Constraint Satisfaction Problems with Tree Duality

CSP:
- Input: relational structures $G, H$ (over some signature $\sigma$)
- Output: Yes if there is a homomorphism $G \rightarrow H$
  No otherwise

("Everything" is finite.)
Computational Complexity

$P =$ problems that admit a polynomial-time algorithm
$NP =$ problems with a polynomial-size certificate for Yes, which can be checked in polynomial time

$NP$-hard $=$ every problem in $NP$ reducible to it
$NP$-complete $=$ $NP$-hard $\cap$ $NP$

Examples:
$P \ni \text{2-colouring of graphs}$

$\text{3-colouring of graphs is } NP$-complete

$\text{CSP is } NP$-complete
CSP is NP-complete

→ Exponential algorithms

→ Poly-time algorithms that don’t always work

→ Restrict the constraints that are allowed (and classify complexity)

\[ \text{CSP}(H) : H \text{ is fixed} \]

Input: G

Question: \( G \Rightarrow H ? \)

We’ve seen \( H = K_2, K_3 \)
**CSP (H)**: H is fixed

**Input**: G

**Question**: G \( \rightarrow \) H ?

More examples:

- one ternary relation; dom H = \{0, 1\}
  \[x + y + z = 1 \quad \text{(in GF(2))}\]

- dom H = \{0, 1\}; four ternary relations to express clauses such as \((x \lor \neg y \lor \neg z)\)
  
  3-SAT
CSP\((\mathcal{H})\):

\[ \rightarrow \text{Sometimes P, sometimes NP-complete, sometimes...???} \]

\[ \rightarrow \text{Hell, Nešetřil, 1990: For symmetric graphs:} \]

\[ \bullet \text{P if } \mathcal{H} \text{ is bipartite or has a loop} \]
\[ \bullet \text{NP-complete otherwise} \]

**Example:** \( \mathcal{H} = C_5 \)

\[ \text{Observe: } G \rightarrow \begin{array}{c}
\text{Graph 1} \\
\end{array} \iff G^V \rightarrow \begin{array}{c}
\text{Graph 2} \\
\end{array} \]

\[ G \rightarrow H^V \iff G^{1/3} \rightarrow H \]

... adjoint functors
CSP ($H$):

- Sometimes $P$, sometimes NP-complete, sometimes... ???

- **Hell, Nešetřil, 1990**: For symmetric graphs:
  - $P$ if $H$ is bipartite or has a loop
  - NP-complete otherwise

- **Feder, Vardi, 1998**: conjectured dichotomy
  - studied “width-1” problems
  - lots of other things

- **Hell, Nešetřil, Zhu, 1996**: “tree duality” problems
  - “bounded treewidth duality” for digraphs
H has tree duality if

\[ \text{whenever } G \rightarrow H, \]

[Diagram: an upward arrow from G to H]

then there is a \( \sigma \)-tree \( F \) s.t.

\[ F \rightarrow G \text{ and } F \rightarrow H. \]

there exists \( F \) consisting of \( \sigma \)-trees only, s.t.

- whenever \( G \rightarrow H \), then \( \exists F \in F, F \rightarrow G \)
- \( F \in F \Rightarrow F \rightarrow H \)

\( F \) is a complete set of obstructions for \( H \)

Examples:

\[ \{ \begin{array}{c}
\uparrow \\
\end{array} \} \text{ for } H = \begin{array}{c}
\uparrow \end{array} \]

\[ \{ \begin{array}{c}
\downarrow \\
\end{array} \} \text{ for } H = \begin{array}{c}
\uparrow \end{array} \]

\[ \{ \begin{array}{c}
\uparrow, \downarrow, \leftarrow, \rightarrow,
\end{array} \} \text{ for } H = \begin{array}{c}
\uparrow \end{array} \]
Arc consistency

$L : \text{dom } G \rightarrow 2^{\text{dom } H}$ is consistent with an arc/tuple $(x_1, \ldots, x_k) \in R^G$ (Reσ) if

\[ \forall i \forall y_i \in L(x_i) \exists y_1, y_2, \ldots, y_{i-1}, y_{i+1}, y_{i+2}, \ldots, y_k \in \text{dom } H, \]

each $y_j \in L(x_j)$, s.t. $(y_1, \ldots, y_k) \in R^H$.

**Algorithm** for CSP(H)

**Input**: $G$

1. Initialise $L(x) := \text{dom } H \ \forall x \in \text{dom } G$

2. While $\forall x, L(x) \neq \emptyset$:
   - If $L$ is inconsistent with some $(x_1, \ldots, x_k) \in R^G$, remove corresponding $y_i$ from $L(x_i)$.
   - If $L$ is consistent with all tuples of $G$, stop.
Algorithm for CSP(H)

Input: G

1. Initialise \( L(x) := \text{dom } H \quad \forall x \in \text{dom } G \)

2. While \( \forall x, L(x) \neq \emptyset \):
   - If \( L \) is inconsistent with some \( (x_1, \ldots, x_k) \in R^G \), remove corresponding \( y_i \) from \( L(x_i) \).
   - If \( L \) is consistent with all tuples of \( G \), stop.

- If some \( L(x) = \emptyset \), then \( G \not\rightarrow H \).
- If each \( |L(x)| = 1 \), then \( G \not\rightarrow H \).
- If \( G \) is a tree, non-empty \( L \) gives a homomorphism \( G \rightarrow H \);
  - if \( F \) is a tree and \( g : F \rightarrow G \), then \( L(g(z)) \) gives a hom. \( F \rightarrow H \).
  Hence the algorithm is correct for CSP(H) with tree duality!
- Without tree duality, non-empty lists do not guarantee \( G \not\rightarrow H \).
But **how do we find out** if \( H \) has tree duality?

**Feder, Vardi**: power structure \( \mathcal{U}(H) \):

- \( \text{dom~} \mathcal{U}(H) = 2^{\text{dom~} H} \setminus \{\emptyset\} \)
- \((A_1, \ldots, A_k) \in \mathcal{U}(H) \) if
  \[
  \forall i \forall x_i \in A_i \exists x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k \text{ such that each } x_j \in A_j \text{ s.t. } (a_1, \ldots, x_k) \in \mathcal{U}(H).
  \]

**H has tree duality** if and only if \( \mathcal{U}(H) \to H \).
Which $F$'s are complete sets of obstructions for CSP($H$)?

Simplified setup: Consider $\sigma$ containing only binary relation symbols; assume all elements of $F$ are caterpillars:

$$
\begin{array}{c}
\circ \rightarrow \circ \rightarrow \circ \rightarrow \circ \rightarrow \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\circ \circ \circ \circ \circ \circ \circ \circ \circ \\
\end{array}
= F
$$

associate a word with $F: \overrightarrow{Bg} \circ \overrightarrow{BGr} \overrightarrow{Rb} \overrightarrow{Bg} \overrightarrow{r} \rightarrow$ language $L(F)$

P.L. Erdős, C. Tardif, G. Tardos (2013):

1. If $L(F)$ is regular, then $F$ is a complete set of obstructions for some (finite) $H$.
2. If $F$ is a c.s.o. for $H$, then $L(\uparrow F)$ is a regular language.

Extends to any $\sigma$ and non-caterpillars.
$F$ is a c.s.o. for some finite $H$
\[ \text{Forb}(\mathcal{F}) = \text{CSP}(H) \]

\[
\begin{array}{c}
\text{Erdős, Pálvölgyi,} \\
\text{Tardif, Tardos}
\end{array}
\]

$F$ is a regular set of $\sigma$-trees
\[
\begin{array}{c}
\text{Hubička, Nešetřil}
\end{array}
\]

There is an infinite universal "limit" structure $L$, \( \text{CSP}(H) = \text{Age}(L) \), and $L$ is $\omega$-categorical.

\[
\begin{array}{c}
\text{J.F.}
\end{array}
\]

There is a finite signature $\tau \supseteq \sigma$, extending $\sigma$ by unary relation symbols, and a universal $\tau$-structure $L^*$, s.t. $L$ is the $\sigma$-reduct of $L^*$ and $L^*$ is a Ramsey structure ($\equiv \text{Aut}(L^*)$ is extremely amenable).
Adjunct Functors

\[ G \rightarrow \star \quad \iff \quad G^{1/3} \rightarrow \bigcirc \]
\[ G \rightarrow H^3 \quad \iff \quad G^{1/3} \rightarrow H \]

A. Pultr, 1970: The right adjuncts in the category of \( \sigma \)-structures are given by:

\[ A, \quad B_R \quad \text{for each } R \in \sigma \text{ of arity } k, \]

hom's \( \varepsilon_i : A \rightarrow B_R \quad \text{for } i = 1, 2, \ldots, k. \)

\[ H \rightarrow \Gamma(H): \quad \text{dom } \Gamma(H) = \text{Hom} \left( A, H \right) \]

for \( R \in \sigma \): \[ R^{\Gamma(H)} = \text{Hom} \left( B_R, H \right), \]

\[ g : B_R \rightarrow H \]

\( g \) is the tuple \( \left( g \circ \varepsilon_1, g \circ \varepsilon_2, \ldots, g \circ \varepsilon_k \right). \)

Example above: \[ A = \circ \quad \varepsilon_1 \quad \circ \quad \varepsilon_2 \quad = B \]
Let $\Lambda, \Gamma$ be functors $\text{Rel}(\sigma) \to \text{Rel}(\sigma)$; $\Lambda \to \Gamma$.

Then $\forall G, H$: $\Lambda(G) \to H \iff G \to \Gamma(H)$

$\Lambda(G)$ can be constructed in polynomial time.

**Therefore:**

* If $\text{CSP}(\Gamma(H))$ is NP-complete, then so is $\text{CSP}(H)$.

* If $\text{CSP}(H)$ is poly-time, then so is $\text{CSP}(\Gamma(H))$.

**But also:**

* If $\text{CSP}(H)$ has tree duality, then so does $\text{CSP}(\Gamma(H))$.  
  (J.F., C. Tardif, 2009)
But also: • If CSP(H) has tree duality, then so does CSP(Γ(H)).

(J.F., C.Tardif, 2009)

Example: arc graph $S$

\[ A = \begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 0 & 0
\end{pmatrix} = B \]

In this case we know a description of the obstructions:

- If $\mathcal{F}$ is a complete set of tree obstructions for $H$, then Sproink($\mathcal{F}$) is a c.s.o. for $S(H)$.

\[ \mathcal{F} = \left\{ \begin{array}{c}
\end{array} \right\} \quad \text{Sproink}(\mathcal{F}) = \left\{ \begin{array}{c}
\end{array} \right\} \cdot \sim \]

E.g. $H = \begin{array}{c}
\end{array}$, $S(H) = \begin{array}{c}
\end{array} \sim \begin{array}{c}
\end{array}$
Finite Duality

H has finite duality if it admits a finite c.s.o.

J. Nešetřil, C. Tardif, 2000:
- If H admits a finite c.s.o., then it admits a finite c.s.o. of trees. (i.e., finite duality \(\Rightarrow\) tree duality)
- Any finite set \(\mathcal{F}\) of trees is a c.s.o. for some dual \(H = D(\mathcal{F})\).

A. Atserias 2005 / B. Rossman 2005:
- \(\text{CSP}(H)\) is first-order definable
  \(\iff\) H has finite duality.
Digraphs: “$\rightarrow$” (≡ existence of a homomorphism) is a pre-order on the set of all digraphs.

$F$: finite c.s.o. of tree obstructions for $H$  
$\Rightarrow Fu\{H\}$ is a finite maximal antichain (or $F$ is)

CSP($H_1,\ldots,H_n$): Does $G$ admit a homomorphism to any of $H_1,\ldots,H_n$?

If CSP($H_1,\ldots,H_n$) admits a finite c.s.o. $F$, then all elements of $F$ are forests.

J.F. J. Nešetřil, C. Tardif, 2008:  
The finite maximal antichains in “$\rightarrow$” are exactly the sets $Fu\{H_i: 1\leq i \leq n; \forall F \in F; H_i \not\rightarrow F\}$ where $F$ is a finite c.s.o. for CSP($H_1,\ldots,H_n$).
To summarise:

CSP(H) with tree duality is interesting (to me) because it reaches out to many various areas:
- Ramsey theory
- regular languages (+ Datalog)
- logic (first-order definability)
- categories (adjoints)
- universal algebra (which I didn’t talk about)