On the Subexponential Time Complexity of CSPs

Iyad Kanj
School of Computing, DePaul University

Joint work with Ronald de Haan & Stefan Szeider: Vienna University of Technology
Mathieu Liedloff & Ioan Todinca: Université d'Orléans

Outline

1. CSP
   - Problem Definition
   - Restrictions & Structural Parameters
   - Results & Assumptions
   - Summary of the Results for CSP
2. CSP with Global Constraints
   - Problem Definition & Motivation
   - Results for CSP with cTable Constraints
   - Results for CSP with Cardinality Constraints
3. Open Questions

Outline

1. CSP
   - Problem Definition
   - Restrictions & Structural Parameters
   - Results & Assumptions
   - Summary of the Results for CSP
2. CSP with Global Constraints
   - Problem Definition & Motivation
   - Results for CSP with cTable Constraints
   - Results for CSP with Cardinality Constraints
3. Open Questions

Outline

1. CSP
   - Problem Definition
   - Restrictions & Structural Parameters
   - Results & Assumptions
   - Summary of the Results for CSP
2. CSP with Global Constraints
   - Problem Definition & Motivation
   - Results for CSP with cTable Constraints
   - Results for CSP with Cardinality Constraints
3. Open Questions

Problem Definition

- An instance I of the CONSTRAINT SATISFACTION PROBLEM (CSP) is a triple $(V, D, \mathcal{C})$ where:
  - V is a finite set of *variables*
  - D is a finite set of *domain* values
  - $\mathcal{C}$ is a finite set of *constraints*
- Each constraint in $\mathcal{C}$ is a pair (S, R) where:
  - S is a non-empty sequence of distinct variables
  - R is a relation over D whose arity matches the length of S

Problem Definition

- An *assignment* or *instantiation* is a mapping that assigns every variable in V a value in D
- An assignment τ *satisfies* a constraint $C = ((x_1, \ldots, x_n), R)$ if $(\tau(x_1), \ldots, \tau(x_n)) \in R$, and τ satisfies I if it satisfies all constraints in I
- I is *consistent* or *satisfiable* if it is satisfied by some assignment
- CSP is the problem of deciding whether a given instance of CSP is consistent

Restrictions & Structural Parameters

- BOOLEAN CSP: the CSP with the *Boolean domain* $\{0, 1\}$
- r-CSP: the restriction of CSP to instances in which the arity of each constraint is at most r
- tuples: the total number of tuples in I, which is $\sum_{(S,R)\in\mathcal{C}} |R|$

Restrictions & Structural Parameters

- cons: the total number of constraints in I
- dom: the domain size
- deg: the maximum number of constraints any variable appears in
- arity: the maximum number of variables any constraint contains

Restrictions & Structural Parameters

- *primal graph*: the variables are the vertices; two vertices are adjacent iff they occur together in the scope of a constraint
- *incidence graph*: bipartite graph where one partition is the set of variables and the other is the set of constraints; a variable is adjacent to a constraint iff the variable occurs in the constraint
- tw: the treewidth of the primal graph
- tw^*: treewidth of the incidence graph

Subexponential Time

- A CSP instance with n variables can be solved in $\mathcal{O}^*(\mathrm{dom}^n)$ time by brute-force
- Significant work has been concerned with improving this trivial upper bound
- All the improvements over the trivial brute-force search give exponential running times in which the exponent is linear in n
- Can the factor dom^n be reduced to a *subexponential factor* $\mathrm{dom}^{o(n)}$, possibly considering various natural $\mathcal{NP}$-hard restrictions of the problem?

Our Work & Assumptions

- We studied the subexponential-time complexity of CSP w.r.t. restrictions on its structural parameters
- For several natural CSP parameters, we obtain threshold functions that precisely dictate its subexponential-time complexity
- This allows us to draw a detailed landscape of the subexponential-time complexity of CSP with respect to the parameters under consideration, in parallel to similar studies for CNF-SAT
- Most of the lower bound results are derived under common assumptions in complexity theory: ETH, W-hierarchy does not collapse, CNF-SAT $\notin 2^{o(n)} m^{\mathcal{O}(1)}$

CNF-SAT and Boolean CSP

- We first establish relations between the subexponential-time complexity of CSP and that of CNF-SAT
- This relation is then exploited to provide characterizations of the subexponential-time complexity of CSP and its variants
- Naturally, CNF-SAT can be modeled as a CSP problem:
  - the set of variables is the same
  - each clause is a constraint containing the satisfying tuples

CNF-Sat and Boolean CSP

- CSP of *bounded domain size* and *bounded arity* has a subexponential-time algorithm if and only if ETH fails:

Theorem

Boolean r-CSP *is in SUBEXP if and only if ETH fails*

CNF-SAT and Boolean CSP

- When we drop the bound on the domain, the problem seems to become "harder":

Theorem

If 2-CSP *is in SUBEXP then* CLIQUE *is solvable in time* $n^{o(k)}$

CNF-SAT and Boolean CSP

- We can show the following relation between CNF-SAT and BOOLEAN CSP (unbounded arity):

Theorem

If BOOLEAN CSP *is in nonuniform SUBEXP then so is* CNF-SAT

  - If the number of clauses m in the CNF-SAT instance is subexponential in n, 2^{cn} for some $0 < c < 1$, then we can use Schuler's width-reduction algorithm, followed by representing each clause as a constraint, which runs in subexponential time
  - Otherwise, m is exponential in n, the instance can be solved in polynomial time, but the exponent of the polynomial depends on c

Summary of the Results for CSP

CSP $\in$ SUBEXP	**CSP $\notin$ SUBEXP (assuming ETH)**
tuples $\in o(n)$	tuples $\in \Omega(n)$
cons $\in \mathcal{O}(1)$ (in P)	cons $\in \omega(1)$
deg $= 1$ (in P)	deg ≥ 2
arity $= 1$ or arity $= 2$ and dom ≤ 2 (in P)	arity ≥ 2 and dom ≥ 3
tw $\in o(n)$	tw $\in \Omega(n)$
tw* $\in \mathcal{O}(1)$ (in P)	tw* $\in \omega(1)$

CSP with Global Constraints: Definitions

- In CSP the constraints are given extensionally as tables
- In CSP with global constraints the constraints are given intensionally
- The CSPs with global constraints we focus on are:
  - CSP with cardinality constraints:
    - CSP with *AllDifferent* constraints denoted $\text{CSP}^{\neq}$
    - CSP with *NValue* constraints denoted $\text{CSP}^{=}$
    - CSP with *AtLeastNValue* constraints denoted $\text{CSP}^{\geq}$
    - CSP with *AtMostNValue* constraints denoted $\text{CSP}^{\leq}$
    - Some other variants/combinations ...
  - CSP with compressed tuples — *cTable* constraints — denoted CSP^{c}

CSP with Global Constraints: Definitions

In CSP with global constraints the goal remains to find an assignment that satisfies all constraints

- $\text{CSP}^{\neq}$: a constraint is satisfied if all its variables are assigned different values
- $\text{CSP}^{=}$: a constraint C is satisfied if the number of distinct values assigned to the variables in C is exactly n_C, for a given integer n_C
- $\text{CSP}^{\leq}$: a constraint C is satisfied if the number of distinct values is $\leq n_C$
- $\text{CSP}^{\geq}$: a constraint C is satisfied if the number of distinct values is $\geq n_C$

CSP with Global Constraints: Definitions

- A *cTable* constraint is a pair (S, U) where $S = (v_1, \ldots, v_r)$ is a sequence of variables, and U is a set of *compressed tuples*
  - Each compressed tuple is a sequence $(V_1, \ldots, V_r)$, where $V_i \subseteq D(v_i)$
  - A compressed tuple $(V_1, \ldots, V_r)$ represents all the tuples $(d_1, \ldots, d_r)$ with $d_i \in V_i$
  - By "decompression" one can compute from (S, U) an equivalent table constraint (S, R)
  - A *cTable* constraint is satisfied if the assignment to its variables is a tuple in the decompressed table of the constraint

All the above CSPs with global constraints are $\mathcal{NP}$-complete

Why Study CSP with Global Constraints?

- CSP with global constraints model $\mathcal{NP}$-hard problems arising in various areas
- It is often preferred to represent a constraint more succinctly than listing all the tuples of the constraint relation
- CSP^c admits a potentially exponential reduction in the space compared to an extensional table constraint
- (Hyper)Graph Coloring problems can be (easily) modeled as CSPs with cardinality constraints
- $\text{CSP}^=$, $\text{CSP}^{\neq}$, $\text{CSP}^{\geq}$, and $\text{CSP}^{\leq}$ are heavily used in constraint programming

Highlight of the Results for CSP^c

- CSP^c is a generalization of CSP
- We know that if BOOLEAN CSP is in SUBEXP then ETH fails
- We provide evidence that BOOLEAN CSP^c may be harder w.r.t. subexponential-time complexity than BOOLEAN CSP:

Proposition

Unless $W[2] = \mathbb{FPT}$, BOOLEAN CSP^c is not in SUBEXP

- The above implies that if BOOLEAN CSP^c is in SUBEXP then so is CNF-SAT

Highlight of the Results for CSP^c

- We obtain the following results for CSP^c, which match those for CSP:

CSP^c $\in$ SUBEXP	**CSP^c $\notin$ SUBEXP (assuming ETH)**
tuples $\in o(n)$	tuples $\in \Omega(n)$
cons $\in \mathcal{O}(1)$ (in P)	cons $\in \omega(1)$
deg $= 1$ (in P)	deg ≥ 2
tw $\in o(n)$	tw $\in \Omega(n)$
tw* $\in \mathcal{O}(1)$ (in P)	tw* $\in \omega(1)$

Highlight of the Results for CSP$^{\neq}$

- By a simple reduction to List Coloring we have:

Proposition

CSP$^{\neq}$ *can be solved in time* $\mathcal{O}^*(2^n)$

- Therefore, for any domain size $\mathrm{dom} = \omega(1)$, since $2^n = \mathrm{dom}^{o(n)}$ we have:

Corollary

The CSP$^{\neq}$ *with* $\mathrm{dom} = \omega(1)$ *is in SUBEXP*

- Therefore, we can focus on CSP$^{\neq}$ restricted to domain size d, where $d > 2$ is a constant ($d = 2$ is in $\mathcal{P}$)

Highlight of the Results for CSP$^{\neq}$

- We highlight the following tight results we obtain for CSP$^{\neq}$:

CSP$^{\neq}$ $\in$ SUBEXP	**CSP$^{\neq}$ $\notin$ SUBEXP (assuming ETH)**
dom $= \omega(1)$	dom $= d \geq 3$
—	cons $\in \Omega(n)$
tw $\in o(n)$	tw $\in \Omega(n)$
tw* $\in o(n)$	tw* $\in \Omega(n)$

Highlight of the Results for $\text{CSP}^=$, $\text{CSP}^\leq$, $\text{CSP}^\geq$

- W.r.t. tw, we have this tight result for $\text{CSP}^=$, $\text{CSP}^\geq$, and $\text{CSP}^\leq$:

Theorem

$\text{CSP}^=$, $\text{CSP}^\geq$, *and* $\text{CSP}^\leq$ *with* $\text{tw} = o(n)$ *are in SUBEXP, and unless ETH fails,* $\text{CSP}^=$, $\text{CSP}^\geq$, *and* $\text{CSP}^\leq$ *with* $\text{tw} = \Omega(n)$ *are not in SUBEXP*

Highlight of the Results for $\mathrm{CSP}^{=}$, $\mathrm{CSP}^{\leq}$, $\mathrm{CSP}^{\geq}$

- W.r.t. cons, we have:

Theorem

Unless ETH fails, $\mathrm{CSP}^{=}$, $\mathrm{CSP}^{\neq}$, $\mathrm{CSP}^{\geq}$, *and* $\mathrm{CSP}^{\leq}$ *with* $\mathrm{cons} = \Omega(n)$ *and* $\mathrm{dom} = O(1)$ *are not in SUBEXP*

Theorem

$\mathrm{CSP}^{=}$, $\mathrm{CSP}^{\neq}$, $\mathrm{CSP}^{\geq}$, *and* $\mathrm{CSP}^{\leq}$ *with* $\mathrm{cons} = o(n)$ *are in SUBEXP*

Open Questions

- We showed that CSP with cardinality constraints is solvable in subexponential time when $\text{cons} = o(n)$
- If $\text{cons} = \Omega(n)$, we could only show that these problems are not solvable in subexponential time (assuming ETH) when $\text{dom} = O(1)$

Open Questions

- What happens when dom $= \omega(1)$ and cons $= \Omega(n)$? In particular, what happens when both cons and dom are linear in n?
  - Can we show in such case that the problem is solvable in subexponential time?
  - Or, can we use the recent lower bound results for Graph Homomorphism by Fomin et al. to rule out the existence of subexponential-time algorithms in such case?