13th International Conference on Relational and Algebraic Methods in Computer Science (RAMiCS 13)

Relation Algebras, Matrices, and Multi-Valued Decision Diagrams

Francis Atampore and Dr. Michael Winter

Brock University, Department of Computer Science
13th International Conference on Relational and Algebraic Methods in Computer Science (RAMiCS 13)

September 20, 2012

Presentation Outline

Concrete Relation Algebra

Definition

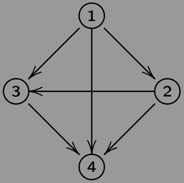
A concrete relation R between two sets A and B is a subset of the Cartesian product $A \times B$, where

$$A \times B = \{(a, b) : a \in A, b \in B\}$$

[3] [1]

Example

Let $A = \{1, 2, 3, 4\}$ and let R be a relation defined on A such that $R = \{(a, b) \mid a < b\}$, then the relation can be visualized as set given by:
$R = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}$.
R can also be visualized as a graph, a table or a boolean matrix as shown in the figures below:



R	1	2	3	4
1		×	×	×
2			×	×
3				×
4				

$$\begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

RelView System

"RELVIEW, developed in Kiel (formerly in Munich). RELVIEW is an interactive tool for computer-supported manipulation of relations represented as Boolean matrices or directed graphs, especially for prototyping relational specifications and programs " [2].

- RELVIEW implements relation in the form of boolean matrices using Ordered Binary Decision Diagrams (OBDDs).
- OBDDs are an efficient data structure used to implement very large Boolean functions.
- The first versions of RELVIEW were implemented using only arrays.
- A BDD is a directed acyclic graph (DAG) with decision nodes and two terminal nodes (0 and 1). Each decision node is labeled by a Boolean variable and has two child nodes (low child and high child).
- A BDD is called 'ordered' if different variables appear in the same order on all paths from the root to the leaf node. A BDD is said to be 'reduced' if there is no isomorphic subgraphs .

Heterogeneous relation algebras

A variant of the theory of relation which originated from category theory is called heterogeneous relation algebras. In this approach, relations have different source and target.

Definition

A (heterogeneous abstract) relation algebra is a locally small category $\mathcal{R}$ consisting of a class $Obj_{\mathcal{R}}$ of objects and a set $\mathcal{R}[A, B]$ of morphisms for all $A, B \in Obj_{\mathcal{R}}$. The morphisms are usually called relations. Composition is denoted by ";", identities are denoted by $\mathbb{I} \in \mathcal{R}[A, B]$, conversion
$\smile_{AB}: \mathcal{R}[A, B] \to \mathcal{R}[B, A]$
The operations satisfy the following rules:

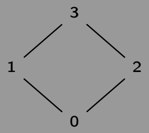
1. Every set $\mathcal{R}[A, B]$ carries the structure of a complete atomic boolean algebra with operations, $\sqcup_{AB}$, $\sqcap_{AB}$, $\overline{}_{AB}$, zero element $\perp\!\!\!\perp_{AB}$, universal element $\top\!\!\!\top_{AB}$, and inclusion ordering $\subseteq_{AB}$.
2. The Schröder equivalences

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

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

Example of Heterogeneous Relation Algebras

- Two Boolean algebra $\mathcal{L}_2$ and $\mathcal{L}_4$ with 2 and 4 elements.



-

Identity	
A	1
B	1

Bottom	
A, A	0
B, B	0
B, A	0
A, B	0

Top	
A, A	3
B, B	3
B, A	1
A, B	1

- Similarly, we can produce the tables for meet, composition, converse and complement.

- Union Operation

A, B	0	1
0	0	1
1	1	1

B, A	0	1
0	0	1
1	1	1

A, A	0	1	2	3
0	0	1	2	3
1	1	1	3	3
2	0	3	2	3
3	3	3	3	3

B, B	0	1	2	3
0	0	1	2	3
1	1	1	3	3
2	0	3	2	3
3	3	3	3	3

Problems and Motivations

1. Splittings exist in matrix algebras
2. Standard and Non-Standard models of relation algebras
    - (Point axiom) For every relation $R \neq \perp\!\!\!\perp$ there exit two points x, y such that $xy^{\smile} \subseteq R$
    - Concrete relations between sets form the standard model of relation algebras. But the theory has non-standard models.
    - RELVIEW works with the standard model. This system cannot visualize computations in non-standard models of the abstract theory of heterogeneous relations
    - That is, not all relation algebras can be represented as the algebra of Boolean matrices.

Example

- The relation $\top\!\!\!\top ; \top\!\!\!\top = \top\!\!\!\top$ is always true in RELVIEW since it uses the standard model. These properties might not be true in some non-standard models, hence RELVIEW is not a complete system for relation algebra.
- An advanced 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. $(P(A) \times P(B) \cong P(A + B))$

Matrix Algebras

Definition

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

- The class of objects of $\mathcal{R}^+$ is the collection of all functions from an arbitrary set I to $Obj_{\mathcal{R}}$.
- For every pair $f : I \to Obj_{\mathcal{R}}, g : J \to Obj_{\mathcal{R}}$ of objects from $\mathcal{R}^+$, the set of morphisms $\mathcal{R}^+[f,g]$ is the set of all functions $R : I \times J \to Mor_{\mathcal{R}}$ such that $R(i,j) \in \mathcal{R}[f(i), g(j)]$ holds.
- For $R \in \mathcal{R}^+[f,g]$ and $S \in \mathcal{R}^+[g,h]$ composition is defined by $(R;S)(i,k) := \bigsqcup R(i,j); S(j,k)$.
- For $R \in \mathcal{R}^+[f,g]$ conversion and negation is defined by $R^{\smile}(j,i) := (R(i,j))^{\smile}$, $\overline{R}(i,j) = \overline{R(i,j)}$
- For $R, S \in \mathcal{R}^+[f,g]$ union and intersection is defined by $(R \sqcup S)(i,j) = R(i,j) \sqcup S(i,j)$, $(R \sqcap S)(i,j) = R(i,j) \sqcap S(i,j)$
- The identity, zero and universal elements are defined by

$$\mathbb{I}_{f(i_1,i_2)} := \begin{cases} \perp\!\!\!\perp_{f(i_1)f(i_2)} & : i_1 \neq i_2 \\ \mathbb{I}_{f(i_1)} & : i_1 = i_2 \end{cases}$$

$$\perp\!\!\!\perp_{fg(i,j)} := \perp\!\!\!\perp_{f(i)g(j)}, \quad \mathbb{T}_{fg(i,j)} := \mathbb{T}_{f(i)g(j)}$$

[3].

A Pseudo Representation Theorem

Lemma

$\mathfrak{R}^+$ is a relation algebra

Definition

An object A of a relation algebra is called integral iff $\perp\!\!\!\perp_{AA} \neq \top\!\!\!\top_{AA}$ and for all $Q, R \in \mathcal{R}[A, A]$, the equation $Q; R = \perp\!\!\!\perp_{AA}$ implies either $Q = \perp\!\!\!\perp_{AA}$ or $R = \perp\!\!\!\perp_{AA}$.

Definition

Let $\{A_i | i \in I\}$ be a set of objects indexed by a set I. An object $\sum_{i\in I} A_i$. together with relations $\iota_j \in \mathcal{R}[A_j, \sum_{i\in I} A_i]$ for all $j \in I$ is called a relational sum of $\{A_i | i \in I\}$ iff for all $i, j \in$ I with $i \neq j$ the following holds: $\iota_i;\breve{\iota_i} = \mathbb{I}_{A_i}$ $\quad \iota_i;\breve{\iota_j} = \perp\!\!\!\perp_{A_iA_j}$ $\quad \bigsqcup_{i\in I} \iota_i;\breve{\iota_i} = \mathbb{I}_{\sum_{i\in I} A_i}$

Theorem

Let $\mathcal{R}$ be a relation algebra with relational sums and subobjects and $\mathcal{B}$ be the basis of $\mathcal{R}$. Then $\mathcal{R}$ and $\mathcal{B}^+$ are equivalent [7].

Definition

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

$$\psi; \psi^{\smile} = \mathbb{I}_{\boldsymbol{B}}, \qquad \psi^{\smile}; \psi = \Xi.$$

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

Definition

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

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

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

Theorem

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

- We call an object B a pre-power of A if there is 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 $\mathrm{syQ}(T, T)$

Theorem

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

Example

Let $M : f \to f$ be 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 $\mathcal{R}$. Define $g : \alpha \to \mathrm{Obj}_{\mathcal{R}}$ and $N : g \to f$ by

$$g(\beta) = A_\beta,$$

$$N(\beta,\gamma) = \begin{cases} R_\beta ; M(\beta,\gamma) & \text{iff } \beta \leq \gamma, \\ \bot_{A_\beta f(\gamma)} & \text{otherwise,} \end{cases}$$

For our example we consider the object $[B_{\boldsymbol{abc}}, B_{\boldsymbol{abc}}, B_{\boldsymbol{abc}}, B_{\boldsymbol{abc}}]$ and following partial equivalence relation in matrix form:

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

Example

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 matrix N as:

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

Objectives

- The proposed system will work with both standard and non-standard model.
- It will implement relation algebras in the form of matrix algebras.
- Make available a library to aid researchers to manipulate relations in their own programs.
- The proposed structure will be implemented in a packaged using C which can be imported into other programs and/or languages. A specific focus will be on a suitable module for the programming language Haskell.
- Understand how relations represented as matrices could be implemented using MDDs.

MDDs

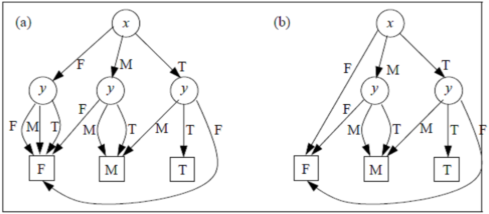
- A multiple-valued decision diagram (MDD) is a natural extension of the reduced ordered binary decision diagrams (ROBDDs) to the multiple-valued case
- MDDs are considered to be more efficient and require smaller memory size than the ROBDDs.
- Let V be a set of finite size r. An r-valued function f 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 $X = \{x_0, x_1, \ldots, x_n\}$. 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
- 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.
- MDD is called reduced if the graph does not contain isomorphic subgraphs and no nodes for which all r children are isomorphic (i.e. there is no redundant nodes in which two edges leaving a node points to the same next node within the graph).

Example MDDs

Let f be a multiple valued logic defined over the truth values T, M, F such that T = True, M = Maybe and F = False.

$\wedge$	F	M	T
F	F	F	F
M	F	M	M
T	F	M	T

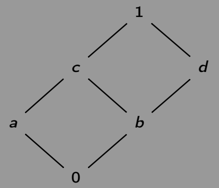
$\vee$	F	M	T
F	F	M	T
M	M	M	T
T	T	T	T



$f = y \wedge x$ (a)MDD; (b)ROMDD

Implementation

- Heyting algebra $\mathcal{L}$



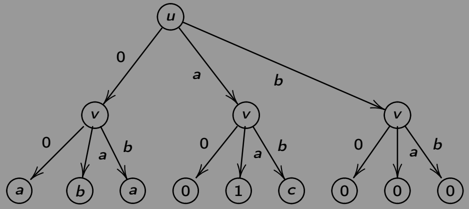
$$\begin{array}{c|cccccc} & 0 & a & b & c & d & 1 \\ \hline 0 & a & b & a & 0 & 0 & 0 \\ a & 0 & 1 & c & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 0 & 0 & 0 \\ c & 0 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

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

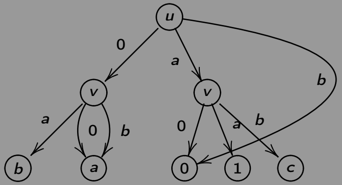
- Example: $c \sqcap d = b$, $a \sqcap b = 0$, $a \sqcap c = a$, $1 \sqcup b = 1$, and $a \sqcup b = c$
-

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

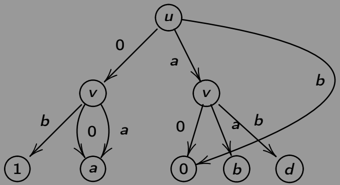
$$T := R \sqcap S = \begin{pmatrix} a & 0 & a \\ 0 & b & b \end{pmatrix}$$



Implementation



- Similarly, we construct the ROMDD for S:



- 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.
- Another version of the apply operation applies a unary function to a each leaf node of a single ROMDD
- abstraction operation: This operation applies a function to all elements of an entire row or column of a matrix

RelMDD-A Library for Manipulating Relations Based on MDDs

- RelMDD is an arbitrary relation algebra manipulator library written in C programming language.
- It is a package that implements relation algebra using the matrix algebra approach and can be imported by other programs and/or languages such as Java and Haskell when programming or manipulating arbitrary relations.
- We implemented RelMDD using the matrix algebra over a suitable basis approach, hence it is capable of manipulating both the class of standard and the non-standard models of relation algebra.
- It has an XML interface for loading the basis of relations into memory.
- It has several function for performing basic operations.
- The implementation is currently restricted to the basic operations of relation algebras, i.e., union, intersection, composition, converse, and complement.
- In our implementation MDDs were implemented using algebraic decision diagrams [2]. By taking this approach we were able to use a well-known package for these diagrams called CUDD [5].

Summary and Conclusion

Summary and Conclusion

- RELVIEW uses only standard procedures for OBDD manipulation.

Summary and Conclusion

- RELVIEW uses only standard procedures for OBDD manipulation.
- Relation algebra is made up of standard and non-standard models

Summary and Conclusion

- RELVIEW uses only standard procedures for OBDD manipulation.
- Relation algebra is made up of standard and non-standard models
- Relation algebra are Matrix algebra, hence we can implement relation algebra in a form of matrix algebra (not restricting ourselves to only boolean matrix.)

Summary and Conclusion

- RELVIEW uses only standard procedures for OBDD manipulation.
- Relation algebra is made up of standard and non-standard models
- Relation algebra are Matrix algebra, hence we can implement relation algebra in a form of matrix algebra (not restricting ourselves to only boolean matrix.)
- In order to implement a complete system for relation algebra we used a more advance representation of functions using ROMDDs.

Future Work And Suggestions

- This work would be extended by creating a suitable module for the programming language Haskell or Java.
- Extending the this system into a programming language for relations
- Integrating the package into the RELVIEW System.
- A future project will add further operations such as sums and splittings.

Reference

- Berghammer R. and Neumann F.: RELVIEW- An OBDD-Based Computer Algebra System for Relations CASC 2005, LNCS 3718, Springer 2005.
- 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).
- Gunther S. And Thomas S: Relations and Graphs ,Discrete Mathematics for Computer Scientists. Springer 1993
- Miller D. and Drechsler R. Implementing a Multiple-Valued Decision Diagram Package ISMVL-98.
- R.Bhuke , R. Berghammar,T. Hufmman, B.Leoniuk , P.Schueider: Application of Relview System
- Shinobu N. and Tsutomu S.: Compact Representation of Logic Functions using Heterogenous MDDs IEE 2003
- Winter M.: Relation Algebras are Matrix Algebras over a Suitable Basis Department of computer science, Univ. of Federal Armed Forces Munich ,Germany 1998

Reference

- www.cs.umbc.edu/ artola/fall02/Relations.ppt - United States
- http://www2.cs.unibw.de/Proj/relmics/html
- Zafor A.: Realm - a system to manipulate relations. Master's thesis, Brock University (2009).
- 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).
- 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).

Thank You