

# Strong Duality in Horn minimization

---

Endre Boros, Ondřej Čepek, Kazuhisa Makino  
Charles University, Prague, Czech Republic

Fundamentals of Computation Theory (FCT 2017)  
Bordeaux, France, September 13, 2017

# Digraphs, implications, Horn 2-CNFs

---



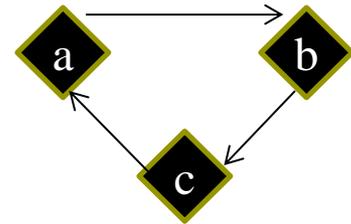
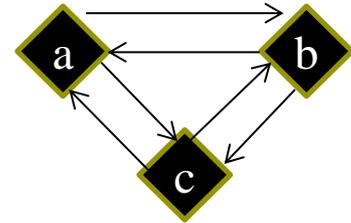
- List of arcs
  - $(a,b), (b,a), (b,c), (c,b)$
- Neighbor lists
  - $a : b$
  - $b : a,c$
  - $c : b$
- Adjacency matrix

0	1	0
1	0	1
0	1	0

# Digraphs, implications, Horn 2-CNFs

---

- Directed paths
- Transitivity
- Transitive closure
  - Can be computed e.g. by DFS
  - Defines equivalence relation
- Transitive reduction
  - Minimum equivalent graph
  - Can be computed in polynomial time [Tarjan 74]
  - Notice: not enough to drop arcs !!



# Digraphs, implications, Horn 2-CNFs

---

- Directed graph  $\leftrightarrow$  Set of simple implications (rules)
- $a \rightarrow b, b \rightarrow a, b \rightarrow c, c \rightarrow b$
- $a \rightarrow b, b \rightarrow ac, c \rightarrow b$
- Transitivity  $\leftrightarrow$  Logical deduction
- Transitive closure  $\leftrightarrow$  Set of all logically deducible implications (rules), implicational closure
- Transitive reduction  $\leftrightarrow$  Minimum set of implications (rules) representing the same knowledge, minimum implicational base

# Digraphs, implications, Horn 2-CNFs

---

- Another equivalent concept: pure Horn 2-CNF

$$(\neg a \vee b) \wedge (\neg b \vee a) \wedge (\neg b \vee c) \wedge (\neg c \vee b)$$

- two quadratic clauses are resolvable if they have exactly one conflicting literal producing a resolvent
- if  $C_1 = \neg a \vee x$ ,  $C_2 = \neg x \vee b$  then  $R(C_1, C_2) = \neg a \vee b$
- Transitivity, Deduction  $\leftrightarrow$  Resolution
- Transitive closure, Implicational closure  $\leftrightarrow$  Resolution closure
- Transitive reduction, Minimum base  $\leftrightarrow$  Shortest equivalent 2-CNF

# Directed hypergraphs, implicational systems, pure Horn CNFs

---

- Generalizations of directed graphs, simple implications, and pure Horn 2-CNFs where antecedents are no longer singletons
- Directed hypergraph:  $(\{a\}, b), (\{b\}, a), (\{a, c\}, d), (\{a, c\}, e)$  or  $\{a\} : b, \{b\} : a, \{a, c\} : d, e$
- Implicational system:  $a \rightarrow b, b \rightarrow a, ac \rightarrow d, ac \rightarrow e$  or  $a \rightarrow b, b \rightarrow a, ac \rightarrow de$
- Pure Horn CNF:  
 $(\neg a \vee b) \wedge (\neg b \vee a) \wedge (\neg a \vee \neg c \vee d) \wedge (\neg a \vee \neg c \vee e)$

# Directed hypergraphs, implicational systems, pure Horn CNFs

---

- Derivation rules
  - Generalized transitivity
  - Logical deduction – Forward chaining
  - Resolution (general form)
- Complexity changes dramatically from the simple case of graphs, simple implications, and 2-CNFs
  - Closures can be of exponential size
  - Reductions are usually hard to compute

# Minimal CNFs: number of clauses

---

- $\Sigma_2$ -complete for general CNFs [Umans 2001]
- NP-complete ( $\Sigma_1$ -complete) for pure Horn CNFs [Ausiello, D'Atri, Sacca 1986]
- NP-complete for cubic pure Horn CNFs [Boros, Gruber 2012, Boros, Čepek, Kučera 2013]
- Polynomial for quadratic CNFs [folklore]
- Polynomial for acyclic and quasi-acyclic Horn CNFs [Hammer, Kogan 1995]
- Polynomial for component-wise quadr. Horn CNFs [Boros, Čepek, Kogan, Kučera 2010]
- All above results true also for total number of literals<sub>8</sub>

# Minimal CNFs: number of source sets

---

- Number of adjacency lists for directed hypergraphs
- Number of rules for implicational systems
- Does not make sense for general CNFs
- Polynomial for pure Horn CNFs [Maier 1980]
- Similar result was independently discovered in the implicational systems community: GD-basis [Guigues, Duquenne 1986]
- Both algorithms can be used for pure Horn CNFs

# Minimal CNFs: number of clauses

## Upper and lower bounds

---

- Given a CNF representing a function  $f$ , can we estimate the number of clauses in a minimal representation of  $f$  ?
- Upper bounds: every CNF representation of  $f$  gives an upper bound
- Lower bounds: ???
- Verification of minimality: ???

# Boolean basics

---

- Clause  $C$  is an implicate of function  $f$  if  $f \leq C$
- $C$  is a prime implicate of  $f$  if dropping any literal means that  $C$  is no longer an implicate of  $f$
- prime CNF, irredundant CNF
- two clauses are resolvable if they have exactly one conflicting literal producing a resolvent
- if  $C_1 = A \vee x$ ,  $C_2 = B \vee \neg x$  then  $R(C_1, C_2) = A \vee B$
- $R(S)$  is a resolution closure of set  $S$  of clauses

# Notation

---

- $P(f)$  is a set of all prime implicants of  $f$
- $P^*(f) = \mathbf{R}(P(f))$
- Recall: completeness of resolution [Quine 1955]:  
for any CNF representation  $S \subseteq P^*(f)$  of a function  $f$   
we have  $\mathbf{R}(S) = P^*(f)$  (and hence  $P(f) \subseteq \mathbf{R}(S)$ )

# Essential sets of implicates: definition

---

- Let  $f$  be a Boolean function. Then  $X \subseteq P^*(f)$  is an essential set of  $f$  if for every two resolvable clauses  $C_1, C_2 \in P^*(f)$  the following implication holds:

$$R(C_1, C_2) \in X \Rightarrow C_1 \in X \text{ or } C_2 \in X$$

- Example 1:  $S \subseteq P^*(f)$  s.t.  $S = \mathbf{R}(S)$ ,  $X = P^*(f) \setminus S$   
(can serve as an alternative definition)
- Example 2:  $t \in \{0,1\}^n$ ,  $X(t) = \{C \in P^*(f) \mid C(t) = 0\}$   
(false-point essential sets)

# Essential sets of implicates: properties

---

- Theorem [Boros, Čepek, Kogan, Kučera 2008]:  
Let  $S \subseteq P^*(f)$  be arbitrary. Then  $S$  represents  $f$  if and only if  $S \cap X \neq \emptyset$  for every nonempty essential set  $X \subseteq P^*(f)$ .
- Observation: If  $P^*(f)$  contains  $k$  pair-wise disjoint nonempty essential sets then every CNF representation of  $f$  consists of at least  $k$  clauses

# Essential sets of implicates: properties

---

- Definition: For a function  $f$  let  $\text{cnf}(f)$  denote the minimum number of clauses in a CNF representation of  $f$  and  $\text{ess}(f)$  the maximum number of pair-wise disjoint nonempty essential sets of  $f$ .
- Corollary (weak duality): For every function  $f$ :  
$$\text{ess}(f) \leq \text{cnf}(f).$$

# Essential sets: verifiable lower bounds

---

- Gap: There exists a cubic Horn function on 4 variables for which  $\text{ess}(f) = 4$  and  $\text{cnf}(f) = 5$ .
- More on the size of this gap can be found in [Hellerstein, Kletenik 2011]
- Definition: For a function  $f$  let  $\text{ess}^*(f)$  denote the maximum number of vectors  $\mathbf{t}$  such that  $\mathbf{X}(\mathbf{t})$ 's are pairwise disjoint nonempty essential sets of  $f$ .
- Theorem: [Čepek, Kučera, Savický 2009]  
For every function  $f$ :  $\text{ess}^*(f) = \text{ess}(f)$ .
- $\text{cnf}(f) \geq \text{ess}^*(f) \geq k$  can be verified by listing  $k$  vectors (falsepoints of  $f$ ) and checking disjointness of  $\mathbf{X}(\mathbf{t})$ 's<sup>16</sup>

# Body disjoint essential sets

---

- Definition: Let  $f$  be a pure Horn function. Two essential sets  $X, Y \subseteq P^*(f)$  are called body disjoint if there is no pair of clauses  $C \in X$  and  $C' \in Y$  such that  $C$  and  $C'$  have the same body (source set).
- Definition: For a pure Horn function  $f$  let  $\text{body}(f)$  denote the minimum number of bodies (source sets) in a CNF representation of  $f$  and let  $\text{bess}(f)$  denote the maximum number of pair-wise body disjoint nonempty essential sets of  $f$ .
- Weak duality: For every pure Horn function  $f$ :  
$$\text{bess}(f) \leq \text{body}(f).$$

# Body disjoint essential sets

---

- Theorem (strong duality): [Boros, Čepek, Makino]  
 $\text{bess}(f) = \text{body}(f)$ .
- Again it is sufficient to consider falsepoints: for any two falsepoints the body-disjointness of their essential sets can be tested in polynomial time.
- This result in some sense explains why the number of bodies (source sets) is the only „measure“ for which pure Horn minimization is poly-time solvable.

# Algorithmic consequences

---

- Strong duality implies a conceptually very simple minimization algorithm
- 1. Right saturation (use forward chaining on every source set to add all logical consequences to the right hand side of the list)
- 2. Drop redundant lists
- Overall (asymptotic) complexity same as previous algorithms, the output is not unique, but fewer steps are required.



---

Thank you.