Relationship between Granger non-causality and network graph of state-space representations

Jozsa, Monika

Chapter 1
Preliminaries

In this thesis we consider multivariate discrete-time stochastic processes where the discrete-time axis is the set of integers $\mathbb{Z}$. Let $(\mathbb{R}, \mathcal{F}, P)$ be a probability space, where $\mathcal{F}$ is a σ-algebra and P is a probability measure on $\mathcal{F}$. Throughout the thesis, all the random variables and stochastic processes are understood with respect to this probability space. We denote the random variable of a process $\mathbf{z}$ at time $t \in \mathbb{Z}$ by $\mathbf{z}(t)$. If $\mathbf{z}(t)$ is k-dimensional (for all $t \in \mathbb{Z}$) then we call $k = \dim(\mathbf{z})$ the dimension of $\mathbf{z}$ and we write $\mathbf{z}(t) \in \mathbb{R}^k$ or $\mathbf{z} \in \mathbb{R}^k$. If $\mathbf{z}_1, \ldots, \mathbf{z}_n$ are vector-valued processes, then $\mathbf{z} = \left[\mathbf{z}_1^T, \ldots, \mathbf{z}_n^T\right]^T$ denotes the process defined by $\mathbf{z}(t) = \left[\mathbf{z}_1^T(t), \ldots, \mathbf{z}_n^T(t)\right]^T$, $t \in \mathbb{Z}$.

By using standard notation, we denote the covariance matrix of two random variables $\mathbf{y}$ and $\mathbf{z}$ by $E[\mathbf{y}\mathbf{z}^T]$ and we denote the conditional expectation of a $\mathbf{y}$ onto a σ-algebra $\mathcal{F}_1 \subseteq \mathcal{F}$ by $E[\mathbf{y}|\mathcal{F}_1]$.

Throughout the thesis, the $n \times n$ identity matrix is denoted by I_n or by I when its dimension is clear from the context. Likewise, the $n \times m$ zero matrix is denoted by $0_{n,m}$ or by 0.

1.1 Hilbert spaces of stochastic processes

The zero-mean square-integrable random variables form a Hilbert space, denoted by $\mathcal{H}$, with the covariance as the scalar product and with the standard multiplication by scalar and addition of random variables, see (Caines, 1988, Chapter 1) and (Gikhman and Skorokhod, 2004, Chapter 4) for more details.

The closed subspace generated by a set $U \subset \mathcal{H}$ is the smallest (with respect to set inclusion) closed subspace of $\mathcal{H}$ which contains U. The closed subspaces in $\mathcal{H}$ form Hilbert spaces themselves with the same inner product as $\mathcal{H}$. For this reason we call a closed subspace generated by some random variables in $\mathcal{H}$ as the Hilbert space generated by those random variables.

Let $\mathbf{z} \in \mathbb{R}^k$ be a zero-mean square-integrable process and consider a time instant $t \in \mathbb{Z}$ as the present time. Then $\mathcal{H}_{t-}^{\mathbf{z}}$, $\mathcal{H}_{t+}^{\mathbf{z}}$, $\mathcal{H}_{t}^{\mathbf{z}}$ denote the Hilbert spaces generated by the past, future and present values of $\mathbf{z}$, i.e., by the sets $\{\ell^T\mathbf{z}(s) \mid s \in \mathbb{Z}, s < t, \ell \in \mathbb{R}^k\}$, $\{\ell^T\mathbf{z}(s) \mid s \in \mathbb{Z}, s \geqslant t, \ell \in \mathbb{R}^k\}$, and $\{\ell^T\mathbf{z}(t) | \ell \in \mathbb{R}^k\}$, respectively.

If $u \in \mathbb{R}$ is a random variable in $\mathcal{H}$ and $U \in \mathcal{H}$ is a closed subspace, then we denote by $E_l[u \mid U]$ the orthogonal projection of u onto U. The orthogonal projection of a multivariate random variable $u \in \mathbb{R}^k$ onto U is defined element-wise and is denoted by $E_l[u|U]$. That is, $E_l[u|U]$ is the random variable with values in $\mathbb{R}^k$ obtained by projecting the one-dimensional coordinates of u onto U. Accordingly, the orthogonality of u to U is meant element-wise. The orthogonal projection of a closed subspace $U \subseteq \mathcal{H}$ onto a closed subspace $V \subseteq \mathcal{H}$ is defined by $E_l[U|V] := \{E_l[u|V], u \in U\}$. Note that for jointly Gaussian processes $\mathbf{y}$ and $\mathbf{z}$, the orthogonal projection $E_l[\mathbf{y}(t)|\mathcal{H}_t^{\mathbf{z}}]$ of $\mathbf{y}(t)$ onto $\mathcal{H}_t^{\mathbf{z}}$ is equivalent to the conditional expectation $E[\mathbf{y}(t)|\sigma(\mathbf{z}(t))]$ of $\mathbf{y}(t)$ given the σ-algebra generated by $\mathbf{z}(t)$.

Lastly, we mention that in subsequent chapters, for subspaces in $\mathcal{H}$, we will use the following operations: the sum of two subspaces $U, V \subseteq \mathcal{H}$ is written by $U + V := \{u + v | u \in U, v \in V\}$ and the orthogonal complement of U in V (with respect to $\mathcal{H}$) by $V \ominus U$; if $U \cap V = \{0\}$ then the direct sum of them is denoted by $U \dot{+} V$; if U and V are orthogonal then we write the orthogonal direct sum as $U \oplus V$.

1.2 LTI–SS representations

The results of Chapters 2–4 are based on linear stochastic realization theory. Therefore, to provide background material, we introduce the linear stochastic systems that are studied in these chapters and give a brief overview of basic results in the field (see (Lindquist and Picci, 2015)).

1.2.1 Introduction to LTI–SS representations

Below, we provide an introduction of linear time-invariant state-space (LTI–SS) representations. To begin with, we define the class of processes we will work with.

Definition 1.1 (ZMSIR)**.** A stochastic process is called *zero-mean square-integrable with rational spectrum* (abbreviated by ZMSIR) if it is weakly-stationary, square-integrable, zero-mean, purely non-deterministic, and its spectral density is a proper rational function.

See (Lindquist and Picci, 2015; Rozanov, 1987) for further details on the properties of purely non-deterministic, weakly-stationary (wide-sense stationary in (Rozanov, 1987)) processes with rational spectrum. In the literature, it is common to assume that the ZMSIR processes are *coercive*: Recall from (Lindquist and Picci, 2015, Definition 9.4.1) that $\mathbf{y}$ is *coercive* if its spectrum is strictly positive definite on the unit disk. Coercive and non-coercive processes are discussed separately in the main results of Chapters 2–4.

Next, we define the term LTI–SS representation for the class of ZMSIR processes.

Definition 1.2 (LTI–SS representation)**.** A stochastic *LTI–SS representation* is a stochastic dynamical system of the form

$$\begin{aligned} \mathbf{x}(t+1) &= A\mathbf{x}(t) + B\mathbf{v}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{v}(t), \end{aligned} \tag{1.1}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{p\times n}, D \in \mathbb{R}^{p\times m}$ for $n \geqslant 0$, $m, p > 0$ and where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^p$, $\mathbf{v} \in \mathbb{R}^m$ are ZMSIR processes. The processes $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{v}$ are called state, output and noise processes, respectively. Furthermore, we require that A is stable, or equivalently that all its eigenvalues are inside the open unit circle, and that for any $t, k \in \mathbb{Z}$, $k \geqslant 0$, $E[\mathbf{v}(t)\mathbf{v}^T(t-k-1)] = 0$, $E[\mathbf{v}(t)\mathbf{x}^T(t-k)] = 0$, i.e., $\mathbf{v}(t)$ is white noise and uncorrelated with $\mathbf{x}(t-k)$. An LTI–SS representation with output process $\mathbf{y}$ is called *LTI–SS representation of* $\mathbf{y}$.

In (1.1) the state process $\mathbf{x}$ is uniquely determined by the noise process $\mathbf{v}$ and the system matrices A and B so that $\mathbf{x}(t) = \sum_{k=0}^{\infty} A^k B\mathbf{v}(t-k)$, where the convergence of the infinite sum is understood in the mean square sense. On this basis (1.1) is referred to as LTI–SS representation $(A, B, C, D, \mathbf{v}, \mathbf{y})$ or LTI–SS representation $(A, B, C, D, \mathbf{v})$ of $\mathbf{y}$. Following the classical terminology, we call the dimension of the state process $\mathbf{x}$ the *dimension* of the LTI–SS representation (1.1). Also, an LTI–SS representation (1.1) is called *minimal* if it has minimal dimension among all the LTI–SS representations of $\mathbf{y}$. Notice that we allow (1.1) to have zero dimension. Zero dimensional LTI–SS representations corresponds to representations of white noise processes ($\mathbf{y} = D\mathbf{v}$). Whenever we say that $(A, B, C, D, \mathbf{v})$ is a minimal LTI–SS representation of a white noise process, it means that A, B, C are absent (or they are zero by zero empty matrices). Zero-dimensional representations are considered to be minimal, observable and controllable.

Note that the class of ZMSIR processes coincide with the class of processes that can be represented by LTI–SS representations. For convenience, we will assume that the outputs of LTI–SS representations, i.e., of ZMSIR processes, have a so-called full-rank property. To define full-rank property of ZMSIR processes, we use the following terminology: Recall that $\mathcal{H}^{\mathbf{y}}_{t-}$ denotes the Hilbert space generated by $\{\mathbf{y}(t-k)\}_{k=1}^{\infty}$. We call the process

$$\mathbf{e}(t) := \mathbf{y}(t) - E_l[\mathbf{y}(t)|\mathcal{H}^{\mathbf{y}}_{t-}], \quad t \in \mathbb{Z}$$

the *(forward) innovation process* of $\mathbf{y}$.

By full rank property of ZMSIR processes we mean the following:

Definition 1.3. An output process $\mathbf{y}$ of an LTI–SS representation is called *full rank* if the variance matrix of the innovation process of $\mathbf{y}$ is strictly positive definite.

The following assumption will be in force for the rest of the thesis.

Assumption 1.4. *The output process* $\mathbf{y}$ *of an LTI–SS representation* (1.1) *is full rank.*

Assumption 1.4 is a commonly used technical assumption that can not be assumed without loss of generality. However, we know that if $\mathbf{z}$ is a ZMSIR process with innovation process $\mathbf{e}^{\mathbf{z}}$, then there exists a full column rank matrix M and a full rank process $\mathbf{y}$ such that $\mathbf{z} = M\mathbf{y}$ and $\mathbf{e}^{\mathbf{z}} = M\mathbf{e}^{\mathbf{y}}$, where $\mathbf{e}^{\mathbf{y}}$ is the innovation process of $\mathbf{y}$, see (Lindquist and Picci, 2015, (4.46)).

1.2.2 Realization theory of LTI–SS representations

Stochastic LTI–SS representations of a given process $\mathbf{y}$ are strongly related to deterministic LTI–SS realizations of the covariance sequence $\Lambda_k^{\mathbf{y}} := E[\mathbf{y}(t+k)\mathbf{y}^T(t)]$, $k = 0, 1, 2, \ldots$, see (Lindquist and Picci, 2015, Chapter 6) and (Caines, 1988, Chapter 4) for more details. Below we briefly sketch this relationship, as it plays an important role in deriving results in Chapter 2. Consider an LTI–SS representation $(A, B, C, D, \mathbf{v})$ of $\mathbf{y}$ with state process $\mathbf{x}$. Note that weakly stationarity implies that the (co)variance matrices are time-independent. Denote the noise variance matrix by $\Lambda_0^{\mathbf{v}} = E[\mathbf{v}(t)\mathbf{v}^T(t)]$ and the state variance matrix by $\Lambda_0^{\mathbf{x}} = E[\mathbf{x}(t)\mathbf{x}^T(t)]$. Then, $\Lambda_0^{\mathbf{x}}$ is the unique symmetric solution of the Lyapunov equation $\Sigma = A\Sigma A^T + B\Lambda_0^{\mathbf{v}}B^T$ and the covariance $G := E[\mathbf{y}(t)\mathbf{x}^T(t+1)]$ satisfies

$$G = C\Lambda_0^{\mathbf{x}}A^T + D\Lambda_0^{\mathbf{v}}B^T. \tag{1.2}$$

In light of this, the covariances $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{\infty}$ of $\mathbf{y}$ are equal to the Markov parameters of the deterministic LTI–SS system $(A, G^T, C, \Lambda_0^{\mathbf{y}})$, where recall that $\Lambda_k^{\mathbf{y}} = E[\mathbf{y}(t+k)\mathbf{y}^T(t)]$. More precisely, for $k > 0$

$$\Lambda_k^{\mathbf{y}} = CA^{k-1}G^T. \tag{1.3}$$

Therefore, LTI–SS representations of $\mathbf{y}$ yield deterministic LTI–SS systems whose Markov parameters are the covariances $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{\infty}$ of $\mathbf{y}$. Conversely, deterministic LTI–SS systems whose Markov parameters are the covariances $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{\infty}$ yield LTI–SS representations of $\mathbf{y}$. Recall that we call the process

$$\mathbf{e}(t) := \mathbf{y}(t) - E_l[\mathbf{y}(t)|\mathcal{H}_{t-}^{\mathbf{y}}], \quad t \in \mathbb{Z}$$

the innovation process of $\mathbf{y}$. Assume now that $(A, G^T, C, \Lambda_0^{\mathbf{y}})$ is a stable minimal deterministic LTI–SS system whose Markov parameters are the covariances of $\mathbf{y}$, i.e., (1.3) holds. Call a matrix M minimal symmetric solution of a matrix equation if for any other symmetric solution $\tilde{M}$ the matrix $\tilde{M} - M$ is positive definite. Let $\Sigma_{\mathbf{x}}$ be the minimal symmetric solution of the algebraic Riccati equation

$$\Sigma = A\Sigma A^T + (G^T - A\Sigma C^T)(\Delta(\Sigma))^{-1}(G^T - A\Sigma C^T)^T, \tag{1.4}$$

where $\Delta(\Sigma) = (\Lambda_0^{\mathbf{y}} - C\Sigma C^T)$ and set K as

$$K := (G^T - A\Sigma_{\mathbf{x}} C^T)(\Delta(\Sigma_{\mathbf{x}})^{-1}). \tag{1.5}$$

Then, we know the following about the tuple $(A, K, C, I, \mathbf{e})$.

Proposition 1.5. *(Katayama, 2005, Section 7.7) Let K be as in (1.5) and $\mathbf{e}$ be the innovation process of $\mathbf{y}$. Then the tuple*

$$(A, K, C, I, \mathbf{e}) \tag{1.6}$$

is a minimal LTI–SS representation of $\mathbf{y}$.

If $\mathbf{x}$ is the state of $(A, K, C, I, \mathbf{e})$, then the minimal symmetric solution of (1.4) is $\Sigma_{\mathbf{x}} = E[\mathbf{x}(t)\mathbf{x}^T(t)]$ and $\Delta(\Sigma_{\mathbf{x}}) = E[\mathbf{e}(t)\mathbf{e}^T(t)]$. Furthermore, $K = E[\mathbf{x}(t+1)\mathbf{e}^T(t)]E[\mathbf{e}(t)\mathbf{e}^T(t)]^{-1}$ in (1.5) is the gain of the steady-state Kalman filter (Lindquist and Picci, 2015, Section 6.9). This motivates the following definition:

Definition 1.6. Let $\mathbf{e}, \mathbf{y} \in \mathbb{R}^p$ be ZMSIR processes and $A \in \mathbb{R}^{n\times n}, K \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{p\times n}, D \in \mathbb{R}^{p\times p}$. An LTI–SS representation $(A, K, C, D, \mathbf{e}, \mathbf{y})$ is called *Kalman representation* if $\mathbf{e}$ is the innovation process of $\mathbf{y}$ and $D = I_p$.

A Kalman representation with output process $\mathbf{y}$ is called *Kalman representation of* $\mathbf{y}$. A Kalman representation is minimal, called *minimal Kalman representation*, if it is a minimal LTI–SS representation. The representation in Proposition 1.5 is a minimal Kalman representation, thus from the discussion above we can conclude that

Proposition 1.7. *Every ZMSIR process $\mathbf{y}$ has a minimal Kalman representation.*

Notice that Proposition 1.7 trivially implies that every ZMSIR process has a minimal LTI–SS representation.

An important feature of Kalman representations is that they can be calculated from the covariance sequence of the output process, see Algorithm 1 below. As a consequence, Kalman representations of a process can be calculated from any LTI–SS representation of that process, see Algorithm 2 below.

In Chapters 2–4, we deal with the so-called coercive property of ZMSIR processes, see (Lindquist and Picci, 2015, Definition 9.4.1). In terms of Kalman representations, coercivity of a process $\mathbf{y}$ is equivalent to the invertibility of any Kalman representation $(A, K, C, I, \mathbf{e})$ of $\mathbf{y}$, i.e., with the existence of the inverse matrix $(A - KC)^{-1}$, see (Lindquist and Picci, 2015, Theorem 9.4.2). From this, it is easy to see that if $\mathbf{y}$ is coercive, i.e., $(A - KC)^{-1}$ exists, then the innovation process $\mathbf{e}$ can be expressed as below.

$$\mathbf{e}(t) = \mathbf{y}(t) - \sum_{k=0}^{\infty} C(A - KC)^k K\mathbf{y}(t - k - 1).$$

In view of the foregoing, we present Algorithms 1 and 2.

Algorithm 1 Minimal Kalman representation based on output covariances

Input $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{2N}$: Covariance sequence of $\mathbf{y}$
Output $\{A, K, C, \Lambda_0^{\mathbf{e}}\}$: System matrices of (1.6) and variance of the innovation process of $\mathbf{y}$

Step 1 Define the Hankel and the shifted Hankel matrices

$$H_0 = \begin{bmatrix} \Lambda_1^{\mathbf{y}} & \Lambda_2^{\mathbf{y}} & \cdots & \Lambda_N^{\mathbf{y}} \\ \Lambda_2^{\mathbf{y}} & \Lambda_3^{\mathbf{y}} & \cdots & \Lambda_{N+1}^{\mathbf{y}} \\ \vdots & \vdots & & \vdots \\ \Lambda_N^{\mathbf{y}} & \Lambda_{N+1}^{\mathbf{y}} & \cdots & \Lambda_{2N-1}^{\mathbf{y}} \end{bmatrix} \quad H_1 = \begin{bmatrix} \Lambda_2^{\mathbf{y}} & \Lambda_3^{\mathbf{y}} & \cdots & \Lambda_{N+1}^{\mathbf{y}} \\ \Lambda_3^{\mathbf{y}} & \Lambda_4^{\mathbf{y}} & \cdots & \Lambda_{N+2}^{\mathbf{y}} \\ \vdots & \vdots & & \vdots \\ \Lambda_{N+1}^{\mathbf{y}} & \Lambda_{N+2}^{\mathbf{y}} & \cdots & \Lambda_{2N}^{\mathbf{y}} \end{bmatrix}.$$

Step 2 Calculate the SVD of $H_0 = USV^T$.
Step 3 Let m be such that $\Lambda_0^{\mathbf{y}} \in \mathbb{R}^{m \times m}$ and denote the first m rows of a matrix by $(.)_{1:m}$. Define

$$A = S^{-1/2}U^T H_1 V S^{-1/2}$$
$$C = (US^{1/2})_{1:m} \qquad G = (VS^{1/2})_{1:m}$$

Step 4 Find the minimal symmetric solution $\Sigma_{\mathbf{x}}$ of the Riccati equation (1.4) (see *e.g.*, (Katayama, 2005, Section 7.4.2)).
Step 5 Set K as in (1.5) and define $\Lambda_0^{\mathbf{e}} = \Lambda_0^{\mathbf{y}} - C\Sigma_{\mathbf{x}}C^T$.

Note that Steps 1–3 of Algorithm 1 calculate a minimal deterministic LTI–SS system (A, G^T, C, Λ_0) such that (1.3) holds using the classical Kalman-Ho realization algorithm.

Remark 1.8 (Correctness of Algorithms 1–2)**.** Consider a ZMSIR process $\mathbf{y}$ with co-

Algorithm 2 Minimal Kalman representation based on LTI–SS representation

Input $\{\bar{A}, \bar{B}, \bar{C}, \bar{D}, \Lambda_0^{\mathbf{v}}\}$: System matrices of an LTI–SS representation $(\bar{A}, \bar{B}, \bar{C}, \bar{D}, \mathbf{v})$ of $\mathbf{y}$ and variance of $\mathbf{v}(t)$

Output $\{A, K, C, \Lambda_0^{\mathbf{e}}\}$: System matrices of (1.6) and variance of the innovation process of $\mathbf{y}$

Step 1 Find the solution $\Sigma_{\mathbf{x}}$ of the Lyapunov equation $\Sigma = \bar{A}\Sigma\bar{A}^T + \bar{B}\Lambda_0^{\mathbf{v}}\bar{B}^T$.
Step 2 Define $G : \bar{C}\Sigma_{\mathbf{x}}\bar{A}^T + \bar{D}\Lambda_0^{\mathbf{v}}\bar{B}^T$ and calculate the output covariance matrices $\Lambda_k^{\mathbf{y}} = \bar{C}\bar{A}^{k-1}G^T$ for $k = 0, \ldots, 2n$, where n is such that $\bar{A} \in \mathbb{R}^{n\times n}$.
Step 3 Apply Algorithm 1 with input $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{2n}$ and denote the output by $\{A, K, C, \Lambda_0^{\mathbf{e}}\}$.

variance sequence $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{\infty}$ and an LTI–SS representation $(\bar{A}, \bar{B}, \bar{C}, \bar{D}, \mathbf{v})$ of $\mathbf{y}$. Let $\mathbf{e}$ be the innovation process of $\mathbf{y}$ and N be larger than or equal to the dimension of a minimal LTI–SS representation of $\mathbf{y}$. Then it follows from (Katayama, 2005, Lemma 7.9, Section 7.7) that if $\{A, K, C, \Lambda_0^{\mathbf{e}}\}$ is the output of Algorithm 1 with input $\{\Lambda_k^{\mathbf{y}}\}_{k=0}^{2N}$, then $(A, K, C, I, \mathbf{e})$ is a minimal Kalman representation of $\mathbf{y}$ and $\Lambda_0^{\mathbf{e}} = E[\mathbf{e}(t)\mathbf{e}^T(t)]$. Likewise, if $\{A, K, C, \Lambda_0^{\mathbf{e}}\}$ is the output of Algorithm 2 with input $\{\bar{A}, \bar{B}, \bar{C}, \bar{D}, E[\mathbf{v}(t)\mathbf{v}^T(t)]\}$, then $(A, K, C, I, \mathbf{e})$ is a minimal Kalman representation of $\mathbf{y}$ and $\Lambda_0^{\mathbf{e}} = E[\mathbf{e}(t)\mathbf{e}^T(t)]$.

Remark 1.9. Algorithms 1 and 2 involve matrix multiplication, inversion, calculating SVD and solving Riccati and Lyapunov equations. The computational complexity of all involved matrix operations is polynomial in the sizes of the matrices (Golub and Van Loan, 2013). Also, solving Riccati and Lyapunov equations is polynomial in the size of the solution matrix (Bini et al., 2011). For Algorithm 1, the sizes of the matrices involved are polynomial in the number $2N + 1$ and the size $p = \dim(\mathbf{y})$ of the output covariances, hence its complexity is polynomial in N and p. By similar reasoning, Algorithm 2 has polynomial complexity in the dimensions of the state, output, and noise processes of the input LTI–SS representation $(\bar{A}, \bar{B}, \bar{C}, \bar{D}, \mathbf{v})$.

The algorithms in Chapters 2–4 are based on Algorithms 1–2 and under certain conditions, they also calculate minimal Kalman representations. Minimal Kalman representations have the following useful properties:

Proposition 1.10. *A Kalman representation $(A, K, C, I, \mathbf{e}, \mathbf{y})$ is minimal if and only if (A, K) is controllable and (A, C) is observable.*

Proposition 1.10 provides a characterization of minimality of a Kalman representation $(A, K, C, I, \mathbf{e}, \mathbf{y})$ by minimality of the deterministic system (A, K, C, I). In general, the characterization of minimality in LTI–SS representations is more

involved, and it is related to the minimality of the deterministic LTI–SS system $(A, G^T, C, \Lambda_0^{\mathbf{y}})$ associated with the stochastic LTI–SS representation (see (Lindquist and Picci, 2015, Corollary 6.5.5)). The next proposition shows that minimal Kalman representations are *isomorphic* in the sense defined below:

Definition 1.11 (isomorphism)**.** Consider two Kalman representations $(A, K, C, I, \mathbf{e})$ and $(\tilde{A}, \tilde{K}, \tilde{C}, I, \mathbf{e})$ of a process $\mathbf{y}$. Then they are isomorphic if there exists a non-singular matrix T such that $A = T\tilde{A}T^{-1}$, $K = T\tilde{K}$ and $\tilde{C} = CT^{-1}$.

Proposition 1.12. *(Lindquist and Picci, 2015, Theorem 6.6.1) Any two minimal Kalman representations of a process* $\mathbf{y}$ *are isomorphic.*

Again, in general, the result does not apply for any two LTI–SS representations of $\mathbf{y}$. The statement and its proof can be found in (Lindquist and Picci, 2015, Theorem 6.6.1, Section 6.6) with the modification that here the noise process is not normalized.

1.3 GB–SS representations

This section provides background material for Chapter 6 on general bilinear state-space (GB-SS) representations. We adopt the terminology of (Petreczky and René, 2017) and summarize some of its results. First, some basic notation and terminology are introduced. Then GB-SS representations are defined and a brief summary about realization theory of GB-SS representations is presented (for more details see (Petreczky and René, 2017)).

1.3.1 Introduction to GB–SS representations

To define general bilinear state-space representations, we first introduce the necessary terminology. For the rest of the chapter, we fix a finite set $\{1, 2, \ldots, d\}$, where d is a positive integer, and denote it by Σ.

Consider the discrete-time stochastic dynamical system

$$\begin{aligned} \mathbf{x}(t+1) &= \sum_{\sigma \in \Sigma} (A_\sigma \mathbf{x}(t) + K_\sigma \mathbf{v}(t)) \mathbf{u}_\sigma(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{v}(t), \end{aligned} \tag{1.7}$$

where the state $\mathbf{x}(t) \in \mathbb{R}^n$, noise $\mathbf{v}(t) \in \mathbb{R}^m$, output $\mathbf{y}(t) \in \mathbb{R}^p$, and input processes $\mathbf{u}_\sigma(t) \in \mathbb{R}$, $\sigma \in \Sigma$ are weakly stationary stochastic processes.

In order to be able to define generalized bilinear state-space (abbreviated by GB-SS) representations we need to impose further restrictions on systems of the form

(1.7). More precisely, we adapt GB–SS representations from (Petreczky and René, 2017) which are state-space representation of the form (1.7) that satisfy a number of additional conditions. Note that these conditions are necessary for realization theory of representation of the form (1.7). The following notation and terminology help us to define these conditions.

Let Σ^+ be the set of all finite sequences of elements of Σ, i.e., a typical element of Σ^+ is a sequence of the form $w = \sigma_1 \cdots \sigma_k$, where $\sigma_1, \ldots, \sigma_k \in \Sigma$. We define the concatenation operation on Σ^+ in the standard way: if $w = \sigma_1 \cdots \sigma_k$ and $v = \hat{\sigma}_1 \cdots \hat{\sigma}_l$ where $\sigma_1, \ldots, \sigma_k, \hat{\sigma}_1, ..., \hat{\sigma}_l \in \Sigma$ then the concatenation of w and v, denoted by wv, is defined as the sequence $wv = \sigma_1 \cdots \sigma_k \hat{\sigma}_1 \cdots \hat{\sigma}_l$. In the sequel, it will be convenient to extend Σ^+ by adding a formal unit element $\epsilon \notin \Sigma^+$. We denote this set by $\Sigma^* := \Sigma^+ \cup \{\epsilon\}$. The concatenation operation can be extended to Σ^* as follows: $\epsilon\epsilon = \epsilon$, and for any $w \in \Sigma^+$, $\epsilon w = w\epsilon = w$. Let $w = \sigma_1 \cdots \sigma_k \in \Sigma^+$ and $\sigma \in \Sigma$. Then the length of w is defined by $|w| := k$ and the length of ϵ is defined by $|\epsilon| := 0$. Consider a set of matrices $\{M_\sigma\}_{\sigma \in \Sigma}$ where $M_\sigma \in \mathbb{R}^{n \times n}$, $n \geqslant 1$ for all $\sigma \in \Sigma$ and let $w = \sigma_1 \cdots \sigma_k \in \Sigma^+$, where $\sigma_1, \ldots, \sigma_k \in \Sigma$. Then, we denote the matrix $M_{\sigma_k} \cdots M_{\sigma_1}$ by M_w and we define $M_\epsilon := I$. In addition, for a set of processes $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$ and for $w = \sigma_1 \cdots \sigma_k \in \Sigma^+$, where $\sigma_1, \ldots, \sigma_k \in \Sigma$ we denote the process $\mathbf{u}_{\sigma_k}(t) \cdots \mathbf{u}_{\sigma_1}(t - |w| + 1)$ by $\mathbf{u}_w(t)$ and define $\mathbf{u}_\epsilon(t) :\equiv 1$. In a dynamical system (1.7), the past of the noise, state and output processes that are multiplied by the past of the input processes play an important role in defining GB-SS representations and adapting analytical tools from linear system theory to the study of GB-SS representations. For this reason we define the following processes:

Definition 1.13. Consider a process $\mathbf{r}$ and a set of processes $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$. Let $\sigma \in \Sigma$ and $w = \sigma_1 \cdots \sigma_k \in \Sigma^+$, where $\sigma_1, \ldots, \sigma_k \in \Sigma^*$. Then, we define the process

$$\mathbf{z}^{\mathbf{r}}_w(t) := \mathbf{r}(t - |w|)\mathbf{u}_w(t - 1),$$

which we call the *past* of $\mathbf{r}$ with respect to $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$.

Definition 1.14. Consider a process $\mathbf{r}$ and a set of processes $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$. Let $\sigma \in \Sigma$ and $w = \sigma_1 \cdots \sigma_k \in \Sigma^+$, where $\sigma_1, \ldots, \sigma_k \in \Sigma^*$. Then, we define the process

$$\mathbf{z}^{\mathbf{r}+}_w(t) := \mathbf{r}(t + |w|)\mathbf{u}_w(t + |w| - 1),$$

which we call the *future* of $\mathbf{r}$ with respect to $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$.

Notice that for $w = \epsilon$, both the past $\mathbf{z}^{\mathbf{r}}_\epsilon(t)$ and the future $\mathbf{z}^{\mathbf{r}+}_\epsilon(t)$ of $\mathbf{r}$ with respect to $\{\mathbf{u}_\sigma\}_{\sigma \in \Sigma}$ equal $\mathbf{r}(t)$.

The processes in Definitions 1.13 and 1.14 slightly differ from the parallel past and future processes of a process used in (Petreczky and René, 2017). Note that

we can obtain the processes in Definitions 1.13 and 1.14 by multiplying the parallel processes used in (Petreczky and René, 2017) with a scalar, see e.g., equation (6) in (Petreczky and René, 2017).

In what follows we define classes of processes that serve as a basis in formulating constraints on system (1.7). We begin with defining *admissible sets of processes*. For this, first recall that all the random variables and stochastic processes are understood with respect to a probability space $(\mathbb{R}, \mathcal{F}, P)$. Using the standard notation, the conditional expectation of a random variable $\mathbf{z}$ to a σ-algebra $\mathcal{F}^*$ is denoted by $E[\mathbf{z}|\mathcal{F}^*]$. Furthermore, considering a process $\mathbf{z}$ and a time t, the σ-algebra generated by the random variables in the past is denoted by $\mathcal{F}^{\mathbf{z}}_{t-} = \sigma\left(\{\mathbf{z}(k)\}_{k=-\infty}^{t-1}\right)$.

The definition of admissible sets of processes below will help us in formulating conditions on the input processes of (1.7). As we will see, the set of input processes of a GB–SS representation forms an admissible set of processes.

Definition 1.15 (admissible set of processes)**.** A set of processes $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ is called *admissible* if

- $[\mathbf{u}_v^T, \mathbf{u}_w^T]^T$ is weakly stationary for all $v, w \in \Sigma^*$
- there exist real numbers $\{\alpha_\sigma\}_{\sigma\in\Sigma}$ such that for all $t \in \mathbb{Z}$: $\quad \sum_{\sigma\in\Sigma} \alpha_\sigma \mathbf{u}_\sigma(t) \equiv 1$
- there exist (strictly) positive numbers $\{p_\sigma\}_{\sigma\in\Sigma}$ such that for any $\sigma_1, \sigma_2 \in \Sigma$ and $v_1, v_2 \in \Sigma^*$, where $v_1 v_2 \in \Sigma^+$ the following holds:

$$E[\mathbf{u}_{v_1\sigma_1}(t)\mathbf{u}_{v_2\sigma_2}(t)|\bigvee_{\sigma\in\Sigma}\mathcal{F}^{\mathbf{u}_\sigma}_{t-}] = \begin{cases} p_{\sigma_1}\mathbf{u}_{v_1}(t-1)\mathbf{u}_{v_2}(t-1) & \sigma_1 = \sigma_2 \\ 0 & \sigma_1 \neq \sigma_2 \end{cases}$$

  where $\bigvee_{\sigma\in\Sigma}\mathcal{F}^{\mathbf{u}_\sigma}_{t-}$ denotes the smallest sigma algebra containing all $\mathcal{F}^{\mathbf{u}_\sigma}_{t-}$ for $\sigma \in \Sigma$.

Next, we define the class of processes to which the noise, state and output processes of a GB–SS representation belong. The definition involves the concept of conditionally independent σ-algebras: Recall that two σ-algebras $\mathcal{F}_1$, $\mathcal{F}_2$ are conditionally independent with respect to a third one $\mathcal{F}_3$, if for every event $A_1 \in \mathcal{F}_1$ and $A_2 \in \mathcal{F}_2$ the following holds: $P(A_1 \cap A_2|\mathcal{F}_3) = P(A_1|\mathcal{F}_3)P(A_2|\mathcal{F}_3)$ with probability one (Pearl, 2000). Furthermore, besides the σ-algebra generated by the random variables in the past of $\mathbf{z}$ with respect to a time t, we will work with σ-algebras generated by the random variables in the present and future, denoted by $\mathcal{F}^{\mathbf{z}}_t = \sigma\left(\mathbf{z}(t)\right)$ and $\mathcal{F}^{\mathbf{z}}_{t+} = \sigma\left(\{\mathbf{z}(k)\}_{k=t}^{\infty}\right)$, respectively.

Definition 1.16 (ZMWSSI process)**.** A stochastic process $\mathbf{r}$ is called *zero-mean weakly stationary* with respect to an admissible set of processes $\{\mathbf{u}\}_{\sigma\in\Sigma}$ (ZMWSSI) if

- $\mathcal{F}^{\mathbf{r}}_{(t+1)-}$ and $\mathcal{F}^{\mathbf{u}}_{t+}$ are conditionally independent with respect to $\mathcal{F}^{\mathbf{u}}_{t-}$

- $[\mathbf{r}^T, (\mathbf{z}_v^{\mathbf{r}})^T, (\mathbf{z}_w^{\mathbf{r}})^T]^T$ is zero-mean weakly stationary for all $v, w \in \Sigma^+$.

Note that ZMWSSI abbreviates zero-mean weakly stationary with respect to input. This is because ZMWSSI will describe processes of GB–SS representations where the admissible set of processes will be the set of input processes.

We are now ready to define GB–SS representations:

Definition 1.17 (GB–SS representation)**.** A system of the form (1.7) is called *generalized bilinear state-space* (GB–SS) representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ if the following holds:

- $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ is admissible
- $[\mathbf{x}^T, \mathbf{v}^T]$ is ZMWSSI w.r.t. $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$
- $E[\mathbf{z}_w^{\mathbf{v}}(t)\mathbf{v}^T(t)] = 0$ and $E[\mathbf{z}_w^{\mathbf{x}}(t)\mathbf{v}^T(t)] = 0$ for all $w \in \Sigma^+$
- $E[\mathbf{z}_{\hat{\sigma}}^{\mathbf{x}}(t)\,(\mathbf{z}_\sigma^{\mathbf{v}}(t))^T] = 0$ for all $\hat{\sigma}, \sigma \in \Sigma$
- $\sum_{\sigma\in\Sigma} p_\sigma A_\sigma \otimes A_\sigma$ is stable.

We refer to a GB–SS representation (1.7) as GB–SS representation $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v}, \{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ or as GB–SS representation $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v})$ of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$.

Notice that because $[\mathbf{x}^T, \mathbf{v}^T]$ is ZMWSSI w.r.t. $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ and the output process $\mathbf{y}$ is the linear combination of $\mathbf{x}$ and $\mathbf{v}$, the output process $\mathbf{y}$ is also ZMWSSI w.r.t. $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$. Depending on the choice of the input, the behaviour of a GB–SS representation can significantly vary. The constraint on the input of a GB–SS representation, formulated in Definition 1.15, gives scope to choosing $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ for example in the following ways:

- If $\Sigma = 1$ and $\mathbf{u}_1(t) \equiv 1$, then $\mathbf{u}_1$ is admissible and the GB–SS representation defines an autonomous LTI–SS representation.
- If $\mathbf{u}_\sigma(t)$ is zero-mean, square-integrable, independent and identically distributed (iid) process for all $\sigma \in \Sigma$ and $\mathbf{u}_{\sigma_1}(t)$, $\mathbf{u}_{\sigma_2}(t)$ are independent for all $\sigma_1, \sigma_2 \in \Sigma, \sigma_1 \neq \sigma_2$, then $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ is admissible.
- If $\mathbf{u}_\sigma(t) = \chi(\Theta(t) = \sigma)$ where Θ is an iid process taking values in Σ, then $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ is admissible.

More examples for inputs of GB–SS representations can be found in (Petreczky and René, 2017). Note that Definition 1.15 gives a more strict definition of admissible set of processes than (Petreczky and René, 2017, Definition 1). More specifically, the set of admissible words used in (Petreczky and René, 2017) is the trivial Σ^+ set. The results of Chapter 6 remain valid with the definition of admissible set of processes in (Petreczky and René, 2017), however, we use Definition 1.15 in order to avoid technicalities.

1.3.2 Realization theory of GB–SS representations

In this section, we recall from (Petreczky and René, 2017) a number of results from realization theory of GB-SS representations. First, we introduce a specific GB–SS representation, called innovation GB–SS representation, which is followed by a summary of realization theory of GB–SS representations. In particular, we present a realization algorithm, which calculates innovation GB–SS representations from covariances of the input and output processes. Innovation GB-SS representations and the latter realization algorithm of GB-SS representations will play a crucial role in deriving the main results of Chapter 6.

Before defining innovation GB–SS representations, we recall some basic notations that rely on the theory of Hilbert spaces of zero-mean square-integrable random variables.

In Section 1.1 we have seen the set of zero-mean square-integrable random variables taking values in $\mathbb{R}$ forms a Hilbert-space $\mathcal{H}$. Let $\mathbf{r}$ be a ZMWSSI process with respect to a set of admissible processes $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$. Then the one-dimensional components of the random variables $\mathbf{r}(t)$ and $\mathbf{z}^{\mathbf{r}}_w(t)$, (the past of $\mathbf{r}$ with respect to $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$, see Definition 1.13) belong to $\mathcal{H}$ for all $t \in \mathbb{Z}$. Recall that the Hilbert space generated by the one-dimensional components of $\mathbf{r}(t)$ is denoted by $\mathcal{H}^{\mathbf{z}}_t$. Likewise, we denote the Hilbert space generated by the one-dimensional components of $\{\mathbf{z}^{\mathbf{r}}_w(t)\}_{w\in\Sigma^+}$ by $\mathcal{H}^{\mathbf{z}^{\mathbf{r}}_w}_{t,w\in\Sigma^+}$. Note that $\mathcal{H}^{\mathbf{z}^{\mathbf{r}}_w}_{t,w\in\Sigma^+}$ is the closed sum of the Hilbert spaces generated by the one-dimensional components of $\{\mathbf{z}^{\mathbf{r}}_w(t)\}_{w\in\Sigma^+}$.

Below, we define what we mean by innovation GB–SS representation. Informally, an innovation GB–SS representation has a specific noise process, which we call GB–innovation process. First, we define GB-innovation processes:

Definition 1.18 (GB–innovation process)**.** The *GB–innovation process* of a ZMWSSI process $\mathbf{y}$ with respect to the set of admissible processes $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ is defined by

$$\mathbf{e}(t) := \mathbf{y}(t) - E_l[\mathbf{y}(t)|\mathcal{H}^{\mathbf{z}^{\mathbf{y}}_w}_{t,w\in\Sigma^+}].$$

The class of innovation GB-SS representations is then defined as follows.

Definition 1.19 (innovation GB–SS representation)**.** A GB–SS representation (1.7) is called *innovation GB–SS representation* if the noise process $\mathbf{v}$ is the GB–innovation process $\mathbf{e}(t) = \mathbf{y}(t) - E_l[\mathbf{y}(t)|\mathcal{H}^{\mathbf{z}^{\mathbf{y}}_w}_{t,w\in\Sigma^+}]$ of $\mathbf{y}$ with respect to the input $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ and D is the identity matrix.

Recall that in the specific case when $\Sigma = \{1\}$ and $\mathbf{u}_1(t) \equiv 1$, GB–SS representations define LTI–SS representations. For this case, the GB–innovation process w.r.t. $\mathbf{u}_1$ is the innovation process $\mathbf{e}(t) = \mathbf{y}(t) - E_l[\mathbf{y}(t)|\mathcal{H}^{\mathbf{y}}_{t-}]$, where $\mathcal{H}^{\mathbf{y}}_{t-}$ is the Hilbert

space generated by the elements of the past $\{\mathbf{y}(t-k)\}_{k=1}^{\infty}$ of $\mathbf{y}$. From this, it is easy to see that when $\Sigma = \{1\}$ and $\mathbf{u}_1(t) \equiv 1$, then innovation GB–SS representations define innovation LTI–SS representations (called Kalman representation in (Jozsa et al., 2018b)).

Among innovation GB–SS representations we are particularly interested in minimal ones. As for LTI–SS representations in the previous chapters, we define the *dimension* of a GB–SS representation as the dimension of the space from where the state process takes its values. Then, a GB–SS representation $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ is called *minimal* if it has minimal dimension among all GB–SS representations of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$.

Innovation GB–SS representations, in particular minimal ones, have several advantageous properties. For instance, if there exists a GB–SS representation of the input-output processes $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ then there also exists a minimal innovation GB–SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$, see (Petreczky and René, 2017, Theorem 2). In fact, this minimal innovation GB–SS representation can be calculated algorithmically from $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$. This algorithm is called realization algorithm and we will present it later on in this section.

Recall that we refer to a GB–SS representation (1.7) as GB–SS representation $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v}, \{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ or as GB–SS representation $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v})$ of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$. Note that in the notations above, we do not mention the state process. However, the state process of a GB–SS representation (1.7) is uniquely determined by the noise process and the system matrices. Indeed, for a given $w \in \Sigma^+$, the state process $\mathbf{x}$ can be expressed at time t as below (see (Petreczky and René, 2017, Lemma 1)).

$$\mathbf{x}(t) = \sum_{N=0}^{\infty} \sum_{|w|\leqslant N, \sigma\in\Sigma} A_w K_\sigma \mathbf{z}^{\mathbf{v}}_{\sigma w}(t) \tag{1.8}$$

Next, we present the realization algorithm of GB-SS representations. To this end, we need to define Hankel matrices in GB–SS representations which requires the introduction of a (complete) *lexicographic ordering* $(\prec)$ *on* Σ^*: $v \prec w$ if either $|v| < |w|$ or if $v = \nu_1 \dots \nu_k$, $w = \sigma_1 \dots \sigma_k$ then $\exists\, l \in \{1, \dots, k\}$ such that $\nu_i = \sigma_i$, $i < l$ and $\nu_l < \sigma_l$. Let the ordered elements of Σ^* be $v_1 = \epsilon, v_2 = \sigma_1, \dots$ and define $M(j) = \frac{d^{j+1}-1}{d-1}$ for $d \geqslant 2$ and $M(j) = j+1$ for $d = 1$ as the largest index such that $|v_{M(j)}| \leqslant j$. Now, consider a GB–SS representation (1.7) and denote the covariances between $\mathbf{y}$ and its past $\mathbf{z}^{\mathbf{y}}_w$ w.r.t. the input (see Definition 1.13) by $\Lambda^{\mathbf{y}}_w := E[\mathbf{y}(t)(\mathbf{z}^{\mathbf{y}}_w(t))^T]$. Then the matrices that form the block matrices of the Hankel matrix are given by

$$\Psi^{\mathbf{y}}_w := [\Lambda^{\mathbf{y}}_{1w}, \dots, \Lambda^{\mathbf{y}}_{dw}].$$

Note that the covariances $\Lambda^{\mathbf{y}}_w$, thus also $\Psi^{\mathbf{y}}_w$, can be expressed via the system matrices of a GB-SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ as follows: Let $P_\sigma = E[\mathbf{x}(t)\mathbf{x}^T(t)\mathbf{u}^2_\sigma(t)]$ be the unique symmetric positive semi-definite solution of

$$P_\sigma = p_\sigma\Big(\sum_{\sigma_1\in\Sigma} A_{\sigma_1}P_{\sigma_1}A^T_{\sigma_1} + K_{\sigma_1}Q_{\sigma_1}K^T_{\sigma_1}\Big)$$

and $Q_{\sigma_1} = E[\mathbf{v}(t)\mathbf{v}^T(t)\mathbf{u}^2_\sigma(t)]$. Define $p_\epsilon = 1$ and note that $v_1 = \epsilon$. Then the following holds for all $v \in \Sigma^*$, $\sigma \in \Sigma$

$$\Lambda^{\mathbf{y}}_{\sigma v} = p_v C A_v G_\sigma, \quad \Psi^{\mathbf{y}}_v = p_v C A_v G$$

where $G_\sigma = A_\sigma P_\sigma C^T + K_\sigma Q_\sigma$ and $G = \begin{bmatrix} G_1 & G_2 & \cdots & G_d \end{bmatrix}$.

By using $\Psi^{\mathbf{y}}_w$, we define the (finite) Hankel matrix of $\mathbf{y}$ with respect to $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$ indexed by i, j as follows

$$H^{\mathbf{y}}_{i,j} := \begin{bmatrix} \Psi^{\mathbf{y}}_{v_1v_1} & \Psi^{\mathbf{y}}_{v_2v_1} & \cdots & \Psi^{\mathbf{y}}_{v_{M(j)}v_1} \\ \Psi^{\mathbf{y}}_{v_1v_2} & \Psi^{\mathbf{y}}_{v_2v_2} & \cdots & \Psi^{\mathbf{y}}_{v_{M(j)}v_2} \\ \vdots & \vdots & & \vdots \\ \Psi^{\mathbf{y}}_{v_1v_{M(i)}} & \Psi^{\mathbf{y}}_{v_2v_{M(i)}} & \cdots & \Psi^{\mathbf{y}}_{v_{M(j)}v_{M(i)}} \end{bmatrix}. \tag{1.9}$$

For presenting the realization algorithm, we also need to introduce observability matrices. The observability matrix $\mathscr{O}_k$ of $(\{A_\sigma\}_{\sigma\in\Sigma}, C)$ up to k is defined by

$$\mathscr{O}_k := \begin{bmatrix}(CA_{v_1})^T & (CA_{v_2})^T & \cdots & (CA_{v_k})^T\end{bmatrix}^T\Big].$$

Finally, we will make the technical assumption that the output processes of GB–SS representations are full rank. The full rank property is defined as follows:

Definition 1.20. An output process $\mathbf{y}$ of a GB–SS representation is called *full rank* if for all $\sigma \in \Sigma$ and $t \in \mathbb{Z}$, the matrix $E[\mathbf{e}(t)\mathbf{e}^T(t)\mathbf{u}^2_\sigma(t)]$ is strictly positive definite where $\mathbf{e}$ is the GB–innovation process of $\mathbf{y}$ w.r.t the input $\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}$.

Definition 1.20 is equivalent to saying that the random variable $\mathbf{z}^{\mathbf{e}}_\sigma(t)$ has positive definite variance matrix for all $\sigma \in \Sigma$ and $t \in \mathbb{Z}$. The next assumption will be in force for the rest of the thesis:

Assumption 1.21. *The output process $\mathbf{y}$ of a GB–SS representation (1.7) is full rank.*

Next, we present Algorithm 3, a realization algorithm of GB–SS representations that calculates an innovation GB–SS representation from the covariances of the input-output processes. Algorithm 3 is equivalent to (Petreczky and René, 2017,

Algorithm 2), however, there is a nuance between the two algorithms due to that in (Petreczky and René, 2017) there is a scalar factor of the processes $\mathbf{z}^{\mathbf{y}}_w(t)$ which factor also effects the formulas for the covariances $\Psi^{\mathbf{y}}_w$ and the Hankel matrix $H^{\mathbf{y}}_{i,j}$.

Algorithm 3 Minimal innovation GB–SS representation based on output covariances

Input $\{\Psi^{\mathbf{y}}_w\}_{\{w\in\Sigma^*,|w|\leqslant N\}}$ and $\{E[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^T]\}_{\sigma\in\Sigma}$: Covariance sequence of $\mathbf{y}$ and its past and variances of $\mathbf{z}^{\mathbf{y}}_\sigma$

Output $(\{A_\sigma, K_\sigma, Q_\sigma\}_{\sigma\in\Sigma}, C)$: System matrices of a minimal innovation GB–SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$

Step 1 Form the Hankel matrix $H^{\mathbf{y}}_{N-1,N}$ defined in (1.9)

Step 2 Decompose $H^{\mathbf{y}}_{N-1,N} = \mathbf{OR}$ such that $\mathbf{O} \in \mathbb{R}^{pM(N-1)\times n}$ and $\mathbf{R} \in \mathbb{R}^{n\times pdM(N)}$ have full column and row rank, respectively, and n is the rank of $H^{\mathbf{y}}_{N-1,N}$

Step 3 Take C as the first p rows of $\mathbf{O}$ and $R_{v_i} \in \mathbb{R}^{n\times pd}$ such that $\mathbf{R} = [R_{v_1}\cdots R_{v_{M(N)}}]$

Step 4 Take $G_i \in \mathbb{R}^{n\times p}$ such that $R_{v_1} = [G_1 \cdots G_d]$

Step 5 Let A_σ be the linear least square solution of

$$A_\sigma R_{v_i} = \frac{1}{p_\sigma} R_{v_i\sigma}, \quad i = 1, \ldots, M(N-1)$$

Step 6 Let $P_\sigma = \lim_{k\to\infty} P^k_\sigma$ where $P^0_\sigma = 0$ and P^{i+1}_σ, $i = 0, 1, \ldots$ is s.t.

$$\begin{aligned} Q^i_\sigma &= E[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^T] - CP^i_\sigma C^T \\ K^i_\sigma &= (G_\sigma - A_\sigma P^i_\sigma C^T)(Q^i{}_\sigma)^+ \\ P^{i+1}_\sigma &= p_\sigma\Big(\sum_{\sigma_1\in\Sigma} (A_{\sigma_1} P^i_{\sigma_1}) A^T_{\sigma_1} + K^i_{\sigma_1} Q^i_{\sigma_1} (K^i_{\sigma_1})^T\Big) \end{aligned}$$

Step 7 Let $K_\sigma = \lim_{i\to\infty} K^i_\sigma$ and $Q_\sigma = \lim_{i\to\infty} Q^i_\sigma$.

Assume that the processes $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ have an n-dimensional GB–SS representation. Then, the following statement holds for the output $(\{A_\sigma, K_\sigma, Q_\sigma\}_{\sigma\in\Sigma}, C)$ of Algorithm 3 with input $\{\Psi^{\mathbf{y}}_w\}_{\{w\in\Sigma^*,|w|\leqslant N\}}$ and $\{E[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^T]\}_{\sigma\in\Sigma}$ where $N \geqslant n$:

Lemma 1.22. *(Petreczky and René, 2017, Theorem 3) Denote the GB–innovation process of* $\mathbf{y}$ *by* $\mathbf{e}$. *Then* $E[\mathbf{z}^{\mathbf{e}}_\sigma(t)\,(\mathbf{z}^{\mathbf{e}}_\sigma(t))^T] = Q_\sigma$ *and the tuple* $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, I, \mathbf{e})$ *is a minimal innovation GB–SS representation of* $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$, *i.e.,*

$$\begin{aligned} \mathbf{x}(t+1) &= \sum_{\sigma\in\Sigma} (A_\sigma \mathbf{x}(t) + K_\sigma \mathbf{e}(t))\mathbf{u}_\sigma(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + \mathbf{e}(t), \end{aligned} \tag{1.10}$$

Furthermore, we also know the following formula on the state process, see the proof of (Petreczky and René, 2017, Lemma 18) or Lemma 6.9 in Appendix 6.A:

Lemma 1.23. *The state process of the GB–SS representation* (1.10) *is in the form of*

$$\mathbf{x}(t) = \begin{bmatrix} p_{v_1} I_p & 0 & \cdots & 0 \\ 0 & p_{v_2} I_p & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & p_{v_{M(n)}} I_p \end{bmatrix}^{-1} \mathscr{O}^+_{M(n)} E_l[Z^{\mathbf{y}}_n(t) | \mathcal{H}^{\mathbf{z}^{\mathbf{y}}_w}_{t,w\in\Sigma^+}], \tag{1.11}$$

where I_p is the $p \times p$ identity matrix and

$$Z^{\mathbf{y}}_n(t) = \left[(\mathbf{z}^{\mathbf{y}+}_{v_1}(t))^T \ \dots \ (\mathbf{z}^{\mathbf{y}+}_{v_{M(n)}}(t))^T \right]$$

is a vector of the future of $\mathbf{y}(t)$ with respect to the input (see Definition 1.14) and $\mathscr{O}^+_{M(n)}$ denotes the Moore-Penrose pseudo inverse of the observability matrix $\mathscr{O}_{M(n)}$ of $(\{A_\sigma\}_{\sigma\in\Sigma}, C)$ up to n.

Remark 1.24. The variance matrices $\{E[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^T]\}_{\sigma\in\Sigma}$ of the processes $\{\mathbf{z}^{\mathbf{y}}_\sigma\}_{\sigma\in\Sigma}$ can be expressed by the system matrices of a GB-SS representation $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v})$ of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ as

$$E[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^T] = CP_\sigma C^T + DQ_\sigma D^T.$$

Hence, the inputs of Algorithm 3 can be computed from any GB-SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$. As a result, Algorithm 3 can be modified to compute a minimal innovation GB-SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ from any GB–SS representation of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ in a similar manner as the realization algorithm Algorithm 1 of LTI–SS representations was modified as Algorithm 2. When applied to LTI–SS representation, Algorithm 3 is essentially equivalent to the well-known covariance realization algorithm (Lindquist and Picci, 2015).

The last result that we recall from realization theory of GB–SS representations helps us to relate Algorithm 3 to another realization algorithm in the Chapter 6 (see Remark 6.8). We first define isomorphism between GB–SS representations:

Definition 1.25 (isomorphism)**.** Consider two GB–SS representations $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v})$ with state process $\mathbf{x}$ and $(\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma\in\Sigma}, \hat{C}, \hat{D}, \mathbf{v})$ with state process $\hat{\mathbf{x}}$ of the same input-output processes. Then we call them *isomorphic* if there exists a non-singular matrix T such that

$$CT^{-1} = \hat{C}, \quad TA_\sigma T^{-1} = \hat{A}_\sigma, \quad TK_\sigma = \hat{K}_\sigma, \quad D = \hat{D}, \quad T\mathbf{x}(t) = \hat{\mathbf{x}}(t).$$

The next lemma claims that minimal innovation GB-SS representations are isomorphic.

Lemma 1.26. *(Petreczky and René, 2017, Theorem 2) Any two minimal innovation GB–SS representations of the processes* $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$ *are isomorphic.*

Note that in general, Lemma 1.26 does not apply for any two GB–SS representations of $(\{\mathbf{u}_\sigma\}_{\sigma\in\Sigma}, \mathbf{y})$.

1.4 Network graphs

In this thesis, the output processes of LTI–SS representations, transfer matrices and GB–SS representations are partitioned into components. Based on this partitioning, the latter dynamical systems are also decomposed into subsystems. Consider a process $\mathbf{y}$ partitioned into n components such that $\mathbf{y} = [\mathbf{y}_1^T, \dots, \mathbf{y}_n^T]^T$ where $\mathbf{y}_i \in \mathbb{R}^{r_i}$. Then, the dynamical system that represents $\mathbf{y}$ can be seen as the network of subsystems, where each subsystem generates a component $\mathbf{y}_i$ of $\mathbf{y}$. In what follows, we define the network of these subsystems in each of the dynamical systems mentioned above, with the help of a directed graph, called network graph.

1.4.1 Network graphs of LTI–SS representations

Below, we discuss what we mean by network graphs of LTI–SS representations. Consider an LTI–SS representation of $\mathbf{y}$ as follows

$$\begin{aligned}
\begin{bmatrix}\mathbf{x}_1(t+1)\\ \dots \\ \mathbf{x}_n(t+1)\end{bmatrix} &= \begin{bmatrix}A_{11} & \dots & A_{1n}\\ \vdots & \ddots & \vdots \\ A_{n1} & \dots & A_{nn}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1(t)\\ \dots \\ \mathbf{x}_n(t)\end{bmatrix} + \begin{bmatrix}B_{11} & \dots & B_{1n}\\ \vdots & \ddots & \vdots \\ B_{n1} & \dots & B_{nn}\end{bmatrix}\begin{bmatrix}\mathbf{v}_1(t)\\ \dots \\ \mathbf{v}_n(t)\end{bmatrix}\\
\begin{bmatrix}\mathbf{y}_1(t)\\ \dots \\ \mathbf{y}_n(t)\end{bmatrix} &= \begin{bmatrix}C_{11} & \dots & C_{1n}\\ \vdots & \ddots & \vdots \\ C_{n1} & \dots & C_{nn}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1(t)\\ \dots \\ \mathbf{x}_n(t)\end{bmatrix} + \begin{bmatrix}D_{11} & \dots & D_{1n}\\ \vdots & \ddots & \vdots \\ D_{n1} & \dots & D_{nn}\end{bmatrix}\begin{bmatrix}\mathbf{v}_1(t)\\ \dots \\ \mathbf{v}_n(t)\end{bmatrix},
\end{aligned} \tag{1.12}$$

where $A_{ij} \in \mathbb{R}^{p_i\times p_j}, B_{ij} \in \mathbb{R}^{p_i\times q_j}, C_{ij} \in \mathbb{R}^{r_i\times p_j}, D_{ij} \in \mathbb{R}^{r_i\times q_j}$ for some positive integers p_i, q_i and r_i. In (1.12), the sub-representation generating $\mathbf{y}_i$ for $i = 1, \dots, n$,

denoted by S_i, is given by

$$\begin{aligned} \mathbf{x}_i(t+1) &= \sum_{j|A_{ij} \text{ or } B_{ij}\neq 0} A_{ij}\mathbf{x}_j(t) + B_{ij}\mathbf{v}_j(t) \\ \mathbf{y}_i(t) &= \sum_{j|C_{ij} \text{ or } D_{ij}\neq 0} C_{ij}\mathbf{x}_j(t) + D_{ij}\mathbf{v}_j(t). \end{aligned} \tag{1.13}$$

Then, informally, we say that information flows from S_i to S_j, if $\mathbf{x}_i$ and $\mathbf{v}_i$ serve as an input for S_j. It is easy to see that the LTI–SS representation (1.12) is the network of the (non-autonomous) sub-representations (1.13).

To provide illustrative explanation of the results in Chapters 2–4, we will use the term *network graph of LTI–SS representations*. Network graphs are not used in the formulation of the results, they only serve to illustrate them. By network graph of the LTI–SS representation (1.12) we mean the directed graph $G = (V, E)$ where $V = \{1, \dots, n\}$, $E \subseteq V \times V$ and $(i, j) \in E$ if $\mathbf{x}_i$ and $\mathbf{v}_i$ serve as an input for S_j. It is determined by the zero-block matrices among the block matrices $\{A_{ij}, B_{ij}, C_{ij}, D_{ij}\}_{i,j=1}^n$. More precisely, for any two nodes $i, j \in V$ there is no edge from j to i, i.e., $(j, i) \notin E$, if and only if

$$A_{ij} = 0, B_{ij} = 0, C_{ij} = 0, D_{ij} = 0.$$

Notice that the network graph of the LTI–SS representation (1.12) depends on the numbers $\{p_i, q_i, r_i\}_{i=1}^n$.

In Chapters 2–4 network graphs of LTI–SS representations are one of the following graphs: the two node graph with one directed edge, star graphs with one root node and an arbitrary number of leaves and the transitive acyclic directed graphs. Notice that star graphs form a subclass among transitive acyclic directed graphs as well as the two node graph with one directed edge is a special case of star graphs.

1.4.2 Network graphs of transfer matrices

Next, we introduce network graphs of transfer matrices. Let $G(z) = C(sI - A)^{-1}B + D$ be a transfer matrix of an LTI–SS representation $(A, B, C, D, \mathbf{v}, \mathbf{y})$ (Anderson and Moore, 1979, Appendix C & D) such that

$$\begin{bmatrix} \mathbf{y}_1(t) \\ \mathbf{y}_2(t) \\ \vdots \\ \mathbf{y}_n(t) \end{bmatrix} = \begin{bmatrix} G_{11}(z) & G_{12}(z) & \dots & G_{1n}(z) \\ G_{21}(z) & G_{22}(z) & \dots & G_{2n}(z) \\ \vdots & \vdots & & \vdots \\ G_{n1}(z) & G_{n2}(z) & \dots & G_{nn}(z) \end{bmatrix} \begin{bmatrix} \mathbf{v}_1(t) \\ \mathbf{v}_2(t) \\ \vdots \\ \mathbf{v}_n(t) \end{bmatrix}, \tag{1.14}$$

where $G_{ij}(z) \in \mathbb{R}^{r_i \times q_j}$ is an $r_i \times q_j$ rational transfer matrix. Then the network graph of $G(z)$ is the directed graph $G = (V, E)$ where $V = \{1, 2, \ldots, n\}$ and $(i, j) \in E$ if and only if $G_{ji}(z) \neq 0$. Notice that the network graph of a transfer matrix depends on how we define the numbers $\{r_i, q_i\}_{i=1}^n$.

In Chapter 5, we study network graphs of transfer matrices that belong to the class of transitive acyclic directed graphs.

1.4.3 Network graphs of GB–SS representations

At last, the network graphs of GB–SS representations are introduced. Consider a GB-SS representation of $\mathbf{y}$ as follows

$$\begin{aligned}
\begin{bmatrix} \mathbf{x}_1(t+1) \\ \cdots \\ \mathbf{x}_n(t+1) \end{bmatrix} &= \sum_{\sigma \in \Sigma} \left(\begin{bmatrix} A_{\sigma,11} & \cdots & A_{\sigma,1n} \\ \vdots & \ddots & \vdots \\ A_{\sigma,n1} & \cdots & A_{\sigma,nn} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1(t) \\ \cdots \\ \mathbf{x}_n(t) \end{bmatrix} + \begin{bmatrix} B_{\sigma,11} & \cdots & B_{\sigma,1n} \\ \vdots & \ddots & \vdots \\ B_{\sigma,n1} & \cdots & B_{\sigma,nn} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1(t) \\ \cdots \\ \mathbf{v}_n(t) \end{bmatrix} \right) \mathbf{u}_\sigma(t) \\
\begin{bmatrix} \mathbf{y}_1(t) \\ \cdots \\ \mathbf{y}_n(t) \end{bmatrix} &= \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1(t) \\ \cdots \\ \mathbf{x}_n(t) \end{bmatrix} + \begin{bmatrix} D_{11} & \cdots & D_{1n} \\ \vdots & \ddots & \vdots \\ D_{n1} & \cdots & D_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1(t) \\ \cdots \\ \mathbf{v}_n(t) \end{bmatrix},
\end{aligned} \tag{1.15}$$

where for all $\sigma \in \Sigma$ $A_{\sigma,ij} \in \mathbb{R}^{p_i \times p_j}, B_{\sigma,ij} \in \mathbb{R}^{p_i \times q_j}, C_{ij} \in \mathbb{R}^{r_i \times p_j}, D_{ij} \in \mathbb{R}^{r_i \times q_j}$ for some positive integers $\sum_{i=1}^n p_i$, $\sum_{i=1}^n q_i$ and $\sum_{i=1}^n r_i$. In (1.15), the sub-representation generating $\mathbf{y}_i$ for $i = 1, \ldots, n$, denoted by S_i, is given by

$$\begin{aligned}
\mathbf{x}_i(t+1) &= \sum_{\sigma \in \Sigma} \left(\sum_{j | A_{\sigma,ij} \text{ or } B_{\sigma,ij} \neq 0} A_{\sigma,ij} \mathbf{x}_j(t) + B_{ij} \mathbf{v}_j(t) \right) \mathbf{u}_\sigma(t) \\
\mathbf{y}_i(t) &= \sum_{j | C_{ij} \text{ or } D_{ij} \neq 0} C_{ij} \mathbf{x}_j(t) + D_{ij} \mathbf{v}_j(t).
\end{aligned} \tag{1.16}$$

Informally, we say that information flows from S_i to S_j, if $\mathbf{x}_i$ and $\mathbf{v}_i$ appear in the equations of S_j. It is then easy to see that the GB–SS representation (1.15) is the network of the sub-representations (1.16).

To help the understanding of the results in Chapter 6, we will use the term *network graph of GB–SS representations*. Just as it was done for LTI–SS representations, this term is not used in the formulation of the results, however, it is a useful tool to illustrate the results. By network graph of the GB–SS representation (1.15) we mean the directed graph $G = (V, E)$ where $V = \{1, \ldots, n\}$, $E \subseteq V \times V$ and $(i, j) \in E$ if $\mathbf{x}_i$ and $\mathbf{v}_i$ appear in the equations of S_j. The network graph of a Kalman representation is determined by the zero-block matrices among the block matrices $\{A_{\sigma,ij}, B_{\sigma,ij}, C_{ij}, D_{ij}\}_{i,j=1}^n$. That is, for $i, j \in V$ $(j, i) \notin E$ if and only if

$A_{\sigma,ij} = 0, B_{\sigma,ij} = 0$ for all $\sigma \in \Sigma$ and $C_{ij} = 0, D_{ij=0}$. Notice that the network graph of the GB–SS representation (1.15) depends on the numbers $\{p_i, q_i, r_i\}_{i=1}^n$.