Brock University

Department of Computer Science

Relation Algebras, Matrices, and Multi-Valued Decision Diagrams

Francis Atampore and Michael Winter
Technical Report # CS-12-03
April 2012

Brock University
Department of Computer Science
St. Catharines, Ontario Canada L2S 3A1
www.cosc.brocku.ca

Relation Algebras, Matrices, and Multi-Valued Decision Diagrams

Francis Atampore and Michael Winter*

Department of Computer Science
Brock University
St. Catharines, ON, Canada
[fa10qv|mwinter]@brocku.ca

Abstract. In this paper we want to further investigate the usage of matrices as a representation of relations within arbitrary heterogeneous relation algebras. First, we want to show that splittings do exist in matrix algebras assuming that the underlying algebra of the coefficients provides this operation. Second, we want to outline an implementation of matrix algebras using reduced ordered multi-valued decision diagrams. This implementation combines the efficiency of operations based on those data structures with the general matrix approach to arbitrary relation algebras.

1 Introduction

Relation algebras, and category of relations, in particular, have been extremely useful as a formal system in various areas of mathematics and computer science. Applications range from logic [14, 19], fuzzy relations [24], program development [6], program semantics [15, 25], and graph theory [3, 15] to social choice theory [5][1]. Visualizing relations and computing expression in the language of relations can be very helpful while working within abstract theory. The RelView system [2] was developed for exactly this purpose. The system visualizes relations as Boolean matrices. Internally it uses reduced ordered binary decision diagrams (ROBDDs) in order to provide very efficient implementations of the operations on relations.

The RelView system is based on the standard model of relation algebras, i.e., Boolean matrices or, equivalently, sets of pairs. Therefore, this system cannot visualize computations in non-standard models of the abstract theory of heterogeneous relations. This is particularly important if one considers properties that are true in the standard model but not in all models. An example of such a property is given by the composition of two (heterogeneous) universal relations, i.e., two relations that relate every pair of elements between different sets. In the

* The author gratefully acknowledges support from the Natural Sciences and Engineering Research Council of Canada.

[1] This list of topics, papers, and books is not meant to be exhaustive. It is supposed to serve as starting point for further research.

standard model one will always obtain the universal relation, while this might not be the case in some non-standard models. Another example is given by the relationship between the power set of a disjoint union of two sets A and B and the product of the power set of A and the power set of B. In the standard model both constructions lead to isomorphic objects, while this might not be the case in certain non-standard models.

In [21, 22] it has been shown that relations from arbitrary relation algebras can be represented by matrices. Instead of the Boolean values one has to deal with more general coefficients. As a consequence one obtains that all standard operations on relations correspond to known matrix operations. In this paper we want to extend the general matrix approach to additional operations within relation algebras such as splittings and relational powers. It has been shown by multiple examples [4] that splittings are an important construction in the application of relational methods. Having this construction available shows once more that it is sufficient to use matrices as a representation for arbitrary relation algebras. Furthermore, we want to outline an implementation of matrix algebras using reduced ordered multi-valued decision diagrams. Since this implementation uses a similar data structure than RelView it combines an efficient computation of the operations on relations with the general matrix approach for arbitrary relation algebras.

The remainder of this paper is organized as follows. In Section 2 we recall the basic theory of heterogeneous relation algebras. After recalling the pseudo-representation theorem using matrices we will show in Section 3 that splittings in matrix algebras do exist if the underlying algebra of the coefficients provides this kind of operation. Finally, we will outline the implementation of matrix algebras using multi-valued decision diagrams in Section 4.

2 Heterogeneous Relation Algebras

In this section we recall some fundamentals on heterogeneous relation algebras. Heterogeneous relation algebras are a categorical version of Tarski's relation algebras with the additional requirements that the underlying Boolean algebras are complete and atomic. For further details we refer to [10, 15, 17].

We will denote the collection of objects and the collection of morphisms of a category $\mathcal{C}$ by $\mathrm{Obj}_{\mathcal{C}}$ and $\mathrm{Mor}_{\mathcal{C}}$, respectively. Composition is written as ";", which has to be read from left to right, i.e., $f;g$ means "f first, then g". For a morphism f in a category $\mathcal{C}$ with source A and target B we use $f \in \mathcal{C}[A,B]$ and $f : A \to B$ interchangeably. Finally, the identity morphism in $\mathcal{C}[A,A]$ is denoted by $\mathbb{I}_A$.

Definition 1. *A (heterogeneous abstract) relation algebra is a locally small category $\mathcal{R}$. The morphisms are usually called relations. In addition, there is a totally defined unary operation* $\smile_{AB} : \mathcal{R}[A,B] \longrightarrow \mathcal{R}[B,A]$ *between the sets of morphisms, called conversion. The operations satisfy the following rules:*

1. *Every set $\mathcal{R}[A,B]$ carries the structure of a complete atomic Boolean algebra with operations $\sqcup_{AB}, \sqcap_{AB}, \overline{}_{AB}$, zero element $\perp\!\!\!\perp_{AB}$, universal element $\top\!\!\!\top_{AB}$, and inclusion ordering $\sqsubseteq_{AB}$.*
2. *The Schröder equivalences*

$$Q;R \sqsubseteq_{AC} S \iff Q^{\smile};\overline{S} \sqsubseteq_{BC} \overline{R} \iff \overline{S};R^{\smile} \sqsubseteq_{AB} \overline{Q}$$

hold for relations $Q : A \to B, R : B \to C$ and $S : A \to C$.

As usual we omit all indices of elements and operations for brevity if they are not important or clear from the context.

The standard example of a relation algebra is the category **Rel** of sets and binary relations, i.e., sets of ordered pairs, with the usual operations. We will use this example frequently in order to motivate or illustrate definitions and properties of relations. Notice that those relations can also be represented by Boolean matrices as shown in [15–17] and the RelView system [2].

In the following lemma we have summarized several standard properties of relations. We will use them as well as other basic properties such as $\mathbb{I}_A^{\smile} = \mathbb{I}_A$ throughout this paper without mentioning. Proof can be found in any of the following [10, 15–17].

Lemma 1. *Let $\mathcal{R}$ be a relation algebra and $Q : A \to B$, $R, R_1, R_2 : B \to C$ and $S : A \to C$ be relations. Then we have:*

1. $Q;(R_1 \sqcap R_2) \sqsubseteq Q;R_1 \sqcap Q;S_2$,
2. $Q;R \sqcap S \sqsubseteq (Q \sqcap S;R^{\smile});(R \sqcap Q^{\smile};S)$ *(Dedekind formula),*
3. $Q \sqsubseteq Q;Q^{\smile};Q$.

The inclusion $Q;X \sqsubseteq R$ for two relations $Q : A \to B$ and $R : A \to C$ has a greatest solution $Q\backslash R = \overline{Q^{\smile};\overline{R}}$ in X called the right residual of Q and R. Similarly, the inclusion $Y;S \sqsubseteq R$ has a greatest solution $R/S = \overline{\overline{R};S^{\smile}}$ in Y called the left residual of S and R. In **Rel** the two constructions are given by the relations $Q\backslash R = \{(x,y) \mid \forall z : (z,x) \in Q \text{ implies } (z,y) \in R\}$ and $R/S = \{(z,u) \mid \forall y : (u,y) \in S \text{ implies } (z,y) \in R\}$. Using both constructions together we obtain the definition of a symmetric quotient $\mathrm{syQ}(Q,R) = Q\backslash R \sqcap Q^{\smile}/R^{\smile}$. The characteristic property of the symmetric quotient can be stated as

$$X \sqsubseteq \mathrm{syQ}(Q,R) \iff Q;X \sqsubseteq R \text{ and } X;R^{\smile} \sqsubseteq Q^{\smile}.$$

Notice that in **Rel** we have $\mathrm{syQ}(Q,R) = \{(x,y) \mid \forall z : (z,x) \in Q \text{ iff } (z,y) \in R\}$.

We define the notion of a homomorphism between relation algebras as usual, i.e., as a functor that preserves the additional structure.

Definition 2. *Let be $\mathcal{R}$ and $\mathcal{S}$ relation algebras and $F : \mathcal{R} \to \mathcal{S}$ a functor. Then F is called a homomorphism between relation algebras iff*

1. $F(\prod_{i \in I} S_i) = \prod_{i \in I} F(S_i)$,

2. $F(\overline{R}) = \overline{F(R)}$,
3. $F(R^{\smile}) = F(R)^{\smile}$,

hold for all relations R, S_i with $i \in I$.

A pair of homomorphisms $F : \mathcal{R} \to \mathcal{S}, G : \mathcal{S} \to \mathcal{R}$ is called an equivalence iff $F \circ G$ and $G \circ F$ are naturally isomorphic to the corresponding identity functors.

The relational description of disjoint unions of sets is the relational sum [17, 25]. This construction corresponds to the categorical product.

Definition 3. *Let $\{A_i \mid i \in I\}$ be a set of objects indexed by a set I. An object $\sum\limits_{i \in I} A_i$ together with relations $\iota_j \in \mathcal{R}[A_j, \sum\limits_{i \in I} A_i]$ for all $j \in I$ is called a relational sum of $\{A_i \mid i \in I\}$ iff for all $i, j \in I$ with $i \neq j$ the following holds*

$$\iota_i ; \iota_i^{\smile} = \mathbb{I}_{A_i}, \qquad \iota_i ; \iota_j^{\smile} = \perp\!\!\!\perp_{A_i A_j}, \qquad \bigsqcup_{i \in I} \iota_i^{\smile} ; \iota_i = \mathbb{I}_{\sum\limits_{i \in I} A_i}.$$

$\mathcal{R}$ has relational sums iff for every set of objects the relational sum does exist.

For a set of two objects $\{A, B\}$ this definition corresponds to usual definition of the relational sum. As known categorical products and hence relational sums are unique up to isomorphism.

A partial equivalence relation Ξ on a set A is an equivalence relation that does not need to be totally defined. Algebraically, it satisfies $\Xi^{\smile} = \Xi$ (symmetric) and $\Xi ; \Xi = \Xi$ (idempotent). Because of the defining properties partial equivalence relations are also called symmetric idempotent relations, or sid's for short.

Lemma 2. *If $\Xi, \Theta : A \to A$ are partial equivalence relations so that $\Xi ; \Theta \sqsubseteq \Theta$, then $\Xi \sqcap \overline{\Theta}$ is a partial equivalence relation with $(\Xi \sqcap \overline{\Theta}) ; \Xi = \Xi \sqcap \overline{\Theta}$.*

Proof. The relation $\Xi \sqcap \overline{\Theta}$ is symmetric since converse distributes over all operations in question. From $\Xi ; \Theta \sqsubseteq \Theta$ we obtain $\Xi ; \overline{\Theta} = \Xi^{\smile} ; \overline{\Theta} \sqsubseteq \overline{\Theta}$ by using the Schröder equivalences. This implies

$$\begin{aligned} (\Xi \sqcap \overline{\Theta}) ; (\Xi \sqcap \overline{\Theta}) &\sqsubseteq \Xi ; (\Xi \sqcap \overline{\Theta}) \\ &\sqsubseteq \Xi ; \Xi \sqcap \Xi ; \overline{\Theta} \\ &\sqsubseteq \Xi \sqcap \overline{\Theta}. \end{aligned}$$

The converse inclusion is shown by

$$\begin{aligned} \Xi \sqcap \overline{\Theta} &= \Xi ; \Xi \sqcap \overline{\Theta} \\ &\sqsubseteq (\Xi \sqcap \overline{\Theta} ; \Xi^{\smile}) ; (\Xi \sqcap \Xi^{\smile} ; \overline{\Theta}) && \text{Dedekind formula} \\ &= (\Xi \sqcap \overline{\Theta} ; \Xi) ; (\Xi \sqcap \Xi ; \overline{\Theta}) \\ &\sqsubseteq (\Xi \sqcap \overline{\Theta}) ; (\Xi \sqcap \overline{\Theta}), \end{aligned}$$

where the last inclusion follows from $\Xi;\overline{\Theta} \sqsubseteq \overline{\Theta}$ as shown above using converse. Now, consider the following computation

$$\begin{aligned}
(\Xi \sqcap \overline{\Theta});\Xi &\sqsubseteq \Xi;\Xi \sqcap \overline{\Theta};\Xi \\
&\sqsubseteq \Xi \sqcap \overline{\Theta}, && \Xi \text{ idempotent and see above} \\
\Xi \sqcap \overline{\Theta} &= \Xi;\Xi \sqcap \overline{\Theta} && \Xi \text{ idempotent} \\
&\sqsubseteq (\Xi \sqcap \overline{\Theta};\Xi);\Xi && \Xi \text{ symmetric} \\
&\sqsubseteq (\Xi \sqcap \overline{\Theta});\Xi, && \text{see above}
\end{aligned}$$

which shows the additional equation $(\Xi \sqcap \overline{\Theta});\Xi = \Xi \sqcap \overline{\Theta}$. □

Given a partial equivalence relation Ξ in **Rel** one can compute the set of all equivalence classes whenever Ξ is defined. This concept is called a splitting [10].

Definition 4. *Let $\Xi \in \mathcal{R}[A, A]$ be a partial equivalence relation. An object B together with a relation $\psi \in \mathcal{R}[B, A]$ is called a splitting of Ξ iff*

$$\psi;\psi^{\smile} = \mathbb{I}_B, \qquad \psi^{\smile};\psi = \Xi.$$

A relation algebra has splittings iff for all partial equivalence relations a splitting exists.

Notice that splittings are unique up to isomorphism. Furthermore, we may distinguish two special cases of splittings. If Ξ is also reflexive, i.e., an equivalence relation, the splitting corresponds to the construction of the set of equivalence classes. If $\Xi \sqsubseteq \mathbb{I}_A$, then Ξ corresponds to a subset of A. In this case the splitting becomes the subobject induced by Ξ.

The last construction we want to introduce in this section is the abstract counterpart of a power set, called the relational power. The definition is based on the fact that every relation $R : A \to B$ in **Rel** can be transformed into a function $f_R : A \to \mathcal{P}(B)$ where $\mathcal{P}(B)$ denotes the power set of B.

Definition 5. *Let $\mathcal{R}$ be a relation algebra. An object $\mathcal{P}(A)$, together with a relation $\varepsilon_A : A \to \mathcal{P}(A)$ is called a relational power of A iff*

$$\mathrm{syQ}(\varepsilon_A, \varepsilon_A) \sqsubseteq \mathbb{I}_{\mathcal{P}(A)} \quad \textit{and} \quad \mathrm{syQ}(R, \varepsilon_A) \textit{ is total}$$

for all relations $R : B \to A$. $\mathcal{R}$ has relational powers iff the relational power for any object exists.

Notice that in the case of the relation algebra **Rel** and a relation $R : A \to B$ the construction $\mathrm{syQ}(R^{\smile}, \varepsilon_B) : A \to \mathcal{P}(B)$ is the function f_R mentioned above.

For technical reasons we follow [10] and call an object B a pre-power of A if there a relation $T : A \to B$ so that $\mathrm{syQ}(R, T)$ is total for all relations $R : C \to A$, i.e., a relational power is a pre-power with the additional requirement $\mathrm{syQ}(T, T) \sqsubseteq \mathbb{I}$. If $\mathcal{R}$ has splittings, then we obtain a relational power of A from a pre-power by splitting the equivalence relation $\mathrm{syQ}(T, T)$. This fact indicates once more that splittings are an important construction in the theory of relations.

3 Matrix Algebras

Given a heterogeneous relation algebra $\mathcal{R}$, an algebra of matrices with coefficients from $\mathcal{R}$ may be defined.

Definition 6. *Let $\mathcal{R}$ be a relation algebra. The algebra $\mathcal{R}^+$ of matrices with coefficients from $\mathcal{R}$ is defined by:*

1. *The class of objects of $\mathcal{R}^+$ is the collection of all functions from an arbitrary set I to $\mathrm{Obj}_\mathcal{R}$.*
2. *For every pair $f : I \to \mathrm{Obj}_\mathcal{R}, g : J \to \mathrm{Obj}_\mathcal{R}$ of objects from $\mathcal{R}^+$, the set of morphisms $\mathcal{R}^+[f, g]$ is the set of all functions $R : I \times J \to \mathrm{Mor}_\mathcal{R}$ so that $R(i, j) \in \mathcal{R}[f(i), g(j)]$ holds.*
3. *For $R \in \mathcal{R}^+[f, g]$ and $S \in \mathcal{R}^+[g, h]$ composition is defined by*
$$(R; S)(i, k) := \bigsqcup_{j \in J} R(i, j); S(j, k).$$
4. *For $R \in \mathcal{R}^+[f, g]$ conversion and negation are defined by*
$$R^{\smile}(j, i) := (R(i, j))^{\smile}, \qquad \overline{R}(i, j) := \overline{R(i, j)}.$$
5. *For $R, S \in \mathcal{R}^+[f, g]$ union and intersection are defined by*
$$(R \sqcup S)(i, j) := R(i, j) \sqcup S(i, j), \qquad (R \sqcap S)(i, j) := R(i, j) \sqcap S(i, j).$$
6. *The identity, zero and universal elements are defined by*
$$\mathbb{I}_f(i_1, i_2) := \begin{cases} \perp\!\!\!\perp_{f(i_1)f(i_2)} & : i_1 \neq i_2 \\ \mathbb{I}_{f(i_1)} & : i_1 = i_2, \end{cases}$$
$$\perp\!\!\!\perp_{fg}(i, j) := \perp\!\!\!\perp_{f(i)g(j)}, \qquad \top\!\!\!\top_{fg}(i, j) := \top\!\!\!\top_{f(i)g(j)}.$$

Obviously, an object in $\mathcal{R}^+$ may be seen as a (in general non-finite) sequence of objects from $\mathcal{R}$, and a morphism in $\mathcal{R}^+$ may be seen as a (in general non-finite) matrix indexed by objects from $\mathcal{R}$. Notice that any Boolean algebra forms a relation algebra if we define composition as the meet operation and converse is the identity function. We obtain Boolean matrices in $\mathbb{B}^+$ if we choose the Boolean algebra with two values $\mathbb{B} = \{0, 1\}$. These matrices are a natural representation of the relations in **Rel**.

The proof of the following result is an easy exercise and is, therefore, omitted.

Lemma 3. *$\mathcal{R}^+$ is a relation algebra.*

Furthermore, the possibility to build disjoint unions of arbitrary sets indexed by a set gives us the following.

Theorem 1. *$\mathcal{R}^+$ has relational sums.*

Proof. A detailed proof can be found in [10]. We only want to recall how to construct of the sum of a set $\{f_i : J_i \to \mathrm{Obj}_{\mathcal{R}} \mid i \in I\}$ of objects of $\mathcal{R}^+$ here. Let $J = \sum_{i \in I} J_i \to \mathrm{Obj}_{\mathcal{R}}$ be the disjoint union of the sets J_i for $i \in I$. Then the function $h : J \to \mathrm{Obj}_{\mathcal{R}}$ defined by $h(j) := f_i(j)$ iff $j \in J_i$ is also an object of $\mathcal{R}^+$. Now, we define

$$\iota_i(j_1, j_2) := \begin{cases} \perp\!\!\!\perp_{f_i(j_1)h(j_2)} & : j_1 \neq j_2 \\ \mathbb{I}_{f(j_1)} & : j_1 = j_2. \end{cases}$$

An easy verification shows that the above definition gives us the required relational sum. □

In addition to relational sums, matrix algebras also provide the essential part of relational powers.

Theorem 2. *If $\mathcal{R}$ is small, then $\mathcal{R}^+$ has pre-powers.*

Proof. Again, a detailed proof can be found in [10]. We only want to recall how to construct the pre-power of an object $f : I \to \mathrm{Obj}_{\mathcal{R}}$ of $\mathcal{R}^+$. Since $\mathcal{R}$ is small, the collection $J = \{(i, B, R) \mid i \in I \text{ and } B \in \mathrm{Obj}_{\mathcal{R}} \text{ and } R : f(i) \to B\}$ is a set, and, hence, the function $g(i, B, R) = B$ is an object of $\mathcal{R}^+$. Now, define a relation $T : f \to g$ by

$$T(i, (j, B, R)) = \begin{cases} R & \text{iff } i = j, \\ \perp\!\!\!\perp_{f(i)B} & \text{otherwise.} \end{cases}$$

The object g together with the relation T constitutes a pre-power of f. □

In [10] it was shown that every relation algebra $\mathcal{R}$ can be embedded into an algebra $\mathcal{R}_{sid}$ that has splittings. Furthermore, if $\mathcal{R}$ has relational sums, so does $\mathcal{R}_{sid}$. As a special case we obtain that every matrix algebra can be embedded into an algebra with splittings. However, this new algebra does not need to be a matrix algebra again. Altogether, these constructions do not provide any hint when the matrix algebra itself already provides splittings. Such a characterization is important if we want to consider matrix algebras as a general representation and/or visualization of relation algebras. We will come back to this problem later in this section.

Following the notion used in algebra, we call an object A integral if there are no zero divisors within the algebra $\mathcal{R}[A, A]$. The class of integral objects will define the basis of $\mathcal{R}$.

Definition 7. *An object A of a relation algebra $\mathcal{R}$ is called integral iff for all $Q, R \in \mathcal{R}[A, A]$ the equation $Q; R = \perp\!\!\!\perp_{AA}$ implies $Q = \perp\!\!\!\perp_{AA}$ or $R = \perp\!\!\!\perp_{AA}$. $\mathcal{R}$ is called integral iff all objects of $\mathcal{R}$ are integral. The basis $\mathcal{B}_{\mathcal{R}}$ of $\mathcal{R}$ is defined as the full subcategory given by the class of all integral objects.*

As usual, we omit the index $\mathcal{R}$ in $\mathcal{B}_{\mathcal{R}}$ when its meaning is clear from the context. Notice that the basis is normally a lot "smaller" than the original relation algebra. In particular, if $\mathcal{R}$ has relational sums, then all objects in $\mathcal{B}$ are irreducible objects with respect to the sum construction.

The following theorem was shown in [21, 22]. It can be seen as a pseudo-representation theorem indicating that it is completely sufficient to consider matrix algebras over integral relations algebras when considering the standard operations on relations.

Theorem 3. *Let $\mathcal{R}$ be a relation algebra with relational sums and subobjects and $\mathcal{B}$ the basis of $\mathcal{R}$. Then $\mathcal{R}$ and $\mathcal{B}^+$ are equivalent.*

Since matrix algebras also provide relational sums (Theorem 1) and pre-powers (Theorem 2), these algebras also cover both additional constructions. As mentioned already above it has not been shown that we can perform splittings in matrix algebras. In order to prove such a theorem we will use the following conventions. Suppose $M : f \to f$ is a square matrix where $f : I \to \mathrm{Obj}_{\mathcal{R}}$ is an object of $\mathcal{R}^+$. The set I can be well-ordered by the axiom of the choice, and, hence, is isomorphic to some ordinal number α. For simplicity we identify I and α in the rest of this section and call M a matrix of size α. In addition, we will denote by M_β with $\beta \leq \alpha$ the submatrix of M of size β, i.e., $M_\beta : f_\beta \to f_\beta$ with $f_\beta(\gamma) = f(\gamma)$ and $M_\beta(\gamma, \delta) = M(\gamma, \delta)$ for all $\gamma, \delta \leq \beta$. Obviously we have $M_\alpha = M$ and $(M_\beta)_\gamma = M_\gamma$ for $\gamma \leq \beta \leq \alpha$.

The basic idea of splitting relations in matrix form is as follows. Each element on the diagonal of a partial equivalence relation is itself a partial equivalence relation. Those relations are not necessarily independent of each other. The relation $M(\beta, \gamma)$ relates the two partial equivalence relations $M(\beta, \beta)$ and $M(\gamma, \gamma)$. If we split $M(\beta, \beta)$, then we only have to split the remaining part of $M(\gamma, \gamma)$ that was not already covered by the splitting of $M(\beta, \beta)$, i.e., the relation $M(\gamma, \gamma) \sqcap \overline{M(\gamma, \beta); M(\beta, \gamma)}$. We will illustrate this process by an example at the end of this section.

In order to establish a theorem about splittings in matrix algebras we first need to prove some basic properties of partial equivalence relations in matrix form.

Lemma 4. *If $M : f \to f$ is a partial equivalence relation in $\mathcal{R}^+$ of size α, then we have for all β, γ, δ and relations R:*

1. $M(\beta, \gamma); M(\gamma, \delta) \sqsubseteq M(\beta, \delta)$,
2. $M(\beta, \beta); \overline{M(\beta, \gamma)} = \overline{M(\beta, \gamma)}$,
3. $M(\beta, \gamma); \overline{M(\gamma, \delta); R} \sqsubseteq \overline{M(\beta, \delta); R}$.

Proof. 1. This property follows immediately from

$$\begin{aligned} M(\beta, \gamma); M(\gamma, \delta) &\sqsubseteq \bigsqcup_{\gamma \leq \alpha} M(\beta, \gamma); M(\gamma, \delta) \\ &= (M; M)(\gamma, \delta) \\ &= M(\gamma, \delta). && M \text{ idempotent} \end{aligned}$$

2. Property (1) shows "$\sqsubseteq$". We obtain the other inclusion from

$$\begin{aligned}
M(\beta,\gamma) &\sqsubseteq M(\beta,\gamma);M(\beta,\gamma)^{\smile};M(\beta,\gamma) \\
&= M(\beta,\gamma);M^{\smile}(\gamma,\beta);M(\beta,\gamma) \\
&= M(\beta,\gamma);M(\gamma,\beta);M(\beta,\gamma) && M \text{ symmetric} \\
&\sqsubseteq M(\beta,\beta);M(\beta,\gamma) && \text{by (1)}
\end{aligned}$$

3. From (1) we obtain $M(\gamma,\beta);M(\beta,\delta);R \sqsubseteq M(\gamma,\delta);R$. This implies the assertion because M is symmetric. □

The next lemma allows a stepwise definition of partial equivalence relations based on the diagonal elements of M.

Lemma 5. *If $M : f \to f$ is a partial equivalence relation in $\mathcal{R}^+$ of size α, then the relations*

$$\Xi_\beta := M(\beta,\beta) \sqcap \bigsqcap_{\gamma<\beta} \overline{M(\beta,\gamma);M(\gamma,\beta)}$$

for $1 \leq \beta \leq \alpha$ are partial equivalence relations.

Proof. Let us define $\Theta_\beta : f(\beta) \to f(\beta)$ for $1 \leq \beta \leq \alpha$ by

$$\Theta_\beta := \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta).$$

First we want to show that Θ_β is a partial equivalence relation. Symmetry is trivial, and from the computation

$$\begin{aligned}
\Theta_\beta;\Theta_\beta &= (\bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta));(\bigsqcup_{\gamma'<\beta} M(\beta,\gamma');M(\gamma',\beta)) \\
&= \bigsqcup_{\gamma,\gamma'<\beta} M(\beta,\gamma);M(\gamma,\beta);M(\beta,\gamma');M(\gamma',\beta) \\
&\sqsubseteq \bigsqcup_{\gamma,\gamma'<\beta} M(\beta,\gamma);M(\gamma,\gamma');M(\gamma',\beta) && \text{Lemma 4(1)} \\
&\sqsubseteq \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta) && \text{Lemma 4(1)} \\
&= \Theta_\beta, \\
\Theta_\beta &= \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta) \\
&\sqsubseteq \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta);M(\gamma,\beta)^{\smile};M(\gamma,\beta)
\end{aligned}$$

$$
\begin{aligned}
&= \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta);M(\beta,\gamma);M(\gamma,\beta) && M \text{ symmetric}\\
&\sqsubseteq \bigsqcup_{\gamma,\gamma'<\beta} M(\beta,\gamma);M(\gamma,\beta);M(\beta,\gamma');M(\gamma',\beta)\\
&= (\bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta));(\bigsqcup_{\gamma'<\beta} M(\beta,\gamma');M(\gamma',\beta))\\
&= \Theta_\beta;\Theta_\beta
\end{aligned}
$$

we conclude that Θ_β is also idempotent. From

$$
\begin{aligned}
M(\beta,\beta);\Theta_\beta &= \bigsqcup_{\gamma<\beta} M(\beta,\beta);M(\beta,\gamma);M(\gamma,\beta)\\
&= \bigsqcup_{\gamma<\beta} M(\beta,\gamma);M(\gamma,\beta) && \text{Lemma 4(2)}\\
&= \Theta_\beta.
\end{aligned}
$$

and Lemma 2 we obtain that $\Xi_\beta = M(\beta,\beta) \sqcap \overline{\Theta_\beta}$ is also a partial equivalence relation. □

Our final lemma states some properties about the partial equivalence relations Ξ_β and their splittings.

Lemma 6. *Let be $M : f \to f$ a partial equivalence relation in $\mathcal{R}^+$ of size α, and Xi_β for $1 \leq \beta \leq \alpha$ be the partial equivalence relations defined above. Furthermore, suppose $R_\beta : A_\beta \to f(\beta)$ splits Ξ_β in $\mathcal{R}$. Then we have:*

1. $M(\gamma,\beta);R_\beta^{\smile} = \perp\!\!\!\perp_{f(\gamma)A_\beta}$ *for all* $\gamma < \beta$.
2. $M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta) = M(\beta,\gamma);M(\gamma,\beta) \sqcap \prod_{\delta<\gamma} \overline{M(\beta,\delta);M(\delta,\beta)}$.
3. $\bigsqcup_{\gamma\leq\delta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta) = \bigsqcup_{\gamma\leq\delta} M(\beta,\gamma);M(\gamma,\beta)$ *for all* $1 \leq \delta \leq \beta$.

Proof. 1. This property follows immediately from

$$
\begin{aligned}
&M(\gamma,\beta);R_\beta^{\smile}\\
&= M(\gamma,\beta);R_\beta^{\smile};R_\beta;R_\beta^{\smile} && R_\beta;R_\beta^{\smile} = \mathbb{I}_{A_\beta}\\
&= M(\gamma,\beta);\Xi_\beta;R_\beta^{\smile} && R_\beta^{\smile};R_\beta = \Xi_\beta\\
&\sqsubseteq M(\gamma,\beta);(M(\beta,\beta) \sqcap \overline{M(\beta,\gamma);M(\gamma,\beta)}) && \gamma<\beta\\
&\sqsubseteq M(\gamma,\beta);M(\beta,\beta) \sqcap M(\gamma,\beta);\overline{M(\beta,\gamma);M(\gamma,\beta)}\\
&= M(\gamma,\beta) \sqcap M(\gamma,\beta);\overline{M(\beta,\gamma);M(\gamma,\beta)} && \text{Lemma 4(2)}\\
&\sqsubseteq M(\gamma,\beta) \sqcap \overline{M(\gamma,\beta)} && \text{Lemma 4(2),(3)}\\
&= \perp\!\!\!\perp_{f(\gamma)A_\beta}.
\end{aligned}
$$

2. The following sequence of inclusions

$$
\begin{aligned}
& M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta) \\
& = M(\beta,\gamma);\left(M(\gamma,\gamma) \sqcap \bigsqcap_{\delta<\gamma} \overline{M(\gamma,\delta);M(\delta,\gamma)}\right);M(\gamma,\beta) \\
& \sqsubseteq M(\beta,\gamma);M(\gamma,\gamma);M(\gamma,\beta) \sqcap \bigsqcap_{\delta<\gamma} M(\beta,\gamma);\overline{M(\gamma,\delta);M(\delta,\gamma)};M(\gamma,\beta) \\
& \sqsubseteq M(\beta,\gamma);M(\gamma,\beta) \sqcap \bigsqcap_{\delta<\gamma} \overline{M(\beta,\delta);M(\delta,\gamma)} && \text{Lemma 4(2),(3)} \\
& = M(\beta,\gamma);M(\gamma,\gamma);M(\gamma,\beta) \sqcap \bigsqcap_{\delta<\gamma} \overline{M(\beta,\delta);M(\delta,\gamma)} && \text{Lemma 4(2)} \\
& \sqsubseteq M(\beta,\gamma);\left(M(\gamma,\gamma);M(\gamma,\beta) \sqcap \bigsqcap_{\delta<\gamma} M(\gamma,\beta);\overline{M(\beta,\delta);M(\delta,\gamma)}\right) \\
& \sqsubseteq M(\beta,\gamma);\left(M(\gamma,\gamma) \sqcap \bigsqcap_{\delta<\gamma} M(\gamma,\beta);\overline{M(\beta,\delta);M(\delta,\gamma)};M(\beta,\gamma)\right);M(\gamma,\beta) \\
& \sqsubseteq M(\beta,\gamma);\left(M(\gamma,\gamma) \sqcap \bigsqcap_{\delta<\gamma} \overline{M(\gamma,\delta);M(\delta,\gamma)}\right);M(\gamma,\beta) && \text{Lemma 4(3)} \\
& = M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta)
\end{aligned}
$$

shows the second property.

3. This property is shown by transfinite induction on δ. If $\delta = 1$, then we obtain $M(\beta,1);\Xi_1;M(1,\beta) = M(\beta,1);M(1,1);M(1,\beta) = M(\beta,1);M(1,\beta)$ from Lemma 4(2). Now, suppose $\delta = \delta'+1$. Then we have

$$
\begin{aligned}
& \bigsqcup_{\gamma\le\delta'+1} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta) \\
& = M(\beta,\delta'+1);\Xi_{\delta'+1};M(\delta'+1,\beta) \sqcup \bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta) \\
& = M(\beta,\delta'+1);\Xi_{\delta'+1};M(\delta'+1,\beta) \sqcup \bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);M(\gamma,\beta) && \text{Ind. hyp.} \\
& = \left(M(\beta,\delta'+1);M(\delta'+1,\beta) \sqcap \bigsqcap_{\gamma<\delta'+1} \overline{M(\beta,\gamma);M(\gamma,\beta)}\right) && \text{by (2)} \\
& \quad \sqcup \bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);M(\gamma,\beta)
\end{aligned}
$$

$$= M(\beta,\delta'+1);M(\delta'+1,\beta) \sqcup \bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);M(\gamma,\beta)$$
$$= \bigsqcup_{\gamma\le\delta'+1} M(\beta,\gamma);M(\gamma,\beta).$$

If δ is a limit ordinal, then we immediately compute

$$\begin{aligned}
&\bigsqcup_{\gamma\le\delta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta)\\
&= M(\beta,\delta);\Xi_\delta;M(\delta,\beta) \sqcup \bigsqcup_{\gamma<\delta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta)\\
&= M(\beta,\delta);\Xi_\delta;M(\delta,\beta) \sqcup \bigsqcup_{\delta'<\delta}\bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta)\\
&= \left(M(\beta,\delta);M(\delta,\beta) \sqcap \bigsqcap_{\gamma<\delta}\overline{M(\beta,\gamma);M(\gamma,\beta)}\right) && \text{by (2)}\\
&\quad \sqcup \bigsqcup_{\delta'<\delta}\bigsqcup_{\gamma\le\delta'} M(\beta,\gamma);M(\gamma,\beta) && \text{Ind. hyp.}\\
&= \left(M(\beta,\delta);M(\delta,\beta) \sqcap \bigsqcap_{\gamma<\delta}\overline{M(\beta,\gamma);M(\gamma,\beta)}\right) \sqcup \bigsqcup_{\gamma<\delta} M(\beta,\gamma);M(\gamma,\beta)\\
&= M(\beta,\delta);M(\delta,\beta) \sqcup \bigsqcup_{\gamma<\delta} M(\beta,\gamma);M(\gamma,\beta)\\
&= \bigsqcup_{\gamma\le\delta} M(\beta,\gamma);M(\gamma,\beta).
\end{aligned}$$

This completes the proof. □

We are now ready to prove our main theorem of this section.

Theorem 4. *Let $\mathcal{R}$ be relation algebra with splittings. Then $\mathcal{R}^+$ has splittings.*

Proof. Let be $M : f \to f$ a partial equivalence relation in $\mathcal{R}^+$ of size α, and let $R_\beta : A_\beta \to f(\beta)$ be a splitting of Ξ_β in $\mathcal{R}$. Now, and define $g : \alpha \to \mathrm{Obj}_\mathcal{R}$ and $N : g \to f$ by

$$g(\beta) = A_\beta,$$
$$N(\beta,\gamma) = \begin{cases} R_\beta;M(\beta,\gamma) & \text{iff } \beta\le\gamma,\\ \perp\!\!\!\perp_{A_\beta f(\gamma)} & \text{otherwise,}\end{cases}$$

with $\beta, \gamma \leq \alpha$. We have

$$
\begin{aligned}
(N; N^{\smile})(\beta, \delta) &= \bigsqcup_{\gamma \leq \alpha} N(\beta, \gamma); N^{\smile}(\gamma, \delta) \\
&= \bigsqcup_{\gamma \leq \alpha} N(\beta, \gamma); N(\delta, \gamma)^{\smile} \\
&= \bigsqcup_{\max(\beta,\delta) \leq \gamma \leq \alpha} N(\beta, \gamma); N(\delta, \gamma)^{\smile} && \text{Def. of } N \\
&= \bigsqcup_{\max(\beta,\delta) \leq \gamma \leq \alpha} R_\beta; M(\beta, \gamma); (R_\delta; M(\delta, \gamma))^{\smile} \\
&= \bigsqcup_{\max(\beta,\delta) \leq \gamma \leq \alpha} R_\beta; M(\beta, \gamma); M(\gamma, \delta); R_\delta^{\smile} && M \text{ symmetric} \\
&= R_\beta; \left(\bigsqcup_{\max(\beta,\delta) \leq \gamma \leq \alpha} M(\beta, \gamma); M(\gamma, \delta) \right); R_\delta^{\smile}
\end{aligned}
$$

If $\beta \neq \delta$, then we obtain

$$
\begin{aligned}
& R_\beta; \left(\bigsqcup_{\max(\beta,\delta) \leq \gamma \leq \alpha} M(\beta, \gamma); M(\gamma, \delta) \right); R_\delta^{\smile} \\
& \sqsubseteq R_\beta; \left(\bigsqcup_{\gamma \leq \alpha} M(\beta, \gamma); M(\gamma, \delta) \right); R_\delta^{\smile} \\
& = R_\beta; (M; M)(\beta, \delta); R_\delta^{\smile} \\
& = R_\beta; M(\beta, \delta); R_\delta^{\smile} && M \text{ idempotent} \\
& = \perp\!\!\!\perp_{A_\beta A_\delta},
\end{aligned}
$$

where the last equality follows from Lemma 6(1) since we have either $\beta < \delta$ or $\delta < \beta$. If $\beta = \delta$, then we compute

$$
\begin{aligned}
& R_\beta; \left(\bigsqcup_{\beta \leq \gamma \leq \alpha} M(\beta, \gamma); M(\gamma, \beta) \right); R_\beta^{\smile} \\
& = R_\beta; M(\beta, \beta); R_\beta^{\smile} && \text{Lemma 2(1) for } \gamma = \beta \text{ and (2) otherwise} \\
& = R_\beta; R_\beta^{\smile}; R_\beta; M(\beta, \beta); R_\beta^{\smile} && R_\beta; R_\beta^{\smile} = \mathbb{I}_{A_\beta} \\
& = R_\beta; \Xi_\beta; M(\beta, \beta); R_\beta^{\smile} && R_\beta^{\smile}; R_\beta = \Xi_\beta \\
& = R_\beta; \Xi_\beta; R_\beta^{\smile} && \text{Lemma 2} \\
& = R_\beta; R_\beta^{\smile}; R_\beta; R_\beta^{\smile} && R_\beta^{\smile}; R_\beta = \Xi_\beta \\
& = \mathbb{I}_{A_\beta},
\end{aligned}
$$

i.e., we have just shown that $N;N^{\smile} = \mathbb{I}_g$. In order to verify that $N^{\smile};N = M$ consider

$$
\begin{aligned}
(N^{\smile};N)(\beta,\delta) &= \bigsqcup_{\gamma\leq\alpha} N^{\smile}(\beta,\gamma);N(\gamma,\delta) \\
&= \bigsqcup_{\gamma\leq\alpha} N(\gamma,\beta)^{\smile};N(\gamma,\delta) \\
&= \bigsqcup_{\gamma\leq\min(\beta,\gamma)} N(\gamma,\beta)^{\smile};N(\gamma,\delta) && \text{Def. } N \\
&= \bigsqcup_{\gamma\leq\min(\beta,\gamma)} M(\beta,\gamma);R_\gamma^{\smile};R_\gamma;M(\gamma,\delta) && \text{Def. } N \\
&= \bigsqcup_{\gamma\leq\min(\beta,\gamma)} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\delta).
\end{aligned}
$$

We immediately obtain from Lemma 4(1)

$$M(\beta,\gamma);\Xi_\gamma;M(\gamma,\delta) \sqsubseteq M(\beta,\gamma);M(\gamma,\gamma);M(\gamma,\delta) \sqsubseteq M(\beta,\delta),$$

and, hence, $(N^{\smile};N)(\beta,\delta) \sqsubseteq M(\beta,\delta)$. For the converse inclusion assume $\beta \leq \delta$. Then we have

$$
\begin{aligned}
&M(\beta,\delta) \\
&= M(\beta,\beta);M(\beta,\delta) && \text{Lemma 4(2)} \\
&= \left(\bigsqcup_{\gamma\leq\beta} M(\beta,\gamma);M(\gamma,\beta)\right);M(\beta,\delta) && \text{Lemma 4(3) and (2) for } \gamma=\beta \\
&= \left(\bigsqcup_{\gamma\leq\beta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta)\right);M(\beta,\delta) && \text{Lemma 6(3)} \\
&= \bigsqcup_{\gamma\leq\beta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\beta);M(\beta,\delta) \\
&\sqsubseteq \bigsqcup_{\gamma\leq\beta} M(\beta,\gamma);\Xi_\gamma;M(\gamma,\delta). && \text{Lemma 4(3)}
\end{aligned}
$$

The case $\delta \leq \beta$ is shown analogously. This completes the proof. □

We want to illustrate the previous theorem by an example. In this example we use $\mathcal{B}$-fuzzy relations where $\mathcal{B}$ is a Boolean algebra. This is a special case of so-called $\mathcal{L}$-fuzzy relations where $\mathcal{L}$ is a Heyting algebra. For further details on these kind of fuzzy relations we refer to [24]. As already mentioned above every Boolean algebra is also a relation algebra where composition is given by the meet operation and converse is the identity. Let B_{abc} be the Boolean algebra with the three atoms a, b, c. We will denote arbitrary elements of B_{abc} by the sequence of atoms below that element, e.g., ab or bc or abc, or 0 for the least element. In order to create a relation algebra based on B_{abc} that has splittings we need to consider also the Boolean algebras of all elements smaller or equal a given element x of B_{abc}. We will denote this Boolean algebra by B_x. Now, the objects of the relation algebra are the Boolean algebras B_x for every $x \in B_{abc}$, and the morphisms between B_x and B_y are given by the Boolean algebra $B_{x \cap y}$, where $x \cap y$ is the intersection of the two sets of atoms x and y. For our example we consider the object $[B_{abc}, B_{abc}, B_{abc}, B_{abc}]$ and following partial equivalence relation in matrix form:

$$M = \begin{array}{c} \\ B_{abc} \\ B_{abc} \\ B_{abc} \\ B_{abc} \end{array} \begin{array}{c} \begin{array}{cccc} B_{abc} & B_{abc} & B_{abc} & B_{abc} \end{array} \\ \begin{pmatrix} ab & 0 & b & 0 \\ 0 & ab & 0 & a \\ b & 0 & bc & 0 \\ 0 & a & 0 & a \end{pmatrix} \end{array}$$

Following the proof of the theorem above we obtain the following four partial equivalence relations $\Xi : B_{abc} \to B_{abc}$.

$$\begin{aligned} \Xi_1 &:= ab, \\ \Xi_2 &:= ab \sqcap \overline{0 \sqcap 0} = ab \sqcap abc = ab, \\ \Xi_3 &:= bc \sqcap \overline{b \sqcap b} \sqcap \overline{0 \sqcap 0} = bc \sqcap ac \sqcap abc = c, \\ \Xi_4 &:= a \sqcap \overline{0 \sqcap 0} \sqcap \overline{a \sqcap a} \sqcap \overline{0 \sqcap 0} = a \sqcap abc \sqcap bc \sqcap abc = 0. \end{aligned}$$

The splittings for each of those relations is given by

$$R_1 = R_2 := ab : B_{ab} \to B_{abc}, \quad R_3 := c : B_c \to B_{abc}, \quad R_4 := 0 : B_0 \to B_{abc}.$$

Notice that the source object of each of those relations is different from B_{abc}. We obtain the matric N as:

$$N = \begin{array}{c} \\ B_{ab} \\ B_{ab} \\ B_c \\ B_0 \end{array} \begin{array}{c} \begin{array}{cccc} B_{abc} & B_{abc} & B_{abc} & B_{abc} \end{array} \\ \begin{pmatrix} ab & 0 & b & 0 \\ 0 & ab & 0 & a \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

An easy computation shows that N indeed splits M. Moreover, since B_0 is the trivial Boolean algebra with $0 = 1$ the last row of the matrix can actually be dropped from N. In the abstract language of relation algebras this corresponds to move from an object $A + 0$ to A where 0 is a null object, i.e., an object that is neutral (up to isomorphism) with respect to the relational sum.

4 Matrix algebras and Multi-valued Decision Diagrams

In this section we want to introduce multi-valued Decision Diagrams (MDDs) and how they can be used to implement heterogeneous relation using the matrix algebra approach. Decision diagrams are one of the contemporary symbolic data structures used to represent logic functions. A multiple-valued decision diagram is a natural extension of reduced ordered decision diagrams (ROBDD) [8] to the multi-valued case. MDDs are considered to be more efficient, and they perform better than ROBDDs with respect to memory size and path length [12].

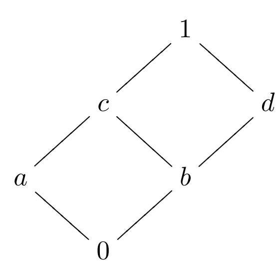
Let V be a set of finite size r. An r-valued function f is a function mapping V^n for some n to V. We will identify the n input values of f using a set of variables $X = \{x_0, x_1, ..., x_n\}$. Each x_i as well as $f(X)$ is r-valued, i.e., it represents an element from V. The function f can be represented by a multi-valued decision diagram. Such a decision diagram is directed acyclic graph (DAG) with up to r terminal nodes each labeled by a distinct value from V. Every non-terminal node is labeled by an input variable x_i and has r outgoing edges [11].

An MDD is ordered (OMDD) if there is an order on the set of variables X so that for every path from the root to a leave node all variables appear in that order. Furthermore, a MDD is called reduced if the graph does not contain isomorphic subgraphs and no nodes for which all r children are isomorphic. A MDD that is ordered and reduced is called a reduced ordered multi-valued decision diagram (ROMDD). Both ROBDDs and ROMDDs have widely been studied. Most of the techniques used when implementing a package for the creation and manipulation of ROMDDs are those already known from the binary case. These techniques includes edge negation, adjacent level interchange, operator nodes and logical operation [13].

MDDs are usually traversed in one of the following three ways:

1. A depth-first traversal starting at the top node and moving along the edges from each node to the descendants or child nodes. This technique is a very well-known conventional graph traversal.
2. ROMDDs can be traversed horizontally by moving from one node to another of all nodes labeled by the same variable. This corresponds to a specific breath-first traversal.
3. ROMDDs can also be traversed by applying both techniques described above at the same time.

We want to demonstrate by an example how relations can be implemented using MDDs. We will assume that relations are given as matrices. The elements of the matrices become the leaf or terminal nodes of the MDD after encoding the domain and the range of the relation by a suitable set of variables. In our example we want to use so-called $\mathcal{L}$-fuzzy relations as already mentioned in the previous section. Now consider the following Heyting algebra $\mathcal{L}$:



From the figure above we can derive operation tables for the meet and the join operation of the lattice $\mathcal{L}$. For example, we have $c \sqcap d = b$, $a \sqcap b = 0$, $a \sqcap c = a$, $1 \sqcup b = 1$, and $a \sqcup b = c$.

Now suppose that R and S are two $\mathcal{L}$-fuzzy relations represented by the following matrices, i.e., elements from $\mathcal{L}^+$:

$$R = \begin{pmatrix} a & b & a \\ 0 & 1 & c \end{pmatrix} \quad S = \begin{pmatrix} a & a & 1 \\ 0 & b & d \end{pmatrix}$$

Using the meet operation of the underlying lattice $\mathcal{L}$ we can compute the intersection (or meet) of R and S as:

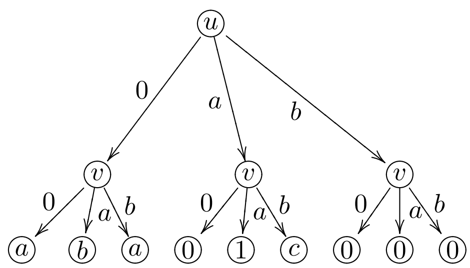
$$T := R \sqcap S = \begin{pmatrix} a & 0 & a \\ 0 & b & b \end{pmatrix}$$

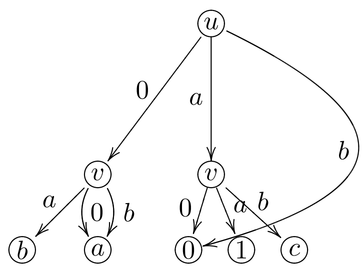
In order to implement these relations using MDDs we have to encode the corresponding matrices first as functions and then as graphs. We will adopt the method used in the RelView system for the binary to MDDs. First we have to encode the row and column indices by variables ranging over the lattice $\mathbb{L}$. Since $\mathbb{L}$ has 6 elements we need only one variable for the rows and the columns of R and S, respectively. However, in order to obtain a totally defined function we have to enlarge the matrices so that its size is a power of the number of elements, i.e., the corresponding function is defined for every possible input for each variable. We will fill the new entries with 0's. The figure below shows R enlarged to a proper size with its rows and columns labeled by values, i.e., by elements of the lattice $\mathbb{L}$, for the row variable u and the column variable v. In addition, the figure shows the corresponding function f_R encoding R where _ stands for an arbitrary parameter not listed before:

$$\begin{array}{c c} & \begin{array}{cccccc} 0 & a & b & c & d & 1 \end{array} \\ \begin{array}{c} 0 \\ a \\ b \\ c \\ d \\ 1 \end{array} & \begin{pmatrix} a & b & a & 0 & 0 & 0 \\ 0 & 1 & c & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

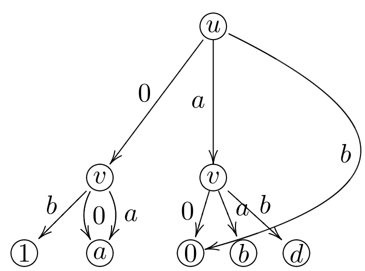
u	v	$f_R(u,v)$
0	0	a
0	a	b
0	b	a
a	0	0
a	a	1
a	b	c
_	_	0

Now based on the variable ordering $u < v$ we can produce a MDD representing f_R as shown on the left in the figure below. In addition, we present the corresponding reduced graph on the right-hand side of the figure. Notice that we only display the essential part of R in both graphs for brevity. The additional 0's are skipped:





Similarly, we construct the ROMDD for S shown below:



In order to compute the ROMDD for T from the ROMDDs for R and S we use an operation called `apply` on ROMDDs. This operation applies a function to a leaf node of R and the corresponding leaf node of S. Corresponding leaf nodes are determined by the same path from the root to the leaf in both graphs, i.e., the same spot in each matrix. The function that is applied to the leaf nodes is given by a table as mentioned above. For example, if we follow the edges labeled 0a (in that order) in the graph for R we get to a leaf labeled with b. In the graph of S we obtain a. The meet operation of $\mathbb{L}$ that is passed as parameter of `apply` will give 0, the leaf node of the path $0a$ in the graph of T.

The operation `apply` can be used to implement several operations of relations on their ROMDD representation. Another version of the `apply` operation applies an unary function to a each leaf node of a single ROMDD. This version can be used to implement relation algebraic operations such as complement and transpose. In addition to these apply operations we also use a so-called abstraction operation. This operation applies a function to all elements of an entire row or column of a matrix. It behave similar to a `fold` in a functional programming language, and it reduces the size of the matrix. For instance, if T is a $n \times m$ matrix and we apply the abstraction operation with the abstraction variable set to the entire set of row encoding variables, then the result of abstraction is a row vector, that is $1 \times m$ matrix. The product of two matrices can be computed by a combination of the apply and abstraction operations. We refer to [1, 13] for more details on operations on MDDs.

4.1 RelMDD - A Library for Manipulating Relations based on MDDs

The RelView system [2] implements relations in **Rel** using Boolean matrix represented as binary decision diagrams (BDDs) [7]. Our library RelMDD implements arbitrary heterogeneous relation algebras using the matrix algebra approach represented by ROMDDs. RelMDD is a library written in the programming language C. It is a package that can be imported by other programs and/or languages such as Java and Haskell when programming or manipulating arbitrary relations. The implementation is currently restricted to the basic operations of relation algebras, i.e., union, intersection, composition, converse, and complement. By design the package is capable of manipulating relations from both the classes of models, standard models and non-standard models of relation algebras. In our implementation MDDs were implemented using algebraic decision diagrams [1]. By taking this approach we were able to use a well-known package for these diagrams called CUDD [18].

5 Conclusion and Future Work

In this paper we have shown that splittings do exist in matrix algebras assuming that the underlying algebra of the coefficients provides this operation. This shows once more that it is sufficient to use matrices as a representation for arbitrary relation algebras. In addition, we have outlined an implementation of matrix algebras using ROMDDs. This implementation combines two major advantages over a regular array implementation of matrices. First of all, it is suitable for arbitrary relation algebras and is not restricted to the standard model **Rel**. In addition, it uses an advanced data structure that is known to work more efficient. The RelView system, in particular, has proven that an implementation of relations using decision diagrams is of great benefit.

The package implements all standard operations on relations. A future project will add further operations such as sums and splittings. The latter will then also allow to compute relational powers and so-called weak relational products [23]. Another project will be a suitable module for the programming language Haskell that makes the RelMDD package available in this language.

References

1. Bahar, I.R., Frohm, E.A., Gaona, C.M., Hachtel, G.D., Macii, E., Pardo, A., Somenzi, F.: Algebraic Decision Diagrams amd their Applications. Proceedings of the International Conference on Computer-Aided Design, IEEE 188-191 (1993).
2. Berghammer R., Neumann F.: RELVIEW - An OBDD-Based Computer Algebra System for Relations. CASC 2005, LNCS 3718, 40-50 (2005).
3. Berghammer, R.: Applying relation algebra and RelView to solve problems on orders and lattices. Acta Informatica 45 (3), 211-236 (2008).
4. Berghammer, R., Winter, M.: Embedding mappings and splittings with applications. Acta Informatica 47(2), 77-110 (2010).

5. Berghammer, R., Rusinowska, A., Swart, H.C.M. de: Applying relational algebra and RelView to measures in a social network. European Journal of Operational Research 202, 182-195 (2010).
6. Bird, R., de Moor, O.: Algebra of Programming. Prentice Hall (1997).
7. Brace, K.S., Rudell, L.R., Bryant E.R.: Efficient Implementation of a BDD Package. ACM/IEEE Design Automation Conference, IEEE 40-42 (1990).
8. Bryant, R.E.: Graph-based algorithms for Boolean function manipulation. IEEE Trans. on Computers 35(8), 677-691 (1986).
9. Chin, L.H., Tarski, A.: Distributive and modular laws in the arithmetic of relation algebras. University of California Press, Berkley and Los Angeles (1951).
10. Freyd, P., Scedrov, A.: Categories, Allegories. North-Holland (1990).
11. Miller, D.M., Drechsler, R.: Implementing a Multiple-Valued Decision Diagram Package. Proceedings of the 28th International Symposium on Multiple-Valued Logic (ISMVL '98), IEEE 52-57 (1998).
12. Nagayama, S., Sasao, T.: Code Generation for Embedded Systems using Heterogeneous MDDs. 12th Workshop on Synthesis and System Integration of Mixed Information Technologies (SASIMI 2003), 258-264 (2003).
13. Nagayama, S., Sasao, T.: Compact Representations of Logic Functions using Heterogeneous MDDs. Proceedings of the 33rd International Symposium on Multiple-Valued Logic (ISML'03), IEEE 3168-3175 (2003).
14. Balbiani, P., Orlowska, E.: A hierarchy of modal logics with relative accessibility relations. Journal of Applied Non-Classical Logics 9(2-3), (1999).
15. Schmidt, G., Ströhlein, T.: Relationen und Graphen. Springer (1989); English version: Relations and Graphs. Discrete Mathematics for Computer Scientists, EATCS Monographs on Theoret. Comput. Sci., Springer (1993).
16. Schmidt, G.: Relational Mathematics. Encyclopedia of Mathematics and its Applications 132, Cambridge University Press (2010).
17. Schmidt, G., Hattensperger, C., Winter, M.: Heterogeneous Relation Algebras. *In:* Brink C., Kahl W., Schmidt G. (eds.), Relational Methods in Computer Science, Advances in Computer Science, Springer Vienna (1997).
18. Somenzi, F.: CUDD: CU decision diagram package; Release 2.5.0. Department of Electrical, Computer, and Energy Engineering, University of Colorado. `http://vlsi.colorado.edu/ fabio/CUDD/cuddIntro.html` (2012).
19. Tarski, A., Givant, S.: A Formalization of Set Theory without Variables. Amer. Math. Soc. Colloq. Publ. 41 (1987).
20. Winter, M.: Strukturtheorie heterogener Relationenalgebren mit Anwendung auf Nichtdetermismus in Programmiersprachen. Dissertationsverlag NG Kopierladen GmbH, München (1998).
21. Winter, M.: Relation Algebras are Matrix Algebras over a suitable Basis. University of the Federal Armed Forces Munich, Report Nr. 1998-05 (1998).
22. Winter, M.: A Pseudo Representation Theorem for various Categories of Relations. TAC Theory and Applications of Categories 7(2), 23-37 (2000).
23. Winter, M.: Weak relational products. Relations and Kleene Algebra in Computer Science (RelMiCS'06/AKA'06), LNCS 4136, 417-431 (2006).
24. Winter, M.: Goguen Categories - A Categorical Approach to L-fuzzy Relations. Trends in Logic 25 (2007).
25. Zierer, H.: Relation Algebraic Domain Constructions. Theoret. Comput. Sci. 87, 163-188 (1991).