Refining Schaefer's Theorem

Heribert Vollmer

Institut für Theoretische Informatik
Universität Hannover

Joint work with: E. Allender, M. Bauland, N. Immerman, H. Schnoor

Γ – a finite set of Boolean constraint relations

CSP(Γ):

Input: a propositional Γ-formula F in CNF

Question: Is F satisfiable?

Γ – a finite set of Boolean constraint relations

CSP(Γ):

Input: a propositional Γ-formula F in CNF

Question: Is F satisfiable?

Basic Goal: Determine the computational complexity of CSP(Γ) as a function of Γ!

The Galois Connection

$\mathrm{Pol}(\Gamma)$ is the set of all polymorphisms of Γ, i.e., the set of all Boolean functions that preserve every relation in Γ.

- $\mathrm{Pol}(\Gamma)$ is a clone, i.e., a set of Boolean functions that contains all projections and is closed under composition (Post).

Pol(Γ) is the set of all polymorphisms of Γ, i.e., the set of all Boolean functions that preserve every relation in Γ.

- Pol(Γ) is a clone, i.e., a set of Boolean functions that contains all projections and is closed under composition (Post).

Inv(B) is the set of all invariants of B, i.e., the set of all Boolean relations that are preserved by every function in B.

- Inv(B) is a relational clone, i.e., a set of Boolean relations that contains the equality relation and is closed under primitive positive definitions, i.e., if ϕ is an Inv(B)-formula and $R(x_1, \ldots, x_n) \equiv \exists y_1 \ldots y_\ell \; \phi(x_1, \ldots, x_n, y_1, \ldots, y_\ell)$ then $R \in \mathrm{Inv}(B)$.

Schaefer's Theorem

Let $\langle \Gamma \rangle$ be the relational clone generated by Γ.

$\mathrm{Inv}\big(\mathrm{Pol}(\Gamma)\big) = \langle \Gamma \rangle$ ("expressive power" of Γ).

Let $\langle\Gamma\rangle$ be the relational clone generated by Γ.

$\mathrm{Inv}\big(\mathrm{Pol}(\Gamma)\big) = \langle\Gamma\rangle$ ("expressive power" of Γ).

- If $\langle\Gamma\rangle = \langle\Gamma'\rangle$, then $\mathrm{CSP}(\Gamma) \equiv_m^{\log} \mathrm{CSP}(\Gamma')$, i.e., the complexity of $\mathrm{CSP}(\Gamma)$ depends only on $\mathrm{Pol}(\Gamma)$.

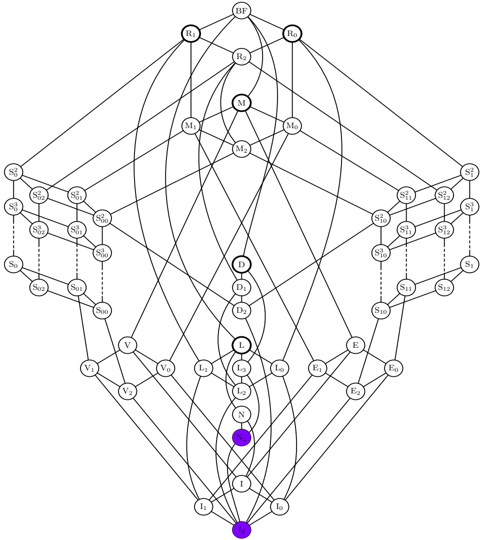
We only have to study co-clones in order to obtain a full classification.

Let $\langle\Gamma\rangle$ be the relational clone generated by Γ.
$\mathrm{Inv}\big(\mathrm{Pol}(\Gamma)\big) = \langle\Gamma\rangle$ ("expressive power" of Γ).

- If $\langle\Gamma\rangle = \langle\Gamma'\rangle$, then $\mathrm{CSP}(\Gamma) \equiv_m^{\log} \mathrm{CSP}(\Gamma')$,
  i.e., the complexity of $\mathrm{CSP}(\Gamma)$ depends only on $\mathrm{Pol}(\Gamma)$.

We only have to study co-clones in order to obtain a full classification.

- If $\langle\Gamma\rangle \supseteq \mathrm{Inv}\big(\mathrm{N}_2\big)$ then $\mathrm{CSP}(\Gamma)$ is NP-complete, otherwise $\mathrm{CSP}(\Gamma)$ is in P [Schaefer].



BF
R_1
R_0
R_2
M
M_1
M_0
M_2
S_0^2
S_{02}^2
S_{01}^2
S_{00}^2
S_0^3
S_{02}^3
S_{01}^3
S_{00}^3
S_0
S_{02}
S_{01}
S_{00}
S_1^2
S_{11}^2
S_{12}^2
S_{10}^2
S_1^3
S_{11}^3
S_{12}^3
S_{10}^3
S_1
S_{11}
S_{12}
S_{10}
D
D_1
D_2
V
V_1
V_0
V_2
L
L_1
L_3
L_0
L_2
E
E_1
E_0
E_2
N
N_2
I
I_1
I_0
I_2

Towards a Finer Classification

Goal: Determine logspace-degree of CSP(Γ) for every constraint language Γ!

Goal: Determine logspace-degree of $\mathrm{CSP}(\Gamma)$ for every constraint language Γ!

Examples:

- If $\Gamma \subseteq \mathrm{Inv}(\mathrm{L}_2)$, then $\mathrm{CSP}(\Gamma) \in \oplus\mathrm{L}$.
  ($\mathrm{Inv}(\mathrm{L}_2) = \mathrm{Inv}\big(\{x \oplus y \oplus z\}\big)$: "affine" constraints)

Towards a Finer Classification

Goal: Determine logspace-degree of CSP(Γ) for every constraint language Γ!

Examples:

- If $\Gamma \subseteq \mathrm{Inv}(\mathrm{L}_2)$, then $\mathrm{CSP}(\Gamma) \in \oplus\mathrm{L}$.
  ($\mathrm{Inv}(\mathrm{L}_2) = \mathrm{Inv}\big(\{x \oplus y \oplus z\}\big)$: "affine" constraints)
- If $\langle\Gamma\rangle \supseteq \mathrm{Inv}(\mathrm{L}_3)$ then CSP(Γ) is hard for $\oplus$L.
  ($\mathrm{Inv}(\mathrm{L}_3) = \langle x \oplus y \oplus z \oplus w\rangle$, Reduction from circuit value problem for $\oplus$-circuits)

Goal: Determine logspace-degree of $\mathrm{CSP}(\Gamma)$ for every constraint language Γ!

Examples:

- If $\Gamma \subseteq \mathrm{Inv}(\mathrm{L}_2)$, then $\mathrm{CSP}(\Gamma) \in \oplus\mathrm{L}$.
  ($\mathrm{Inv}(\mathrm{L}_2) = \mathrm{Inv}\big(\{x \oplus y \oplus z\}\big)$: "affine" constraints)
- If $\langle\Gamma\rangle \supseteq \mathrm{Inv}(\mathrm{L}_3)$ then $\mathrm{CSP}(\Gamma)$ is hard for $\oplus\mathrm{L}$.
  ($\mathrm{Inv}(\mathrm{L}_3) = \langle x \oplus y \oplus z \oplus w\rangle$, Reduction from circuit value problem for $\oplus$-circuits)

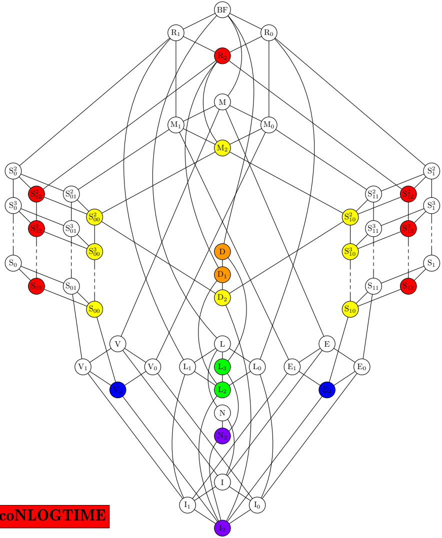
Thus:

- If $\mathrm{Inv}(\mathrm{L}_3) \subseteq \langle\Gamma\rangle \subseteq \mathrm{Inv}(\mathrm{L}_2)$, then $\mathrm{CSP}(\Gamma)$ is $\oplus\mathrm{L}$-complete.

Towards a Finer Classification

- If $I_0 \subseteq \mathrm{Pol}(\Gamma)$ or $I_1 \subseteq \mathrm{Pol}(\Gamma)$, then every constraint formula over Γ is satisfiable, and therefore $\mathrm{CSP}(\Gamma)$ is trivial.
- If $\mathrm{Pol}(\Gamma) \in \{I_2, N_2\}$, then $\mathrm{CSP}(\Gamma)$ is NP-complete.
- If $\mathrm{Pol}(\Gamma) \in \{V_2, E_2\}$, then $\mathrm{CSP}(\Gamma)$ is P-complete.
- If $\mathrm{Pol}(\Gamma) \in \{L_2, L_3\}$, then $\mathrm{CSP}(\Gamma)$ is $\oplus$L-complete.
- If $S_{00} \subseteq \mathrm{Pol}(\Gamma) \subseteq S_{00}^2$ or $S_{10} \subseteq \mathrm{Pol}(\Gamma) \subseteq S_{10}^2$ or $\mathrm{Pol}(\Gamma) \in \{D_2, M_2\}$, then $\mathrm{CSP}(\Gamma)$ is NL-complete.
- If $\mathrm{Pol}(\Gamma) \in \{D_1, D\}$ or $S_{02} \subseteq \mathrm{Pol}(\Gamma) \subseteq R_2$ or $S_{12} \subseteq \mathrm{Pol}(\Gamma) \subseteq R_2$, then $\mathrm{CSP}(\Gamma)$ is in L.

Classification of CSP-Satisfiability



NP complete
P complete
NL complete
$\oplus$L complete
L complete
L complete / coNLOGTIME

If $\Gamma' \subseteq \langle \Gamma \rangle$ then $\mathbf{CSP}(\Gamma') \leq_m^{\log} \mathbf{CSP}(\Gamma)$

Let F be a Γ'-formula. Construct F' as follows:

If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathbf{CSP}(\Gamma') \leq_m^{\log} \mathbf{CSP}(\Gamma)$

Let F be a Γ'-formula. Construct F' as follows:

- Replace every constraint from $\langle\Gamma\rangle$ by its defining existentially quantified $\big(\Gamma \cup \{=\}\big)$-formula.

If $\Gamma' \subseteq \langle\Gamma\rangle$ then CSP$(\Gamma') \leq_m^{\log}$ CSP(Γ)

Let F be a Γ'-formula. Construct F' as follows:

- Replace every constraint from $\langle\Gamma\rangle$ by its defining existentially quantified $\big(\Gamma \cup \{=\}\big)$-formula.
- Delete existential quantifiers.

If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathbf{CSP}(\Gamma') \leq_m^{\log} \mathbf{CSP}(\Gamma)$

Let F be a Γ'-formula. Construct F' as follows:

- Replace every constraint from $\langle\Gamma\rangle$ by its defining existentially quantified $\big(\Gamma \cup \{=\}\big)$-formula.
- Delete existential quantifiers.
- Delete equality clauses and replace all variables that are connected via a chain of equality constraints by a common new variable (undirected graph accessibility problem).

If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathrm{CSP}(\Gamma') \leq_m^{\log} \mathrm{CSP}(\Gamma)$

Let F be a Γ'-formula. Construct F' as follows:

- Replace every constraint from $\langle\Gamma\rangle$ by its defining existentially quantified $\big(\Gamma \cup \{=\}\big)$-formula.
- Delete existential quantifiers.
- Delete equality clauses and replace all variables that are connected via a chain of equality constraints by a common new variable (undirected graph accessibility problem).

F' is a Γ-formula.

Then: F is satisfiable iff F' is satisfiable.

If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathbf{CSP}(\Gamma') \leq_m^{\log} \mathbf{CSP}(\Gamma)$

Let F be a Γ'-formula. Construct F' as follows:

- Replace every constraint from $\langle\Gamma\rangle$ by its defining existentially quantified $\big(\Gamma \cup \{=\}\big)$-formula.
- Delete existential quantifiers.
- Delete equality clauses and replace all variables that are connected via a chain of equality constraints by a common new variable (undirected graph accessibility problem).

F' is a Γ-formula.

Then: F is satisfiable iff F' is satisfiable.

Question: Stricter reductions?

Example 1: $\Gamma_1 = \{x, \overline{x}\}$:

A Γ_1-formula F is unsatisfiable iff it contains clauses x and $\overline{x}$ for some x, hence $\mathrm{CSP}(\Gamma_1) \in \mathrm{coNLOGTIME}$.

Within LOGSPACE

Example 1: $\Gamma_1 = \{x, \overline{x}\}$:

A Γ_1-formula F is unsatisfiable iff it contains clauses x and $\overline{x}$ for some x, hence $\mathrm{CSP}(\Gamma_1) \in$ coNLOGTIME.

Example 2: $\Gamma_2 = \{x, \overline{x}, =\}$:

Then $\mathrm{CSP}(\Gamma_2)$ can express undirected graph reachability as follows: Given G, s, t, construct F to consist of clauses $\overline{s}$, t, and $u = v$ for every edge $(u, v) \in G$.

Then t is reachable in G from s iff F is unsatisfiable, Example 1: $\Gamma_1 = \{x, \overline{x}\}$:

A Γ_1-formula F is unsatisfiable iff it contains clauses x and $\overline{x}$ for some x, hence $\mathrm{CSP}(\Gamma_1) \in \mathrm{coNLOGTIME}$.

Example 2: $\Gamma_2 = \{x, \overline{x}, =\}$:

Then $\mathrm{CSP}(\Gamma_2)$ can express undirected graph reachability as follows: Given G, s, t, construct F to consist of clauses $\overline{s}$, t, and $u = v$ for every edge $(u, v) \in G$.

Then t is reachable in G from s iff F is unsatisfiable, hence $\mathrm{CSP}(\Gamma_2)$ is hard for L (under AC^0-reductions/FO-reductions).

Within LOGSPACE

Example 1: $\Gamma_1 = \{x, \overline{x}\}$:

A Γ_1-formula F is unsatisfiable iff it contains clauses x and $\overline{x}$ for some x, hence $\mathrm{CSP}(\Gamma_1) \in \mathrm{coNLOGTIME}$.

Example 2: $\Gamma_2 = \{x, \overline{x}, =\}$:

Then $\mathrm{CSP}(\Gamma_2)$ can express undirected graph reachability as follows: Given G, s, t, construct F to consist of clauses $\overline{s}$, t, and $u = v$ for every edge $(u, v) \in G$.

Then t is reachable in G from s iff F is unsatisfiable, hence $\mathrm{CSP}(\Gamma_2)$ is hard for L (under AC^0-reductions/FO-reductions).

Thus: Provably different complexity: $\mathrm{CSP}(\Gamma_2) \not\leq_m^{\mathrm{AC}^0} \mathrm{CSP}(\Gamma_1)$,

but $\mathrm{Pol}(\Gamma_1) = \mathrm{Pol}(\Gamma_2) \quad (= \mathrm{R}_2)$.

- If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathsf{CSP}(\Gamma') \leq_m^{\mathsf{AC}^0} \mathsf{CSP}\big(\Gamma \cup \{=\}\big) \leq_m^{\log} \mathsf{CSP}(\Gamma)$.

- If $\Gamma' \subseteq \langle\Gamma\rangle$ then $\mathrm{CSP}(\Gamma') \leq_m^{\mathrm{AC}^0} \mathrm{CSP}\big(\Gamma \cup \{=\}\big) \leq_m^{\log} \mathrm{CSP}(\Gamma)$.

Say that Γ can express equality if equality constraint can be defined by an existentially quantified Γ-formula.

- If Γ can express equality then $\mathrm{CSP}\big(\Gamma \cup \{=\}\big) \leq_m^{\mathrm{AC}^0} \mathrm{CSP}(\Gamma)$.

There is an algorithm that detects if Γ can express equality.

The Equality Constraint

- If $\Gamma' \subseteq \langle \Gamma \rangle$ then $\mathrm{CSP}(\Gamma') \leq_m^{\mathrm{AC}^0} \mathrm{CSP}\big(\Gamma \cup \{=\}\big) \leq_m^{\log} \mathrm{CSP}(\Gamma)$.

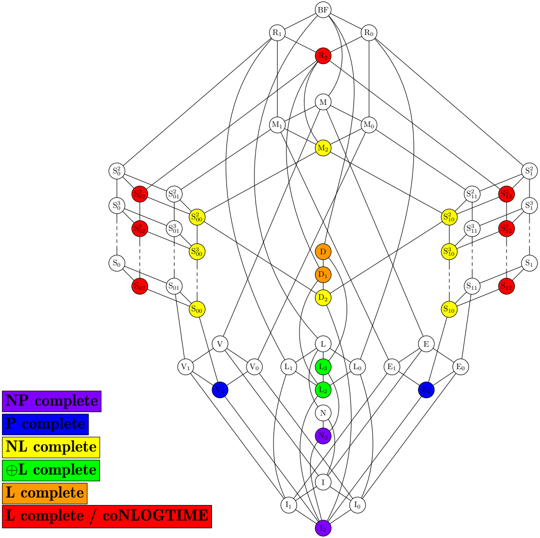
Say that Γ can express equality if equality constraint can be defined by an existentially quantified Γ-formula.

- If Γ can express equality then $\mathrm{CSP}\big(\Gamma \cup \{=\}\big) \leq_m^{\mathrm{AC}^0} \mathrm{CSP}(\Gamma)$.

There is an algorithm that detects if Γ can express equality.

- If Γ can express equality then $\mathrm{CSP}(\Gamma)$ is hard for L, otherwise $\mathrm{CSP}(\Gamma) \in \mathrm{coNLOGTIME}$.

- If $\mathrm{Pol}(\Gamma) \in \{D_1, D\}$, then $\mathrm{CSP}(\Gamma)$ is L-complete.
- If $S_{02} \subseteq \mathrm{Pol}(\Gamma) \subseteq R_2$ or $S_{12} \subseteq \mathrm{Pol}(\Gamma) \subseteq R_2$, then either $\mathrm{CSP}(\Gamma)$ is in coNLOGTIME, or $\mathrm{CSP}(\Gamma)$ is L-complete. There is an algorithm deciding which case occurs.


BF
R_1
R_0
R_2
M
M_1
M_0
M_2
S_0^2
S_{02}^2
S_{01}^2
S_{00}^2
S_0^3
S_{02}^3
S_{01}^3
S_{00}^3
S_0
S_{02}
S_{01}
S_{00}
S_1^2
S_{12}^2
S_{11}^2
S_{10}^2
S_1^3
S_{12}^3
S_{11}^3
S_{10}^3
S_1
S_{12}
S_{11}
S_{10}
D
D_1
D_2
V
V_1
V_0
V_2
L
L_1
L_3
L_0
L_2
E
E_1
E_0
E_2
N
N_2
I
I_1
I_0
I_2
NP complete
P complete
NL complete
$\oplus$L complete
L complete
L complete / coNLOGTIME

Post's lattice: $L_2 \subseteq R_2$, hence $\mathrm{Inv}(R_2) \subseteq \mathrm{Inv}(L_2)$.

Hence:

- Undirected graph accessibility is in $\oplus$L, in other words: SL $\subseteq \oplus$L [Karchmer, Wigderson, 1993].

The Power of $\oplus$L

Post's lattice: $L_2 \subseteq R_2$, hence $\mathrm{Inv}(R_2) \subseteq \mathrm{Inv}(L_2)$.

Hence:

- Undirected graph accessibility is in $\oplus$L, in other words: SL $\subseteq$ $\oplus$L [Karchmer, Wigderson, 1993].

(Today we even know SL $\subseteq$ L.)

Myhill's Isomorphism Theorem:

- If $A \leq_1 B$ and $B \leq_1 A$ then A and B are isomorphic via a recursive bijection.

Myhill's Isomorphism Theorem:

- If $A \leq_1 B$ and $B \leq_1 A$ then A and B are isomorphic via a recursive bijection.

In the polynomial time setting:

Isomorphism conjecture/Berman-Hartmanis-Conjecture:

- If $A \leq_m^{\mathrm{p}} B$ and $B \leq_m^{\mathrm{p}} A$ then A and B are isomorphic via a polynomial-time computable bijection.

Implies $\mathrm{P} \neq \mathrm{NP}$.

Isomorphism Theorem holds for $\leq_m^{\mathrm{AC}^0}$-reducibility:

Isomorphism Theorem holds for $\leq_m^{AC^0}$-reducibility:

- For every constraint language Γ, $CSP(\Gamma)$ is AC^0-isomorphic either to $0\Sigma^\star$ or to the standard complete set for one of the complexity classes NP, P, $\oplus$L, NL, or L.

There are only six different CSP-problems!

Why study Boolean CSP?

Provide a reasonably accurate bird's eye view of complexity theory [Creignou, Khanna, Sudan]:

- inclusions among complexity classes
- relations among reducibility notions
- structure of complete problems

Why study Boolean CSP?

Provide a reasonably accurate bird's eye view of complexity theory [Creignou, Khanna, Sudan]:

- inclusions among complexity classes
- relations among reducibility notions
- structure of complete problems

- playground for the study of many issues related to counting classes
- CSP isomorphism problems yield good candidates for "intermediate problems"

References

The classification presented in this talk is given in

- E. Allender, M. Bauland, N. Immerman, H. Schnoor, H. Vollmer. The complexity of satisfiability problems: refining Schaefer's theorem; Proceedings 30th Mathematical Foundations of Computer Science, Springer Lecture Notes in Computer Science Vol. 3618, pp. 71–82, 2005.

More on counting for Boolean CSP:

- M. Bauland, E. Böhler, N. Creignou, S. Reith, H. Schnoor, H. Vollmer. Quantified constraints: the complexity of decision and counting for bounded alternation; Electronic Colloqium on Computational Complexity, TR05-024, 2005.

More on isomorphism for Boolean CSP:

- E. Böhler, E. Hemaspaandra, S. Reith, H. Vollmer. The complexity of Boolean constraint isomorphism; Proceedings 21st Symposium on Theoretical Aspects of Computer Science, Springer Lecture Notes in Computer Science Vol. 2996, pp. 164–175, 2004.

Open Questions for Boolean CSP

- Fine classification for Boolean counting problem
- Infinite constraint languages?
- Uniform Boolean CSP?