The Complexity of Constraint Satisfaction Problems

Andrei Krokhin
Durham University

Tutorial, Part I - Here and Now

Two Well-Known Problems

SAT: is a given propositional formula in CNF satisfiable?

$$F = (\neg x \vee y \vee \neg z) \wedge (x \vee y \vee \neg z) \wedge (\neg x \vee \neg y \vee z)$$

LINEAR EQUATIONS: does a given system of linear equations have a solution?

$$\begin{cases} 2x + 2y + 3z = 1 \\ 3x - 2y - 2z = 0 \\ 5x - y + 10z = 2 \end{cases}$$

Outline

1. Constraints and Their Complexity: An introduction
    - The CSP and its forms
    - Complexity of CSP: A roadmap
    - Some algebra, finally ...
2. Universal Algebra for CSP: A general theory
3. UA (and a bit of logic) for CSP: A bigger picture

CSP in AI Setting

Instance: (V, D, C) where

- V is a finite set of variables,
- D is a (finite) set of values,
- C is a set of constraints $\{C_1, \ldots, C_q\}$ where
  - each constraint C_i is a pair $(\overline{s}_i, R_i)$ with
    - scope $\overline{s}_i$ - a list of variables of length m_i, and
    - relation R_i - an m_i-ary relation over D

Question: is there $f : V \rightarrow D$ such that $f(\overline{s}_i) \in R_i$ for all i?

Some Real-World Examples of CSPs

- Drawing up timetable for a conference
- Choosing frequencies for a mobile-phone network
- Fitting a protein structure to measurements
- Laying out components on a circuit board
- Finding a DNA sequence from a set of contigs
- Scheduling a construction project

CSP in Logical Setting

Instance: a first-order formula

$$\varphi(x_1, \ldots, x_n) = R_1(\overline{s}_1) \wedge \ldots \wedge R_q(\overline{s}_q).$$

Question: is φ satisfiable?

The $\overline{s}_i$'s = constraint scopes $\overline{s}_i$

Predicates R_i = constraint relations R_i

Hence, CSP generalizes SAT.

In Database Theory, CSP = Conjunctive-Query Evaluation

CSP in Combinatorial Setting

The Homomorphism Problem:

Given two finite similar relational structures,
$\mathcal{A} = (A;\ R_1^{\mathcal{A}}, \ldots, R_k^{\mathcal{A}})$ and $\mathcal{B} = (B;\ R_1^{\mathcal{B}}, \ldots, R_k^{\mathcal{B}})$,
is there a homomorphism $h : \mathcal{A} \to \mathcal{B}$?

$$\forall i\ [(a_1, \ldots, a_{n_i}) \in R_i^{\mathcal{A}} \Longrightarrow (h(a_1), \ldots, h(a_{n_i})) \in R_i^{\mathcal{B}}]$$

- Think of elements in $\mathcal{A}$ as of variables.
  Tuples in relations in $\mathcal{A}$ = constraint scopes $\overline{s}_i$.
- Think of elements in $\mathcal{B}$ are values.
  Relations in $\mathcal{B}$ = constraint relations R_i.

Hence, CSP generalizes GRAPH HOMOMORPHISM.

Example: 2-SAT in Hom Form

Let $R_{ab}^{\mathcal{B}} = \{0,1\}^2 \setminus \{(a,b)\}$ and $\mathcal{B} = (\{0,1\}; R_{00}^{\mathcal{B}}, R_{01}^{\mathcal{B}}, R_{11}^{\mathcal{B}})$.
Then 2-SAT is precisely CSP($\mathcal{B}$).

An instance of 2-SAT, say

$$F = (\neg x \vee \neg z) \wedge (x \vee y) \wedge (y \vee \neg z) \wedge (u \vee x) \wedge (x \vee \neg u) \ldots$$

becomes a structure $\mathcal{A}$ with base set $\{x, y, z, u, \ldots\}$ and

$R_{00}^{\mathcal{A}} = \{(x,y), (u,x), \ldots\}$

$R_{01}^{\mathcal{A}} = \{(y,z), (x,u), \ldots\}$

$R_{11}^{\mathcal{A}} = \{(x,z), \ldots\}$

Then $h : \mathcal{A} \to \mathcal{B}$ iff h is a satisfying assignment for F.

Forms of CSP: A recap

- Variable-value

  Given finite sets V (variables), D (values), and a set of constraints $\{(\overline{s}_1, R_1), \ldots, (\overline{s}_q, R_q)\}$ over V, is there a function $f : V \to D$ such that $f(\overline{s}_i) \in R_i$ for all i?

- Satisfiability

  Given a formula $\mathcal{P}(x_1, \ldots, x_n) = R_1(\overline{s}_1) \wedge \ldots \wedge R_q(\overline{s}_q)$ (where R_i's are seen as predicates), is $\mathcal{P}$ satisfiable?

- Homomorphism

  Given two finite similar relational structures, $\mathcal{A} = (V;\ R_1^{\mathcal{A}}, \ldots, R_k^{\mathcal{A}})$ and $\mathcal{B} = (D;\ R_1^{\mathcal{B}}, \ldots, R_k^{\mathcal{B}})$, is there a homomorphism $h : \mathcal{A} \to \mathcal{B}$?

The Complexity of CSP

Fact. CSP is **NP**-complete.

Membership in **NP** is trivial.

Complete because contains 3-SAT.

Question: What restrictions make it computationally easy?

Parameterisation of CSP

With any instance of CSP one can associate two natural parameters reflecting

1. Which variables constrain which others, i.e.,
   - constraint scopes, or
   - query language, or
   - LHS structure $\mathcal{A}$ (as in $\mathcal{A} \to \mathcal{B}$).
2. How values for the variables are constrained, i.e.,
   - constraint relations, or
   - relational database, or
   - RHS structure $\mathcal{B}$ (as in $\mathcal{A} \to \mathcal{B}$).

Restricting LHS: "The other side"

For a class $\mathbb{C}$ of structures, let $\mathrm{CSP}(\mathbb{C}, -)$ denote the set of all CSP instances $(\mathcal{A}, \mathcal{B})$ with $\mathcal{A} \in \mathbb{C}$.

Example: if $\mathbb{C} = \{K_n \mid n > 0\}$ is the class of all complete graphs then $\mathrm{CSP}(\mathbb{C}, -)$ is the CLIQUE problem (**NP**-c) .

For any fixed $\mathcal{A}$, $\mathrm{CSP}(\{\mathcal{A}\}, -)$ is in **P**. Simply check each mapping $A \to B$. If $|A| = k$ then $|B|^k$ is polynomial in $|\mathcal{B}|$. Boring.

Theorem 1 (Grohe'07) *Let $\mathbb{C}$ be an arity-bounded class of structures. Under a "reasonable" complexity-theoretic assumption,* $\mathrm{CSP}(\mathbb{C}, -)$ *is in* **P** *iff "all structures in $\mathbb{C}$ look like trees (when you look at $\mathbb{C}$ from far enough)".*

Restricting RHS: Constraint Languages

Fix a finite set D.

Definition 1 *A constraint language is any finite set* Γ *of relations on* D*. The problem* CSP(Γ) *is the restriction of* CSP *where all constraint relations* R_i *must belong to* Γ*.*

Equivalently, fix target structure $\mathcal{B}$ (aka template) and ask whether a given structure $\mathcal{A}$ homomorphically maps to $\mathcal{B}$. Notation: $\mathrm{CSP}(\mathcal{B}) = \{\mathcal{A} \mid \mathcal{A} \to \mathcal{B}\}$.

The structure $\mathcal{B}$ is obtained from Γ by indexing relations.

NB. For a digraph $\mathcal{H}$, CSP($\mathcal{H}$) is known as $\mathcal{H}$-COLOURING, appears in 100s of papers + recent book by Hell & Nešetřil.

Examples

- Let $D = \{0,1\}$ and $R = \{0,1\}^3 \setminus \{(0,0,0),(1,1,1)\}$.
  If $\Gamma = \{R\}$ then CSP(Γ) is Not-All-Equal Sat.
  This problem is **NP**-complete.

- Let $D = \{0,1\}$ and $R = \{(x,y,z) \mid y \wedge z \rightarrow x\}$.
  If $\Gamma = \{R, \{0\}, \{1\}\}$ then CSP(Γ) is Horn 3-Sat.
  This problem is **P**-complete.

- Let $D = \{0,1\}$ and $\Gamma = \{\leq, \{0\}, \{1\}\}$. Then CSP($\Gamma$) is the complement of Path (i.e., Unreachability).
  Think: An instance is satisfiable iff it contains no path of the form $1 = x_1 \leq x_2 \leq \ldots \leq x_n = 0$.
  This problem is **NL**-complete.

More Examples

- If $\Gamma = \{\neq_D\}$ where $\neq_D$ is the disequality relation on D and $|D| = k$ then CSP(Γ) is GRAPH k-COLOURING. Think: elements of D are colours, variables are the nodes, and constraints $x \neq_D y$ are the edges of graph. Belongs to **L** if $k \leq 2$, **NP**-complete for $k \geq 3$.

- Let D with $|D| = p$ have a structure of $\mathbb{Z}_p$, p prime. If $\Gamma = \{R, \{0\}, \{1\}\}$ where $R = \{(x, y, z) \mid x + y = z\}$ then CSP(Γ) is (essentially) the problem of solving LINEAR EQUATIONS over $\mathbb{Z}_p$.

Classification Problems & The Holy Grail

Two main classification problems about problems CSP(Γ):

1. Classify CSP(Γ) w.r.t. computational complexity, (i.e., w.r.t. membership in a given complexity class)
2. Classify CSP($\mathcal{B}$) w.r.t. descriptive complexity, (i.e., w.r.t. definability in a given logic)

Conjecture 1 (Feder,Vardi '98)
Dichotomy Conjecture: for each Γ*, the problem* CSP(Γ) *is either tractable (i.e., in* **P***) or* **NP***-complete.*

Original Motivation for FV Conjecture

Ladner '75 : $\mathbf{P} \neq \mathbf{NP}$ implies that $\mathbf{NP} - (\mathbf{P} \cup \mathbf{NPc}) \neq \emptyset$.

Want: a large(st) "natural" subclass of **NP** where .. $= \emptyset$.

Feder & Vardi define complexity class **MMSNP** obtained from **NP** by simultaneously imposing 3 logical restrictions.

FV: Any 2 of them give **NP** modulo **P**-reductions (.. $\neq \emptyset$).

Theorem 2 (Feder,Vardi '98; Kun '07)
1) The class $\{\mathrm{CSP}(\Gamma)\}$ *is a proper sublclass of* **MMSNP**.
2) The two classes are the same modulo **P**-*reductions.*

Hence, Dichotomy for CSP $\Rightarrow$ Dichotomy for **MMSNP**.

The Three Approaches

The three main approaches to our classification problems are:

- via Combinatorics (Graphs & Posets)
  - Interesting, but only a hint in this tutorial
- via Logic and Games
  - Some in 3rd lecture, not much (phew-w-w...)
- via Algebra
  - Hey, that's what we like !!!

Combinatorics Approach: Encoding CSP

Theorem 3 (FV'98) *For every structure $\mathcal{B}$ there exist*

- *a poset $P_{\mathcal{B}}$;*
- *a bipartite graph $G_{\mathcal{B}}$;*
- *a digraph $H_{\mathcal{B}}$*

such that these problems are polynomially equivalent:

- CSP($\mathcal{B}$),
- *poset-retraction($P_{\mathcal{B}}$),*
- *bipartite graph-retraction($G_{\mathcal{B}}$),*
- *digraph-homomorphism($H_{\mathcal{B}}$).*

Logic and Games Approach

One can view CSP($\mathcal{B}$) as the membership problem for the class of structures $\mathcal{A}$ such that $\mathcal{A} \to \mathcal{B}$.

Typical result describes the class CSP($\mathcal{B}$)

- by a logical specification (e.g., formula in a nice logic) that can be checked easily against a given structure, or
- as a class of structures $\mathcal{A}$ for which there exists an (easily detectable) winning strategy in a certain game on $\mathcal{A}$ and $\mathcal{B}$.

Examples: in my 3rd lecture

A UAlgebraic Approach: Intuition

Intuition: The more one can express in Γ the harder CSP(Γ).

Example: Suppose Γ contains two binary relations, R_1 and R_2. Consider the following (part of) instance

$$((x, z), R_1),\ ((z, y), R_2).$$

- The implicit constraint on (x, y) is $R_3 = R_1 \circ R_2$.
- It may not belong to Γ, but
- CSP(Γ) and CSP($\Gamma \cup \{R_3\}$) are logspace equivalent.

Question: Where does this lead us to?

Relational Clones

Definition 2 *For a set of relations* Γ *on* D*, let* $\langle\Gamma\rangle$ *denote the set of all relations that can be expressed by primitive positive (p.p.-) formulas over* Γ*, that is, using*

- *relations in* $\Gamma \cup \{=_D\}$*,*
- *conjunction,*
- *existential quantification.*

Example: $R_1(x,y,z) = \exists u[R_2(x,u) \wedge R_3(u,y) \wedge y = z]$.

The set $\langle\Gamma\rangle$ is the relational clone generated by Γ.

Relational Clones cont'd

Theorem 4 (Jeavons '98)
If Γ_1 *and* Γ_2 *are constraint languages such that* $\langle\Gamma_1\rangle \subseteq \langle\Gamma_2\rangle$ *then* $\mathrm{CSP}(\Gamma_1)$ *is logspace reducible to* $\mathrm{CSP}(\Gamma_2)$.

Proof. Reduction goes as follows:

1. Take an instance $R_1(\overline{s}_1) \wedge \ldots \wedge R_q(\overline{s}_q)$ where $R_i \in \Gamma_1$.
2. Since $R_i \in \langle\Gamma_2\rangle$, replace each $R_i(\overline{s}_1)$ by the corresponding p.p.-formula over Γ_2
3. Remove quantifiers, renaming variables as necessary.
4. Identify variables connected by equality constraints.
5. Remove equality constraints.

Example

Assume $\Gamma_1 = \{R_1\}$ and $\Gamma_2 = \{R_2, R_3\}$, and $R_1 \in \langle\Gamma_2\rangle$.
0) Fix expression for R_1, for example,
$R_1(x, y, z) = \exists u[R_2(x, u) \wedge R_3(u, y) \wedge y = z]$.

1) Take an instance $R_1(x, y, z) \wedge R_1(z, t, y)$.

2) Transform it into equivalent formula
$\exists u[R_2(x, u) \wedge R_3(u, y) \wedge y = z] \wedge \exists u[R_2(z, u) \wedge R_3(u, t) \wedge t = y]$.

3) Remove quantifiers, renaming the quantified variables
$R_2(x, u_1) \wedge R_3(u_1, y) \wedge y = z \wedge R_2(z, u_2) \wedge R_3(u_2, t) \wedge t = y$.

4-5) Identify z, t with y and remove equality constraints
$R_2(x, u_1) \wedge R_3(u_1, y) \wedge R_2(y, u_2) \wedge R_3(u_2, y)$.

Invariance and Polymorphisms

Definition 3 *An m-ary relation R is invariant under an n-ary operation f (or f is a polymorphism of R) if, for any tuples $\bar{a}_1 = (a_{11}, \ldots, a_{1m}), \ldots, \bar{a}_n = (a_{n1}, \ldots, a_{nm}) \in R$, the tuple obtained by applying f componentwise belongs to R.*

$$
\begin{array}{ccccccc}
 & f & & f & & f & & \\
(& a_{11} & , & \ldots & , & a_{1m} &) & \in R \\
 & \vdots & & \vdots & & \vdots & & \vdots \\
(& a_{n1} & , & \ldots & , & a_{nm} &) & \in R \\
\hline
(& f(a_{11}, \ldots, a_{n1}) & , & \ldots & , & f(a_{1m}, \ldots, a_{nm}) &) & \in R
\end{array}
$$

Example

Example 1 *Consider the relation, R, defined by*

$$R = \{(0,0,0),(1,0,0),(0,0,1)\}$$

- *the binary operation* min *is a polymorphism of R. For example,*

$$\begin{array}{ccccccccl} & \min & & \min & & \min & & & \\ (& 1 & , & 0 & , & 0 &) & \in R \\ (& 0 & , & 0 & , & 1 &) & \in R \\ \hline (& 0 & , & 0 & , & 0 &) & \in R \end{array}$$

- *the binary operation* max *is not.*

Galois Correspondence

Let $\mathrm{Pol}(\Gamma)$ be the set of all polymorphisms of Γ.

If F is a set of operations on D, let
$\mathrm{Inv}(F) = \{R \mid R$ is invariant under all operations in $F\}$,
and let $\langle F\rangle$ be the set of all operations obtained from F via superpositions $f(f_1, \dots, f_n)$.
Then $\langle F\rangle$ is called the clone generated by F.

Theorem 5 (Geiger '68; Bodnarchuk et al. '69)

- *For any constraint language* Γ, $\langle\Gamma\rangle = \mathrm{Inv}(\mathrm{Pol}(\Gamma))$.
- *For any set* F *of operations,* $\langle F\rangle = \mathrm{Pol}(\mathrm{Inv}(F))$.

Clones in Control of Complexity

Theorem 6 (Jeavons '98) *If* Γ_1 *are* Γ_2 *are constraint languages such that* $\mathrm{Pol}(\Gamma_1) \subseteq \mathrm{Pol}(\Gamma_2)$ *then* $\mathrm{CSP}(\Gamma_2)$ *is logspace reducible to* $\mathrm{CSP}(\Gamma_1)$.

Proof. The operator Inv() is anti-monotone, so $\mathrm{Pol}(\Gamma_1) \subseteq \mathrm{Pol}(\Gamma_2)$ implies

$$\langle \Gamma_2 \rangle = \mathrm{Inv}(\mathrm{Pol}(\Gamma_2)) \subseteq \mathrm{Inv}(\mathrm{Pol}(\Gamma_1)) = \langle \Gamma_1 \rangle.$$

We already know that $\langle \Gamma_2 \rangle \subseteq \langle \Gamma_1 \rangle$ implies the conclusion of the theorem.

Corollary 1 *If* $\mathrm{Pol}(\Gamma_1) = \mathrm{Pol}(\Gamma_2)$ *then* $\mathrm{CSP}(\Gamma_1)$ *and* $\mathrm{CSP}(\Gamma_2)$ *are logspace equivalent.*

One Striking Feature: Reductions For Free

How do you show that a problem X is **NP**-complete?

You construct a reduction (an explicit transformation) to X from some **NP**-complete problem (say SAT).

You don't have to do this for CSP(Γ) !!!

Just show that $\text{Pol}(\Gamma) \subseteq \text{Pol}(\Gamma')$ for some Γ' with **NP**-complete CSP(Γ').

Think about it: it may be very hard to actually construct a reduction, but you do some apparently unrelated algebra and show that it exists, which is all you need.