A brief history of the Constraint Satisfaction Problem

Alexandr Kazda

CU Boulder

January 14th 2021

Introduction

- This semester: Math motivated by the Constraint Satisfaction Problem
- Overlaps with computer science...
- ...but focused on nice mathematics, we will ignore e.g. SAT solvers used in practice
- Grad students: You get credit for giving a talk
- Talks are not recorded
- Some international presence, but I want a “small local seminar” feel
- Seminar time???

Formula satisfiability (SAT)

- Input: A formula of the form

$$\Psi(x_1, x_2, \dots, x_n) = (x_7 \vee \neg x_5 \vee x_1) \wedge (\neg x_7 \vee x_1) \wedge (x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4) \wedge \dots$$

- Decision version: Does there exist an assignment $\{x_1, \dots, x_n\} \to \{0, 1\}$ satisfying Ψ?
- Search version: Find an assignment $\{x_1, \dots, x_n\} \to \{0, 1\}$ satisfying Ψ if one exists
- 3-SAT: Like SAT, but 3 literals per clause:

$$\Psi(x_1, x_2, \dots, x_n, t) = (x_7 \vee \neg x_5 \vee x_1) \wedge (\neg x_7 \vee x_1 \vee x_1) \wedge (x_1 \vee x_2 \vee t) \\ \wedge (\neg t \vee \neg x_3 \vee \neg x_4) \wedge \dots$$

- SAT and 3-SAT: The first problems shown to be NP-complete (S. Cook, The complexity of theorem-proving procedures, 1971)

Modifying 3-SAT

- 3-SAT rewritten

$$\Psi(x_1, x_2, \ldots, x_n) = \bigwedge_{i=1}^{m} R_i(x_{i1}, x_{i2}, x_{i3})$$

- Where R_i is one of 8 predicates; example is $F(a, b, c) = a \vee \neg b \vee c$
- What if we take a different set of predicates($=$ Constraint language)?
- Example: Say our formulas will be

$$\Psi(x_1, x_2, \ldots, x_n) = \bigwedge_{i=1}^{m} (x_{i1} \vee x_{i2} \vee x_{i3}),$$

- $x_i = 1$ for all i always satisfies Ψ
- Deciding satsifiability is easy!

Linear equations as formula satisfiability

- More interesting example:

$$\begin{aligned} S(a,b,c) &= 1 && \text{iff } a+b+c=0 \pmod 2 \\ C_0(a) &= 1 && \text{iff } a=0 \\ C_1(a) &= 1 && \text{iff } a=1 \end{aligned}$$

- This constraint language lets us write systems of linear equations over $\mathbb{Z}_2$
- Example

$$\begin{aligned} x_1 + x_2 + x_4 + x_5 &= 1 \\ x_1 + x_3 &= 0 \end{aligned}$$

- is equivalent to the formula (note the extra variables $t_1, \ldots, t_4$)

$$S(x_1, x_2, t_1) \wedge S(t_1, x_4, t_2) \wedge S(t_2, x_5, t_3) \wedge C_1(t_3) \wedge S(x_1, x_3, t_4) \wedge C_0(t_4)$$

Horn 3-SAT

- Alfred Horn, On Sentences Which are True of Direct Unions of Algebras, 1951.
- Constraint language $C_0(a)$, $C_1(a)$, and $R(a, b, c) = \neg a \vee \neg b \vee c$,
- Observe: $R(a, b, c) = (a \wedge b) \Rightarrow c$
- Example:

$$R(x_1, x_2, x_3) \wedge C_1(x_1) \wedge C_1(x_2) \wedge C_0(x_3)$$

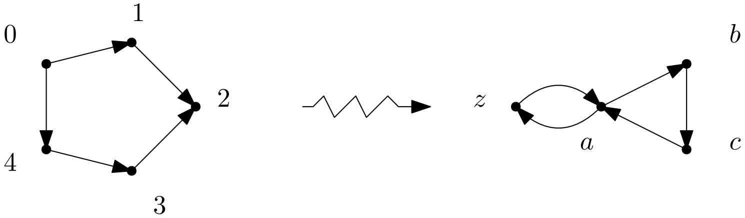
- x_1 and x_2 are forced to be 1
- Use R to propagate the forced 1 from x_1, x_2 to x_3
- We see that x_3 has to be 0 and 1 at the same time – inconsistent
- This is (roughly) how local consistency checking works
- Consistency checking solves Horn 3-SAT

Schaefer's dichotomy

- Complete classification for variables over $\{0,1\}$ by T. Schaefer, The Complexity of Satisfiability Problems, 1978
- Depending on the constraint language, the problem is either NP-complete or in P
- We have a dichotomy (assume $P \neq NP$)
- The easy cases are:
  - Always satisfiable Ψ,
  - always unsatisfiable Ψ,
  - linear equations,
  - problems solvable by local consistency

Digraph homomorphisms

- G, H be directed graphs (digraphs)
- Homomorphism is a map $V(G) \to V(H)$ that maps edges to edges



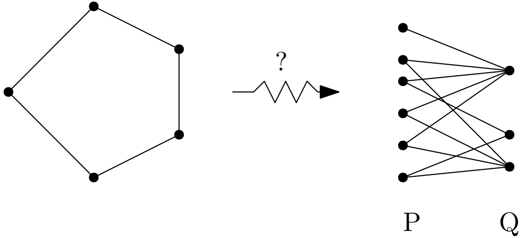
- Given G, H, how to decide if there is a homomorphism?
- There is a formula for that!

$$E(x_0, x_1) \wedge E(x_1, x_2) \wedge E(x_3, x_2) \wedge E(x_4, x_3) \wedge E(x_0, x_4)$$

- Different H gives a different predicate "E"
- CSP(H) be the problem "Given G, decide if $G \to H$."
- Can generalize this to CSP($\mathbb{A}$) where $\mathbb{A}$ is some relational structure $\mathbb{A} = (A; R_1, \dots, R_n)$

Hell-Nešetřil's dichotomy

- CSP(H) be the problem "Given G, decide if $G \to H$."
- Let H be a symmetric graph ($E(a,b) \Leftrightarrow E(b,a)$)
- If H is bipartite, CSP(H) is easy:



- For any other symmetric graph H, CSP(H) is NP-complete
- P. Hell, J. Nešetřil, On the complexity of H-coloring, 1990
- Again P vs. NP-complete

Monotone Monadic SNP without inequality I

- So far we always got problems in P or NP-complete
- If P $\neq$ NP then there are infinitely many intermediate classes between P and NP (R. Ladner, On the Structure of Polynomial Time Reducibility, 1975)
- T. Feder, M. Vardi, Monotone Monadic SNP and Constraint Satisfaction, 1993 (journal version 1998)
- MMSNP without inequality is a subclass of NP
- Feder and Vardi conjecture: MMSNP without inequality contains no intermediate problems

Monotone Monadic SNP without inequality II

- Feder and Vardi: Each MMSNP without inequality is computationally equivalent to CSP($\mathbb{A}$) for some $\mathbb{A}$ finite
- Dichotomy conjecture: Each CSP($\mathbb{A}$) is either in P or NP-complete; no intermediate problems
- Thus complexity CSP($\mathbb{A}$) is a way to characterize the complexity of a sizeable part of NP

Algebraic approach to CSP($\mathbb{A}$) I

- Around 2000: A. Bulatov, D. Cohen, P. Jeavons, M. Gyssens, A. Krokhin, J. Pearson
- Reductions from logic and universal algebra
- Example: If $\mathbb{A} = (A; R_1, R_2)$ and $\mathbb{B} = (A; S)$ where

$$S(a,b) = \exists c,\ R_1(a,c) \wedge R_2(c,b,b),$$

  then CSP($\mathbb{B}$) reduces to CSP($\mathbb{A}$)
- How? Given CSP($\mathbb{B}$) formula such as

$$S(x_1,x_2) \wedge S(x_2,x_3) \wedge S(x_1,x_1),$$

  add new variables and rewrite S's:

$$\begin{aligned} & R_1(x_1,y_1) \wedge R_2(y_1,x_2,x_2) \\ \wedge\ & R_1(x_2,y_2) \wedge R_2(y_2,x_3,x_3) \\ \wedge\ & R_1(x_1,y_3) \wedge R_2(y_3,x_1,x_1) \end{aligned}$$

Algebraic approach to CSP($\mathbb{A}$) II

- A few other reductions give us that complexity of CSP($\mathbb{A}$) depends only on the set of polymorphisms of $\mathbb{A}$
- Polymorphims: Mappings $\mathbb{A}^n \to \mathbb{A}$ that preserve the relations of $\mathbb{A}$
- Pol($\mathbb{A}$) is a clone of operations: Contains projections and is closed under composition
- If Pol($\mathbb{A}$) $\subseteq$ Pol($\mathbb{B}$) then CSP($\mathbb{B}$) reduces to CSP($\mathbb{A}$)
- Later improved to Pol($\mathbb{A}$) $\to$ Pol($\mathbb{B}$) where $\to$ preserves identities
- Later improved to Pol($\mathbb{A}$) $\to$ Pol($\mathbb{B}$) where $\to$ preserves identities without composition
- Universal algebraic approach to CSP

Example of a polymorphism

- Polymorphisms of $\mathbb{A} \approx$ higher arity symmetries of $\mathbb{A}$
- Consider $\mathbb{A} = (\{0,1\}; S, C_0, C_1)$ with

$$\begin{aligned} S(a,b,c) &= 1 && \text{iff } a+b+c = 0 \pmod 2 \\ C_0(a) &= 1 && \text{iff } a = 0 \\ C_1(a) &= 1 && \text{iff } a = 1 \end{aligned}$$

- This has the polymorphism $p(a,b,c) = a+b+c \pmod 2$
- $C_0(a), C_0(b), C_0(c) \Rightarrow p(a,b,c) = 0 \Rightarrow C_0(p(a,b,c)$
-
$$S\begin{pmatrix} a \\ a' \\ a'' \end{pmatrix}, S\begin{pmatrix} b \\ b' \\ b'' \end{pmatrix}, S\begin{pmatrix} c \\ c' \\ c'' \end{pmatrix} \Rightarrow S\begin{pmatrix} p(a,b,c) \\ p(a',b',c') \\ p(a'',b'',c'') \end{pmatrix}$$
- Whenever $\mathrm{Pol}(\mathbb{A})$ contains p such that $p(x,x,y) = y$ and $p(x,y,y) = x$ for all x, y then $\mathrm{CSP}(\mathbb{A})$ is in P (A. Bulatov, Mal'tsev Constraints Are Tractable, 2002)

The hard cases of CSP

- The smaller Pol($\mathbb{A}$), the harder CSP($\mathbb{A}$)
- Smallest possible Pol($\mathbb{A}$): Only projections

$$\pi_i(x_1, \ldots, x_n) = x_i$$

- If CSP($\mathbb{A}$) contains only projection-like operations, then CSP($\mathbb{A}$) $\to$ Pol(3-SAT)
- Then 3-SAT reduces to CSP($\mathbb{A}$) $\Rightarrow$ CSP($\mathbb{A}$) is NP-complete
- Algebraic dichotomy conjecture: If Pol($\mathbb{A}$) contains an operation that is not projection-like, then CSP($\mathbb{A}$) in P

Towards dichotomy for CSP

- It remains "only" to give a P-time algorithm for any CSP($\mathbb{A}$) when $\mathbb{A}$ has a nontrivial polymorphism
- L. Barto, M. Kozik: Characterized CSP($\mathbb{A}$) solvable by local consistency methods (published 2014)
- "Local consistency works iff CSP($\mathbb{A}$) cannot simulate linear equations."
- Group theory-like (or Gaussian elimination-like) algorithm for a big class of CSPs (Paweł Idziak, Petar Marković, Ralph McKenzie, Matthew Valeriote, and Ross Willard, Tractability and learnability arising from algebras with few subpowers, 2007)
- Attempts were made at unifying the two approaches (the most sophisticated by Miklós Maróti)...
- ...but there was only minimal overall progress

The proofs of dichotomy

- The year is 2017...
- A. Bulatov and D. Zhuk independently announce their CSP algorithms that work for any $\mathbb{A}$ with nontrivial $\mathrm{Pol}(\mathbb{A})$
- Published at the FOCS 2017 conference
- Ongoing project: Simplify the proofs and extract new mathematics from them
- Another proof was announced by A. Rafiey, J. Kinne and T. Feder, but R. Willard found a counterexample

What's next

- Valued CSP: Find a minimum of a sum of functions such as

$$f(x_1, x_2) + f(x_1, x_3) + g(x_3)$$

- For f and g with values 0 and ∞ we get CSP
- VCSP dichotomy conditional on CSP dichotomy proven in 2015 (Vladimir Kolmogorov, Andrei Krokhin and Michal Rolínek, The Complexity of General-Valued CSPs, 2015)
- (V)CSP with infinite templates (M. Bodirsky, M. Pinsker and friends)
- PCSP($\mathbb{A}, \mathbb{B}$): Assume $\mathbb{A} \to \mathbb{B}$. Input is a structure $\mathbb{C}$ and the goal is to decide between $\mathbb{C} \to \mathbb{A}$ and $\mathbb{C} \not\to \mathbb{B}$
- Example: Distinguish 3-colorable graphs from graphs that are not even 100-colorable
- Complexity of PCSP is an open problem

Further reading

- L. Barto, A. Krokhin, R. Willard, Polymorphisms, and how to use them, in "The Constraint Satisfaction Problem: Complexity and Approximability", Dagstuhl Follow-Ups, vol. 7, 1–44, 2017 http://drops.dagstuhl.de/opus/volltexte/2017/6959/pdf/DFU-Vol7-15301-1.pdf
- Andrei A. Bulatov. 2018. Constraint satisfaction problems: complexity and algorithms. ACM SIGLOG News 5, 4 (October 2018), 4–24. DOI: https://doi.org/10.1145/3292048.3292050