CMSC 858F: Algorithmic Lower Bounds: Fun with Hardness Proofs
Spring 2022
Chapter 8: Parametrized Complexity

Instructor: Mohammad T. Hajiaghayi
Scribe: Joel Rajakumar

April 22, 2022

1 Summary

This chapter discusses subclasses within NP that characterize each decision problem's dependence on the input parameter k. We end up with the W hierarchy, and we can describe this hierarchy through each class's complete problems or through properties of their circuit representations.

1. W[0], also known as FPT
   (a) Definition (in terms of problem difficulty): contains problems that can be solved in $O(f(k)n^{O(1)})$ time
   (b) Definition (in terms of circuit representation): contains problems whose circuit representation contains no large fan-in gates
   (c) Kernelization: FPT problems can be solved in $O(n^{O(1)} + 2^{f(k)})$ time through a $O(n^{O(1)})$ preprocessing step that removes dependency on n
   (d) Complete Problems:
      i. Vertex cover
      ii. Planar Dominating Set
2. W[1]
   (a) Definition (in terms of problem difficulty): contains problems that can be parameter-reduced to the k-step-NDM problem
   (b) Definition (in terms of circuit representation): contains problems whose circuit representation contains at most one large fan-in gate

   (c) Complete Problems:
      i. Clique (complete for regular graphs as well)
      ii. Independent Set (complete for regular graphs as well)
      iii. Shortest Common Subsequence
      iv. Restricted Shortest Common Subsequence
      v. Partial Vertex Cover
      vi. Multicolored Clique
      vii. Multicolored Independent Set
      viii. Floodit

3. W[2]
   (a) Definition (in terms of problem difficulty): contains problems that can be parameter-reduced to the Dominating Set problem
   (b) Definition (in terms of circuit representation): contains problems whose circuit representation contains at most two large fan-in gates
   (c) Complete Problems:
      i. Dominating Set
      ii. Weighted Circuit SAT

4. W[P]: contains problems that can be parameter-reduced to Circuit SAT

5. W[SAT]: contains problems that can be parameter-reduced to SAT

6. XP: contains problems that can be solved in $O(f(k)n^{O(g(k))}$ time

2 Suggestions

1. In this chapter, the complexity classes FPT, $W[1]$, and $W[2]$ are introduced by going through harder and harder problems (in terms of parametrized complexity) and trying to find lower bounds on hardness. I definitely found this presentation very helpful compared to other texts as it motivated the construction of these classes. However, I felt like the introduction of the k-step Nondeterministic Turing Machinery (k-step-NTM) may not need to be so early and could be skipped till later. Specifically, I think you can straight away define W[1] as the complexity class for which Clique and IS are complete. The fact that parametrized Clique and IS have no known polynomial algorithm already motivates the construction of such a parametrized complexity class that is harder than FPT and complete for Clique and IS. You have done something similar to what I am suggesting in the next section, where you have defined the complexity class W[2] as the class which is complete for Dominating Set solely based on the intuition that Dominating Set is harder than Clique and IS (but without using the W[2] complete Turing Machinery problem - "k-step-NTM with

multiple tapes"). I think that it would be more clear if the Turing machine problems were listed later on or in a separate section as explained in the next point

2. In this chapter, the parametrized complexity classes are first introduced through their complete problems. Then, towards the end, you introduced an equivalent definition of these complexity classes through properties of their circuit representations. I think you can add another section after, where you introduce another equivalent definition of these complexity classes in terms of variants of the Turing machine halting problems, i.e.
   - W[1]: k-step-NTM
   - W[i]: k-step-multi-NTM (multiple (i) tapes)
   - W[P]: Bounded-NTM (explained in suggested problem 11)
   - W[SAT]: I assume there may exist some variant of the halting problem that is complete for this class, but I do not know
3. Is it true that all problems in FPT are kernelizable? This is not clear from the text because FPT is defined after kernelization, and it is defined without reference to kernelization. I only was able to infer this due to the comment at the end of theorem 8.2.1: "this will soon be called Fixed Parameter Tractable." I think you should either switch the order of 8.2 and 8.3 (because the definition of FPT immediately follows the discussion in 8.1), or you should make this another lemma/theorem
4. I think you should refer to all problems with "k-" as a prefix for clarity (i.e. "k-clique")
5. It would be useful to have a diagram/chart of the different problems in each class of the W heirarchy (similar to my list in the summary)
6. I think it would be useful to discuss/mention a problem which is outside of XP and in NP
7. In some other texts, k-step-NTM is refered to as a "halting problem" or an "acceptance problem," for example k-step-Turing Machine Acceptance. I, personally, would find this labelling more recognizeable and understandable than "Turing Machinery"
8. FPT is the class of problems that can be solved in $O(f(k)n^{O(1)})$ time while XP is the class of problems that can be solved in $O(f(k)n^{O(g(k))}$. I found myself curious what is the runtime big-O of the intermediate classes W[1] and W[2]. Is this well understood? If so, it would be useful to mention this.

3 Problems

1. Prove that the Odd Set problem is W[1]-complete
   - INPUT: Set system F over a universe U
   - QUESTION: Is there a set S of at most k elements such that $|S \cap F|$ is odd for every $F \in F$
   - BONUS: Prove that the Exact Odd Set problem is also W[1]-complete ($|S| = k$)
   - SOURCE: http://www.cs.bme.hu/ dmarx/papers/marx-telaviv2017-hardness.pdf
2. Prove that the Exact Even Set problem is W[1]-complete
   - INPUT: Set system F over a universe U
   - QUESTION: Is there a set S of exactly k elements such that $|S \cap F|$ is even for every $F \in F$
   - BONUS: The parametrized complexity of the non-exact Even Set problem is open
   - SOURCE: http://www.cs.bme.hu/ dmarx/papers/marx-telaviv2017-hardness.pdf
3. Prove that Independent Set is W[1]-complete for 2-interval graphs (Exercise in Downey & Fellows textbook, "Parametrized Complexity")
4. Prove that Hitting Set is W[2] complete
5. Prove that Independent Dominating Set is W[2] complete
6. Prove that Red Blue Dominating Set is W[2] complete
   - INPUT: A bipartite graph $G(V_B \cup V_R, E)$
   - QUESTION: Does G have a subset $D \subseteq V_B$ of at most k "blue" vertices such that each "red" vertex from V_R is adjacent to a vertex in D
7. Prove that Maximal Irredundant Set is W[2] complete
   - INPUT: A graph $G(V, E)$
   - QUESTION: Does G have a set X of k vertices such that for each member x of X, there is a $y \in G$ such that either y = x or (x, y) is an edge, and y is not adjacent to nor a member of S, and, furthermore, S is maximal with this property?
   - SOURCE: Exercise in Downey & Fellows textbook, "Parametrized Complexity"

8. Fix integers $\alpha, c \geq 1$ with $d \geq \alpha^c$. Prove that the problem of Clique restricted to d-regular graphs is W[1]-complete (Exercise in Downey & Fellows textbook, "Parametrized Complexity")

9. Prove that the Kernel problem is W[2]-complete

   - INPUT: A directed graph $G(V, E)$
   - QUESTION: Decide whether G has a kernel of k elements
   - SOURCE: Stated as theorem in Flum & Grohe textbook, "Parametrized Complexity Theory"

10. Prove that the Bounded-NTM-Halt problem is W[P]-complete

    - INPUT: A nondeterministic Turing machine M and $n \in \mathbb{N}$ in unary
    - QUESTION: Does M accept the empty string in at most n steps and using at most k nondeterministic steps?
    - SOURCE: Stated as theorem in Flum & Grohe textbook, "Parametrized Complexity Theory"