1} & \pm b_2 \dots b_{n-1} & \pm b_3 \dots b_{n-1} & \dots & \dots & 1 \end{bmatrix}. \tag{5}$$

This bi-diagonal matrix B only exhibits $n-1$ free parameters, which also applies to B^{-1}. Looking at Equation 5 we can also immediately see that B^{-1} is a lower triangular semi-separable matrix.

3.3.2 Tri-diagonal Matrices

As a next we consider a tri-diagonal matrix T. It is easy to check that except for special cases the inverse of a tri-diagonal matrix is no longer tri-diagonal, but it is semi-separable. In such cases we have

$$T = \begin{bmatrix} t_{11} & t_{12} & 0 & \dots & 0 \\ t_{21} & t_{22} & t_{23} & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & & & t_{n-1n} \\ 0 & \dots & 0 & t_{nn-1} & t_{nn} \end{bmatrix} \quad \rightarrow \quad T^{-1} = \begin{bmatrix} \star & \star & \star & \dots & \star \\ \star & \star & \star & \dots & \star \\ \star & \star & \star & \dots & \star \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \star & \star & \star & \dots & \star \end{bmatrix} \quad \leftarrow \text{semi-separable}$$

The tri-diagonal matrix T has $3n-2$ free parameters. The number of parameters does not change upon inversion. Therefore, even if T^{-1} does not show it directly, it also has only $3n-2$ free parameters.

hier kommt noch etwas um klarer zu zeigen, dass T^{-1} semi-separable ist.

3.4 QR Decomposition of Semi-separable Matrices

Now, we shall have a quick look on basic observations concerning the QR decomposition of semi-separable matrices. Computing the QR decomposition of T amounts to determining the orthogonal factor Q ($Q^TQ = 1$) and the upper triangular factor R as

$$T = Q \cdot R.$$

For that purpose, let's take a look at a 5×5 example of the lower semi-separable matrix as introduced in Equation 2 as an illustrative example. We start the QR decomposition with a Givens rotation appied to rows 4 and 5 with the goal to eliminate the 51 entry of T, i.e. we aim at

$$\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} \begin{bmatrix} p_1q_1 & & & & \\ p_2q_1 & p_2q_2 & & & \\ p_3q_1 & p_3q_2 & p_3q_3 & & \\ \hline p_4q_1 & p_4q_2 & p_4q_3 & p_4q_4 & \\ p_5q_1 & p_5q_2 & p_5q_3 & p_5q_4 & p_5q_5 \end{bmatrix} \overset{G_{45}^T}{\rightarrow} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} \begin{bmatrix} p_1q_1 & & & & \\ p_2q_1 & p_2q_2 & & & \\ p_3q_1 & p_3q_2 & p_3q_3 & & \\ \hline \star & \star & \star & \star & \star \\ 0 & 0 & 0 & 0 & \star \end{bmatrix},$$

where G_{45} denotes a Givens rotation acting on rows 4 and 5 and $\star$ denoting any arbitrary number (e.g. 42). Extracting the rows #4 and #5 from the matrix we can see that the row vectors made up by the first 4 entries in rows #4 and #5 are linear dependent, i.e. we have

$$\begin{matrix} 4 \\ 5 \end{matrix} \begin{bmatrix} p_4q_1 & p_4q_2 & p_4q_3 & p_4q_4 & 0 \\ p_5q_1 & p_5q_2 & p_5q_3 & p_5q_4 & p_5q_5 \end{bmatrix} = \begin{bmatrix} p_4 & 0 \\ 0 & p_5 \end{bmatrix} \cdot \left[\begin{array}{cccc|c} q_1 & q_2 & q_3 & q_4 & 0 \\ q_1 & q_2 & q_3 & q_4 & q_5 \end{array}\right].$$

Therefore, eliminating the 51 entry will produce additional zeros in row 5 basically for free, i.e. we see

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \left[\begin{array}{cccc|c} q_1 & q_2 & q_3 & q_4 & 0 \\ q_1 & q_2 & q_3 & q_4 & q_5 \end{array}\right] = \left[\begin{array}{cccc|c} q_1 & q_2 & q_3 & q_4 & \frac{q_5}{2} \\ 0 & 0 & 0 & 0 & -\frac{q_5}{2} \end{array}\right]$$

happening.

If T is a semi-separable matrix, then the lower triangular part of the matrix Q produced by a QR decomposition of T also has semi-separable structure.

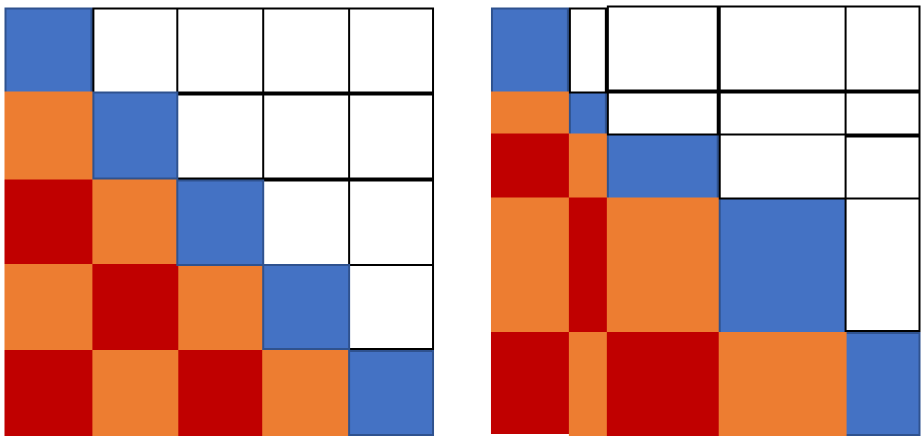

Figure 1: Subdivision of lower triangular (causal) matrix T, left - regular subdivision, right - irregular subdivision

4 Quasi-separable Matrix

4.1 Opening Comments

A further type of structured matrices, which also belongs to the family of semi-separable matrices is the class of *sequentially semi-separable matrices*, which is also called *quasi-separable matrices* and which we will discuss in the following sections, and which is the most prominent type of matrix structure we will handle in this course.

4.2 Simple Task

We use the simple task of matrix-vector multiplication

$$y = T \cdot u, \quad T \in \mathcal{R}^{n \times n}, \quad u, y \in \mathcal{R}^{n}, \tag{6}$$

but we equally well consider matrix-matrix multiplication, all conceivable matrix factorization tasks as well as matrix inversion. All these matrix operations can benefit from any semi-separable structure a matrix T may exhibit. We also consider all shapes of matrices and we do not restrict our discussion to quadratic or even just invertible matrices.

4.3 Lower Triangular Matrix

As a first step I just consider lower triangular matrices and then move on to full matrices. The lower triangular matrix can exhibit a regular or an irregular block structure as shown in Figure 1.

Looking only at a small example of a 5×5 lower triangular matrix a *sequentially semi-separable* matrix is defined as being representable as

$$T_{SSS} = \begin{bmatrix} D_1 & 0 & 0 & 0 & 0 \\ C_2B_1 & D_2 & 0 & 0 & 0 \\ C_3A_2B_1 & C_3B_2 & D_3 & 0 & 0 \\ C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & D_4 & 0 \\ C_5A_4A_3A_2B_1 & C_5A_4A_3B_2 & C_5A_4B_3 & C_5B_4 & D_5 \end{bmatrix}, \tag{7}$$

where all the matrices A_k, B_k, C_k and D_k for $k = 1, \ldots 5$ may take on a wide range of dimensions (including 0-dimensional matrices) as long as the products of these matrices appearing in Equation 7 are well defined. This representation is the core piece for sequentially semi-separable structures. For now we assume that we know the individual matrices A_k, B_k, C_k and D_k for $k = 1, \ldots 5$ showing up in this parameterization.

If all the matrices A_k in Equation 7 are the identity, i.e. we have

$$T_{SSS} = \begin{bmatrix} D_1 & 0 & 0 & 0 & 0 \\ C_2B_1 & D_2 & 0 & 0 & 0 \\ C_3B_1 & C_3B_2 & D_3 & 0 & 0 \\ C_4B_1 & C_4B_2 & C_4B_3 & D_4 & 0 \\ C_5B_1 & C_5B_2 & C_5B_3 & C_5B_4 & D_5 \end{bmatrix} = \text{tril}\left(\begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{bmatrix} \begin{bmatrix} B_1 & B_2 & B_3 & B_4 & B_5 \end{bmatrix} \right). \tag{8}$$

We can see that the matrix resulting from this simplification happens to be a lower triangular semi-separable matrix. That indicates that the sequentially semi-separable matrices contain the semi-separable matrices as a special case.

4.4 Full Matrix

The notion of sequentially semi-separable matrix structure also applies to full matrices. A full matrix is considered to consist of a lower triangular part and a strictly upper triangular part. The concept that we have seen for the lower-triangular matrix can be extended to the strictly upper triangular part as

$$T_{SSS} = \begin{bmatrix} D_1 & G_1F_2 & G_1E_2F_3 & G_1E_2E_3F_4 & G_1E_2E_3E_4F_5 \\ C_2B_1 & D_2 & G_2F_3 & G_2E_3F_4 & G_2E_3E_4F_5 \\ C_3A_2B_1 & C_3B_2 & D_3 & G_3F_4 & G_3E_4F_5 \\ C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & D_4 & G_4F5 \\ C_5A_4A_3A_2B_1 & C_5A_4A_3B_2 & C_5A_4B_3 & C_5B_4 & D_5 \end{bmatrix}. \tag{9}$$

We can see that the lower triangular part of this matrix T_{SSS} is identical to the lower triangular matrix introduced in the previous section. For the strictly upper triangular part we have introduced new parameters E_k, F_k, H_k, which are composed in a similar way than the A_k, B_k, C_k for their r the strictly lower triangular part. The diagonal block D_k are on their own so that we can separate the matrix into three separate components as

$$T_{SSS} = \mathcal{L}(A_k, B_k, C_k) + \mathcal{D}(D_k) + \mathcal{U}(E_k, F_k, H_k), \quad k = 1, 2, \ldots N$$

where $\mathcal{L}, \mathcal{D}, \mathcal{U}$ denote the lower, diagonal and upper part, respectively. Looking at Figure 2 we can see that the diagonal blocks D_k (see the blue coloured boxes) are not restricted to be quadratic.

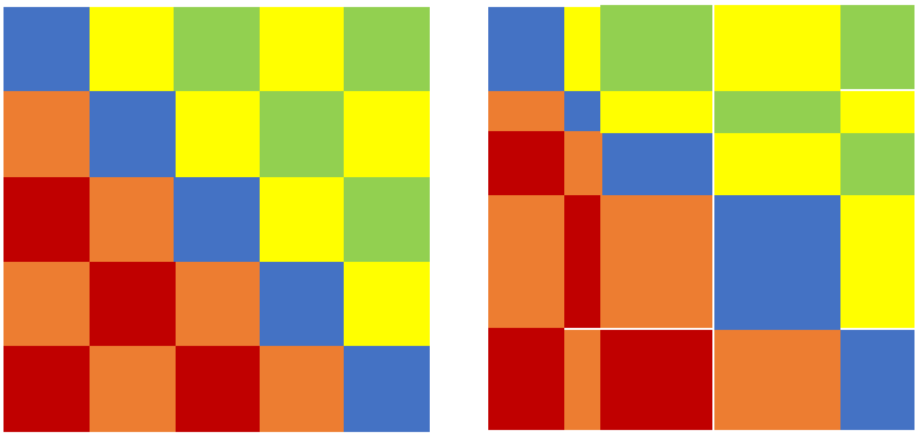

Figure 2: Sub-blocks of the matrix T can have regular shape (left figure) or irregular shape (right figure). The sequentially semi-separable character is visible by the lower part $\mathcal{L}$ (red/orange rectangles), the diagonal part $\mathcal{D}$(blue rectangles) and the upper part $\mathcal{U}$) (green/yellow rectangles).

4.5 Matrix-Vector Multiplication

We have a closer look at the matrix-vector multiplication using the lower-triangular sequentially semi-separable (SSS) matrix given as

$$y = T_{SSS} \cdot u.$$

Plugging in the little 5×5 lower triangular example matrix

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1 & 0 & 0 & 0 & 0 \\ C_2B_1 & D_2 & 0 & 0 & 0 \\ C_3A_2B_1 & C_3B_2 & D_3 & 0 & 0 \\ C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & D_4 & 0 \\ C_5A_4A_3A_2B_1 & C_5A_4A_3B_2 & C_5A_4B_3 & C_5B_4 & D_5 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}$$

we can expand the product fully as

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1u_1 \\ C_2(B_1u_1) + D_2u_2 \\ C_3A_2(B_1u_1) + C_3(B_2u_2) + D_3u_3 \\ C_4A_3A_2(B_1u_1) + C_4A_3(B_2u_2) + C_4(B_3u_3) + D_4u_4 \\ C_5A_4A_3A_2(B_1u_1) + C_5A_4A_3(B_2u_2) + C_5A_4(B_3u_3) + C_5(B_4u_4) + D_5u_5 \end{bmatrix}.$$

Looking at this example we can deduce a more general scheme for computing the vector entry y_k as

$$y_k = C_kA_{k-1} \dots A_2B_ku_1 + C_kA_{k-1} \dots A_3B_2u_2 + \dots + C_kB_{k-1}u_{k-1} + D_ku_k.$$

Staring at this formula long enough reveals that we can speed up these calculations by introducing intermediate quantities

$$x_1 = B_1u_1, \quad \text{and} \quad x_k = B_ku_k + A_kx_{k-1}, \quad k = 2, \dots n. \tag{10}$$

We can successively insert these intermediate quantities, i.e. next we get

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1u_1 \\ C_2x_1 + D_2u_2 \\ C_3(A_2x_1 + B_2u_2) + D_3u_3 \\ C_4A_3(A_2x_1 + B_2u_2) + C_4(B_3u_3) + D_4u_4 \\ C_5A_4A_3(A_2x_1 + B_2u_2) + C_5A_4(B_3u_3) + C_5(B_4u_4) + D_5u_5 \end{bmatrix}.$$

yet another step of inserting intermediate quantities produces

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1u_1 \\ C_2x_1 + D_2u_2 \\ C_3x_2 + D_3u_3 \\ C_4(A_3x_2 + B_3u_3) + D_4u_4 \\ C_5A_4(A_3x_2 + B_3u_3) + C_5(B_4u_4) + D_5u_5 \end{bmatrix},$$

almost done with plugging in

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1u_1 \\ C_2x_1 + D_2u_2 \\ C_3x_2 + D_3u_3 \\ C_4x_3 + D_4u_4 \\ C_5(A_4x_3 + B_4u_4) + D_5u_5 \end{bmatrix},$$

and finally we arrive at

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} D_1u_1 \\ C_2x_1 + D_2u_2 \\ C_3x_2 + D_3u_3 \\ C_4x_3 + D_4u_4 \\ C_5x_4 + D_5u_5 \end{bmatrix}.$$

At the end of this process we observe that we get the result of our matrix-vector multiplication as

$$y_1 = D_1u_1, \quad y_k = C_kx_{k-1} + D_ku_k, \quad 1 < k < n, \quad \text{and} \quad y_n = C_nx_{n-1} + D_nu_n \tag{11}$$

Making systematic use of intermediate results we can build up an efficient computational scheme that uses less arithmetic operations than a plain straight-forward matrix-vector multiplication. It is also obvious that the efficiency may dependent on the actual size of the involved matrices A_k, B_k, C_k, D_k. That represents the core observation that makes the sequentially semi-separable matrix structure attractive for the design of efficient algorithms. In [3], the authors show that this process provides an efficient algorithm for matrix-vector multiplication ...

From an engineering point of view the ensemble of Equations 10 and 11 look very familiar and can be recognized as a discrete-time state-space representation for a causal, linear, time-varying system (see further [2]). We defer a more detailed discussion of this important observation to a later chapter.

4.6 Computational Model

4.6.1 Direct Realization

In this section we present a simple example for a lower triangular matrix for which we devise a computational model in terms of a graphical representation to depict the computations involved in a matrix-vector multiplication, which is also known as a signal flow graph well established in signal processing. For that purpose look at a matrix T given as

$$T = \begin{bmatrix} 1 & & & & \\ -1/2 & 1 & & & \\ 1/6 & -1/3 & 1 & & \\ -1/24 & 1/12 & -1/4 & 1 & \\ 1/120 & -1/60 & 1/30 & -1/5 & 1 \end{bmatrix} = \begin{bmatrix} D_1 & 0 & 0 & 0 & 0 \\ C_2B_1 & D_2 & 0 & 0 & 0 \\ C_3A_2B_1 & C_3B_2 & D_3 & 0 & 0 \\ C_4A_3A_2B_1 & C_4A_3B_2 & C_4B_3 & D_4 & 0 \\ C_5A_4A_3A_2B_1 & C_5A_4A_3B_2 & C_5A_4B_3 & C_5B_4 & D_5 \end{bmatrix},$$

where we already gave the representation of T in terms of the parameters A_k, B_k, C_k, D_k pertaining to the semi-separable structure.

This is actually used to determine the matrix-vector multiplication as introduced in the previous section. If we follow the lines in the signal flow graph, then we can verify in a straight forward way by inspection that this is actually the case.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ -1/2 & 1 & & & \\ 1/6 & -1/3 & 1 & & \\ -1/24 & 1/12 & -1/4 & 1 & \\ 1/120 & -1/60 & 1/30 & -1/5 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 - 1/2u_1 \\ u_3 - 1/3u_2 + 1/6u_1 \\ u_4 1/4u_e + 1/12u_2 - 1/24u_1 \\ u_5 - 1/5u_4 + 1/30u_3 - 1/60u_2 + 1/120u_1 \end{bmatrix}$$

Looking at the computational structure in Figure 3 we can quickly determine the computational load. We count the non-trivial multiplications (red boxes) as well as the number of latches (denoted by gray boxes with the letter Z), which is gives us the amount of additional memory needed to perform the computation. We can see, that our example requires 10 multiplications and 10 latches. Additionally, we need to store the matrix entries of T, for which we need memory for 10 values.

4.6.2 Elementary Computing Block

If we were to use this matrix to perform the simple matrix-vector multiplication $T \cdot u = y$ we can represent the computational task graphically as a signal flow graph as it is shown in Figure 3. in Figure 4 an elementary computational building block is shown, along with the associated elementary computations.

We can summarize the computations associated with an elementary computing block by means of a simple matrix

$$\Sigma_k = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right].$$

The parameters $A_k B_k, C_k, d_k$ also appear in the elementary signal flow graph shown in Figure 4. For the signal flow graph of our 5×5 example shown in Figure 3 we can read off the corresponding parameters for the elementary building blocks to be

$$\Sigma_1 = \left[\begin{array}{c|c} [\cdot] & [1] \\ \hline [\cdot] & [1] \end{array}\right], \quad \Sigma_2 = \left[\begin{array}{c|c} \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \hline [-1/2] & [1] \end{array}\right], \quad \Sigma_3 = \left[\begin{array}{c|c} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \hline \begin{bmatrix} 1/6 & -1/3 \end{bmatrix} & [1] \end{array}\right]$$

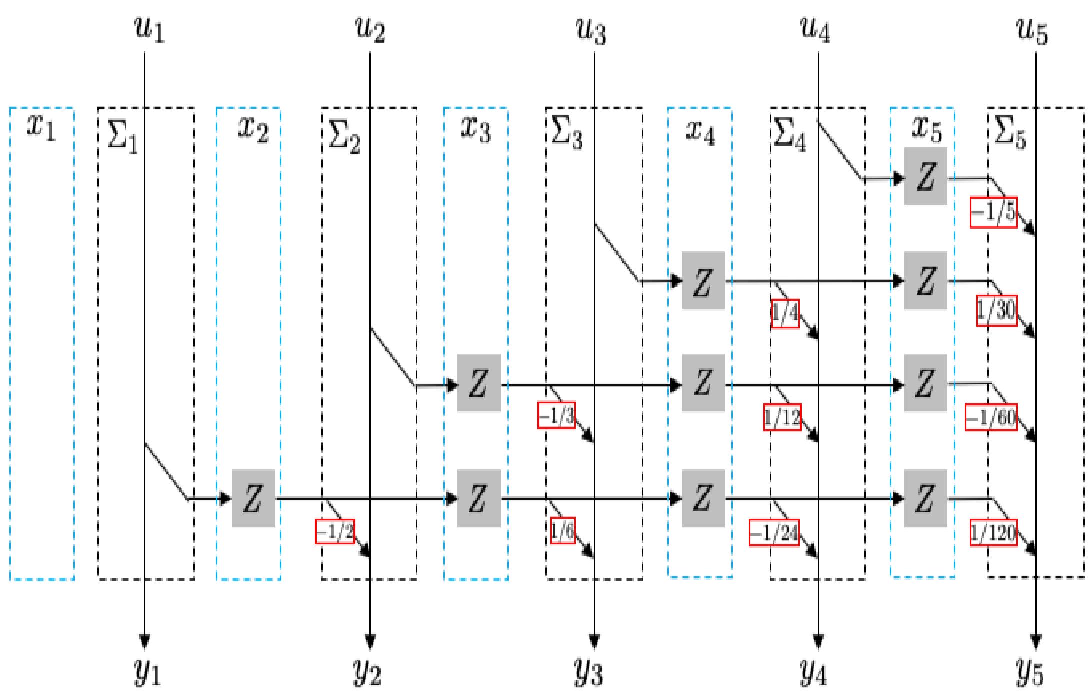

Figure 3: Computational model for representing the 5×5 lower triangular matrix T and the data flow for the computation of $T \cdot u = y$.

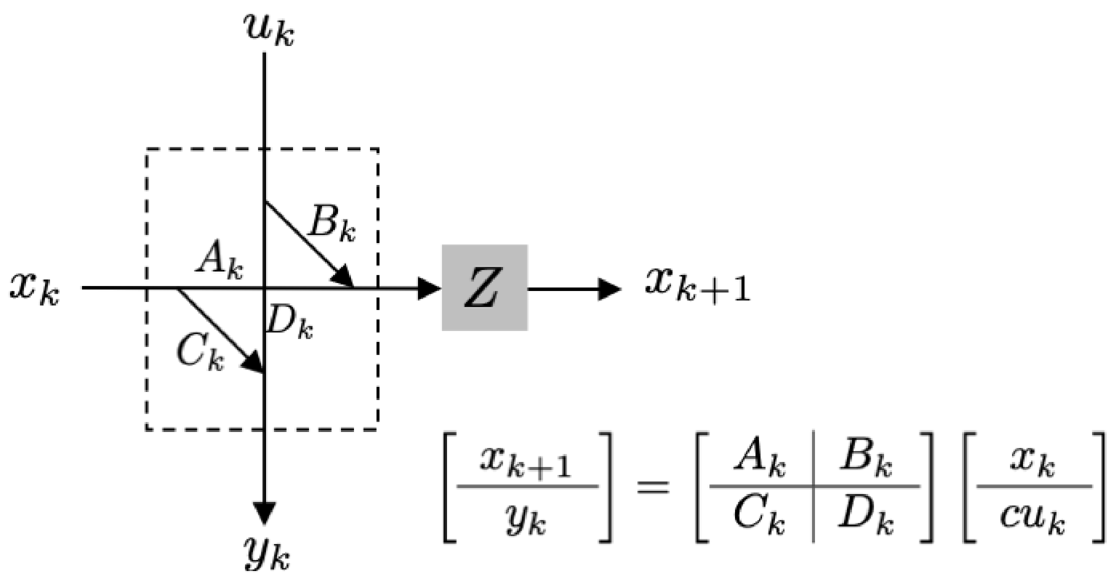

Figure 4: Elementary computational building block

$$\Sigma_4 = \left[\begin{array}{c|c} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \\ \hline \begin{bmatrix} -1/24 & 1/12 & 1/4 \end{bmatrix} & [1] \end{array}\right], \quad \Sigma_5 = \left[\begin{array}{c|c} [\cdot] & [\cdot] \\ \hline \begin{bmatrix} 1/120 & -1/60 & 1/30 & -1/5 \end{bmatrix} & [1] \end{array}\right],$$

where the symbol $[\cdot]$ denotes a matrix with dimension 0.

4.6.3 Alternative Realization

We now have a look at the computational model as shown in Figure 5. While this model looks distinctively different than the one shown in Figure 3 we can check by visual inspection that this structure performs exactly the same matrix-vector multiplication. If we use this model, we can again assess the amount of computational resources needed. We end up with only 7 non-trivial multiplications and only 4 latches. By finding the second structure, we have reduced the number of multiplications by 30% and the amount of memory by 60%. Also, while we needed to store all 10 values of the matrix entries we now only have to store 7, which is another reduction in the memory footprint for storing the matrix T.

We can summarize the computations associated with an elementary computing block by means of a simple matrix

$$\Sigma_k = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right].$$

For the signal flow graph of our 5×5 example shown in Figure 5 we can read off the corresponding

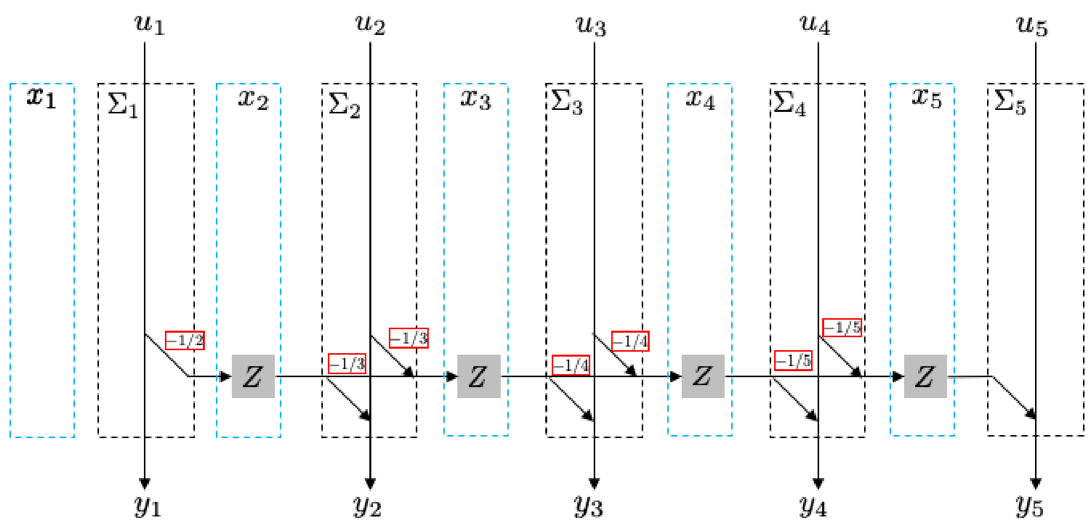

Figure 5: Equivalent Computational model for representing the 5×5 lower triangular matrix T and the data flow for the computation of $T \cdot u = y$.

parameters for the elementary building blocks to be

$$\Sigma_1 = \left[\begin{array}{c|c} [\cdot] & [-1/2] \\ \hline [\cdot] & [1] \end{array}\right], \quad \Sigma_2 = \left[\begin{array}{c|c} [-1/3] & [-1/3] \\ \hline [1] & [1] \end{array}\right], \quad \Sigma_3 = \left[\begin{array}{c|c} [-1/4] & [-1/4] \\ \hline [1] & [1] \end{array}\right]$$

$$\Sigma_4 = \left[\begin{array}{c|c} [-1/5] & [-1/5] \\ \hline [1] & [1] \end{array}\right] \quad \Sigma_5 = \left[\begin{array}{c|c} [\cdot] & [\cdot] \\ \hline [1] & [1] \end{array}\right],$$

where the symbol $[\cdot]$ denotes a matrix with dimension 0.

4.7 Questions

While the amount of savings in terms of multiplications and memory for our little example appears to be only moderate, one needs to consider that the size of the matrices in many real-world applications and in deep machine learning is considerably larger.

We can check the structural properties of sequentially semi-separable matrices and convince ourselves of the efficiency of computation. For a more systematic approach to exploitation of this type of structure we have to address and answer a few questions:

1. Just how efficient can we get with matrix-vector multiplication if we were to fully exploit the sequentially semi-separable structure ?
2. Can we characterize the set of matrices that allow for efficient computations? Can all matrices be represented as SSS?

3. How do we determine the parameters A_k, B_k, C_k, D_k that make up the sequentially semi-separable matrix?
4. Can we systematically find alternative computational models, which offer the benefits of reducing computational complexity. Can we resolve this apparent ambiguity?
5. What is a theoretical framework that allows us the systematic study of SSS matrices?
6. How does the computational demands for computing with SSS matrices scale with the size of the matrices - what is the asymptotic computational complexity?

References

[1] I. Gohberg, T. Kailath and I. Koltracht. Linear complexity algorithms for semiseparable matrices. Integral Equations and Operator Theory, vol. 8, pp. 780-804, Birkhauser Verlag, 1985.

[2] P. Dewilde, A.-J. van der Veen. *Time-Varying Systems and Computations*. Kluwer, 1989.

[3] S. Chandrasekaran, P. Dewilde, M. Gu, T. Pals, A.-J. van der Veen, D. White. Fast Stable Solvers for Sequentially Semi-Separable Linear System of Equations. Lawrence Livermore Nationall Laboratory. Report UCRL-CR-151499, 2003.

[4] R.Vandebril, M. van Barel, N. Mastonardi. Matrix Computations and Semiseparable Matrices, Vol.1 Linear Systems. Johns Hopkins University Press, 2008.