Advanced Algorithmics

Strategies for Tackling Hard Problems

Sebastian Wild

Markus Nebel

Lecture 4

2017-05-01

para-$\mathcal{NP}$ is too large a class to yield have interesting complete problems
$\rightsquigarrow$ We must weaken the class. Unfortunately, TM inconvenient here.

$f(k) \cdot p(|x|)$

Definition 3.15 (NRAM, κ-restricted)

An NRAM M is a word-RAM with $w = \mathcal{O}(\log n)$ with the additional operation to nondeterministically guess a number between 0 and a current register content.
An NRAM M that decides a parametrized problem (L, κ) is *κ-restricted* if on input $x \in \Sigma^\star$ with $n = |x|$ and $k = \kappa(x)$

1. it performs at most $f(k) \cdot p(n)$ steps,
2. at most $g(k)$ of them nondeterministic,
3. uses at most $f(k) \cdot p(n)$ registers, and
4. those never contain numbers larger than $f(k) \cdot p(n)$.

for computable functions f and g. ◀

Definition 3.16 (W[P])

The class $\mathcal{W}[P]$ is the set of all parametrized problems (L, κ) decidable by a κ-restricted NRAM. ◀

A first $\mathcal{W}[P]$-complete problem?

Define hardness and completeness for $\mathcal{W}[P]$ using $\leq_{fpt}$.

What could be the mother of all $\mathcal{W}[P]$-complete problems?

Some parametrized version of SAT? Parameter #variables does not work. (Why?)

p-SAT $\in$ FPT

What can be guessed using k numbers in $[n]$?

$\hookrightarrow$ Subset of variables of size k

Recall: Weight of variable assignment $v : [n] \to \{0, 1\}$
is the #1's assigned.

Definition 3.17 (Weighted Satisfiability)

Given: Boolean formula φ and integer $k \in \mathbb{N}$

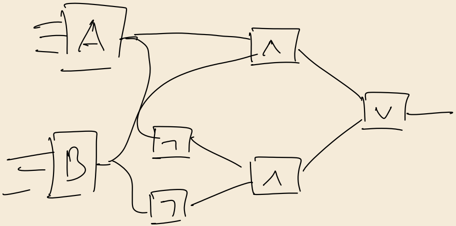
Parameter: k

Question: $\exists$ satisfying assignment with weight $= k$?

φ could be as

- formula (e.g., CNF, d-CNF, ...) $A \leftrightarrow B \equiv (A \wedge B) \vee (\neg A \wedge \neg B)$

- circuit



Theorem 3.18 (p-WSAT[CIRC] is $\mathcal{W}[P]$-complete)
The weighted satisfiability problem for boolean circuits parametrized by the weight is $\mathcal{W}[P]$-complete.

Proof Idea: Simulate NRAM by multi-tape TM
Simulate multi-tape TM by boolean circuit polynomial in size.

trickery on circuits to convert that circuit C to D so that

C satisfiable $\Leftrightarrow$ D k-satisfiable
weight-k assignment

Circuit satisfiability still too strong to show hardness of many interesting problems.
$\rightsquigarrow$ We must weaken the class *further*.

Definition 3.19 (tail-nondeterministic NRAM)

A κ-restricted NRAM M for a problem (L, κ) is called *tail-nondeterministic* if all nondeterministic steps occur only among the last $h(\kappa(x))$ steps.

Definition 3.20 (W[1])

The class $\mathcal{W}[1]$ consists of all parametrized decision problems (L, κ) that are decided by a tail-nondeterministic κ-restricted NRAM.

As before, define hardness and completeness for $\mathcal{W}[1]$ w.r.t. $\leq_{fpt}$.

Same questions: What is a first W[1]-complete problem.

Definition 3.21 (k-step Halting Problem)

Given: A nondeterministic (single-tape) Turing machine M, an input x and $k \in \mathbb{N}$ be given.
Parameter: k
Question: Does M accepts x after at most k computation steps?

Remark 3.22

- M is part of input, so state space and tape alphabet are not fixed!

$\rightsquigarrow$ up to n different non-deterministic choices in *each* step. (n is size of encoding of M)

$\rightsquigarrow$ Trivial algorithm has to simulate up to n^{k+1} steps of M.

- Equivalent here to halting problem for $x = \varepsilon$, since we can hard-wire the given input into the states of a TM M' constructed from M.

<u>Proof:</u> "$\in W[1]$" Given TM M, x, k construct NRAM A

A simulates M for k steps and copies output

"W[1]-hard" Assume NRAM A (tail-nondet $\varkappa$-restricted) given that decides $(L, \varkappa)$

Given A, $\tilde{x}$:

generate as part of the reduction an input x that hardcodes the result of deterministic part on $\tilde{x}$ of computation

$\Rightarrow$ Write down the state of A before the first nondet step as x

$\Rightarrow$ Generate TM M that simulates $h(\varkappa(\tilde{x}))$ steps of A

$k = f(h(\varkappa(\tilde{x})))$ $\square$

Theorem 3.23 (k-step halting problem $\mathcal{W}[1]$-complete)
The k-step Halting Problem (for single-tape TM) parametrized by k is $\mathcal{W}[1]$-complete. ◀

More natural problems?

2-CNF-SAT $\in$ P

Definition 3.24 (p-WSAT[2CNF])

Given: Boolean formula φ in 2-CNF and integer $k \in \mathbb{N}$
Parameter: k
Question: $\exists$ satisfying assignment with weight $= k$?

Theorem 3.25

p-WSAT[2CNF] is $\mathcal{W}[1]$-complete.

Theorem 3.26

p-WSAT[2CNF$^-$] is $\mathcal{W}[1]$-complete.

$(\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_2 \vee \bar{x}_3) \wedge \cdots$

p-WSAT[2CNF$^-$] means: *all* literals *negated*.

$W[1] \neq FPT$ widely believed

Thm: p-Independent-Set is W[1]-complete

"$\in W[1]$" given $G=(V,E)$ $k \in \mathbb{N}$

NRAM guess k numbers $=$ nodes for each k-subset check the k^2 pairs for no edge

"W[1]-hard" $p\text{-WSAT}[2CNF^-] \leq_{fpt}$ p-Indepdnt-Set

$$\varphi = \bigwedge_{(i,j)\in I} (\overline{x_i} \vee \overline{x_j}) \qquad n^2 \qquad G = ([n], \{\{i,j\} : (i,j) \in I\})$$

k $\qquad$ k

Thm: p-Clique is W[1]-complete

as above, replace edge $\leftrightarrow$ no edge

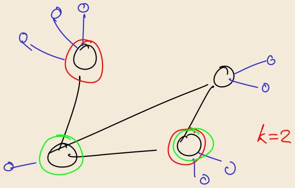
Definition: p-Partial-Vertex-Cover

Given: $G=(V,E)$, $t \in \mathbb{N}$, $k \in \mathbb{N}$

Parameters: k

Question: $\exists\, C \subseteq V$: $|C| = k$ s.t. C covers at least t edges?

Thm: p-PVC is W[1]-hard.



Proof: p-Independ-Set $\leq_{fpt}$ p-PVC

$G=(V,E)$, k

$G'=(V',E')$ $\qquad V' = V \cup \bigcup_{v\in V} \{w_{v,i} : i = 1,\dots,|V|-\deg(v)\}$

$t = k\cdot|V|$ $\qquad E' = E \cup \bigcup_{v\in V} \{\{v, w_{v,i}\} : i = \cdot\ —\ \}$

p-Vertex-Cover $\in$ FPT

Pick k edges, for each recursive test both endpoints

$\rightsquigarrow 2^k$, check all $\square$