The Complexity of the Counting CSP

Víctor Dalmau

Universitat Pompeu Fabra

(Non Uniform) Counting CSP

Def: (Homomorphism formulation)

Let $\mathbf{B}$ be a (finite) structure. $\#\mathrm{CSP}(\mathbf{B})$ is the comp. problem:

- Input: structure $\mathbf{A}$
- Output: $\#$ homomorphisms from $\mathbf{A}$ to $\mathbf{B}$

(Non Uniform) Counting CSP

Def: (Homomorphism formulation)
Let $\mathbf{B}$ be a (finite) structure. $\#\mathrm{CSP}(\mathbf{B})$ is the comp. problem:

- Input: structure $\mathbf{A}$
- Output: $\#$ homomorphisms from $\mathbf{A}$ to $\mathbf{B}$

Def: (AI formulation)
Let Γ be a set of relations over B. $\#\mathrm{CSP}(\Gamma)$ is the comp. problem:

- Input: CSP instance $P = (V, B, C)$ with constraint relations in Γ
- Output: $\#$ solutions of P

Computational Complexity

Def: A *function problem* is a function $f : \Sigma^* \longrightarrow \mathbb{N}$

Computational Complexity

Def: A *function problem* is a function $f : \Sigma^* \longrightarrow \mathbb{N}$

Complexity Classes

- $f \in$ FP if is computed by a determ. polynomial-time TM

Computational Complexity

Def: A *function problem* is a function $f : \Sigma^* \to \mathbb{N}$

Complexity Classes

- $f \in$ FP if is computed by a determ. polynomial-time TM

- $f \in$ #P [Valiant 79] if exists a non deterministic polynomial-time TM M s.t. $f(x)$ is the $\#$ of accepting computations of M with input x.

Computational Complexity

Def: A *function problem* is a function $f : \Sigma^* \to \mathbb{N}$

Complexity Classes

- $f \in$ FP if is computed by a determ. polynomial-time TM
- $f \in \#$P [Valiant 79] if exists a non deterministic polynomial-time TM M s.t. $f(x)$ is the $\#$ of accepting computations of M with input x.

Note:

$$\text{FP} = \#\text{P} \quad \Rightarrow \quad \text{P} = \text{NP}$$

Computational Complexity (cont'd)

Def: (Turing Reduction)

f *reduces to* g ($f \leq_{\mathrm{TM}} g$) if f can be computed by a deterministic polynomial time TM with g as oracle.

Computational Complexity (cont'd)

Def: (Turing Reduction)

f *reduces to* g ($f \leq_{\text{TM}} g$) if f can be computed by a deterministic polynomial time TM with g as oracle.

Def: ($\#$P-hard)

f is $\#$P-hard if every problem in $\#$P reduces to it.

Computational Complexity (cont'd)

Def: (Turing Reduction)

f *reduces to* g ($f \leq_{\text{TM}} g$) if f can be computed by a deterministic polynomial time TM with g as oracle.

Def: ($\#$P-hard)

f is $\#$P-hard if every problem in $\#$P reduces to it.

If $f \leq_{\text{TM}} g$ then

$$g \in \text{FP} \quad \Rightarrow \quad f \in \text{FP}$$

$$f \in \#\text{P-hard} \quad \Rightarrow \quad g \in \#\text{P-hard}$$

Seminal results

Theorem: [Creinou, Hermann 96]
Let $\mathbf{B}$ be a 2-element structure. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is invariant under $x + y + z$. Otherwise is $\#$P-complete.

Seminal results

Theorem: [Creinou, Hermann 96]
Let $\mathbf{B}$ be a 2-element structure. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is invariant under $x + y + z$. Otherwise is $\#$P-complete.

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if all its connected components are

1. a single vertex, or
2. a complete graph with all loops, or
3. a complete bipartite graph.

Otherwise is $\#$P-complete.

Algebraic approach

[Bulatov, D. 07]

The alg. approach to CSP can be parallelized for $\#\mathrm{CSP}$

$\#$-tractability is preserved under:

1. taking relational clones (or alternatively under pp-definability)

2. subalgebras, homomorphic images and direct powers

3. restriction to idempotent term operations (or alternatively under adding constants)

Algebraic Approach (first stage)

Lemma:

$$R \in \langle \Gamma \rangle \quad \Rightarrow \quad \#\mathrm{CSP}(\Gamma \cup \{R\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Algebraic Approach (first stage)

Lemma:

$$R \in \langle \Gamma \rangle \quad \Rightarrow \quad \#\mathrm{CSP}(\Gamma \cup \{R\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Proof

- R is obtained without existential quantification.

Algebraic Approach (first stage)

Lemma:

$$R \in \langle \Gamma \rangle \quad \Rightarrow \quad \#\mathrm{CSP}(\Gamma \cup \{R\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Proof

- R is obtained without existential quantification. Trivial

Algebraic Approach (first stage)

Lemma:

$$R \in \langle \Gamma \rangle \quad \Rightarrow \quad \#\mathrm{CSP}(\Gamma \cup \{R\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Proof

- R is obtained without existential quantification. Trivial
- $R \equiv \exists \overline{y} S(\overline{x}, \overline{y})$, $S \in \Gamma$.

Algebraic Approach (first stage)

Lemma:

$$R \in \langle \Gamma \rangle \quad \Rightarrow \quad \#\mathrm{CSP}(\Gamma \cup \{R\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Proof

- R is obtained without existential quantification. Trivial
- $R \equiv \exists \overline{y} S(\overline{x}, \overline{y})$, $S \in \Gamma$. By interpolation

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{R\})$:

$$P = P', R(\overline{x_1}), \ldots, R(\overline{x_m})$$

where P' does not contain R

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{R\})$:

$$P = P', R(\overline{x_1}), \ldots, R(\overline{x_m})$$

where P' does not contain R

For $k > 0$, let P^k be the instance of $\#\mathrm{CSP}(\Gamma)$:

$$\begin{array}{lccc} P^k = P', & S(\overline{x_1}, \overline{y_{1,1}}), & \ldots, & S(\overline{x_m}, \overline{y_{1,m}}) \\ & \vdots & & \vdots \\ & S(\overline{x_1}, \overline{y_{k,1}}), & \ldots, & S(\overline{x_m}, \overline{y_{k,m}}) \end{array}$$

Proof (II). Consider P

Let $R = \{\overline{b_1}, \ldots, \overline{b_j}\}$, let φ be a solution of P

Def: $(m_1, \ldots, m_j) \in \mathbb{N}^j$ is the characteristic of φ if

$$m_i = |\{r \in \{1, \ldots, m\} | \varphi(\overline{x_r}) = \overline{b_i}\}| \text{ for every } i = 1, \ldots, j$$

Proof (II). Consider P

Let $R = \{\overline{b_1}, \ldots, \overline{b_j}\}$, let φ be a solution of P

Def: $(m_1, \ldots, m_j) \in \mathbb{N}^j$ is the characteristic of φ if

$$m_i = |\{r \in \{1, \ldots, m\} | \varphi(\overline{x_r}) = \overline{b_i}\}| \text{ for every } i = 1, \ldots, j$$

Def: Sol$(m_1, \ldots, m_j)$ is the set of solutions with characteristic $(m_1, \ldots, m_j)$

$$\#\text{ of solutions of } P = \sum_{m_1 + \cdots + m_j = m} |\text{Sol}(m_1, \ldots, m_j)|$$

We only need to compute $|\text{Sol}(m_1, \ldots, m_j)|$ for all $m_1, \ldots, m_j$

Proof (III). Consider P^k

For P^k define analogously $\mathsf{Sol}^k(m_1, \dots, m_j)$

$$\#\text{ of solutions of } P^k = \sum_{m_1+\dots+m_j=m} |\mathsf{Sol}^k(m_1, \dots, m_j)|$$

Proof (III). Consider P^k

For P^k define analogously $\text{Sol}^k(m_1, \ldots, m_j)$

$$\#\text{ of solutions of } P^k = \sum_{m_1+\cdots+m_j=m} |\text{Sol}^k(m_1, \ldots, m_j)|$$

For $i = 1, \ldots, j$, let e_i be the number of extensions of $\overline{b_i}$ in S

Proof (III). Consider P^k

For P^k define analogously $\mathsf{Sol}^k(m_1, \ldots, m_j)$

$$\# \text{ of solutions of } P^k = \sum_{m_1+\cdots+m_j=m} |\mathsf{Sol}^k(m_1, \ldots, m_j)|$$

For $i = 1, \ldots, j$, let e_i be the number of extensions of $\overline{b_i}$ in S

$$|\mathsf{Sol}^k(m_1, \ldots, m_j)| = |\mathsf{Sol}(m_1, \ldots, m_j)|(e_1^{m_1} \cdots e_j^{m_j})^k$$

Proof (IV)

The values of $|\mathsf{Sol}(m_1,\ldots,m_j)|$ are obtained solving the linear system

$$
\begin{aligned}
N_1 = &\quad \textstyle\sum_{m_1+\cdots+m_j=m} |\mathsf{Sol}(m_1,\ldots,m_j)|(e_1^{m_1}\ldots,e_j^{m_j}) \\
\vdots & \qquad\qquad\qquad\qquad \vdots \\
N_r = & \quad \textstyle\sum_{m_1+\cdots+m_j=m} |\mathsf{Sol}(m_1,\ldots,m_j)|(e_1^{m_1}\ldots,e_j^{m_j})^r
\end{aligned}
$$

with $N_k = \#$ of solutions of $P^k \quad (l = 1,\ldots,r)$
$r = \#$ of choices for $m_1,\ldots,m_j$

Note that the matrix is Vandermonde

Algebraic Approach (Second Stage)

Def: An algebra $\mathcal{B} = (B, F)$ is #-tractable if so is $\mathrm{Inv}(F)$

Lemma: If (B, F) is #-tractable then so is every of its:

- subalgebras. Trivial
- direct powers. Trivial.
- homomorphic images. By interpolation.

Algebraic Approach (third stage)

Lemma: If (B, F) is #-tractable then so is (B, F_{id})

$F_{\mathrm{id}} =$ idempotent term operations of F

Algebraic Approach (third stage)

Lemma: If (B, F) is #-tractable then so is (B, F_{id})

$F_{\text{id}} =$ idempotent term operations of F

Alternative formulation

Lemma: For every finite Γ

$$\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\}) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$$

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\})$

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\})$

Construct an instance Q (from P) of $\#\mathrm{CSP}(\Gamma)$ in the following way:

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\})$

Construct an instance Q (from P) of $\#\mathrm{CSP}(\Gamma)$ in the following way:

- Add a new variable v_b for every $b \in B$

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\})$

Construct an instance Q (from P) of $\#\mathrm{CSP}(\Gamma)$ in the following way:

- Add a new variable v_b for every $b \in B$
- Add constraint $R(v_{b_1}, \ldots, v_{b_r})$ for every $R \in \Gamma$ and every $(b_1, \ldots, b_r) \in R$ (i.e., we add a "copy" of Γ)

Proof (I)

Let P be an instance of $\#\mathrm{CSP}(\Gamma \cup \{\{b\} : b \in B\})$

Construct an instance Q (from P) of $\#\mathrm{CSP}(\Gamma)$ in the following way:

- Add a new variable v_b for every $b \in B$
- Add constraint $R(v_{b_1}, \ldots, v_{b_r})$ for every $R \in \Gamma$ and every $(b_1, \ldots, b_r) \in R$ (i.e., we add a "copy" of Γ)
- Replace every constraint $\{b\}(a)$ by $a = v_b$

Proof (II)

The $\#$ of solutions of P is obtained by:

Proof (II)

The $\#$ of solutions of P is obtained by:

- First stage. Compute the number N of solutions of Q that are injective on $\{v_b : b \in B\}$. How? Next slide

Proof (II)

The $\#$ of solutions of P is obtained by:

- First stage. Compute the number N of solutions of Q that are injective on $\{v_b : b \in B\}$. How? Next slide
- Second stage (as in A. Krokhin talk)

$$\#\text{ solutions of } P = \frac{N}{\#\text{ automorphisms of } \Gamma}$$

Proof (III). Finding N

For every partition θ of B, Q^θ is obtained adding to Q the constraints $v_b = v_{b'}$, for every $b\theta b'$.

Proof (III). Finding N

For every partition θ of B, Q^θ is obtained adding to Q the constraints $v_b = v_{b'}$, for every $b\theta b'$.

N is obtained using the Möbius inversion formula:

$$N = \sum_\theta w(\theta) \cdot (\# \text{ solutions of } Q^\theta)$$

where

- $w(0_B) = 1$
- $w(\theta) = -\sum_{\theta' \lneq \theta} w(\theta')$

A necessary condition: Mal’tsev algebras

Theorem: [Bulatov, D. 07]
If (B, F) does not have a Mal’tsev term operation then it is $\#$P-complete.

Sketch of the proof

- $\#\mathrm{CSP}(R_1) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$ for some R_1 reflex. & not sym.
  Proof: Direct from [Hageman, Mitschke 73]

Sketch of the proof

- $\#\mathrm{CSP}(R_1) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$ for some R_1 reflex. & not sym.
  Proof: Direct from [Hageman, Mitschke 73]
- $\#\mathrm{CSP}(R_2) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_1)$ for some R_2 in normal form
  Proof: R_2 is pp-definable from R_1

  $R_2 \subseteq B^2$ in NF if $R = B^2 \setminus B_0 \times B_1$ with $B_0 \cap B_1 \neq \emptyset$

Sketch of the proof

- $\#\mathrm{CSP}(R_1) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$ for some R_1 reflex. & not sym.
  Proof: Direct from [Hageman, Mitschke 73]

- $\#\mathrm{CSP}(R_2) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_1)$ for some R_2 in normal form
  Proof: R_2 is pp-definable from R_1

  $R_2 \subseteq B^2$ in NF if $R = B^2 \setminus B_0 \times B_1$ with $B_0 \cap B_1 \neq \emptyset$

- $\#\mathrm{CSP}(R_3) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_2)$ for some R_3 in NF and $|B_0| = |B_1| = 1$, $|B \setminus (B_0 \cup B_1)| \leq 1$.
  Proof: By interpolation

Sketch of the proof

- $\#\mathrm{CSP}(R_1) \leq_{\mathsf{TM}} \#\mathrm{CSP}(\Gamma)$ for some R_1 reflex. & not sym.
  Proof: Direct from [Hageman, Mitschke 73]

- $\#\mathrm{CSP}(R_2) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_1)$ for some R_2 in normal form
  Proof: R_2 is pp-definable from R_1

  $R_2 \subseteq B^2$ in NF if $R = B^2 \setminus B_0 \times B_1$ with $B_0 \cap B_1 \neq \emptyset$

- $\#\mathrm{CSP}(R_3) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_2)$ for some R_3 in NF and $|B_0| = |B_1| = 1$, $|B \setminus (B_0 \cup B_1)| \leq 1$.
  Proof: By interpolation

- $\#\mathrm{CSP}(\leq) \leq_{\mathsf{TM}} \#\mathrm{CSP}(R_3)$ where $\leq$ is the boolean implication.
  Proof: By interpolation

2-element case revisited

Theorem: [Creignou, Hermann 96]
Let $\mathbf{B}$ be a 2-element structure. If $\mathbf{B}$ is invariant under $x + y + z$ then $\#\mathrm{CSP}(\mathbf{B})$ is in FP. Otherwise is $\#$P-complete.

2-element case revisited

Theorem: [Creignou, Hermann 96]
Let $\mathbf{B}$ be a 2-element structure. If $\mathbf{B}$ is invariant under $x + y + z$ then $\#\mathrm{CSP}(\mathbf{B})$ is in FP. Otherwise is $\#$P-complete.

Proof:

- The $\#$-tractability part is straightforward

2-element case revisited

Theorem: [Creignou, Hermann 96]
Let $\mathbf{B}$ be a 2-element structure. If $\mathbf{B}$ is invariant under $x+y+z$ then $\#\mathrm{CSP}(\mathbf{B})$ is in FP. Otherwise is $\#$P-complete.

Proof:

- The $\#$-tractability part is straightforward
- The $\#$P-hardness part is a consequence of

  Theorem [Post 41] If a 2-element algebra has a Mal'tsev term then it also has $x+y+z$

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: ($\#$P-hardness)

- If G contains a loop then all elements have loops

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: (#P-hardness)

- If G contains a loop then all elements have loops
  Proof: Let a with a loop and b adjacent to a

$$(b, a)$$
$$(a, a)$$
$$(a, b)$$

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: (#P-hardness)

- If G contains a loop then all elements have loops
  Proof: Let a with a loop and b adjacent to a

$$\begin{array}{c} (b,a) \\ (a,a) \\ (a,b) \\ \hline (b,b) \end{array}$$

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: ($\#$P-hardness)

- If G contains a loop then its is a complete graph

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: (#P-hardness)

- If G contains a loop then its is a complete graph
  Proof: Let a, b, c a path

$$\frac{\begin{array}{c}(a, a)\\(b, a)\\(b, c)\end{array}}{(a, c)}$$

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: ($\#$P-hardness)

- If G does not contain a loop then is bipartite

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: (#P-hardness)

- If G does not contain a loop then is bipartite
  Proof: Let $a_1, a_2, a_3, a_4, \ldots, a_n = a_1$ be an odd cycle

$$\begin{array}{c} (a_1, a_2) \\ (a_3, a_2) \\ (a_3, a_4) \\ \hline (a_1, a_4) \end{array}$$

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: (#P-hardness)

- If G does not contain a loop then is a complete bipartite graph

Graphs revisited

Theorem: [Dyer, Greenhill 00]
Let $\mathbf{B}$ be a connected graph. Then $\#\mathrm{CSP}(\mathbf{B})$ is in FP if $\mathbf{B}$ is and isolated node, a complete graph with all loops, or a complete bipartite graph.

Proof: ($\#$P-hardness)

- If G does not contain a loop then is a complete bipartite graph
  Proof:

$$\frac{\begin{array}{c}(a_1, b_1)\\(a_1, b_2)\\(a_2, b_2)\end{array}}{(a_2, b_1)}$$

Indeed,

Theorem: Let $\mathbf{B}$ be a graph or a 2-element structure. Then $\#\mathrm{CSP}(\mathbf{B})$ in FP if $\mathbf{B}$ is invariant under a Mal'tsev operation and $\#$P-complete otherwise.

Indeed,

Theorem: Let $\mathbf{B}$ be a graph or a 2-element structure. Then $\#\mathrm{CSP}(\mathbf{B})$ in FP if $\mathbf{B}$ is invariant under a Mal'tsev operation and $\#$P-complete otherwise.

The existence of a Mal'tsev term alone is not enough to guarantee tractability even in the case of directed acyclic graphs.

Indeed,

Theorem: Let $\mathbf{B}$ be a graph or a 2-element structure. Then $\#\mathrm{CSP}(\mathbf{B})$ in FP if $\mathbf{B}$ is invariant under a Mal'tsev operation and $\#$P-complete otherwise.

The existence of a Mal'tsev term alone is not enough to guarantee tractability even in the case of directed acyclic graphs.

Partial classifications:
[Dyer, Golberg, Paterson 05] give a complete classification for DAGs
[Klima, Larose, Tesson] give a complete classification for systems of equations over semigroups

Second necessary condition: singularity

Let α, β equivalence relations with classes $A_1, \ldots, A_k$ and $B_1, \ldots, B_l$

Second necessary condition: singularity

Let α, β equivalence relations with classes $A_1, \ldots, A_k$ and $B_1, \ldots, B_l$

Theorem: [Bulatov, Grohe 05]
If

$$\text{rank}(M(\alpha, \beta)) > \#\text{ of classes of } \alpha \vee \beta$$

then $\#\mathrm{CSP}(\{\alpha, \beta\})$ is $\#$-complete.

$M(\alpha, \beta)$ is the $k \times l$ matrix wth $M(\alpha, \beta)_{i,j} = |A_i \cap B_j|$

Second necessary condition: singularity

Let α, β equivalence relations with classes $A_1, \ldots, A_k$ and $B_1, \ldots, B_l$

Theorem: [Bulatov, Grohe 05]
If

$$\text{rank}(M(\alpha, \beta)) > \# \text{ of classes of } \alpha \vee \beta$$

then $\#\mathrm{CSP}(\{\alpha, \beta\})$ is $\#$-complete.

$M(\alpha, \beta)$ is the $k \times l$ matrix wth $M(\alpha, \beta)_{i,j} = |A_i \cap B_j|$

Def: An algebra is *congruence singular* if for any two of its congruences the previous condition is satisfied.

Complete classification

Fact: If $\mathbb{V}(\mathcal{B})$ is congruence singular then $\mathcal{B}$ has a Mal'tsev term.

Complete classification

Fact: If $\mathbb{V}(\mathcal{B})$ is congruence singular then $\mathcal{B}$ has a Mal'tsev term.

Putting toguether all results we have

Theorem: An algebra $\mathcal{B}$ is $\#$P-complete if $\mathbb{V}(\mathcal{B}_{\mathrm{id}})$ is not congruence singular.

Complete classification

Fact: If $\mathbb{V}(\mathcal{B})$ is congruence singular then $\mathcal{B}$ has a Mal'tsev term.

Putting toguether all results we have

Theorem: An algebra $\mathcal{B}$ is $\#$P-complete if $\mathbb{V}(\mathcal{B}_{\mathrm{id}})$ is not congruence singular.

Theorem [Bulatov 07]
Otherwise, $\mathcal{B}$ is $\#$-tractable.