Center for Machine Perception presents

Computation in orthomodular lattices and algebras related to fuzzy logics

Mirko Navara (Praha)

Boolean algebras

When two formulas are equivalent? E.g.

Question 1:

$$a \vee (a' \wedge b) \stackrel{?}{=} a \vee b$$

Boolean algebras

When two formulas are equivalent? E.g.

Question 1:

$$a \vee (a' \wedge b) \stackrel{?}{=} a \vee b$$

- this can be decided by brute force in truth tables

a	b	$a \vee (a' \wedge b)$	$a \vee b$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

Boolean algebras

When two formulas are equivalent? E.g.

Question 1:

$$a \vee (a' \wedge b) \stackrel{?}{=} a \vee b$$

- this can be decided by brute force in truth tables

a	b	$a \vee (a' \wedge b)$	$a \vee b$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

better arrangement:

$a \backslash b$	0	1
0	0	1
1	1	1

Boolean algebras 2

Boolean algebras 2

- or transformation to (unique) normal forms

Boolean algebras 2

- or transformation to (unique) normal forms

- testing tautologies, not only by brute force, but

Boolean algebras 2

- or transformation to (unique) normal forms

- testing tautologies, not only by brute force, but

- by resolution principle

Boolean algebras 2

- or transformation to (unique) normal forms
- testing tautologies, not only by brute force, but
- by resolution principle
- simplification of formulas using **distributivity**

Boolean algebras 2

- or transformation to (unique) normal forms
- testing tautologies, not only by brute force, but
- by resolution principle
- simplification of formulas using **distributivity**
    - Karnaugh maps

Boolean algebras 2

- or transformation to (unique) normal forms

- testing tautologies, not only by brute force, but

- by resolution principle

- simplification of formulas using **distributivity**

    - Karnaugh maps

    - Svoboda maps

Boolean algebras 2

- or transformation to (unique) normal forms
- testing tautologies, not only by brute force, but
- by resolution principle
- simplification of formulas using **distributivity**
    - Karnaugh maps
    - Svoboda maps
    - Quine-McCluskey method, etc.

Quine-McCluskey method in Boolean algebras

Repeated use of the law $(\varphi \wedge a) \vee (\varphi \wedge a') = \varphi$

Quine-McCluskey method in Boolean algebras

Repeated use of the law $(\varphi \wedge a) \vee (\varphi \wedge a') = \varphi$

Example:

$$\begin{aligned}
&(a \wedge c) \vee (a \wedge b' \wedge c') \vee \underbrace{(a \wedge b \wedge c' \wedge d) \vee (a \wedge b \wedge c' \wedge d')} \\
&= (a \wedge c) \vee \underbrace{(a \wedge b' \wedge c') \vee \overbrace{(a \wedge b \wedge c')}} \\
&= \underbrace{(a \wedge c) \vee \overbrace{(a \wedge c')}} \\
&= a
\end{aligned}$$

Quine-McCluskey method in many-valued logic

[Petrík 04] Quine-McCluskey method for Gödel logic with all truth constants and crisp equality operation (=Kronecker delta)

Testing equations in many-valued logics - examples

$a \vee (a' \wedge b) \neq a \vee b$ in **Gödel logic with involutive negation**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

Testing equations in many-valued logics - examples

$a \vee (a' \wedge b) \neq a \vee b$ in **Gödel logic with involutive negation**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \oplus (a' \odot b) = a \vee b$ in **Łukasiewicz logic (MV-algebra)**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

Testing equations in many-valued logics - examples

$a \vee (a' \wedge b) \neq a \vee b$ in **Gödel logic with involutive negation**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \oplus (a' \odot b) = a \vee b$ in **Łukasiewicz logic (MV-algebra)**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \oplus (a' \odot b) \neq a \oplus b$ in **Łukasiewicz logic (MV-algebra)**

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1
1	1	1	1

$a \backslash b$	0	$\frac{1}{2}$	1
0	0	$\frac{1}{2}$	1
$\frac{1}{2}$	$\frac{1}{2}$	1	1
1	1	1	1

Semantical testing of tautologies

In **Boolean algebras**:

only a "small" search space: 2^n cases

$n =$ the number of different variables

Semantical testing of tautologies

In **Boolean algebras**:

only a "small" search space: 2^n cases

$n =$ the number of different variables

In **Gödel logic**: $(n+2)^n$ cases

Semantical testing of tautologies

In **Boolean algebras**:
only a "small" search space: 2^n cases
$n =$ the number of different variables

In **Gödel logic**: $(n+2)^n$ cases

In **Gödel logic with involutive negation**: $(2\,n+2)^n$ cases

Semantical testing of tautologies in Łukasiewicz logic (in MV-algebras)

It suffices to consider evaluations in

Semantical testing of tautologies in Łukasiewicz logic (in MV-algebras)

It suffices to consider evaluations in

- the standard MV-algebra $[0, 1]$ [Chang 58]

Semantical testing of tautologies in Łukasiewicz logic (in MV-algebras)

It suffices to consider evaluations in

- the standard MV-algebra $[0,1]$ [Chang 58]
- $\{0, \frac{1}{m}, \frac{2}{m}, \ldots, 1\}$, $\forall m \in \mathbb{N}$ [Chang 58]

Semantical testing of tautologies in Łukasiewicz logic (in MV-algebras)

It suffices to consider evaluations in

- the standard MV-algebra $[0,1]$ [Chang 58]
- $\{0, \frac{1}{m}, \frac{2}{m}, \ldots, 1\}$, $\forall m \in \mathbb{N}$ [Chang 58]
  still infinite;
  we need a **bound** for m

1st bound

M ... the number of all occurrences of variables in the formula

n ... the number of different variables in the formula

1st bound

M ... the number of all occurrences of variables in the formula
n ... the number of different variables in the formula

[Mundici 87]: $m \leq b_0(M) = 2^{(2M)^2} = 2^{4M^2}$

1st bound

M ... the number of all occurrences of variables in the formula
n ... the number of different variables in the formula

[Mundici 87]: $m \leq b_0(M) = 2^{(2M)^2} = 2^{4M^2}$

M	number of truth values-1
1	16
2	65 536
3	68 719 476 736
4	18 446 744 073 709 551 616
5	1267 650 600 228 229 401 496 703 205 376

1st bound

M ... the number of all occurrences of variables in the formula
n ... the number of different variables in the formula

[Mundici 87]: $m \leq b_0(M) = 2^{(2M)^2} = 2^{4M^2}$

M	number of truth values-1
1	16
2	65 536
3	68 719 476 736
4	18 446 744 073 709 551 616
5	1267 650 600 228 229 401 496 703 205 376

Complexity $\sum_{m=1}^{b_0(M)} (m+1)^n$

1st bound

M ... the number of all occurrences of variables in the formula
n ... the number of different variables in the formula

[Mundici 87]: $m \le b_0(M) = 2^{(2M)^2} = 2^{4M^2}$

M	number of truth values-1
1	16
2	65 536
3	68 719 476 736
4	18 446 744 073 709 551 616
5	1267 650 600 228 229 401 496 703 205 376

Complexity $\sum_{m=1}^{b_0(M)} (m+1)^n$

$M\backslash n$	1	2	3
1	152		
2	$2147\,581\,952$	$93\,831\,434\,829\,824$	
3	$2.361 \cdot 10^{21}$	$1.081 \cdot 10^{32}$	$5.575 \cdot 10^{42}$

2nd bound

"The importance of being a good teacher."

2nd bound

"The importance of being a good teacher."

[Aguzzoli, Ciabattoni, B. Gerla]: $m = b_1(M) = 2^{M-1}$

2nd bound

“The importance of being a good teacher."

[Aguzzoli, Ciabattoni, B. Gerla]: $m = b_1(M) = 2^{M-1}$

M	number of truth values-1
1	1
2	2
3	4
4	8
5	16
6	32
7	64

2nd bound

“The importance of being a good teacher."

[Aguzzoli, Ciabattoni, B. Gerla]: $m = b_1(M) = 2^{M-1}$

M	number of truth values-1
1	1
2	2
3	4
4	8
5	16
6	32
7	64

Complexity: $(b_1(M) + 1)^n$

$M \backslash n$	1	2	3	4	5
1	2				
2	3	9			
3	5	25	125		
4	9	81	729	6561	
5	17	289	4913	83 521	1419 857
6	33	1089	35 937	1185 921	39 135 393
7	65	4225	274 625	17 850 625	1160 290 625

3rd bound

[Aguzzoli, Ciabattoni, B. Gerla]: $m \leq b(M,n) = \left(\frac{M}{n}\right)^n$

3rd bound

[Aguzzoli, Ciabattoni, B. Gerla]: $m \leq b(M,n) = \left(\frac{M}{n}\right)^n$

$M\backslash n$	1	2	3	4	5
1	1				
2	2	1			
3	3	2	1		
4	4	4	2	1	
5	5	6	4	2	1
6	6	9	8	5	2
7	7	12	12	9	5

3rd bound

[Aguzzoli, Ciabattoni, B. Gerla]: $m \leq b(M,n) = \left(\frac{M}{n}\right)^n$

$M \backslash n$	1	2	3	4	5
1	1				
2	2	1			
3	3	2	1		
4	4	4	2	1	
5	5	6	4	2	1
6	6	9	8	5	2
7	7	12	12	9	5

Complexity $\sum_{m=1}^{b(M,n)} (m+1)^n$

$M \backslash n$	1	2	3	4	5
1	2				
2	5	4			
3	9	13	8		
4	14	54	35	16	
5	20	139	224	97	32
6	27	384	2024	2274	275
7	35	818	8280	25 332	12 200

$M \backslash n$	1	2	3	4	5
1	2				
2	5	4			
3	9	13	8		
4	14	54	35	16	
5	20	139	224	97	32
6	27	384	2024	2274	275
7	35	818	8280	25 332	12 200

This approach is preferable. As a by-product, we find the minimal denominator for which the formula is not a tautology.

$M \backslash n$	1	2	3	4	5
1	2				
2	5	4			
3	9	13	8		
4	14	54	35	16	
5	20	139	224	97	32
6	27	384	2024	2274	275
7	35	818	8280	25 332	12 200

This approach is preferable. As a by-product, we find the minimal denominator for which the formula is not a tautology.

Implemented by [Brůžková 05].

$M \backslash n$	1	2	3	4	5
1	2				
2	5	4			
3	9	13	8		
4	14	54	35	16	
5	20	139	224	97	32
6	27	384	2024	2274	275
7	35	818	8280	25 332	12 200

This approach is preferable. As a by-product, we find the minimal denominator for which the formula is not a tautology.

Implemented by [Brůžková 05].

For 2 variables, this bound is tough [MN].

Semantical testing in many-valued logics 2

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

- Testing of tautologies in basic logic?
  [Hájek; Haniková; Montagna, Pinna, and Tiezzi 03]

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

- Testing of tautologies in basic logic?
  [Hájek; Haniková; Montagna, Pinna, and Tiezzi 03]

Alternative approaches to testing of tautologies:

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

- Testing of tautologies in basic logic?
  [Hájek; Haniková; Montagna, Pinna, and Tiezzi 03]

Alternative approaches to testing of tautologies:

- Linear programming, mixed integer programming

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

- Testing of tautologies in basic logic?
  [Hájek; Haniková; Montagna, Pinna, and Tiezzi 03]

Alternative approaches to testing of tautologies:

- Linear programming, mixed integer programming

The task can be directly translated to a system of linear equalities and inequalities.

Semantical testing in many-valued logics 2

- Testing of satisfiability in Łukasiewicz logic? **Still a problem.**

- Testing of tautologies in basic logic?
  [Hájek; Haniková; Montagna, Pinna, and Tiezzi 03]

Alternative approaches to testing of tautologies:

- Linear programming, mixed integer programming

The task can be directly translated to a system of linear equalities and inequalities.

- Hypersequent calculus by [Ciabattoni, Fermüller, and Metcalfe 05] allows to test tautologies in Gödel and product logics as well.

Semantical testing in many-valued logics 3

- Looking for counterexamples, a random search need not be a bad alternative [Brůžková 05].

Semantical testing in many-valued logics 3

- Looking for counterexamples, a random search need not be a bad alternative [Brůžková 05].

May give a **negative answer**.

Semantical testing in many-valued logics 3

- Looking for counterexamples, a random search need not be a bad alternative [Brůžková 05].

  May give a **negative answer**.

- Syntactical prover [Lehmke 05]
http://ls1-www.cs.uni-dortmund.de/~lehmke/SimpleProver

Semantical testing in many-valued logics 3

- Looking for counterexamples, a random search need not be a bad alternative [Brůžková 05].

May give a **negative answer**.

- Syntactical prover [Lehmke 05]
http://ls1-www.cs.uni-dortmund.de/~lehmke/SimpleProver

May give a **positive answer**.

Testing equations in orthomodular lattices

Testing equations in orthomodular lattices

Mostly based on free algebras.

Training site for free algebras: Boolean algebras

$a \vee (a' \wedge b) = a \vee b$

.

$a \backslash b$	0	1
0	0	1
1	1	1

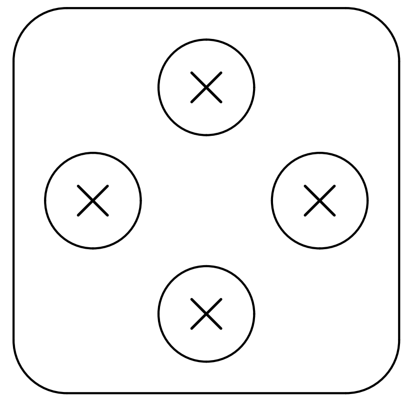
Training site for free algebras: Boolean algebras

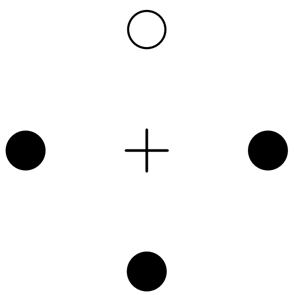
$a \vee (a' \wedge b) = a \vee b$

$\nearrow$

$a \backslash b$	0	1
0	0	1
1	1	1

2^4





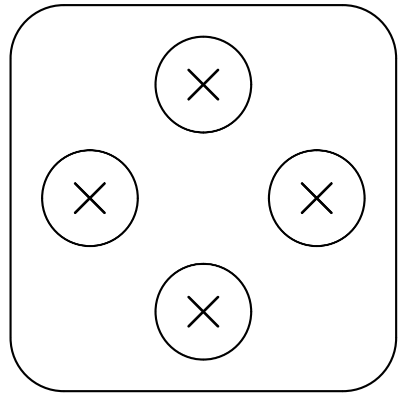
Training site for free algebras: Boolean algebras

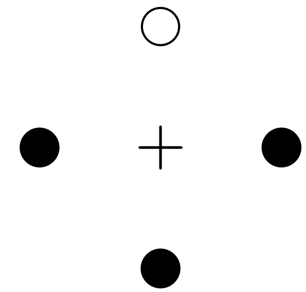
2^4

$a \vee (a' \wedge b) = a \vee b$

$\nearrow$

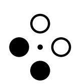
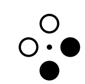
$a \backslash b$	0	1
0	0	1
1	1	1

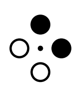
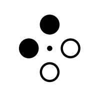


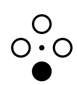
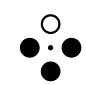


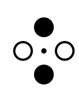
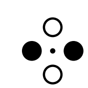
Everything is seen in a **"good" Venn diagram** $=$ **free Boolean algebra with n free generators** $= 2^n$

All $2^4 = 16$ binary Boolean operations represented by subsets of a 4-element set:

$a = a$  b, $\quad b = a$  b

$a' = a$  b, $\quad b' = a$  b

$a \wedge b = a$  b, $\quad a \vee b = a$  b

$(a \wedge b) \vee (a' \wedge b') = a$  b, $\quad (a \wedge b') \vee (a' \wedge b) = a$  b

Training site for free algebras: Boolean algebras

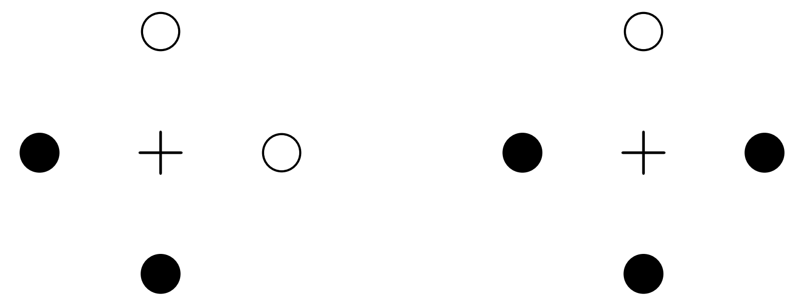
Binary Boolean operations can be combined:

$$
\begin{aligned}
a \vee (a' \wedge b) &= \left(a \underset{\bullet}{\overset{\circ}{\bullet\cdot\circ}} b\right) \underset{\bullet}{\overset{\circ}{\bullet\cdot\bullet}} \left(\left(a \underset{\circ}{\overset{\bullet}{\circ\cdot\bullet}} b\right) \underset{\bullet}{\overset{\circ}{\circ\cdot\circ}} \left(a \underset{\bullet}{\overset{\circ}{\circ\cdot\bullet}} b\right)\right) = \\
&= \left(a \underset{\bullet}{\overset{\circ}{\bullet\cdot\circ}} b\right) \vee \left(\left(a \underset{\circ}{\overset{\bullet}{\circ\cdot\bullet}} b\right) \wedge \left(a \underset{\bullet}{\overset{\circ}{\circ\cdot\bullet}} b\right)\right) = \\
&= \left(a \underset{\bullet}{\overset{\circ}{\bullet\cdot\circ}} b\right) \vee \left(a \underset{\circ}{\overset{\circ}{\circ\cdot\bullet}} b\right) = \\
&= a \underset{\bullet}{\overset{\circ}{\bullet\cdot\bullet}} b = a \vee b
\end{aligned}
$$

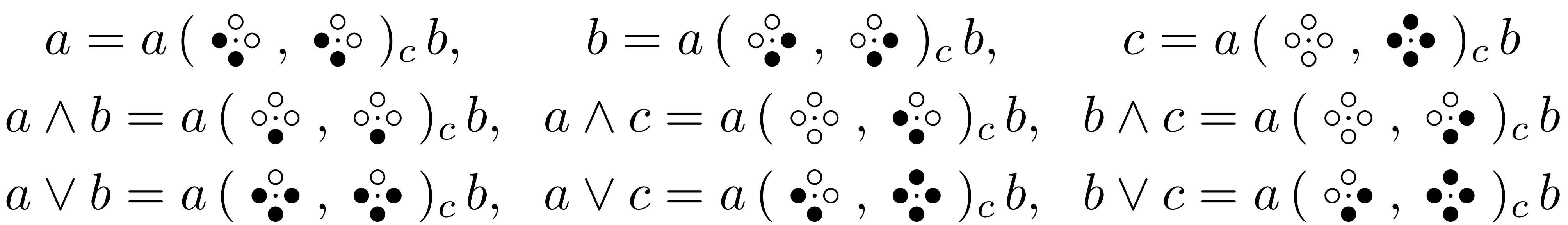
Training site for free algebras: Boolean algebras

Example with 3 variables – distributivity: $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$

c	0	0	1	1
$a \backslash b$	0	1	0	1
0	0	0	0	1
1	1	1	1	1



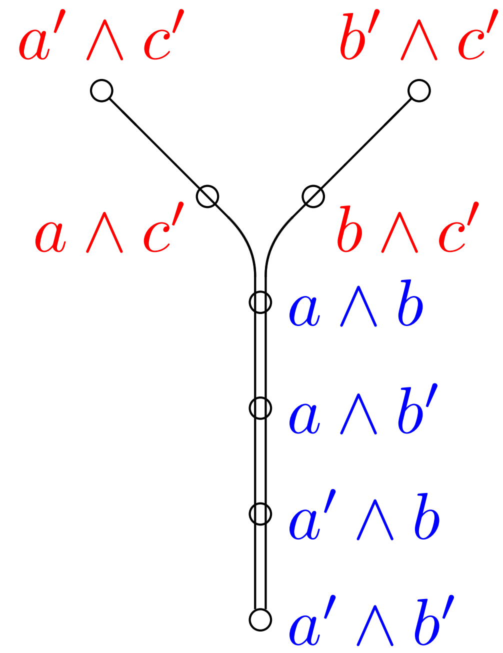
2^3 represented by subsets of an 8-element set:



Testing equations in orthomodular lattices

Free OML with 2 free generators $= F(a,b) \cong 2^4 \times \mathrm{MO2}$

Greechie diagram:

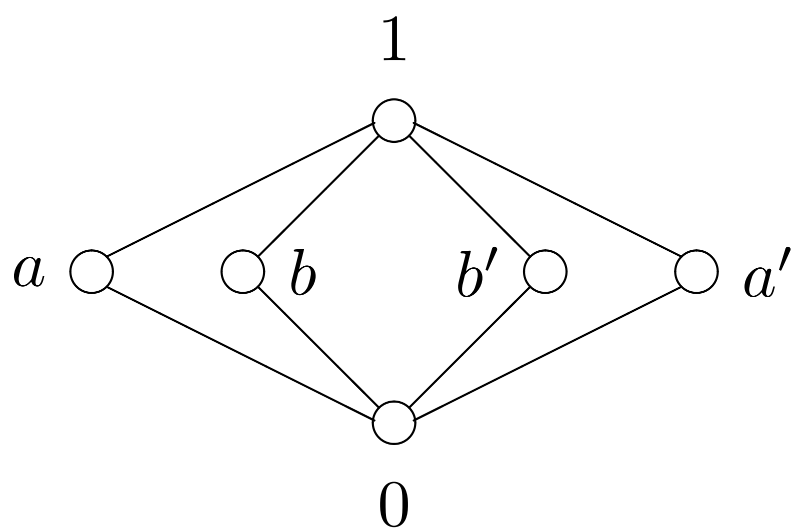


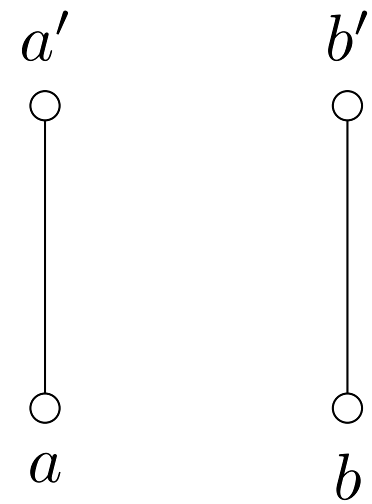
$$c' = (a \wedge b) \vee (a \wedge b') \vee (a' \wedge b) \vee (a' \wedge b')$$

1st factor $= 2^4$ (Boolean algebra)

2nd factor $= \mathrm{MO2}$

Computation in $\mathrm{MO2}$



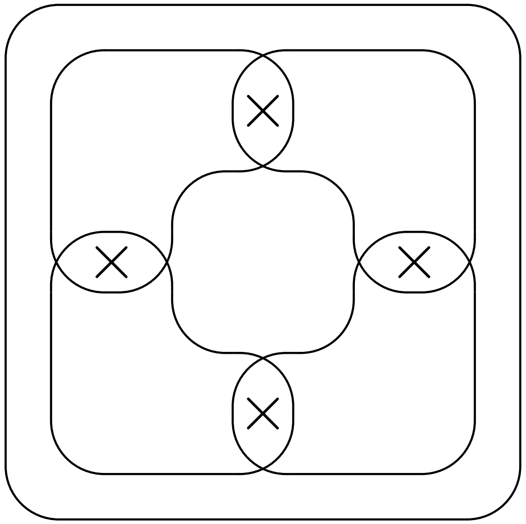


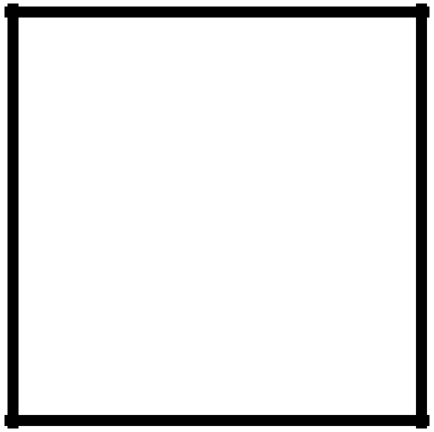
$$a \wedge b = a \wedge b' = a' \wedge b = a' \wedge b' = 0$$

$$a \vee b = a \vee b' = a' \vee b = a' \vee b' = 1$$

Computation in $\mathrm{MO2}$

$\mathrm{MO2}$ is also represented by **some** subsets of a 4-element set:





$0 = a \; \cdot \; b,$ $\quad a = a \llcorner\!\cdot\; b,$ $\quad b = a \;\cdot\!\lrcorner\, b$

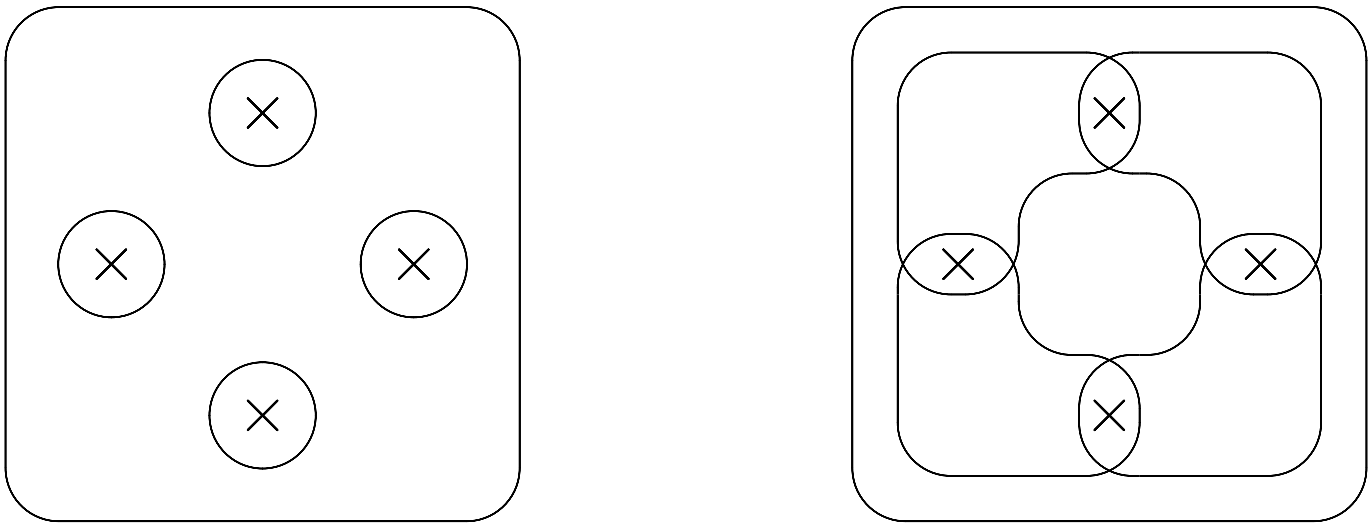
$1 = a \boxdot b,$ $\quad a' = a \;\cdot\!\urcorner\, b,$ $\quad b' = a \ulcorner\!\cdot\; b$

$a \wedge b = a \; \cdot \; b,$ $\quad a \vee b = a \boxdot b$

Testing equations in orthomodular lattices

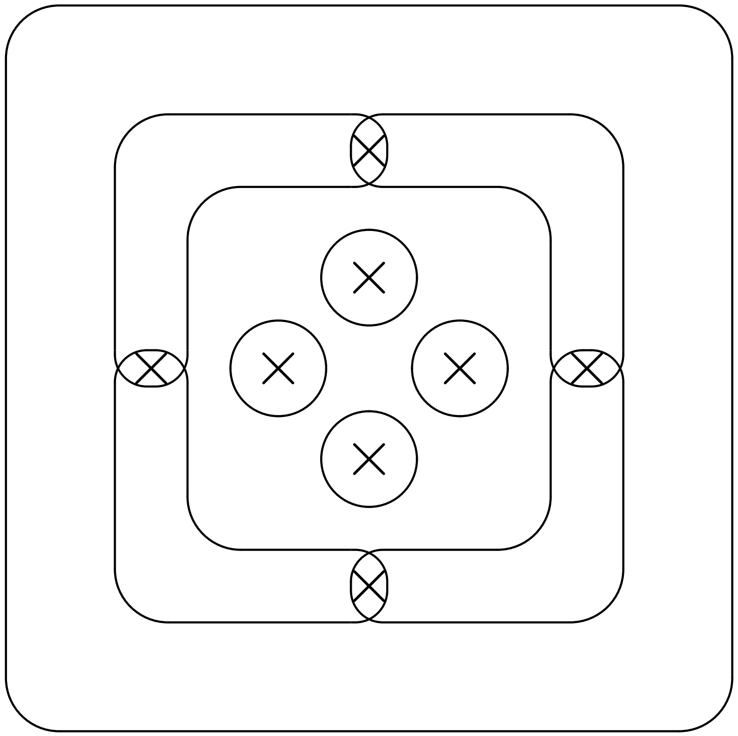
$F(a,b)$ is represented by **some** subsets of an 8-element set:

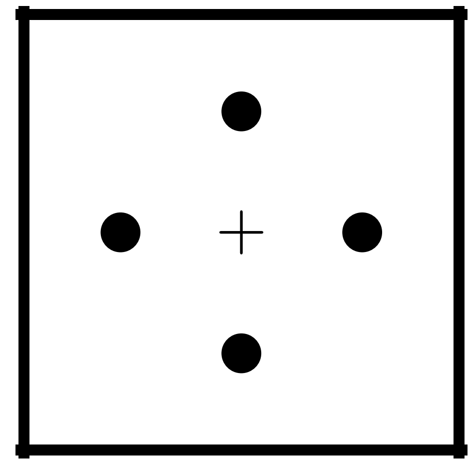
$$2^4 \qquad \times \qquad \mathrm{MO2}$$

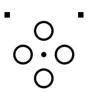
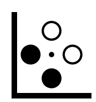
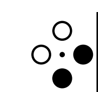


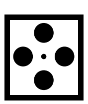
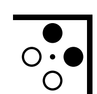
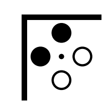
Testing equations in orthomodular lattices

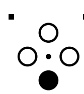
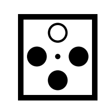
$2^4 \times \mathrm{MO2}$





$0 = a$  b, $a = a$  b, $b = a$  b

$1 = a$  b, $a' = a$  b, $b' = a$  b

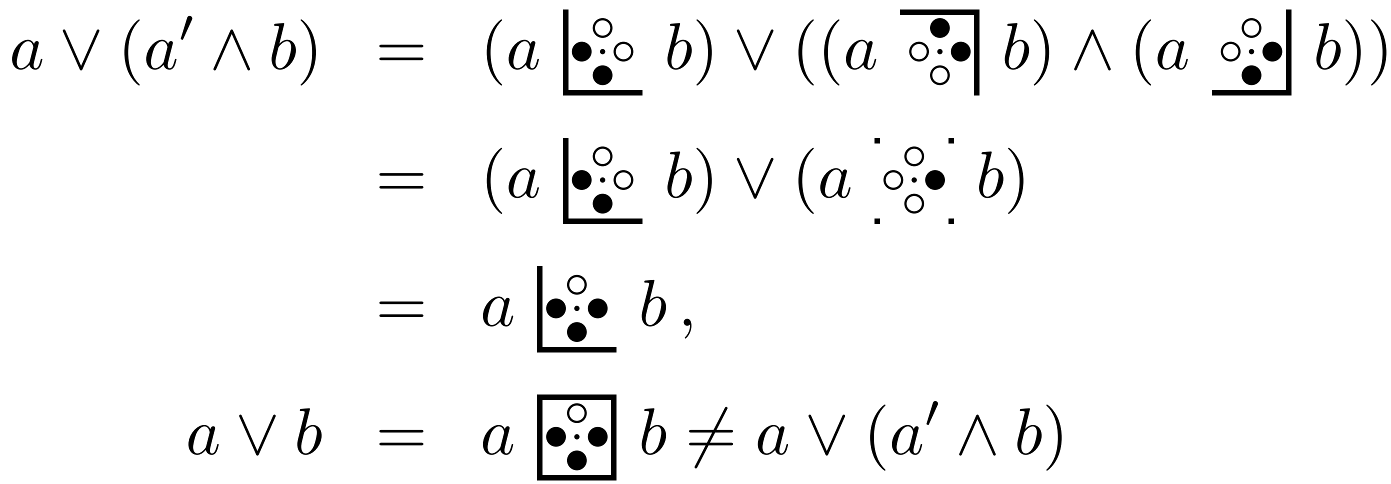
$a \wedge b = a$  b, $a \vee b = a$  b

Testing equations in orthomodular lattices

$$a \vee (a' \wedge b) \stackrel{?}{=} a \vee b$$

Testing equations in orthomodular lattices

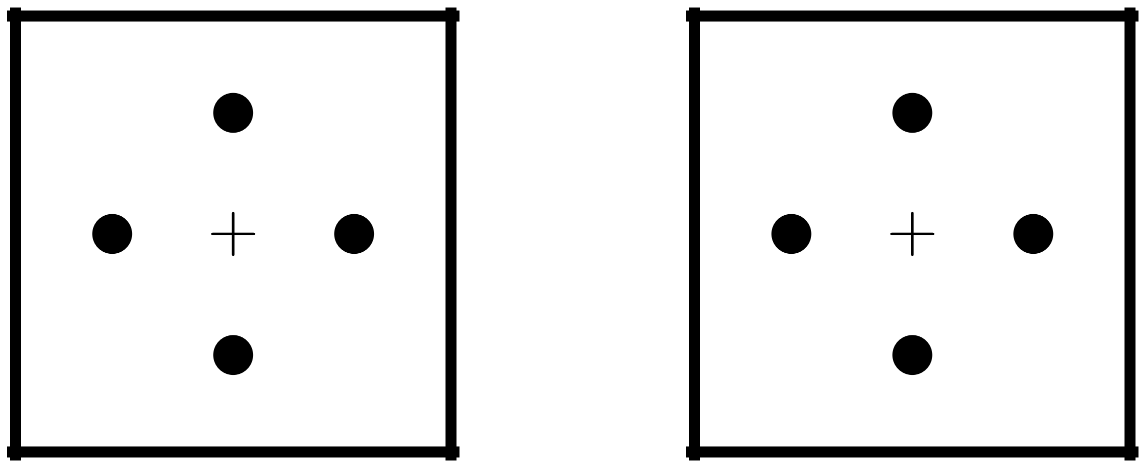
$$a \vee (a' \wedge b) \stackrel{?}{=} a \vee b$$

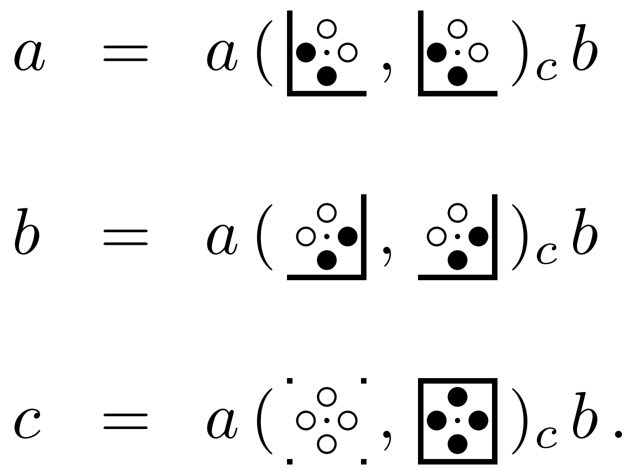


Testing equations in orthomodular lattices

We may admit further variables which commute with all others.

c commutes with $a, b \;\Rightarrow\; F(a,b,c) \cong F(a,b) \times F(a,b)$ is represented by **some** subsets of a 16-element set:





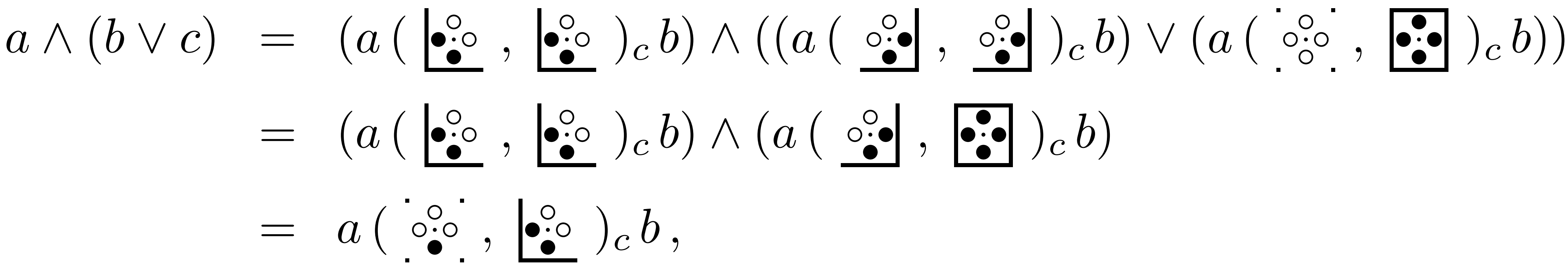
Foulis–Holland Theorem

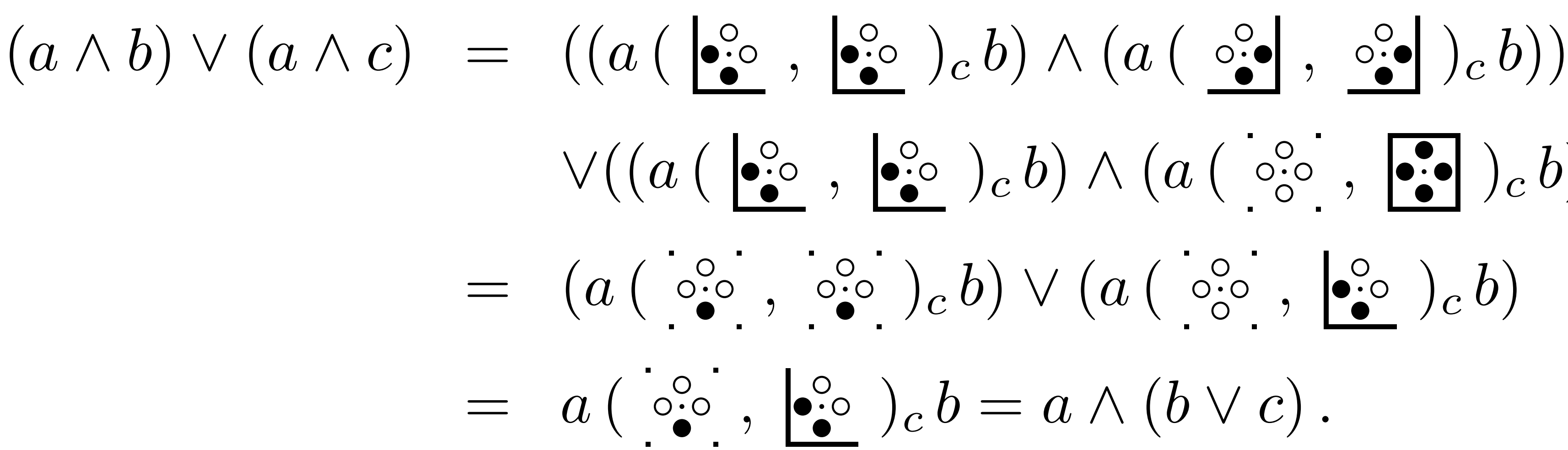
c commutes with $a, b \;\Rightarrow\; a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)\,.$

Foulis–Holland Theorem

c commutes with $a, b \;\Rightarrow\; a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)\,.$

Proof:





Testing equations in orthomodular lattices

Automatic prover: http://www.mat.savba.sk/~hycko/oml

Example: Associativity equations with 2 variables:

$$
\begin{aligned}
(a*a)*b &= a*(a*b) \\
(a*a')*b &= a*(a'*b) \\
(a*b)*b &= a*(b*b) \\
(a*b')*b &= a*(b'*b) \\
(a*b)*a &= a*(b*a) \\
(a*b)*a' &= a*(b*a')
\end{aligned}
$$

All can be tested for one binary OML operation $*$ by a single command, e.g.

B3(54,B3(54,a,a),b)=B3(54,a,B3(54,a,b)) AND
B3(54,B3(54,a,a'),b)=B3(54,a,B3(54,a',b)) AND
B3(54,B3(54,a,b),b)=B3(54,a,B3(54,b,b)) AND
(B3(54,B3(54,a,b'),b)=B3(54,a,B3(54,b',b))) AND
(B3(54,B3(54,a,b),a)=B3(54,a,B3(54,b,a))) AND
(B3(54,B3(54,a,b),a')=B3(54,a,B3(54,b,a')))

B3(92,B3(92,a,a),b)=B3(92,a,B3(92,a,b)) AND

B3(92,B3(92,a,a'),b)=B3(92,a,B3(92,a',b)) AND

B3(92,B3(92,a,b),b)=B3(92,a,B3(92,b,b)) AND

(B3(92,B3(92,a,b'),b)=B3(92,a,B3(92,b',b))) AND

(B3(92,B3(92,a,b),a)=B3(92,a,B3(92,b,a))) AND

(B3(92,B3(92,a,b),a')=B3(92,a,B3(92,b,a')))

B3(92,B3(92,a,a),b)=B3(92,a,B3(92,a,b)) AND

B3(92,B3(92,a,a'),b)=B3(92,a,B3(92,a',b)) AND

B3(92,B3(92,a,b),b)=B3(92,a,B3(92,b,b)) AND

(B3(92,B3(92,a,b'),b)=B3(92,a,B3(92,b',b))) AND

(B3(92,B3(92,a,b),a)=B3(92,a,B3(92,b,a))) AND

(B3(92,B3(92,a,b),a')=B3(92,a,B3(92,b,a')))

Another prover by [Megill and Pavičić].

Focusing technique [Greechie 1977]

Weaker assumption:
Every variable may not commute with at most one other variable.

$\Rightarrow$

the free **lattice** (not the free OML!) generated by these variables is distributive.

Focusing technique [Greechie 1977]

Weaker assumption:
Every variable may not commute with at most one other variable.

$\Rightarrow$

the free **lattice** (not the free OML!) generated by these variables is distributive.

It does not allow to combine a variable and its orthocomplement.

Focusing technique [Greechie 1977]

Weaker assumption:
Every variable may not commute with at most one other variable.

$\Rightarrow$

the free **lattice** (not the free OML!) generated by these variables is distributive.

It does not allow to combine a variable and its orthocomplement.

For $n = 3$,
Greechie focusing technique is applicable to 18 expressions,
our approach to $96^2 = 9216$ expressions.

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Our approach does not cover this case; can it be extended?

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Our approach does not cover this case; can it be extended?

Partial answers: The free OML $F(a, b, c, d)$ with these generators is a product of

- a (big) Boolean algebra,

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Our approach does not cover this case; can it be extended?

Partial answers: The free OML $F(a, b, c, d)$ with these generators is a product of

- a (big) Boolean algebra,
- a product $(\mathrm{MO2})^k$,

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Our approach does not cover this case; can it be extended?

Partial answers: The free OML $F(a, b, c, d)$ with these generators is a product of

- a (big) Boolean algebra,
- a product $(\mathrm{MO2})^k$,
- a more complex factor.

Topics for future research

The focusing technique admits 4 variables a, b, c, d, where only a, b and c, d form non-commuting pairs.

Our approach does not cover this case; can it be extended?

Partial answers: The free OML $F(a, b, c, d)$ with these generators is a product of

- a (big) Boolean algebra,
- a product $(\mathrm{MO2})^k$,
- a more complex factor.

We do not know if it is finite.

Summary of progress

Case study: Foulis–Holland Theorem

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

00's: Computer programs.

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

00's: Computer programs.

10's: ???

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

00's: Computer programs.

10's: ???

(It is up to you.)

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

00's: Computer programs.

10's: ???

(It is up to you.)

TFYA

Summary of progress

Case study: Foulis–Holland Theorem

60's: Separate papers devoted to the result.

70's: Focusing technique solves it together with many other results.

80's: Free OMLs allow to clarify it.

90's: Graphical tool.

00's: Computer programs.

10's: ???

(It is up to you.)

TFYA

(if any)