Bernoulli **24**(4A), 2018, 2693–2720
https://doi.org/10.3150/17-BEJ941

Max-linear models on directed acyclic graphs

NADINE GISSIBL[*] and CLAUDIA KLÜPPELBERG[**]

Center for Mathematical Sciences, Technical University of Munich, Boltzmannstrasse 3, 85748 Garching, Germany. E-mail: [*]n.gissibl@tum.de; [**]cklu@tum.de; url: http://www.statistics.tum.de

We consider a new recursive structural equation model where all variables can be written as max-linear function of their parental node variables and independent noise variables. The model is max-linear in terms of the noise variables, and its causal structure is represented by a directed acyclic graph. We detail the relation between the weights of the recursive structural equation model and the coefficients in its max-linear representation. In particular, we characterize all max-linear models which are generated by a recursive structural equation model, and show that its max-linear coefficient matrix is the solution of a fixed point equation. We also find the minimum directed acyclic graph representing the recursive structural equations of the variables. The model structure introduces a natural order between the node variables and the max-linear coefficients. This yields representations of the vector components, which are based on the minimum number of node and noise variables.

Keywords: directed acyclic graph; graphical model; max-linear model; minimal representation; path analysis; structural equation model

1. Introduction

Graphical models are a popular tool to analyze and visualize conditional independence relations between random variables (see e.g. Koller and Friedman [8] and Lauritzen [9]). Each node in a graph indicates a random variable, and the graph encodes conditional independence relations between the random variables. We focus on directed graphical models, also called Bayesian networks, where edge orientations come along with an intuitive causal interpretation. The conditional independence relations between the random variables, which are encoded by a directed acyclic graph (DAG), can be explored using the (directed) Markov property: each variable is conditionally independent of its non-descendants (excluding the parents) given its parents (cf. [9], Chapter 3.2).

Despite many areas of applications for directed graphical models, ranging from artificial intelligence, decision support systems, and engineering to genetics, geology, medicine, and finance (see, e.g., Pourret et al. [13]), graphical modelling of random vectors has mainly been limited to discrete and Gaussian distributions; see, for example, [8,9]. In the context of risk assessment, risk exposures are usually modelled by continuous variables, however, the assumption of Gaussianity leads invariably to severe underestimation of large risks and therefore to unsuitable models.

Recursive structural equation models (recursive SEMs) offer a possibility to construct directed graphical models; cf. Bollen [2], Pearl [12] and Spirtes et al. [17]. For a given DAG $\mathcal{D} = (V, E)$ with nodes $V = \{1, \dots, d\}$ and edges $E = \{(k, i) : i \in V \text{ and } k \in \mathrm{pa}(i)\}$ define

$$X_i = f_i(\mathbf{X}_{\mathrm{pa}(i)}, Z_i), \qquad i = 1, \dots, d, \tag{1.1}$$

1350-7265 © 2018 ISI/BS

where $\mathrm{pa}(i)$ denotes the parents of node i in $\mathcal{D}$ and f_i is a real-valued measurable function; $Z_1, \dots, Z_d$ are independent noise variables. Thus, a recursive SEM is specified by an underlying causal structure given by a DAG $\mathcal{D}$, the functions f_i, and the distributions of Z_i for $i = 1, \dots, d$. In this setting, the distribution of $\mathbf{X} = (X_1, \dots, X_d)$ is uniquely defined by the distributions of the noise variables and, denoting by $\mathrm{nd}(i)$ the non-descendants of node i,

$$X_i \perp\!\!\!\perp \mathbf{X}_{\mathrm{nd}(i)\setminus\mathrm{pa}(i)} \mid \mathbf{X}_{\mathrm{pa}(i)}, \qquad i = 1, \dots, d; \tag{1.2}$$

i.e., the distribution of $\mathbf{X}$ is Markov relative to $\mathcal{D}$ (see Theorem 1.4.1 and the related discussion in Pearl [12]). Recently, recursive linear SEMs and generalisations in a Gaussian setting have received particular attention; see Bühlmann et al. [3], Ernest et al. [7] and references therein.

Our focus is not on sums but on maxima, where natural candidates for the noise distributions are the extreme value distributions or distributions in their domains of attraction; see, for example, Resnick [14,15]. We define a *recursive max-linear* (ML) *model* $\mathbf{X}$ *on a* DAG $\mathcal{D}$ by

$$X_i := \bigvee_{k\in\mathrm{pa}(i)} c_{ki} X_k \vee c_{ii} Z_i, \qquad i = 1, \dots, d, \tag{1.3}$$

with independent random variables $Z_1, \dots, Z_d$ with support $\mathbb{R}_+ = [0, \infty)$ and positive weights c_{ki} for all $i \in V$ and $k \in \mathrm{pa}(i) \cup \{i\}$.

This new model is motivated by applications to risk analysis, where extreme risks play an essential role and may propagate through a network. In such a risk setting it is natural to require the noise variables to have positive infinite support. Moreover, we may think of the edge weights in (1.3) as relative quantities so that a risk may originate with certain proportions in its different ancestors.

In this paper, we investigate structural properties as well as graph properties of a recursive ML model $\mathbf{X}$ on a DAG $\mathcal{D}$. We will show that $\mathbf{X}$ is a *max-linear* (ML) *model* (for background on ML models in the context of extreme value theory see, for example, de Haan and Ferreira [5], Chapter 6) in the sense that

$$X_i = \bigvee_{j=1}^{d} b_{ji} Z_j, \qquad i = 1, \dots, d, \tag{1.4}$$

with $Z_1, \dots, Z_d$ as in (1.3), and $B = (b_{ij})_{d\times d}$ is a matrix with non-negative entries. We call B *max-linear* (ML) *coefficient matrix* of $\mathbf{X}$ and its entries *max-linear* (ML) *coefficients*.

The ML coefficients of $\mathbf{X}$ can be determined by a path analysis of $\mathcal{D}$. Throughout we write $k \to i$, if there is an edge from k to i in $\mathcal{D}$. We assign a weight to every path $p = [j = k_0 \to k_1 \to \dots \to k_n = i]$, which is the product of the edge weights along p multiplied by the weight of the noise variable Z_j (a concept, which goes back to Wright [19]):

$$d_{ji}(p) = c_{k_0,k_0} c_{k_0,k_1} \dots c_{k_{n-2},k_{n-1}} c_{k_{n-1},k_n} = c_{k_0,k_0} \prod_{l=0}^{n-1} c_{k_l,k_{l+1}}. \tag{1.5}$$

We will show that the ML coefficients are given for $i \in V$ by

$$\begin{aligned} b_{ji} &= \bigvee_{p \in P_{ji}} d_{ji}(p) \qquad \text{for } j \in \operatorname{an}(i), \\ b_{ii} &= c_{ii}, \qquad \text{and} \qquad b_{ji} = 0 \text{ for } j \in V \setminus \bigl(\operatorname{an}(i) \cup \{i\}\bigr), \end{aligned} \tag{1.6}$$

where P_{ji} is the set of paths from j to i and $\operatorname{an}(i)$ the ancestors of i.

The computation in (1.6) corresponds to the algebraic path problem over the max-times semiring $(\mathbb{R}_+, \vee, \cdot)$ (see e.g. Mahr [11] and Rote [16]). We present it in matrix form, using the matrix product over $(\mathbb{R}_+, \vee, \cdot)$. We apply this concept in the two different situations, where the DAG $\mathcal{D}$ is given, and we test, if a given ML coefficient matrix is consistent with $\mathcal{D}$, but also later on, when we check, if a given matrix defines a recursive SEM on some unspecified DAG.

From (1.6), it is clear that not all paths are needed for representing $\mathbf{X}$ as ML model (1.4). This perception leads to a complexity reduction of the model in different ways and in different situations. For every specific component X_i of $\mathbf{X}$ only those paths with terminal node i, which carry the maximum weight, are relevant for its representation (1.4), and we call them *max-weighted paths*. All other paths can be disposed of without changing this representation. It is even sufficient to consider only one in $\mathcal{D}$ max-weighted path from every ancestor of i to i. Consequently, X_i can be represented as component of a recursive ML model on a polytree with node set $\operatorname{an}(i) \cup \{i\}$ and with the same weights and noise variables as in the original representation (1.3).

However, in general none of these individual polytrees represents all components of $\mathbf{X}$ in the sense of (1.3) simultaneously. Still there may be subgraphs of $\mathcal{D}$ and weights such that all components of $\mathbf{X}$ have representation (1.3), and we present all such possible subgraphs and weights. In particular, we characterize the smallest subgraph of this kind, which we call *minimum max-linear* (ML) DAG *of* $\mathbf{X}$, and point out its prominent role.

We show that all DAGs and weights, which represent $\mathbf{X}$ as in (1.3) can be identified from the ML coefficient matrix B of $\mathbf{X}$. In this context, we also give necessary and sufficient conditions on a matrix to be the ML coefficient matrix of some recursive ML model.

It is a simple but important observation that there is a natural order between the components of $\mathbf{X}$; from (1.3) we see immediately that $X_i \geq c_{ki} X_k$ holds for all $i \in V$ and $k \in \operatorname{pa}(i)$. For every component of $\mathbf{X}$ and some $U \subseteq V$ we find lower and upper bounds in terms of $\mathbf{X}_U := (X_l, l \in U)$. Often we do not need all components of $\mathbf{X}_U$ to compute the best bounds of X_i in terms of components of $\mathbf{X}_U$. If $i \in U$, then an upper and lower bound is given by X_i itself; otherwise, for a lower bound, we only need to consider a component X_j of $\mathbf{X}_U$ if $j \in \operatorname{an}(i)$, but no max-weighted path from j to i passes through some node in $U \setminus \{j\}$. A similar result and concept applies for the upper bound of X_i. Thus, the max-weighted paths also lead in this context indirectly to a complexity reduction. We will also use the max-weighted ancestors of i in U to obtain a minimal representation of X_i in terms of $\mathbf{X}_U$ and noise variables.

Our paper is organized as follows. In Section 2, we discuss the max-linearity of a recursive ML model $\mathbf{X}$ and express its ML coefficient matrix in terms of a weighted adjacency matrix of a corresponding DAG. Section 3 introduces the important notion of a max-weighted path and studies its consequences for the ML coefficients. In Section 4, we give necessary and sufficient conditions for a ML model being a recursive ML model on a given DAG. Section 5 is devoted to the minimum ML DAG of $\mathbf{X}$ as the DAG with the minimum number of edges within the class of all DAGs representing $\mathbf{X}$ in the sense of (1.3). In Section 6, given a set of node variables, we investigate which information can be gained for the other components of $\mathbf{X}$. This results in lower and upper bounds for the components. Finally, we derive a minimal representation for the components of $\mathbf{X}$ as max-linear function of a subset of node variables and certain noise variables.

We use the following notation throughout. For a node $i \in V$, the sets $\operatorname{an}(i)$, $\operatorname{pa}(i)$, and $\operatorname{de}(i)$ contain the *ancestors, parents*, and *descendants* of i in $\mathcal{D}$. Furthermore, we use the notation $\operatorname{An}(i) = \operatorname{an}(i) \cup \{i\}$, $\operatorname{Pa}(i) = \operatorname{pa}(i) \cup \{i\}$, and $\operatorname{De}(i) = \operatorname{de}(i) \cup \{i\}$. We write $U \subseteq V$ for a non-empty subset U of nodes, $\mathbf{X}_U = (X_l, l \in U)$, and $U^c = V \setminus U$. All our vectors are row vectors. We also extend the previous notation in a natural way by writing $\operatorname{an}(U) = \bigcup_{i \in U} \operatorname{an}(i)$, $\operatorname{An}(U) = \operatorname{an}(U) \cup U$, and so on. For a matrix B with non-negative entries, we write $\operatorname{sgn}(B)$ for the matrix with entries equal to 1, if the corresponding entry in B is positive and 0 else. We denote by $\mathbf{1}_U$ the indicator function of U, and set $\mathbf{1}_\varnothing \equiv 0$. In general, we consider statements for $i \in \varnothing$ as invalid. For arbitrary (possibly random) $a_i \in \mathbb{R}_+$, we set $\bigvee_{i \in \varnothing} a_i = 0$ and $\bigwedge_{i \in \varnothing} a_i = \infty$.

2. Max-linearity of a recursive max-linear model

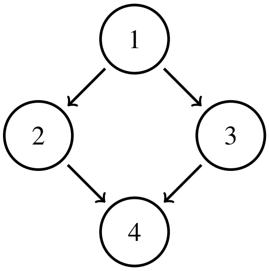
For a recursive ML model $\mathbf{X}$ on a DAG $D = (V, E)$, given by (1.3), we derive its max-linear representation (1.4). We start with our leading example, a recursive ML model on the diamond-shaped DAG depicted below.

Example 2.1 (Max-linear representation of a recursive ML model). Consider a recursive ML model $\mathbf{X} = (X_1, X_2, X_3, X_4)$ on the DAG

$$\mathcal{D} = (V, E) = \big(\{1, 2, 3, 4\}, \big\{(1, 2), (1, 3), (2, 4), (3, 4)\big\}\big)$$

with weights c_{ki} for $i \in V$ and $k \in \operatorname{Pa}(i)$. We obtain for the components of $\mathbf{X}$:

$$\begin{aligned}
X_1 &= c_{11} Z_1 \\
X_2 &= c_{12} X_1 \vee c_{22} Z_2 = c_{12} c_{11} Z_1 \vee c_{22} Z_2 \\
X_3 &= c_{13} X_1 \vee c_{33} Z_3 = c_{13} c_{11} Z_1 \vee c_{33} Z_3 \\
X_4 &= c_{24} X_2 \vee c_{34} X_3 \vee c_{44} Z_4 \\
&= c_{24}(c_{12} c_{11} Z_1 \vee c_{22} Z_2) \\
&\quad \vee c_{34}(c_{13} c_{11} Z_1 \vee c_{33} Z_3) \vee c_{44} Z_4 \\
&= (c_{24} c_{12} c_{11} \vee c_{34} c_{13} c_{11}) Z_1 \\
&\quad \vee c_{24} c_{22} Z_2 \vee c_{34} c_{33} Z_3 \vee c_{44} Z_4.
\end{aligned}$$



Thus $\mathbf{X}$ satisfies (1.4) with ML coefficient matrix

$$B = \begin{bmatrix} c_{11} & c_{11} c_{12} & c_{11} c_{13} & c_{11} c_{12} c_{24} \vee c_{11} c_{13} c_{34} \\ 0 & c_{22} & 0 & c_{22} c_{24} \\ 0 & 0 & c_{33} & c_{33} c_{34} \\ 0 & 0 & 0 & c_{44} \end{bmatrix}.$$

We observe that the ML coefficients satisfy indeed (1.6). Moreover, B is an upper triangular matrix, since $\mathcal{D}$ is well-ordered (cf. Remark 2.3(ii)).

The following result shows that such a representation can be obtained in general: every component of a recursive ML model has a max-linear representation in terms of its ancestral noise variables and an independent one. It provides a general method to calculate the ML coefficients by a path analysis as described in (1.5) and (1.6).

Theorem 2.2. *Let* $\mathbf{X}$ *be a recursive* ML *model on a DAG* $\mathcal{D} = (V, E)$, *and let* $B = (b_{ij})_{d\times d}$ *be the matrix with entries as defined in* (1.6). *Then*

$$X_i = \bigvee_{j\in\mathrm{An}(i)} b_{ji} Z_j, \qquad i = 1, \dots, d; \tag{2.1}$$

i.e., B *is the* ML *coefficient matrix of* $\mathbf{X}$.

Proof. Without loss of generality we assume throughout this proof that $\mathcal{D}$ is well-ordered (cf. Remark 2.3(ii)). We prove the identity (2.1) by induction on the number of nodes of $\mathcal{D}$. For $d = 1$ we have by (1.3)

$$X_1 = c_{11} Z_1 = b_{11} Z_1,$$

where the last equality holds by (1.6). Suppose that (2.1) holds for a recursive ML model $\mathbf{X}$ of dimension d; that is,

$$X_k = \bigvee_{j\in\mathrm{An}(k)} b_{jk} Z_j = \bigvee_{j\in\mathrm{an}(k)} \bigvee_{p\in P_{jk}} d_{jk}(p) Z_j \vee c_{kk} Z_k, \qquad k = 1, \dots, d.$$

Now consider a $(d+1)$-variate recursive ML model. We first investigate the nodes $i \in \{1, \dots, d\}$. Since $\mathcal{D}$ is well-ordered, we have $(d+1) \in V \setminus \mathrm{pa}(i)$. Hence, it suffices to show representation (2.1) with respect to the subgraph $\mathcal{D}[\{1, \dots, d\}] = (\{1, \dots, d\}, E \cap (\{1, \dots, d\} \times \{1, \dots, d\}))$. However, this holds by the induction hypothesis. So we can use this representation for $i \in \{1, \dots, d\}$ and (A.1) to obtain

$$\begin{aligned}
X_{d+1} &= \bigvee_{k\in\mathrm{pa}(d+1)} c_{k,d+1} X_k \vee c_{d+1,d+1} Z_{d+1} \\
&= \bigvee_{k\in\mathrm{pa}(d+1)} \bigvee_{j\in\mathrm{an}(k)} \bigvee_{p\in P_{jk}} c_{k,d+1} d_{jk}(p) Z_j \vee \bigvee_{k\in\mathrm{pa}(d+1)} c_{k,d+1} c_{kk} Z_k \vee c_{d+1,d+1} Z_{d+1} \\
&= \bigvee_{j\in\mathrm{an}(d+1)} \Biggl(\bigvee_{k\in\mathrm{de}(j)\cap\mathrm{pa}(d+1)} \bigvee_{p\in P_{jk}} c_{k,d+1} d_{jk}(p) \\
&\qquad \vee \bigvee_{k\in\mathrm{pa}(d+1)\cap\{j\}} c_{k,d+1} c_{kk} \Biggr) Z_j \vee c_{d+1,d+1} Z_{d+1}.
\end{aligned}$$

Observe that every path from some j to $d+1$ is of the form $p = [j \to \cdots \to k \to d+1]$ for some $k \in \operatorname{de}(j) \cap \operatorname{pa}(d+1)$, or an edge $j \to d+1$ corresponding to $j \in \operatorname{pa}(d+1)$. From (1.5), the path p has weight $d_{j,d+1}(p) = d_{jk}(p)c_{k,d+1}$, and the edge $j \to d+1$ has weight $d_{j,d+1}([j \to d+1]) = c_{jj}c_{j,d+1}$. This yields

$$X_{d+1} = \bigvee_{j \in \operatorname{an}(d+1)} \bigvee_{p \in P_{j,d+1}} d_{j,d+1}(p) Z_j \vee c_{d+1,d+1} Z_{d+1} = \bigvee_{j \in \operatorname{An}(d+1)} b_{j,d+1} Z_j,$$

where we have used that $b_{j,d+1} = \bigvee_{p \in P_{j,d+1}} d_{j,d+1}(p)$ for $j \in \operatorname{an}(d+1)$ and $b_{d+1,d+1} = c_{d+1,d+1}$. □

By (1.6) the ML coefficient b_{ji} of $\mathbf{X}$ is different from zero if and only if $j \in \operatorname{An}(i)$. This information is contained in the *reachability matrix* $R = (r_{ij})_{d \times d}$ of $\mathcal{D}$, which has entries

$$r_{ji} := \begin{cases} 1, & \text{if there is a path from } j \text{ to } i, \text{ or if } j = i, \\ 0, & \text{otherwise.} \end{cases}$$

If the jith entry of R is equal to one, then i *is reachable from* j.

Remark 2.3. Let R be the reachability matrix of $\mathcal{D}$.

(i) The ML coefficient matrix B is a weighted reachability matrix of $\mathcal{D}$; i.e., $R = \operatorname{sgn}(B)$.

(ii) The DAG $\mathcal{D}$ can be *well-ordered*, which means that the set $V = \{1, \dots, d\}$ of nodes can be linearly ordered in a way compatible with $\mathcal{D}$ such that $k \in \operatorname{pa}(i)$ implies $k < i$ (see, e.g., Appendix A of Diestel [6]). If $\mathcal{D}$ is well-ordered, then B and R are upper triangular matrices.

Finding the ML coefficient matrix B from $\mathcal{D}$ and the weights in (1.3) by a path analysis as described in (1.5) and (1.6) would be very inefficient. We may, however, compute B by means of a specific matrix multiplication.

For two non-negative matrices F and G, where the number of columns in F is equal to the number of rows in G, we define the product $\odot : \mathbb{R}_+^{m \times n} \times \mathbb{R}_+^{n \times p} \to \mathbb{R}_+^{m \times p}$ by

$$\bigl(F = (f_{ij})_{m \times n}, G = (g_{ij})_{n \times p}\bigr) \mapsto F \odot G := \left(\bigvee_{k=1}^{n} f_{ik} g_{kj} \right)_{m \times p}. \tag{2.2}$$

The triple $(\mathbb{R}_+, \vee, \cdot)$, which is called max-times or subtropical algebra (see e.g. [11,16]), is an idempotent semiring with 0 as 0-element and 1 as 1-element. The operation $\odot$ is therefore a matrix product over a semiring. Such semirings are fundamental in max-linear systems; for an introduction see Butkovič [4]. The matrix product $\odot$ is associative: for $F \in \mathbb{R}_+^{m \times n}$, $G \in \mathbb{R}_+^{n \times p}$, and $H \in \mathbb{R}_+^{p \times q}$, $F \odot (G \odot H) = (F \odot G) \odot H$, and we have $(F \odot G)^\top = G^\top \odot F^\top$. Denoting by $\mathcal{B}$ all $d \times d$ matrices with non-negative entries and by $\vee$ the componentwise maximum between two matrices, $(\mathcal{B}, \vee, \odot)$ is also a semiring with the null matrix as 0-element and the identity matrix $\mathrm{id}_{d \times d}$ as 1-element. This semiring is, however, not commutative, since $\odot$ is in general not. Consistent with a matrix product, we define powers recursively: $A^{\odot 0} := \mathrm{id}_{d\times d}$ and $A^{\odot n} := A^{\odot(n-1)} \odot A$ for $A \in \mathcal{B}$ and $n \in \mathbb{N}$.

The matrix product $\odot$ allows us to represent the ML coefficient matrix B of $\mathbf{X}$ in terms of the weighted adjacency matrix $(c_{ij}\mathbf{1}_{\mathrm{pa}(j)}(i))_{d\times d}$ of $\mathcal{D}$.

Theorem 2.4. *Let* $\mathbf{X}$ *be a recursive* ML *model on a DAG* $\mathcal{D} = (V, E)$ *with weights* c_{ki} *for* $i \in V$ *and* $k \in \mathrm{Pa}(i)$ *as in* (1.3). *Define the matrices*

$$A := \mathrm{diag}(c_{11}, \dots, c_{dd}),\ A_0 := \bigl(c_{ij}\mathbf{1}_{\mathrm{pa}(j)}(i)\bigr)_{d\times d} \quad \text{and} \quad A_1 := \bigl(c_{ii}c_{ij}\mathbf{1}_{\mathrm{pa}(j)}(i)\bigr)_{d\times d}.$$

Then the ML *coefficient matrix* B *of* $\mathbf{X}$ *from Theorem* 2.2 *has representation*

$$B = A \qquad \text{for } d = 1 \quad \text{and} \quad B = A \vee \bigvee_{k=0}^{d-2} \bigl(A_1 \odot A_0^{\odot k}\bigr) \qquad \text{for } d \geq 2.$$

Proof. For $d = 1$ we know from (1.6) that $b_{11} = c_{11}$. Hence, $B = A$. Now assume that $d \geq 2$. First we show that, if $\mathcal{D}$ has a path of length n (a path consisting of n edges) from node j to node i, then the jith entry of the matrix $A_1 \odot A_0^{\odot(n-1)}$ is equal to the maximum weight of all paths of lengths n from j to i, otherwise it is zero. The proof is by induction on n.

An edge $j \to i$, which is the only path of length $n = 1$, has the weight $d_{ji}([j \to i]) = c_{jj}c_{ji}$. Since the jith entry of the matrix $A_1 \odot A_0^{\odot 0} = A_1 \odot \mathrm{id}_{d\times d} = A_1$ is given by $c_{jj}c_{ji}\mathbf{1}_{\mathrm{pa}(i)}(j)$, the statement is true for $n = 1$.

Denote by $a_{0,ji}$, $a_{n,ji}$, and $a_{n+1,ji}$ the jith entry of A_0, $A_1 \odot A_0^{\odot(n-1)}$, and $A_1 \odot A_0^{\odot n}$, respectively. As $A_1 \odot A_0^{\odot n} = (A_1 \odot A_0^{\odot(n-1)}) \odot A_0$, the jith entry of $A_1 \odot A_0^{\odot n}$ is given by $a_{n+1,ji} = \bigvee_{k=1}^{d} a_{n,jk}a_{0,ki} = \bigvee_{k=1}^{d} a_{n,jk}c_{ki}\mathbf{1}_{\mathrm{pa}(i)}(k)$. We obtain from the induction hypothesis and (1.5) that $a_{n,jk}a_{0,ki}$ is zero, if $\mathcal{D}$ does not contain a path of length n from j to k or the edge $k \to i$; otherwise it is equal to the maximum weight of all paths which consist of a path of length n from j to k and the edge $k \to i$. Since every path of length $n + 1$ from j to i is of this form for some $k \in V$, the jith entry of $A_1 \odot A_0^{\odot n}$ is indeed equal to the maximum weight of all paths of length $n + 1$ from j to i if there exists such a path, otherwise it is zero.

Finally, again by (1.6), for $i \in V$ and $j \in \mathrm{an}(i)$, the ML coefficient b_{ji} is equal to the maximum weight of all paths from j to i, and note that due to acyclicity, a path in $\mathcal{D}$ is at most of length $d - 1$. Thus, if $j \in \mathrm{an}(i)$ then the jith entry of $\bigvee_{k=0}^{d-2} A_1 \odot A_0^{\odot k}$ is equal to b_{ji}, otherwise it is zero. Since by (1.6), $b_{ii} = c_{ii}$ and $b_{ji} = 0$ for $j \in V \setminus \mathrm{An}(i)$, the ML coefficient matrix B is given by

$$B = A \vee A_1 \vee (A_1 \odot A_0) \vee \bigl(A_1 \odot A_0^{\odot 2}\bigr) \vee \cdots \vee \bigl(A_1 \odot A_0^{\odot(d-2)}\bigr). \qquad \square$$

The following has been shown in the proof of Theorem 2.4.

Corollary 2.5. *If* $\mathcal{D}$ *has a path of length* n *from* j *to* i, *the* ji*th entry of the matrix* $A_1 \odot A_0^{\odot(n-1)}$ *is equal to the maximum weight of all paths of length* n *from* j *to* i, *otherwise the entry is zero.*

Summarizing the noise variables of $\mathbf{X}$ into the vector $\mathbf{Z} = (Z_1, \ldots, Z_d)$, the representation (2.1) of $\mathbf{X}$ can be written by means of the product $\odot$ as

$$\mathbf{X} = \mathbf{Z} \odot B = \left(\bigvee_{j=1}^{d} b_{ji} Z_j, i = 1, \ldots, d \right) = \left(\bigvee_{j \in \mathrm{An}(i)} b_{ji} Z_j, i = 1, \ldots, d \right).$$

Consequently, the definition of the matrix product $\odot$ modifies and extends the definition given in [18], Section 2.1, equation (2).

3. Max-weighted paths

Given a recursive ML model $\mathbf{X}$ on a DAG $\mathcal{D} = (V, E)$ with weights c_{ki} for $i \in V$, $k \in \mathrm{Pa}(i)$ and ML coefficient matrix $B = (b_{ij})_{d \times d}$, we investigate the paths of $\mathcal{D}$ and their particular weights, the implications on the ML coefficients as well as induced subgraph structures leading to reduced representations of (1.3).

From (1.6) and (2.1), we know that a path p from j to i, whose weight $d_{ji}(p)$ is strictly smaller than b_{ji}, does not have any influence on the distribution of $\mathbf{X}$. This fact suggests the following definition.

Definition 3.1. Let $\mathbf{X}$ be a recursive ML model on a DAG $\mathcal{D} = (V, E)$ with path weights as in (1.5) and ML coefficient matrix B. We call a path p from j to i a *max-weighted path* if $b_{ji} = d_{ji}(p)$.

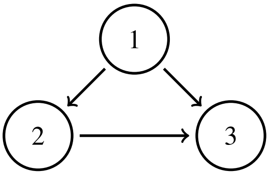
A prominent example, where all paths are max-weighted, is the following.

Example 3.2 (Polytree). A *polytree* is a DAG whose underlying undirected graph has no cycles; polytrees have at most one path between any pair of nodes. Thus, assuming that $\mathbf{X}$ is a recursive ML model on a polytree, all paths must be max-weighted.

The following example indicates the importance and consequences of max-weighted paths.

Example 3.3 (Max-weighted paths, graph reduction). Consider a recursive ML model $\mathbf{X} = (X_1, X_2, X_3)$ on the DAG

$$\mathcal{D} = (V, E) = \big(\{1, 2, 3\}, \big\{(1, 2), (1, 3), (2, 3)\big\}\big)$$



with weights c_{ki} for $i \in V$ and $k \in \mathrm{Pa}(i)$ and ML coefficient matrix B. We distinguish between two situations:

(1) If $c_{13} > c_{12}c_{23}$, then the edge 1 → 3 is the unique max-weighted path from 1 to 3.

(2) If, however, $c_{13} \le c_{12}c_{23}$, then $b_{13} = c_{11}c_{12}c_{23} = \frac{b_{12}b_{23}}{b_{22}}$ and the path $[1 \to 2 \to 3]$ is max-weighted. We obtain in this case

$$X_3 = b_{13}Z_1 \vee b_{23}Z_2 \vee b_{33}Z_3 = \frac{b_{23}}{b_{22}}(b_{12}Z_1 \vee b_{22}Z_2) \vee b_{33}Z_3 = c_{23}X_2 \vee b_{33}Z_3.$$

Thus, $\mathbf{X}$ is also a recursive ML model on the DAG

$$\mathcal{D}^B := \big(\{1,2,3\}, \big\{(1,2),(2,3)\big\}\big).$$

Here $\mathcal{D}^B$ is the DAG with the minimum number of edges such that $\operatorname{sgn}(B)$ is its reachability matrix. By Remark 2.3(i) there cannot be a smaller DAG representing $\mathbf{X}$ in the sense of (1.3).

We present some immediate consequences of the path weights in (1.5) and the definition of max-weighted paths.

Remark 3.4. (i) If there is only one path between two nodes, it is max-weighted.

(ii) Every subpath of a max-weighted path is also max-weighted.

(iii) Every path, which results from a max-weighted path by replacing a subpath with another max-weighted subpath, is also max-weighted.

To find for some $i \in V$ and $j \in \operatorname{an}(i)$ the ML coefficient b_{ji} it suffices to know the weight c_{jj} of the noise variable Z_j and the edge weights along one arbitrary max-weighted path from j to i, since every max-weighted path from j to i has the same weight. This allows us to represent every component of $\mathbf{X}$ as component of a recursive ML model on a subgraph of $\mathcal{D}$. For this purpose, we introduce the following definition.

Definition 3.5. Let $\mathbf{X}$ be a recursive ML model on a DAG $\mathcal{D} = (V, E)$, and let $\overline{\mathcal{D}} = (\overline{V}, \overline{E})$ be a subgraph of $\mathcal{D}$. We denote by $\overline{\operatorname{pa}}(i)$ the parents of node i in $\overline{\mathcal{D}}$ and define

$$Y_i := \bigvee_{k \in \overline{\operatorname{pa}}(i)} c_{ki}Y_k \vee c_{ii}Z_i, \qquad i \in \overline{V},$$

with the same weights and noise variables as in the representation of $\mathbf{X}$ in (1.3). We call the resulting recursive ML model $\mathbf{Y} = (Y_l, l \in \overline{V})$ *recursive* ML *submodel of* $\mathbf{X}$ *induced by* $\overline{\mathcal{D}}$.

We summarize some immediate properties of $\mathbf{Y}$.

Remark 3.6. Let $i \in V$ with ancestors $\operatorname{an}(i)$ in $\mathcal{D}$. Denote by $\overline{B} = (\overline{b}_{ij})_{|\overline{V}|\times|\overline{V}|}$ the ML coefficient matrix of $\mathbf{Y}$.

(i) Every path in $\overline{\mathcal{D}}$ has the same weight (1.5) as in $\mathcal{D}$.

(ii) A path of $\overline{\mathcal{D}}$, which is in $\mathcal{D}$ a max-weighted path, is also in $\overline{\mathcal{D}}$ max-weighted.

(iii) For $j \in \operatorname{an}(i)$, $\overline{\mathcal{D}}$ has one in $\mathcal{D}$ max-weighted path from j to i if and only if $\overline{b}_{ji} = b_{ji}$.

(iv) $\overline{\mathcal{D}}$ has one in $\mathcal{D}$ max-weighted path from every $j \in \operatorname{an}(i)$ to i if and only if $X_i = Y_i$.

By Remark 3.4(ii), for every $i \in V$, there exists a polytree $\mathcal{D}_i$ of $\mathcal{D}$ with node set $\mathrm{An}(i)$, which has exactly one in $\mathcal{D}$ max-weighted path from every ancestor of i to i. There may even exist several such polytrees (cf. Example 3.8 below). We learn from the construction of $\mathcal{D}_i$ and Remark 3.4(ii) that indeed every path of $\mathcal{D}_i$ is in $\mathcal{D}$ max-weighted. Therefore, some component X_j of $\mathbf{X}$ coincides by Remark 3.6(iv) with the corresponding one of the recursive ML submodel of $\mathbf{X}$ induced by $\mathcal{D}_i$ if and only if $\mathcal{D}_i$ has at least one path from every ancestor of j in $\mathcal{D}$ to j. By construction of $\mathcal{D}_i$ this property holds obviously for X_i. We summarize this result as follows.

Proposition 3.7. *Let* $\mathbf{X}$ *be a recursive* ML *model on a* DAG $\mathcal{D} = (V, E)$. *For some* $i \in V$ *and* $\mathrm{An}(i)$ *in* $\mathcal{D}$ *let* $\mathcal{D}_i$ *be a polytree with node set* $\mathrm{An}(i)$ *such that* $\mathcal{D}_i$ *has one in* $\mathcal{D}$ *max-weighted path from every* $j \in \mathrm{an}(i)$ *to* i. *Let* $\mathbf{Y}_i = (Y_l, l \in \mathrm{An}(i))$ *be the recursive* ML *submodel of* $\mathbf{X}$ *induced by* $\mathcal{D}_i$. *Then for all* $j \in \mathrm{An}(i)$, *which have the same ancestors in* $\mathcal{D}_i$ *and* $\mathcal{D}$, *we have* $X_j = Y_j$.

We discuss the recursive ML model from Example 2.1 in the context of Definition 3.1 and Proposition 3.7.

Example 3.8 (Continuation of Example 2.1: max-weighted paths, polytrees, conditional independence). For the polytrees as in Proposition 3.7, we identify all max-weighted paths ending in node 4. By Remark 3.4(i), the paths $[2 \to 4]$ and $[3 \to 4]$ are max-weighted. For the weights of the paths $[1 \to 2 \to 4]$ and $[1 \to 3 \to 4]$, we have three situations:

$$c_{11}c_{12}c_{24} = c_{11}c_{13}c_{34}, \qquad c_{11}c_{12}c_{24} > c_{11}c_{13}c_{34}, \quad \text{and} \quad c_{11}c_{12}c_{24} < c_{11}c_{13}c_{34}.$$

In the first situation, both paths from 1 to 4 are max-weighted. Thus, there are two different polytrees having one in $\mathcal{D}$ max-weighted path from every ancestor of 4 to 4, namely,

$$\mathcal{D}_{4,1} = \bigl(\{1,2,3,4\}, \bigl\{(1,2),(2,4),(3,4)\bigr\}\bigr) \quad \text{and}$$
$$\mathcal{D}_{4,2} = \bigl(\{1,2,3,4\}, \bigl\{(1,3),(2,4),(3,4)\bigr\}\bigr).$$

In the second situation, the path $[1 \to 2 \to 4]$ is the unique max-weighted path from 1 to 4 and, hence, $\mathcal{D}_{4,1}$ is the unique polytree as in Proposition 3.7 for node 4. The third situation is symmetric to the second, such that $\mathcal{D}_{4,2}$ is also such a unique polytree.

Now let $\mathbf{Y}_1 = (Y_{1,1}, Y_{1,2}, Y_{1,3}, Y_{1,4})$ and $\mathbf{Y}_2 = (Y_{2,1}, Y_{2,2}, Y_{2,3}, Y_{2,4})$ be the recursive ML submodels of $\mathbf{X}$ induced by $\mathcal{D}_{4,1}$ and $\mathcal{D}_{4,2}$. The distributions of $\mathbf{X}$, $\mathbf{Y}_1$, and $\mathbf{Y}_2$ are Markov relative to $\mathcal{D}$, $\mathcal{D}_{4,1}$, and $\mathcal{D}_{4,2}$, respectively. For a DAG, the local Markov property as specified in (1.2), is by Proposition 4 of Lauritzen et al. [10] equivalent to the global Markov property (for a definition see Corollary 3.23 of [9]). Using this property, we find

$$Y_{1,1} \perp\!\!\!\perp Y_{1,4} \mid Y_{1,2} \quad \text{and} \quad Y_{2,1} \perp\!\!\!\perp Y_{2,4} \mid Y_{2,3.}$$

If the path $[1 \to 2 \to 4]$ is max-weighted, we have by Proposition 3.7 that

$$Y_{1,1} = X_1, \qquad Y_{1,2} = X_2, \quad \text{and} \quad Y_{1,4} = X_4,$$

hence, $X_1 \perp\!\!\!\perp X_4 \mid X_2$. Accordingly, if $[1 \to 3 \to 4]$ is max-weighted, then

$$Y_{2,1} = X_1, \qquad Y_{2,3} = X_3, \quad \text{and} \quad Y_{2,4} = X_4,$$

and $X_1 \perp\!\!\!\perp X_4 \mid X_3$ holds. Since the only conditional independence property encoded in $\mathcal{D}$ by the (global) Markov property is $X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$, we can identify additional conditional independence properties of $\mathbf{X}$ from the polytrees in Proposition 3.7.

Remark 3.9. (i) Assume the situation of Proposition 3.7. Let V_i be the set of all nodes in $\mathrm{An}(i)$, which have the same ancestors in $\mathcal{D}$ and $\mathcal{D}_i$. Since the distributions of $\mathbf{X}$ and $\mathbf{Y}$ are Markov relative to $\mathcal{D}$ and $\mathcal{D}_i$, respectively, conditional independence properties of $\mathbf{X}$ are encoded in $\mathcal{D}$ and of $\mathbf{Y}$ in $\mathcal{D}_i$. By Proposition 3.7, the conditional independence relations between subvectors of $\mathbf{Y}_{V_i} = (Y_l, l \in V_i)$, which we can read off from $\mathcal{D}_i$, hold also between the corresponding subvectors of $\mathbf{X}$. Since missing edges correspond to conditional independence properties, and $\mathcal{D}_i$ is a subgraph of $\mathcal{D}$, we can often identify additional conditional independence properties of $\mathbf{X}$ from $\mathcal{D}_i$.

(ii) From (i) or Example 3.8 we learn that a recursive ML model on a DAG $\mathcal{D}$ is in general not *faithful*; that is, not all its conditional independence properties are encoded in $\mathcal{D}$ by the (global) Markov property.

As can be seen from Examples 3.3 and 3.8, any reduction of a recursive ML model depends on the existence of max-weighted paths that pass through specific nodes. The following result shows how we can obtain this information from its ML coefficient matrix.

Theorem 3.10. *Let X be a recursive* ML *model on a DAG $\mathcal{D} = (V, E)$ with* ML *coefficient matrix B. Let further $U \subseteq V$, $i \in V$ and $j \in \mathrm{an}(i)$, and recall from Remark* 2.3(i) *that $b_{ji} > 0$.*

(a) *There is a max-weighted path from j to i, which passes through some node in U if and only if*

$$b_{ji} = \bigvee_{k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)} \frac{b_{jk} b_{ki}}{b_{kk}}. \tag{3.1}$$

(b) *No max-weighted path from j to i passes through some node in U if and only if*

$$b_{ji} > \bigvee_{k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)} \frac{b_{jk} b_{ki}}{b_{kk}}. \tag{3.2}$$

This holds also for $U = \varnothing$.

Proof. First assume that $\mathrm{De}(j) \cap U \cap \mathrm{An}(i) = \varnothing$. Thus no path, hence also no max-weighted path, from j to i passes through some node in U, and it suffices to verify (b). Since the right-hand side of (3.2) is zero if and only if $\mathrm{De}(j) \cap U \cap \mathrm{An}(i) = \varnothing$, and the ML coefficient b_{ji} is positive, (b) is proven for this case (including the case that $U = \varnothing$).

Now assume that $\mathrm{De}(j) \cap U \cap \mathrm{An}(i) = \{k\}$, which implies that there is a path from j to i passing through $k \in U$. If $k = i$ or $k = j$, there is obviously a max-weighted path from j to i passing through i or j and (3.1) is always valid.

Next, assume that $k \in V \setminus \{i, j\}$, and let p_1 and p_2 be max-weighted paths from j to k and from k to i; respectively. Denote by p the path from j to i consisting of the subpaths p_1 and p_2. By (1.5) and the definition of a max-weighted path we obtain

$$d_{ji}(p) = \frac{1}{c_{kk}} d_{jk}(p_1) d_{ki}(p_2) = \frac{b_{jk} b_{ki}}{b_{kk}}.$$

Since p is max-weighted if and only if $b_{ji} = d_{ji}(p)$, and this is not the case if and only if $b_{ji} > d_{ji}(p)$, we have shown (a) and (b) for the situation of $\mathrm{De}(j) \cap U \cap \mathrm{An}(i) = \{k\}$. In particular, it follows that $b_{ji} \geq \frac{b_{jk} b_{ki}}{b_{kk}}$ for all $k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)$.

Assume now that $\mathrm{De}(j) \cap U \cap \mathrm{An}(i)$ contains more than one element, and that a max-weighted path from j to i passes through some node $k \in U$. We know from above that this is equivalent to

$$b_{ji} = \frac{b_{jk} b_{ki}}{b_{kk}} \quad \text{and} \quad b_{ji} \geq \frac{b_{jl} b_{li}}{b_{ll}} \qquad \text{for all } l \in \big(\mathrm{De}(j) \cap U \cap \mathrm{An}(i)\big) \setminus \{k\},$$

which is again equivalent to (3.1). Similarly, we obtain (b). □

Remark 3.11. Recall the matrix product $\odot$ from (2.2) and let R be the reachability matrix of $\mathcal{D}$. We obtain from $R = \mathrm{sgn}(B)$ (Remark 2.3(i)) that for $i, j \in V$

$$\bigvee_{k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} = \bigvee_{k=1}^{d} \frac{b_{jk} b_{ki}}{b_{kk}} \mathbf{1}_U(k) =: \bigvee_{k=1}^{d} b_{jk} b_{U,ki}$$

is the jith entry of the matrix $B \odot B_U$ with $B_U = (b_{U,ij})_{d \times d}$. Thus, we may decide whether there is a max-weighted path between two nodes that passes through some node in U by comparing the entries of the matrices B and $B \odot B_U$. Such use of the matrix product $\odot$ can be made at various points throughout the paper, for instance, in Remark 5.2(ii), Theorem 5.3, and Lemma 6.3(b).

The following corollary gives an important property of the ML coefficients. The first part has been shown in the proof of Theorem 3.10, the second part follows from Remark 2.3(i).

Corollary 3.12. *For all $i \in V$, $k \in \mathrm{An}(i)$, and $j \in \mathrm{An}(k)$, $b_{ji} \geq \frac{b_{jk} b_{ki}}{b_{kk}} > 0$. Indeed, $b_{ji} \geq \frac{b_{jk} b_{ki}}{b_{kk}}$ holds for all $i, j, k \in V$.*

We learn immediately from (1.3) that $c_{ki} X_k \leq X_i$ for all $i \in V$ and $k \in \mathrm{pa}(i)$. From Corollary 3.12, we find such inequalities also for components, whose nodes are not connected by an edge but by a path of arbitrary length.

Corollary 3.13. *For all $i \in V$ and $j \in \mathrm{An}(i)$, we have $\frac{b_{ji}}{b_{jj}} X_j \leq X_i$.*

Proof. Note that $\mathrm{An}(j) \subseteq \mathrm{An}(i)$. Using the max-linear representation (2.1) of X_i and X_j as well as Corollary 3.12, we obtain

$$X_i = \bigvee_{l \in \mathrm{An}(i)} b_{li} Z_l \geq \bigvee_{l \in \mathrm{An}(j)} b_{li} Z_l \geq \bigvee_{l \in \mathrm{An}(j)} \frac{b_{lj} b_{ji}}{b_{jj}} Z_j = \frac{b_{ji}}{b_{jj}} \bigvee_{l \in \mathrm{An}(j)} b_{lj} Z_j = \frac{b_{ji}}{b_{jj}} X_j. \qquad \square$$

4. ML coefficients leading to a recursive ML model on a given DAG

Recall the definition of a ML model given in (1.4). From Theorem 2.2, we know that every recursive ML model is max-linear. In this section we provide necessary and sufficient conditions on a ML model to be a recursive ML model on a given DAG $\mathcal{D}$.

It can be shown that every ML model, which is a recursive SEM as given in (1.1) with unspecified functions $f_1, \ldots, f_d$, must be a recursive ML model. That a recursive ML model is also a recursive SEM follows immediately from its recursive definition. To summarize, a ML model can be represented as a recursive SEM (1.1) on a DAG $\mathcal{D}$ if and only if it has a recursive ML representation (1.3) relative to the same DAG $\mathcal{D}$.

Motivated by Remark 2.3(i), in what follows we assume that $\mathrm{sgn}(B)$ is the reachability matrix R of $\mathcal{D}$. In our investigation the DAG with the minimum number of edges, such that $R = \mathrm{sgn}(B)$, will play an important role. This has already been indicated in Example 3.3.

We give a general definition of the DAG with the minimum number of edges that represents the same reachability relation as a given DAG.

Definition 4.1. Let $\mathcal{D} = (V, E)$ be a DAG. The DAG $\mathcal{D}^{\mathrm{tr}} = (V, E^{\mathrm{tr}})$ is the *transitive reduction* of $\mathcal{D}$ if the following holds:

(a) $\mathcal{D}^{\mathrm{tr}}$ has a path from node j to node i if and only if $\mathcal{D}$ has a path from j to i, and
(b) there is no graph with less edges than $\mathcal{D}^{\mathrm{tr}}$ satisfying condition (a).

Since we work with finite DAGs throughout, the transitive reduction is unique and is also a subgraph of the original DAG. The transitive reduction of a DAG can be obtained by successively examining its edges, in any order, and deleting an edge $k \to i$, if it contains a path from k to i which does not include this edge. For these properties and further details see, e.g., Aho et al. [1]. In what follows, we need the notion of $\mathrm{pa}^{\mathrm{tr}}(i)$, the parents of i in $\mathcal{D}^{\mathrm{tr}}$.

We present necessary and sufficient conditions on B to be the ML coefficient matrix of a recursive ML model on $\mathcal{D}$.

Theorem 4.2. *Let $\mathcal{D} = (V, E)$ be a* DAG *with reachability matrix R and* $\mathbf{X}$ *a* ML *model as in* (1.4) *with* ML *coefficient matrix B such that* $\mathrm{sgn}(B) = R$. *Define*

$$A := \mathrm{diag}(b_{11}, \ldots, b_{dd}) \quad \textit{and} \quad A_0 := \left(\frac{b_{ij}}{b_{ii}} \mathbf{1}_{\mathrm{pa}(j)}(i) \right)_{d \times d}.$$

Then **X** *is a recursive* ML *model on $\mathcal{D}$ if and only if the following fixed point equation holds*:

$$B = A \vee B \odot A_0, \tag{4.1}$$

where $\odot$ is the matrix product from (2.2). *In this case*,

$$X_i = \bigvee_{k \in \mathrm{pa}(i)} \frac{b_{ki}}{b_{kk}} X_k \vee b_{ii} Z_i, \qquad i = 1, \dots, d.$$

Proof. First, we investigate the fixed point equation (4.1) and compute the jith entry of $B \odot A_0$. By definition, together with $\mathrm{sgn}(B) = R$, it is equal to

$$\bigvee_{k=1}^{d} \frac{b_{jk} b_{ki}}{b_{kk}} \mathbf{1}_{\mathrm{pa}(i)}(k) = \bigvee_{k \in \mathrm{De}(j) \cap \mathrm{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}}.$$

We have $\mathrm{De}(j) \cap \mathrm{pa}(i) = \varnothing$ for $j \in V \setminus \mathrm{an}(i)$ and $\mathrm{De}(j) \cap \mathrm{pa}(i) = \mathrm{de}(j) \cap \mathrm{pa}(i)$ for $j \in \mathrm{an}(i) \setminus \mathrm{pa}(i)$. Moreover, for $j \in \mathrm{pa}^{\mathrm{tr}}(i)$, using that $\mathrm{de}(j) \cap \mathrm{pa}(i) = \varnothing$, we obtain $\mathrm{De}(j) \cap \mathrm{pa}(i) = \{i\}$. Thus, taking also the matrix A into account, (4.1) is equivalent to

$$b_{ji} = \begin{cases} 0, & \text{if } j \in V \setminus \mathrm{An}(i), \\ b_{ii}, & \text{if } j = i, \\ \displaystyle\bigvee_{k \in \mathrm{de}(j) \cap \mathrm{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}}, & \text{if } j \in \mathrm{an}(i) \setminus \mathrm{pa}(i), \\ b_{ji} \vee \displaystyle\bigvee_{k \in \mathrm{de}(j) \cap \mathrm{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}}, & \text{if } j \in \mathrm{pa}(i) \setminus \mathrm{pa}^{\mathrm{tr}}(i), \\ b_{ji}, & \text{if } j \in \mathrm{pa}^{\mathrm{tr}}(i) \end{cases}$$

for all $i, j \in V$. In this equation, the first row is automatically satisfied, since $R = \mathrm{sgn}(B)$, also the second and the last one hold trivially. To summarize, the fixed point equation (4.1) is satisfied if and only if for all $i \in V$ the following identities hold:

$$b_{ji} = \bigvee_{k \in \mathrm{de}(j) \cap \mathrm{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \qquad \text{for all } j \in \mathrm{an}(i) \setminus \mathrm{pa}(i), \tag{4.2}$$

$$b_{ji} = b_{ji} \vee \bigvee_{k \in \mathrm{de}(j) \cap \mathrm{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \qquad \text{for all } j \in \mathrm{pa}(i) \setminus \mathrm{pa}^{\mathrm{tr}}(i). \tag{4.3}$$

Thus, it suffices to show that **X** is a recursive ML model on $\mathcal{D}$ if and only if (4.2) and (4.3) hold for all $i \in V$.

First, assume that **X** is a recursive ML model on $\mathcal{D}$, and let $i \in V$ and $j \in \mathrm{an}(i)$. Since every path from j to i passes through at least one parent node of i, there must be a max-weighted path from j to i passing through some node in $\mathrm{pa}(i)$. Using (3.1) with $U = \mathrm{pa}(i)$ and noting that $j \in \operatorname{De}(j) \cap U \cap \operatorname{An}(i) = \operatorname{De}(j) \cap \operatorname{pa}(i)$, we find for $j \in \operatorname{an}(i) \setminus \operatorname{pa}(i)$ equation (4.2) and for $j \in \operatorname{pa}(i) \setminus \operatorname{pa}^{\mathrm{tr}}(i)$ equation (4.3).

For the converse statement, assume that (4.2) and (4.3) hold. For $j \in \operatorname{pa}^{\mathrm{tr}}(i)$ we have $\operatorname{de}(j) \cap \operatorname{pa}(i) = \varnothing$, such that the right-hand side of (4.3) is equal to b_{ji}. Thus (4.3) holds for all $j \in \operatorname{pa}(i)$. Since $\operatorname{sgn}(B) = R$, we have $X_i = \bigvee_{j=1}^{d} b_{ji} Z_j = \bigvee_{j \in \operatorname{An}(i)} b_{ji} Z_j$. We split up the index set and use (4.2) in the first place and (4.3) for all $j \in \operatorname{pa}(i)$ in the second place to obtain

$$\begin{aligned}
X_i &= \bigvee_{j \in \operatorname{an}(i) \setminus \operatorname{pa}(i)} b_{ji} Z_j \vee \bigvee_{j \in \operatorname{pa}(i)} b_{ji} Z_j \vee b_{ii} Z_i \\
&= \bigvee_{j \in \operatorname{an}(i) \setminus \operatorname{pa}(i)} \bigvee_{k \in \operatorname{de}(j) \cap \operatorname{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j \\
&\quad \vee \bigvee_{j \in \operatorname{pa}(i)} b_{ji} Z_j \vee \bigvee_{j \in \operatorname{pa}(i)} \bigvee_{k \in \operatorname{de}(j) \cap \operatorname{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j \vee b_{ii} Z_i \\
&= \bigvee_{j \in \operatorname{an}(i)} \bigvee_{k \in \operatorname{de}(j) \cap \operatorname{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j \vee \bigvee_{j \in \operatorname{pa}(i)} b_{ji} Z_j \vee b_{ii} Z_i .
\end{aligned}$$

Interchanging the first two maximum operators by (A.1) yields

$$\begin{aligned}
X_i &= \bigvee_{k \in \operatorname{pa}(i)} \bigvee_{j \in \operatorname{an}(k)} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j \vee \bigvee_{k \in \operatorname{pa}(i)} b_{ki} Z_k \vee b_{ii} Z_i \\
&= \bigvee_{k \in \operatorname{pa}(i)} \frac{b_{ki}}{b_{kk}} \biggl(\bigvee_{j \in \operatorname{an}(k)} b_{jk} Z_j \vee b_{kk} Z_k \biggr) \vee b_{ii} Z_i \\
&= \bigvee_{k \in \operatorname{pa}(i)} \frac{b_{ki}}{b_{kk}} X_k \vee b_{ii} Z_i .
\end{aligned}$$

□

In the proof of Theorem 4.2 we have shown that, under the required conditions, the fixed point equation (4.1) holds if and only if (4.2) and (4.3) hold for all nodes. We summarize this in part (a) of the following corollary. Part (b) has also been verified in the proof of Theorem 4.2. The final statement is based on the fact that for $k \in \operatorname{pa}(i)$ we have $\operatorname{de}(k) \cap \operatorname{pa}(i) = \varnothing$ if and only if $k \in \operatorname{pa}^{\mathrm{tr}}(i)$.

Corollary 4.3. (a) *Assume the situation of Theorem* 4.2. *Then* **X** *is a recursive* ML *model on* $\mathcal{D}$ *if and only if for every* $i \in V$,

$$b_{ji} = \bigvee_{k \in \operatorname{de}(j) \cap \operatorname{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \quad \text{for all } j \in \operatorname{an}(i) \setminus \operatorname{pa}(i) \tag{4.4}$$

$$b_{ji} \geq \bigvee_{k \in \operatorname{de}(j) \cap \operatorname{pa}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \quad \text{for all } j \in \operatorname{pa}(i) \setminus \operatorname{pa}^{\mathrm{tr}}(i). \tag{4.5}$$

(b) *Let* **X** *be a recursive* ML *model on a* DAG $\mathcal{D}=(V,E)$ *with* ML *coefficient matrix* B. *Then for every* $i \in V$ *and* $k \in \mathrm{pa}(i)$,

$$b_{ki} \geq \bigvee_{l \in \mathrm{de}(k) \cap \mathrm{pa}(i)} \frac{b_{kl} b_{li}}{b_{ll}}.$$

Moreover, the right-hand side is equal to 0 *if and only if* $k \in \mathrm{pa}^{\mathrm{tr}}(i)$, *and in this case the inequality is strict.*

By (4.4) and (4.5) exactly those ML coefficients b_{ji} for $i \in V$ and $j \in \mathrm{an}(i)$, such that $j \to i$ is an edge in $\mathcal{D}^{\mathrm{tr}}$, do not have to meet any specific conditions apart from being positive.

In summary, given a DAG $\mathcal{D}$ with d nodes, both Theorem 4.2 and Corollary 4.3(a) characterize all ML coefficient matrices of any recursive ML model possible on $\mathcal{D}$ as all non-negative $d \times d$ matrices that are weighted reachability matrices of $\mathcal{D}$ and satisfy (4.1), equivalently (4.4) and (4.5). If we can verify these two properties for a non-negative $d \times d$ matrix B, then it is the ML coefficient matrix of a recursive ML model on $\mathcal{D}$, and weights in its representation (1.3) are given by $c_{ki} = \frac{b_{ki}}{b_{kk}}$ for $k \in \mathrm{pa}(i)$ and $c_{ii} = b_{ii}$.

5. Graph reduction for a recursive max-linear model

From Proposition 3.7, we know that every component of a recursive ML model **X** on a DAG $\mathcal{D}=(V,E)$ satisfies (1.3) on a subgraph of $\mathcal{D}$. These subgraphs, however, usually vary from one component to another. On the other hand, we know from Example 3.3 that the whole vector **X** may also be a recursive ML model on a subgraph of $\mathcal{D}$. This raises the question of finding the smallest subgraph of $\mathcal{D}$ such that **X** is a recursive ML model on this DAG. We define and characterize this minimum DAG before we point out its prominent role in the class of all DAGs representing **X** in the sense of (1.3).

Definition 5.1. Let **X** be a recursive ML model on a DAG $\mathcal{D}=(V,E)$ with ML coefficient matrix B. We call the DAG

$$\mathcal{D}^B = \left(V, E^B\right) := \left(V, \left\{(k,i) \in E : b_{ki} > \bigvee_{l \in \mathrm{de}(k) \cap \mathrm{pa}(i)} \frac{b_{kl} b_{li}}{b_{ll}}\right\}\right) \tag{5.1}$$

the *minimum max-linear* (ML) DAG *of* **X**.

We summarize some properties of $\mathcal{D}^B$ as follows.

Remark 5.2. (i) The minimum ML DAG $\mathcal{D}^B = (V, E^B)$ is a subgraph of the original DAG $\mathcal{D}=(V,E)$. Observe from Corollary 4.3(b) that the transitive reduction $\mathcal{D}^{\mathrm{tr}} = (V, E^{\mathrm{tr}})$ of $\mathcal{D}$ is also a subgraph of $\mathcal{D}$. In summary, we have $E^{\mathrm{tr}} \subseteq E^B \subseteq E$. This implies that the DAGs $\mathcal{D}^B$ and $\mathcal{D}$ have the same reachability matrix, which is $\mathrm{sgn}(B)$ by Remark 2.3(i).

(ii) By Theorem 3.10(b) the minimum ML DAG $\mathcal{D}^B$ contains exactly those edges $k \to i$ of $\mathcal{D}$, where no max-weighted path from k to i passes through some node in $\mathrm{pa}(i) \setminus \{k\}$. This means that $\mathcal{D}^B$ has an edge $k \to i$ if and only if it is the only max-weighted path from k to i in $\mathcal{D}$. The DAG $\mathcal{D}^B$ can be obtained from $\mathcal{D}$ by deleting an edge $k \to i$, if $\mathcal{D}$ contains a max-weighted path from k to i, which does not include this edge. Note the analogy to finding the transitive reduction $\mathcal{D}^{\mathrm{tr}}$ of $\mathcal{D}$ described below Definition 4.1. An algorithm is by comparison of ML coefficients and motivated by Corollary 4.3(b): for all $i \in V$ and $k \in \mathrm{pa}(i) \setminus \mathrm{pa}^{\mathrm{tr}}(i)$ remove the edge $k \to i$ from $\mathcal{D}$ if

$$b_{ki} = \bigvee_{l \in \mathrm{de}(k) \cap \mathrm{pa}(i)} \frac{b_{kl} b_{li}}{b_{ll}}.$$

The method described in Remark 5.2(ii) determines $\mathcal{D}^B$ from $\mathcal{D}$ and the ML coefficient matrix B. Indeed, we can also identify $\mathcal{D}^B$ directly from B without knowing $\mathcal{D}$.

Theorem 5.3. *Let* $\mathbf{X}$ *be a recursive* ML *model with* ML *coefficient matrix* B. *Then the minimum* ML *DAG of* $\mathbf{X}$ *can be represented as*

$$\mathcal{D}^B = \left(V, \left\{ (k,i) \in V \times V : k \neq i \textit{ and } b_{ki} > \bigvee_{\substack{l=1 \\ l \neq i,k}}^{d} \frac{b_{kl} b_{li}}{b_{ll}} \right\} \right); \tag{5.2}$$

in particular, $\mathcal{D}^B$ *is identifiable from* B.

Proof. Let $\mathcal{D}$ be a DAG, which represents $\mathbf{X}$ in the sense of (1.3). Such a DAG exists by the definition of a recursive ML model. We show that the edge set in (5.2) coincides with E^B as defined in (5.1). Assume first that (k,i) is contained in the edge set in (5.2). Since $\mathrm{sgn}(B)$ is the reachability matrix of $\mathcal{D}$ (cf. Remark 2.3(i)), we have

$$b_{ki} > \bigvee_{\substack{l=1 \\ l \neq i,k}}^{d} \frac{b_{kl} b_{li}}{b_{ll}} = \bigvee_{l \in \mathrm{de}(k) \cap \mathrm{an}(i)} \frac{b_{kl} b_{li}}{b_{ll}}. \tag{5.3}$$

Since the right-hand side of (5.3) is non-negative, we must have $b_{ki} > 0$ and, hence, $k \in \mathrm{an}(i)$. By Theorem 3.10(b) no max-weighted path from k to i passes through some node in $V \setminus \{i,k\}$. Thus the edge $k \to i$ must be the only max-weighted path from k to i and, hence, by Remark 5.2(ii) it must be an edge in E^B as in (5.1).

For the converse, let $(k,i) \in E^B$. Since by Remark 5.2(ii) this edge is the only max-weighted path from k to i, no max-weighted path passes through some node in $V \setminus \{i,k\}$. This is by Theorem 3.10(b) equivalent to (5.3) and (k,i) belongs to the edge set in (5.2). □

We characterize all DAGs and specify all weights such that $\mathbf{X}$ satisfies (1.3). The minimum ML DAG $\mathcal{D}^B$ of $\mathbf{X}$ will be the smallest DAG of this kind and has unique weights in representation (1.3) in the sense that all irrelevant weights are set to zero. We can add any edge $k \to i$ into

$\mathcal{D}^B$ with weight $c_{ki} \in (0, \frac{b_{ki}}{b_{kk}}]$ representing $\mathbf{X}$ again in the sense of (1.3) as long as the graph represents the same reachability relation as $\mathcal{D}^B$. As a consequence, to find B by a path analysis as described in (1.5) and (1.6) it suffices to know $\mathcal{D}^B$ and the weights in representation (1.3) relative to $\mathcal{D}^B$.

Theorem 5.4. *Let* $\mathbf{X}$ *be a recursive* ML *model with* ML *coefficient matrix* B. *Let further* $\mathcal{D}^B = (V, E^B)$ *be the minimum* ML DAG *of* $\mathbf{X}$ *and* $\mathrm{pa}^B(i)$ *the parents of node* i *in* $\mathcal{D}^B$.

(a) *The minimum* ML DAG $\mathcal{D}^B$ *of* $\mathbf{X}$ *is the* DAG *with the minimum number of edges such that* $\mathbf{X}$ *satisfies* (1.3). *The weights in* (1.3) *are uniquely given by* $c_{ii} = b_{ii}$ *and* $c_{ki} = \frac{b_{ki}}{b_{kk}}$ *for* $i \in V$ *and* $k \in \mathrm{pa}^B(i)$.

(b) *Every* DAG *with node set* V *that has at least the edges of* $\mathcal{D}^B$ *and the same reachability matrix as* $\mathcal{D}^B$ *represents* $\mathbf{X}$ *in the sense of* (1.3) *with weights given for all* $i \in V$ *by*

$$c_{ii} = b_{ii}, \qquad c_{ki} = \frac{b_{ki}}{b_{kk}} \qquad \textit{for } k \in \mathrm{pa}^B(i) \quad \textit{and}$$

$$c_{ki} \in \left(0, \frac{b_{ki}}{b_{kk}}\right] \qquad \textit{for } k \in \mathrm{pa}(i) \setminus \mathrm{pa}^B(i).$$

There are no further DAG*s and weights such that* $\mathbf{X}$ *has representation* (1.3).

Proof. (a) Let $\mathcal{D}$ be a DAG and c_{ki} for $i \in V$ and $k \in \mathrm{Pa}(i)$ weights such that $\mathbf{X}$ has representation (1.3). By Remark 5.2(i) $\mathcal{D}^B$ is a subgraph of $\mathcal{D}$.

First, we prove that $\mathbf{X}$ is a recursive ML model on $\mathcal{D}^B$ with weights c_{ki} for $i \in V$ and $k \in \mathrm{Pa}^B(i)$ by showing that all components of $\mathbf{X}$ coincide with those of the recursive ML submodel of $\mathbf{X}$ induced by $\mathcal{D}^B$ (see Definition 3.5). By Remark 3.6(iv), it suffices to verify for all $i \in V$ and $j \in \mathrm{an}(i)$ that $\mathcal{D}^B$ has one in $\mathcal{D}$ max-weighted path from j to i. Among all max-weighted paths from j to i in $\mathcal{D}$, let p be one with maximum length, and assume that p includes an edge, say $k \to l$, which is not contained in $\mathcal{D}^B$. The DAG $\mathcal{D}$ has by Remark 5.2(ii), however, a max-weighted path p_1 from k to l, which does not include the edge $k \to l$. Note that p_1 consists of more edges than the path $[k \to l]$. Thus by replacing in p the edge $k \to l$ by p_1 we obtain by Remark 3.4(iii) a max-weighted path from j to i consisting of more edges than p. Since this a contradiction to the fact that p has maximum length among all max-weighted paths from j to i, p must be in $\mathcal{D}^B$.

Since every edge $k \to i$ in $\mathcal{D}^B$ is by Remark 5.2(ii) the only max-weighted path from k to i in $\mathcal{D}$, the weights in (1.3) are uniquely given, and we have by Definition 3.1 and (1.5) that $b_{ki} = c_{kk}c_{ki} = b_{kk}c_{ki}$, which implies $c_{ki} = \frac{b_{ki}}{b_{kk}}$. For the same reason, there cannot be a DAG with less edges than $\mathcal{D}^B$ such that $\mathbf{X}$ has representation (1.3).

(b) First we show that $\mathbf{X}$ satisfies (1.3) relative to a DAG $\mathcal{D}$ with the properties and weights c_{ki} for $i \in V$ and $k \in \mathrm{Pa}(i)$ (the parents in $\mathcal{D}$). Note that the DAG $\mathcal{D}^B$ is a subgraph of $\mathcal{D}$ and both DAGs have the same reachability relation. Since $\mathbf{X}$ is by part (a) a recursive ML model on $\mathcal{D}^B$, we may use Corollary 3.13 with the ancestors in $\mathcal{D}^B$: for every $i \in V$ and $k \in \mathrm{pa}(i)$, since k

is an ancestor of i in $\mathcal{D}^B$ and $\frac{b_{ki}}{b_{kk}} \geq c_{ki}$, we have

$$X_i \geq \frac{b_{ki}}{b_{kk}} X_k \geq c_{ki} X_k.$$

With this we obtain from representation (1.3) of X_i relative to $\mathcal{D}^B$ that

$$X_i = \bigvee_{k \in \mathrm{pa}^B(i)} c_{ki} X_i \vee c_{ii} Z_i = \bigvee_{k \in \mathrm{pa}^B(i)} c_{ki} X_k \vee \bigvee_{k \in \mathrm{pa}(i) \setminus \mathrm{pa}^B(i)} c_{ki} X_k \vee c_{ii} Z_i,$$

which is (1.3) relative to $\mathcal{D}$.

It remains to show that there are no further DAGs and weights such that $\mathbf{X}$ has representation (1.3). From Remark 5.2(i), every DAG that represents $\mathbf{X}$ in the sense of (1.3) must have the same reachability matrix as $\mathcal{D}^B$ and must contain at least the edges of $\mathcal{D}^B$. By (1.5) and (1.6) the weights in representation (1.3) of $\mathbf{X}$ have to satisfy $c_{ki} \leq \frac{b_{ki}}{b_{kk}}$ for all $i \in V$ and $k \in \mathrm{pa}(i)$. The statement follows, since the weights c_{ki} are by part (a) uniquely with respect to $\mathcal{D}^B$. □

As explained before Theorem 5.4, we can add edges into $\mathcal{D}^B$, while keeping the same reachability relation and still having representation (1.3) for $\mathbf{X}$. In what follows, we will use the DAG with the maximum number of edges with this properties.

Definition 5.5. Let $\mathcal{D} = (V, E)$ be a DAG. The *transitive closure* $\mathcal{D}^{\mathrm{tc}} = (V, E^{\mathrm{tc}})$ of $\mathcal{D}$ is the DAG that has an edge $j \to i$ if and only if $\mathcal{D}$ has a path from j to i.

The transitive reduction is essentially the inverse operation of the transitive closure: for the transitive reduction one reduces the number of edges and for the transitive closure one adds edges, while maintaining the identical reachability relation. The transitive reduction of a DAG $\mathcal{D}$ is a subgraph of $\mathcal{D}$, and $\mathcal{D}$ is again a subgraph of the transitive closure. Moreover, all DAGs with the same reachability matrix have the same transitive reduction and the same transitive closure, and every node has in all such DAGs the same ancestors and descendants.

The following is a consequence of Theorem 5.4(b) and Remark 2.3(i).

Corollary 5.6. *The recursive* ML *model* $\mathbf{X}$ *is also a recursive* ML *model on the transitive closure of every* DAG *with reachability matrix* $\mathrm{sgn}(B)$.

We use this corollary to obtain necessary and sufficient conditions on a ML coefficient matrix B as in (1.4) to be the ML coefficient matrix of a recursive ML model. In contrast to Theorem 4.2 and Corollary 4.3(a), we do not require that B belongs to a specific given DAG.

Theorem 5.7. *Let* $\mathbf{X}$ *be a* ML *model as in* (1.4) *with* ML *coefficient matrix* B *such that* $\mathrm{sgn}(B)$ *is the reachability matrix of some* DAG. *Define*

$$A := \mathrm{diag}(b_{11}, \dots, b_{dd}), \qquad B_0 := \left(\frac{b_{ij}}{b_{ii}}\right)_{d \times d} \quad \textit{and} \quad A_0^{\mathrm{tc}} := B_0 - \mathrm{id}_{d \times d},$$

where $\mathrm{id}_{d\times d}$ *denotes the identity matrix. Then* $\mathbf{X}$ *is a recursive* ML *model if and only if the following fixed point equation holds*:

$$B = B \odot B_0, \qquad \textit{which is equivalent to } B = A \vee B \odot A_0^{\mathrm{tc}}, \tag{5.4}$$

where $\odot$ *is the matrix product from* (2.2).

Proof. Let $\mathcal{D}^{\mathrm{tc}}$ be the transitive closure of a DAG with node set $V = \{1, \dots, d\}$ and reachability matrix $\operatorname{sgn}(B)$. For $i \in V$ we denote by by $\mathrm{pa}(i)$ and $\mathrm{an}(i)$ the parents and ancestors of node i in $\mathcal{D}^{\mathrm{tc}}$, respectively, and observe from the definition of $\mathcal{D}^{\mathrm{tc}}$ that $\mathrm{an}(i) = \mathrm{pa}(i)$ for all $i \in V$.

First, we show that $\mathbf{X}$ is a recursive ML model if and only if the fixed point equation $B = A \vee B \odot A_0^{\mathrm{tc}}$ holds. By Corollary 5.6 $\mathbf{X}$ is a recursive ML model if and only if it is a recursive ML model on $\mathcal{D}^{\mathrm{tc}}$. By Theorem 4.2, it suffices to show that A_0^{tc} is equal to the weighted adjacency matrix $A_0 = (\frac{b_{ij}}{b_{ii}} \mathbf{1}_{\mathrm{pa}(j)}(i))_{d\times d}$. Since B_0 is a weighted reachability matrix of $\mathcal{D}^{\mathrm{tc}}$, we obtain

$$A_0^{\mathrm{tc}} = B_0 - \mathrm{id}_{d\times d} = \left(\frac{b_{ij}}{b_{ii}} \mathbf{1}_{\mathrm{an}(j)}(i)\right)_{d\times d} = \left(\frac{b_{ij}}{b_{ii}} \mathbf{1}_{\mathrm{pa}(j)}(i)\right)_{d\times d}.$$

It remains to show that $B \odot B_0 = A \vee B \odot A_0^{\mathrm{tc}}$. By the definition of the matrix product $\odot$ the jith entry of $A \vee B \odot A_0^{\mathrm{tc}}$ is equal to

$$\begin{aligned} b_{ji}\mathbf{1}_{\{i\}}(j) \vee \bigvee_{k=1}^{d} b_{jk}\left(\frac{b_{ki}}{b_{kk}} - \mathbf{1}_{\{i\}}(k)\right) &= b_{ji}\mathbf{1}_{\{i\}}(j) \vee \bigvee_{\substack{k=1\\k\neq i}}^{d} \frac{b_{jk}b_{ki}}{b_{kk}} \\ &= b_{ji}\mathbf{1}_{\{i\}}(j) \vee \bigvee_{\substack{k=1\\k\neq i,j}}^{d} \frac{b_{jk}b_{ki}}{b_{kk}} \vee b_{ji}\mathbf{1}_{V\setminus\{i\}}(j) \\ &= \bigvee_{k=1}^{d} \frac{b_{jk}b_{ki}}{b_{kk}}, \end{aligned}$$

which is the jith entry of the matrix $B \odot B_0$. □

A non-negative symmetric matrix is by Theorem 5.7 a ML coefficient matrix of a recursive ML model if and only if it is a weighted reachability matrix of a DAG and satisfies (5.4). Assume that we have verified these properties for a matrix B. In order to find now all recursive ML models which have ML coefficient matrix B we can first use (5.2) to derive the minimum ML DAG $\mathcal{D}^B$ from B and then Theorem 5.4(b) to find all DAGs and weights as in (1.3) such that (1.6) holds.

6. Backward and forward information in a recursive ML model

We investigate relations between the components of a recursive ML model $\mathbf{X}$ on a DAG $\mathcal{D} = (V, E)$ with ML coefficient matrix B. More precisely, we apply our previous results to identify those components of $\mathbf{X}$ which provide maximal information on some other component.

We know already from Corollary 3.13 that $X_i \le \frac{b_{ii}}{b_{il}} X_l$ for all $i \in V$ and $l \in \mathrm{De}(i)$ so that for some node set $U \subseteq V$ and all $i \in V$,

$$\bigvee_{j \in \mathrm{An}(i) \cap U} \frac{b_{ji}}{b_{jj}} X_j \le X_i \le \bigwedge_{l \in \mathrm{De}(i) \cap U} \frac{b_{ii}}{b_{il}} X_l. \tag{6.1}$$

The values of the bounds in (6.1) can often be found as the maximum and minimum over a smaller number of nodes in U. We illustrate this by the following example.

Example 6.1 (Continuation of Examples 2.1 and 3.8: bounds). For $U = \{1, 2\}$ and $i = 4$ we find by (6.1) the lower bound

$$\frac{b_{14}}{b_{11}} X_1 \vee \frac{b_{24}}{b_{22}} X_2 \le X_4. \tag{6.2}$$

We discuss the lower bound and distinguish between two cases.

First, assume that the path $[1 \to 2 \to 4]$ is max-weighted, which is by Theorem 3.10(a) equivalent to $b_{14} = \frac{b_{12} b_{24}}{b_{22}}$. From Corollary 3.13 or (6.1) we obtain

$$\frac{b_{12}}{b_{11}} X_1 \le X_2, \quad \text{equivalently} \quad \frac{b_{14}}{b_{11}} X_1 \le \frac{b_{24}}{b_{22}} X_2.$$

Therefore, the lower bound of X_4 in (6.2) is always $\frac{b_{24}}{b_{22}} X_2$.

Now assume that the path $[1 \to 2 \to 4]$ is not max-weighted. Since this is the only path from 1 to 4 passing through node 2, this is by Theorem 3.10(b) equivalent to $b_{14} > \frac{b_{12} b_{24}}{b_{22}}$. From the max-linear representation (2.1) of X_1 and X_2 we have $\frac{b_{24}}{b_{22}} X_2 < \frac{b_{14}}{b_{11}} X_1$ if and only if

$$\frac{b_{12} b_{24}}{b_{22}} Z_1 \vee b_{24} Z_2 < b_{14} Z_1, \quad \text{equivalently} \quad b_{24} Z_2 < b_{14} Z_1.$$

The event $\{b_{24} Z_2 < b_{14} Z_1\}$ has positive probability, since Z_1 and Z_2 are independent with support $\mathbb{R}_+$, giving $\frac{b_{14}}{b_{11}} X_1$ as lower bound. But also the event $\{\frac{b_{14}}{b_{11}} X_1 \le \frac{b_{24}}{b_{22}} X_2\}$ has positive probability, giving the lower bound $\frac{b_{24}}{b_{22}} X_2$. Thus, only in the first case the number of nodes in the lower bound in (6.1) can be reduces.

We will find that a node $j \in \mathrm{An}(i) \cap U$ is relevant for the lower bound in (6.1) if no max-weighted path from j to i passes through some other node in U. Observe that this includes the observation made in Example 6.1. The nodes in the upper bound of (6.1) have a similar characterization. We present a formal definition of these particular ancestors and descendants, characterize them below in Lemma 6.3, and give an example afterwards.

Definition 6.2. Let $\mathbf{X}$ be a recursive ML model on a DAG $\mathcal{D} = (V, E)$, $U \subseteq V$ and $i \in V$.

(a) We call a node $j \in \mathrm{An}(i) \cap U$ *lowest max-weighted ancestor of i in U*, if no max-weighted path from j to i passes through some node in $U \setminus \{j\}$. We denote the set of the lowest max-weighted ancestors of i in U by $\mathrm{An}^U_{\mathrm{low}}(i)$.

(b) We call a node $l \in \mathrm{De}(i) \cap U$ *highest max-weighted descendant of i in U*, if no max-weighted path from i to l passes through some node in $U \setminus \{l\}$. We denote the set of the highest max-weighted descendants of i in U by $\mathrm{De}^U_{\mathrm{high}}(i)$.

For $i \in U$ we find that the only lowest max-weighted ancestor and the only highest max-weighted descendant of i in U is the node i itself. For $i \in U^c = V \setminus U$ a simple characterization of $\mathrm{An}^U_{\mathrm{low}}(i)$ and $\mathrm{De}^U_{\mathrm{high}}(i)$ is given next; this allows us to identify these nodes via the ML coefficient matrix of $\mathbf{X}$.

Lemma 6.3. *Let $\mathbf{X}$ be a recursive* ML *model on a* DAG $\mathcal{D} = (V, E)$, $U \subseteq V$ *and* $i \in V$.

(a) *If $i \in U$, then* $\mathrm{An}^U_{\mathrm{low}}(i) = \mathrm{De}^U_{\mathrm{high}}(i) = \{i\}$.

(b) *If $i \in U^c$, then*

$$\mathrm{An}^U_{\mathrm{low}}(i) = \left\{ j \in \mathrm{an}(i) \cap U : b_{ji} > \bigvee_{k \in \mathrm{de}(j) \cap U \cap \mathrm{an}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \right\}, \tag{6.3}$$

$$\mathrm{De}^U_{\mathrm{high}}(i) = \left\{ l \in \mathrm{de}(i) \cap U : b_{il} > \bigvee_{k \in \mathrm{de}(i) \cap U \cap \mathrm{an}(l)} \frac{b_{ik} b_{kl}}{b_{kk}} \right\}. \tag{6.4}$$

Proof. (a) follows immediately from the definition.

(b) Since $i \in U^c$, we have by Definition 6.2(a) that $\mathrm{An}^U_{\mathrm{low}}(i) \subseteq \mathrm{an}(i) \cap U$. For $j \in \mathrm{an}(i) \cap U$ we know from Theorem 3.10(b) that no max-weighted path from j to i passes through some node in $U \setminus \{j\}$ if and only if

$$b_{ji} > \bigvee_{k \in \mathrm{de}(j) \cap U \cap \mathrm{an}(i)} \frac{b_{jk} b_{ki}}{b_{kk}},$$

where we have used that $i \in U^c$. Similarly, we obtain (6.4). □

Example 6.4 (Continuation of Examples 2.1, 3.8, and 6.1: $\mathrm{An}^U_{\mathrm{low}}(4)$). In order to find the lowest max-weighted ancestors of node 4 in $U = \{1, 2\}$, first observe that the only max-weighted path $[2 \to 4]$ from 2 to 4 does not pass through any node in $U \setminus \{2\}$. Therefore, we have by Definition 6.2(a) that $2 \in \mathrm{An}^U_{\mathrm{low}}(4)$. For node 1 we consider – as in Example 6.1 – two cases and use (6.3):

(1) If $b_{14} = \frac{b_{12} b_{24}}{b_{22}}$, then $\mathrm{An}^U_{\mathrm{low}}(4) = \{2\}$.

(2) If $b_{14} > \frac{b_{12} b_{24}}{b_{22}}$, then $\mathrm{An}^U_{\mathrm{low}}(4) = \{1, 2\}$.

Comparing this with Example 6.1 shows that the lower bound of X_4 in (6.2) is always realized by some lowest max-weighted ancestor of node 4 in U.

We prove that the lower and upper bounds in (6.1) are always realized by some lowest max-weighted ancestor and highest max-weighted descendant in U, respectively. For the lower bound, this is based on the fact that between all nodes and their ancestors in U there is always a max-weighted path, which contains a lowest max-weighted ancestor in U. For the upper bound, we use the existence of a max-weighted path between all nodes and their descendants in U that passes through some highest max-weighted descendant in U. Before we state the modified lower and upper bounds in Proposition 6.6, we provide a useful characterization for a path analysis, which includes these statements.

Lemma 6.5. *Let* $\mathbf{X}$ *be a recursive* ML *model on a* DAG $\mathcal{D} = (V, E)$. *Furthermore, let* $U \subseteq V$, $i \in V$, $j \in \mathrm{an}(i)$, *and* $l \in \mathrm{de}(i)$.

(a) *$\mathcal{D}$ has a max-weighted path from j to i passing through some node in U if and only if it has a max-weighted path from j to i passing through some node in* $\mathrm{An}^U_{\mathrm{low}}(i)$.

(b) *$\mathcal{D}$ has a max-weighted path from i to l passing through some node in U if and only if it has a max-weighted path from i to l passing through some node in* $\mathrm{De}^U_{\mathrm{high}}(i)$.

Proof. We only show (a); part (b) can be proved analogously. Assume that a max-weighted path from j to i passes through some node in $\mathrm{An}^U_{\mathrm{low}}(i)$. Since $\mathrm{An}^U_{\mathrm{low}}(i) \subseteq U$, there is obviously also a max-weighted path from j to i that passes through some node in U.

For the converse, we may assume that $i \in U^c$, since by Lemma 6.3(a) $\mathrm{An}^U_{\mathrm{low}}(i) = \{i\}$ for $i \in U$ and hence every max-weighted path contains a node in $\mathrm{An}^U_{\mathrm{low}}(i)$. Among all max-weighted paths from j to i let p be one with maximum number of nodes in U. Denote by k_1 the lowest node on p contained in U; i.e., the subpath of p from k_1 to i contains no other node of U. Assume that $k_1 \notin \mathrm{An}^U_{\mathrm{low}}(i)$. Since $k_1 \in U$ and $i \in U^c$, there is by Definition 6.2(a) a max-weighted path p_1 from k_1 to i that passes through some node $k_2 \in U$ with $k_2 \neq k_1$. Thus, by replacing in p the subpath from k_1 to i by p_1 we obtain by Remark 3.4(iii) a max-weighted path from j to i containing more nodes in U than p. This is however a contradiction. Hence, $k_1 \in \mathrm{An}^U_{\mathrm{low}}(i)$, and p is a max-weighted path from j to i that passes through some node in $\mathrm{An}^U_{\mathrm{low}}(i)$. □

Proposition 6.6. *Let* $\mathbf{X}$ *be a recursive* ML *model on a* DAG $\mathcal{D} = (V, E)$ *with* ML *coefficient matrix B. Let $U \subseteq V$ and $i \in V$. Then*

$$\begin{aligned} \bigvee_{j \in \mathrm{An}(i) \cap U} \frac{b_{ji}}{b_{jj}} X_j &= \bigvee_{j \in \mathrm{An}^U_{\mathrm{low}}(i)} \frac{b_{ji}}{b_{jj}} X_j \quad \textit{and} \\ \bigwedge_{l \in \mathrm{De}(i) \cap U} \frac{b_{ii}}{b_{il}} X_l &= \bigwedge_{l \in \mathrm{De}^U_{\mathrm{high}}(i)} \frac{b_{ii}}{b_{il}} X_l. \end{aligned} \tag{6.5}$$

Proof. Note from Definition 6.2(a) that $\mathrm{An}^U_{\mathrm{low}}(i) \subseteq \mathrm{An}(i) \cap U$. To show the first equality take some $k \in (\mathrm{An}(i) \cap U) \setminus \mathrm{An}^U_{\mathrm{low}}(i)$. Observe from Lemma 6.3(a) that $k \neq i$ and, hence, $k \in \mathrm{an}(i) \cap U$. By Lemma 6.5(a) there must be a max-weighted path from k to i, which passes through some node $j \in \mathrm{An}^U_{\mathrm{low}}(i)$. By (3.1) and Corollary 3.13, we obtain

$$\frac{b_{ki}}{b_{kk}} X_k = \frac{b_{kj} b_{ji}}{b_{kk} b_{jj}} X_k \leq \frac{b_{ji}}{b_{jj}} X_j. \tag{6.6}$$

Since for all $k \in (\mathrm{An}(i) \cap U) \setminus \mathrm{An}^U_{\mathrm{low}}(i)$ there exists some $j \in \mathrm{An}^U_{\mathrm{low}}(i)$ such that (6.6) holds, the first equality of (6.5) follows. The second equality may be verified analogously. □

So far, for every component of $\mathbf{X}$, we have identified a lower and upper bound in terms of the components of $\mathbf{X}_U = (X_l, l \in U)$. However, we cannot say anything about the quality of the bounds. For instance, we do not know in which situation a component attains one of the bounds. We clarify this by writing all components of $\mathbf{X}$ as max-linear functions of the components of $\mathbf{X}_U$ and certain noise variables. There are many such representations, since we can always include non-relevant ancestral components with appropriate ML coefficients as we know from Corollary 3.13. To find the relevant components of $\mathbf{X}_U$ and noise variables, we focus on those with the minimum number of components of $\mathbf{X}_U$ and the minimum number of noise variables. For $i \in V$, we denote by $\mathrm{an}^U_{\mathrm{nmw}}(i)$ the set of all $j \in \mathrm{an}(i)$ such that no max-weighted path from j to i passes through some node in U. By Theorem 3.10(b), we have

$$\mathrm{an}^U_{\mathrm{nmw}}(i) = \left\{ j \in \mathrm{an}(i) : b_{ji} > \bigvee_{k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \right\}. \tag{6.7}$$

Since $j \in \mathrm{an}(i) \setminus \mathrm{an}^U_{\mathrm{nmw}}(i)$ if and only if there is a max-weighted path from j to i passing through some node in U, we obtain from Theorem 3.10(a)

$$\mathrm{an}(i) \setminus \mathrm{an}^U_{\mathrm{nmw}}(i) = \left\{ j \in \mathrm{an}(i) : b_{ji} = \bigvee_{k \in \mathrm{De}(j) \cap U \cap \mathrm{An}(i)} \frac{b_{jk} b_{ki}}{b_{kk}} \right\}. \tag{6.8}$$

Theorem 6.7. *Let* $\mathbf{X}$ *be a recursive* ML *model on a* DAG $\mathcal{D}$ *with* ML *coefficient matrix* B, *and let* $U \subseteq V$. *Furthermore, let* $\mathrm{An}^U_{\mathrm{low}}(i)$ *be the lowest max-weighted ancestors of node* i *in* U *as in Definition* 6.2(a), *and define* $\mathrm{An}^U_{\mathrm{nmw}}(i) := (\mathrm{an}^U_{\mathrm{nmw}}(i) \cup \{i\}) \cap U^c$. *Then for every* $i \in V$,

$$X_i = \bigvee_{k \in \mathrm{An}^U_{\mathrm{low}}(i)} \frac{b_{ki}}{b_{kk}} X_k \vee \bigvee_{j \in \mathrm{An}^U_{\mathrm{nmw}}(i)} b_{ji} Z_j. \tag{6.9}$$

This representation of X_i *as a max-linear function of the components of* $\mathbf{X}_U$ *and noise variables involves the minimum number of components of* $\mathbf{X}_U$ *and the minimum number of noise variables.*

Proof. We distinguish between nodes $i \in U$ and $i \in U^c$. For $i \in U$ we know from Lemma 6.3(a) that $\mathrm{An}^U_{\mathrm{low}}(i) = \{i\}$. Furthermore, we have $\mathrm{An}^U_{\mathrm{nmw}}(i) = \varnothing$, since $i \in U$ and every path, hence also every max-weighted path, from some $j \in \mathrm{an}(i)$ to i passes through some node in U, namely i itself. Thus we obtain (6.9). The second statement is obvious.

Now assume that $i \in U^c$, and note that in this case $\mathrm{An}^U_{\mathrm{nmw}}(i) = \mathrm{an}^U_{\mathrm{nmw}}(i) \cup \{i\}$. Applying the first equality in (6.5) and (2.1) as well as (A.2) in a second step to interchange the first two maximum operators, we have

$$\begin{aligned}\bigvee_{k \in \mathrm{An}^U_{\mathrm{low}}(i)} \frac{b_{ki}}{b_{kk}} X_k &= \bigvee_{k \in \mathrm{an}(i) \cap U} \frac{b_{ki}}{b_{kk}} \Big(\bigvee_{j \in \mathrm{An}(k)} b_{jk} Z_j \Big) \\ &= \bigvee_{j \in \mathrm{an}(i)} \bigvee_{k \in \mathrm{De}(j) \cap \mathrm{an}(i) \cap U} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j.\end{aligned} \tag{6.10}$$

We split up the set $\mathrm{an}(i)$ into $\mathrm{an}^U_{\mathrm{nmw}}(i)$ and $\mathrm{an}(i) \setminus \mathrm{an}^U_{\mathrm{nmw}}(i)$ as well as the set $\mathrm{An}^U_{\mathrm{nmw}}(i)$ into $\mathrm{an}^U_{\mathrm{nmw}}(i)$ and $\{i\}$ to obtain that the right-hand side of (6.9) is equal to

$$\begin{aligned}&\bigvee_{j \in \mathrm{an}(i) \setminus \mathrm{an}^U_{\mathrm{nmw}}(i)} \bigvee_{k \in \mathrm{De}(j) \cap \mathrm{an}(i) \cap U} \frac{b_{jk} b_{ki}}{b_{kk}} Z_j \\ &\vee \bigvee_{j \in \mathrm{an}^U_{\mathrm{nmw}}(i)} \Big(\bigvee_{k \in \mathrm{De}(j) \cap \mathrm{an}(i) \cap U} \frac{b_{jk} b_{ki}}{b_{kk}} \vee b_{ji} \Big) Z_j \vee b_{ii} Z_i.\end{aligned}$$

Noting that $i \in U^c$ when using (6.8) and (6.7) yields

$$\bigvee_{j \in \mathrm{an}(i) \setminus \mathrm{an}^U_{\mathrm{nmw}}(i)} b_{ji} Z_j \vee \bigvee_{j \in \mathrm{an}^U_{\mathrm{nmw}}(i)} b_{ji} Z_j \vee b_{ii} Z_i = \bigvee_{j \in \mathrm{An}(i)} b_{ji} Z_j = X_i.$$

In order to verify that for $i \in U^c$ (6.9) is the representation of X_i with the minimum number of components of $\mathbf{X}_U$ and the minimum number of noise variables, we prove that each term on the right-hand side of (6.9) has to appear, since otherwise some noise variable Z_j in representation (2.1) of X_i would have a weight strictly less than b_{ji}. We compare the noise variables on the right-hand sides of (6.9) and (6.10). Since $b_{ii} Z_i$ does not appear in (6.10), it has to appear in (6.9). For $j \in \mathrm{an}^U_{\mathrm{nmw}}(i)$ it follows from (6.7) that if Z_j appears in (6.10), then with a coefficient strictly less than b_{ji}. The maximum over $\mathrm{An}^U_{\mathrm{nmw}}(i)$ must therefore appear in (6.9). Definition 6.2(a) implies that no max-weighted path from $j \in \mathrm{An}^U_{\mathrm{low}}(i)$ to i passes through some node in $\mathrm{de}(j) \cap \mathrm{an}(i) \cap U$. Thus observe from (6.10) and (3.2) that only the term $\frac{b_{ji}}{b_{jj}} X_j$ provides Z_j with the weight b_{ji} in (6.9) and the term $\frac{b_{ji}}{b_{jj}} X_j$ has to appear in (6.9). □

We use Theorem 6.7 to obtain for every component X_i of $\mathbf{X}$ a minimal representation in terms of the components of $\mathbf{X}_{\mathrm{pa}(i)}$ and noise variables.

Corollary 6.8. *Let $\mathcal{D}^B$ be the minimum* ML DAG *of* $\mathbf{X}$ *as in Definition 5.1 with parents* $\mathrm{pa}^B(i)$ *of node i in $\mathcal{D}^B$. Then for all $i \in V$ we have* $\mathrm{An}^{\mathrm{pa}(i)}_{\mathrm{low}}(i) = \mathrm{pa}^B(i)$ *and*

$$X_i = \bigvee_{k \in \mathrm{pa}^B(i)} \frac{b_{ki}}{b_{kk}} X_k \vee b_{ii} Z_i = \bigvee_{k \in \mathrm{pa}^B(i)} c_{ki} X_k \vee c_{ii} Z_i. \tag{6.11}$$

Proof. Recall from (5.1) that

$$\mathrm{pa}^B(i) = \left\{ k \in \mathrm{pa}(i) : b_{ki} > \bigvee_{l\,\mathrm{de}(k)\cap \mathrm{pa}(i)} \frac{b_{kl}b_{li}}{b_{ll}} \right\},$$

and observe from this and (6.3) that $\mathrm{An}_{\mathrm{low}}^{\mathrm{pa}(i)}(i) = \mathrm{pa}^B(i)$. Since every path from $j \in \mathrm{an}(i)$ to i passes through some node in $\mathrm{pa}(i)$, there is always a max-weighted path from j to i containing some node of $\mathrm{pa}(i)$. Hence, $\mathrm{An}_{\mathrm{nmw}}^{\mathrm{pa}(i)}(i) = (\mathrm{an}_{\mathrm{nmw}}^{\mathrm{pa}(i)}(i) \cup \{i\}) \cap (\mathrm{pa}(i))^c = \{i\}$. Thus, we obtain from (6.9) the first equality in (6.11). For the second, recall from Theorem 5.4(a) that $b_{ii} = c_{ii}$ and $\frac{b_{ki}}{b_{kk}} = c_{ki}$ for $k \in \mathrm{pa}^B(i)$. □

Remark 6.9. Representation (6.11) complements Theorem 5.4(a); we find again that the minimum ML DAG $\mathcal{D}^B$ yields the minimal representation of $\mathbf{X}$ as a recursive ML model.

The following example illustrates and discusses representation (6.9).

Example 6.10 (Continuation of Examples 2.1, 3.8, 6.1, and 6.4: minimal representation of X_4 by X_1, X_2 and X_2). We consider again $U = \{1, 2\}$ and $i = 4$. Obviously, there are max-weighted paths from 1 and 2 to 4 passing through some node in $U = \{1, 2\}$. Hence, $1, 2 \in \mathrm{an}(4) \setminus \mathrm{an}_{\mathrm{nmw}}^U(4)$. Since no max-weighted path from 3 to 4 passes through 1 or 2, we have $\mathrm{An}_{\mathrm{nmw}}^U(4) = (\mathrm{an}_{\mathrm{nmw}}^U(4) \cup \{4\}) \cap U^c = \{3, 4\}$. In Example 6.4, we have already determined the set $\mathrm{An}_{\mathrm{low}}^U(4)$ depending on the ML coefficients. Thus, we distinguish again between two cases:

(1) If $b_{14} = \frac{b_{12}b_{24}}{b_{22}}$, then $X_4 = \frac{b_{24}}{b_{22}} X_2 \vee b_{34}Z_3 \vee b_{44}Z_4$.

We want to remark that the conditional independence properties of $\mathbf{X}$ are reflected in this representation: from Example 3.8 we know that $X_1 \perp\!\!\!\perp X_4 \mid X_2$, if the path $[1 \to 2 \to 3]$ is max-weighted, which is the case here. So it is obvious that X_1 does not need to appear in the minimal representation of X_4 as max-linear function of X_1 and X_2.

(2) If $b_{14} > \frac{b_{12}b_{24}}{b_{22}}$, then $X_4 = \frac{b_{14}}{b_{11}} X_1 \vee \frac{b_{24}}{b_{22}} X_2 \vee b_{34}Z_3 \vee b_{44}Z_4$.

In particular, $\frac{b_{14}}{b_{11}} X_1 > \frac{b_{24}}{b_{22}} X_2$ is possible with positive probability; in (1) this is not possible (see Example 6.1).

For $U = \{2\}$ and $i = 4$ we have $\mathrm{An}_{\mathrm{low}}^U(4) = \{2\}$. Similarly as above we obtain that $2 \in \mathrm{an}(4) \setminus \mathrm{an}_{\mathrm{nmw}}^U(4)$ and $3, 4 \in \mathrm{An}_{\mathrm{nmw}}^U(4)$. It remains to discuss node 1, which gives rise to the same two cases as above:

(1) If the path $[1 \to 2 \to 4]$ is max-weighted, then $X_4 = \frac{b_{24}}{b_{22}} X_2 \vee b_{34}Z_3 \vee b_{44}Z_4$.

(2) If the path $[1 \to 2 \to 4]$ is not max-weighted, then $X_4 = \frac{b_{24}}{b_{22}} X_2 \vee b_{14}Z_1 \vee b_{34}Z_3 \vee b_{44}Z_4$.

Such minimal representations become relevant, when $\mathbf{X}$ is partially observed. If, for example, X_2 is observed and B is known, then the prediction problem of X_4 can be solved by (conditional) simulation of the relevant noise variables and direct computation of X_4. In case (1) we need to simulate independent Z_3, Z_4, whereas in case (2) additionally Z_1 has to be simulated conditioned on X_2. We will discuss such prediction problems in a subsequent paper.

Appendix A: An auxiliary lemma

Lemma A.1. *Let $\mathcal{D} = (V, E)$ be a* DAG *and $U \subseteq V$. For non-negative functions $a(i, j, k)$, $i, j, k \in V$, we have for all $i \in V$,*

$$\bigvee_{k\in\mathrm{pa}(i)} \bigvee_{j\in\mathrm{an}(k)} a(i, j, k) = \bigvee_{j\in\mathrm{an}(i)} \bigvee_{k\in\mathrm{de}(j)\cap\mathrm{pa}(i)} a(i, j, k), \tag{A.1}$$

$$\bigvee_{k\in\mathrm{an}(i)\cap U} \bigvee_{j\in\mathrm{An}(k)} a(i, j, k) = \bigvee_{j\in\mathrm{an}(i)} \bigvee_{k\in\mathrm{De}(j)\cap\mathrm{an}(i)\cap U} a(i, j, k). \tag{A.2}$$

Proof. Since we take maxima, we only have to prove that each combination of nodes (k, j) on the left-hand side appears also on the right-hand side and vice versa. In order to prove (A.1), it suffices to show that

$$k \in \mathrm{pa}(i) \text{ and } j \in \mathrm{an}(k) \quad \text{if and only if} \quad j \in \mathrm{an}(i) \text{ and } k \in \mathrm{de}(j) \cap \mathrm{pa}(i).$$

By observing that $\mathrm{an}(\mathrm{pa}(i)) \subseteq \mathrm{an}(i)$ and $j \in \mathrm{an}(k)$ if and only if $k \in \mathrm{de}(j)$ this equivalence is obvious. Equation (A.2) is proved in the same way. □

Acknowledgements

We thank Steffen Lauritzen for fruitful discussions and his constructive suggestions, which improved our manuscript. Nadine Gissibl had the pleasure of spending two months at the Seminar for Statistics at ETH Zurich. She wants to thank all colleagues there for a very pleasant time. She also gratefully acknowledges support from the TUM Graduate School's *International School of Applied Mathematics*.

References

[1] Aho, A.V., Garey, M.R. and Ullman, J.D. (1972). The transitive reduction of a directed graph. *SIAM J. Comput.* **1** 131–137. MR0306032

[2] Bollen, K.A. (1989). *Structural Equations with Latent Variables*. *Wiley Series in Probability and Mathematical Statistics*: *Applied Probability and Statistics*. New York: Wiley. MR0996025

[3] Bühlmann, P., Peters, J. and Ernest, J. (2014). CAM: Causal additive models, high-dimensional order search and penalized regression. *Ann. Statist.* **42** 2526–2556. MR3277670

[4] Butkovič, P. (2010). *Max-linear Systems*: *Theory and Algorithms*. *Springer Monographs in Mathematics*. London: Springer. MR2681232

[5] de Haan, L. and Ferreira, A. (2006). *Extreme Value Theory*: *An Introduction*. *Springer Series in Operations Research and Financial Engineering*. New York: Springer. MR2234156

[6] Diestel, R. (2010). *Graph Theory*, 4th ed. *Graduate Texts in Mathematics* **173**. Heidelberg: Springer. MR2744811

[7] Ernest, J., Rothenhäusler, D. and Bühlmann, P. (2016). Causal inference in partially linear structural equation models: Identifiability and estimation. Preprint. Available at arXiv:1607.05980.

[8] Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models*: *Principles and Techniques*. *Adaptive Computation and Machine Learning*. Cambridge, MA: MIT Press. MR2778120

[9] Lauritzen, S.L. (1996). *Graphical Models*. *Oxford Statistical Science Series* **17**. New York: Oxford Univ. Press,. MR1419991

[10] Lauritzen, S.L., Dawid, A.P., Larsen, B.N. and Leimer, H.-G. (1990). Independence properties of directed Markov fields. *Networks* **20** 491–505. MR1064735

[11] Mahr, B. (1981). A bird's-eye view to path problems. In *Graph-Theoretic Concepts in Computer Science* (*Proc*. *Sixth Internat*. *Workshop*, *Bad Honnef*, 1980). *Lecture Notes in Computer Science* **100** 335–353. Springer, Berlin-New York. MR0621510

[12] Pearl, J. (2009). *Causality*: *Models*, *Reasoning*, *and Inference*, 2nd ed. Cambridge: Cambridge Univ. Press. MR2548166

[13] Pourret, O., Naim, P. and Marcot, B., eds. (2008) *Bayesian Networks*: *A Practical Guide to Applications*. *Statistics in Practice*. Chichester: Wiley. MR2407977

[14] Resnick, S.I. (1987). *Extreme Values*, *Regular Variation*, *and Point Processes*. *Applied Probability. A Series of the Applied Probability Trust* **4**. New York: Springer. MR0900810

[15] Resnick, S.I. (2007). *Heavy-Tail Phenomena*: *Probabilistic and Statistical modeling*. *Springer Series in Operations Research and Financial Engineering*. New York: Springer. MR2271424

[16] Rote, G. (1985). A systolic array algorithm for the algebraic path problem (shortest paths; matrix inversion). *Computing* **34** 191–219. MR0799823

[17] Spirtes, P., Glymour, C. and Scheines, R. (2000). *Causation*, *Prediction*, *and Search*, 2nd ed. *Adaptive Computation and Machine Learning*. Cambridge, MA: MIT Press. MR1815675

[18] Wang, Y. and Stoev, S.A. (2011). Conditional sampling for spectrally discrete max-stable random fields. *Adv. in Appl*. *Probab.* **43** 461–483. MR2848386

[19] Wright, S. (1934). The method of path coefficients. *Ann*. *Math*. *Stat*. **5** 161–215.

Received January 2016 and revised February 2017