1 SAT With Few Clauses

Recall CNF-SAT-CLAUSES = \{ (F, k) \mid F \text{ is a satisfiable CNF with } k \text{ clauses} \}.

Prove that CNF-SAT-CLAUSES is solvable in \( O^*(2^k) \) time (recall the \( O^* \) notation omits polynomial factors).

In fact, one can count the number of SAT assignments to \( F \) in such time.

**Hint:** Try the following strategy. Pick an arbitrary clause \( C \). Either (i) remove \( C \) from the formula, and recursively count the number of SAT assignments, or (ii) set all literals in \( C \) to false, remove \( C \), and recursively count the number of SAT assignments. Show the total number of SAT assignments is a linear combination of the results from (i) and (ii), and that this strategy makes at most \( 2^k \) recursive calls (when combined with appropriate base cases and details!).

2 Some Colorful Problems

The following “colorful” variants of the canonical fine-grained problems are often useful to work with, in reductions.

(a) Let \( OV_d \) be the problem: Given \( n \) vectors in \( \{0, 1\}^d \), is there an orthogonal pair?

Let Colorful-\( OV_d \) be the problem: Given \( n \) red vectors and \( n \) blue vectors in \( \{0, 1\}^d \), is there a red-blue pair which is orthogonal?

Let \( t(n) \leq O(n^2) \). Prove that for every \( d \), if \( OV_d \) is in \( O(t(n)) \) time then Colorful-\( OV_{d-2} \) is in \( O(t(n)) \) time.

(b) Let 3SUM be the problem: Given \( n \) numbers, are there three which sum to zero?

Let Colorful-3SUM be the problem: Given \( n \) numbers, where each number is colored either red, green, or blue, is there a red number \( r \), a blue number \( b \), and a green number \( g \) which all sum to zero?

Let \( t(n) \leq O(n^2) \). Prove that 3SUM is solvable in \( O(t(n)) \) time if and only if Colorful-3SUM is solvable in \( O(t(n)) \) time.

(c) Let Zero-Triangle be the problem: Given an \( n \)-node graph with weights on the edges, are there three edges of the form \((a, b), (b, c), (c, a)\) such that the weights of the three sum to zero?

Let Colorful-Zero-Triangle be the problem: Given an \( n \)-node edge-weighted graph where each node is colored either red, green, or blue, is there a zero-triangle with a red, green, and a blue node?

Let \( t(n) \leq O(n^3) \). Prove that Zero-Triangle is solvable in \( O(t(n)) \) time if and only if Colorful-Zero-Triangle is solvable in \( O(t(n)) \) time.

3 Two More Colorful Problems

Let Edge-Monochromatic-Triangle be the problem: Given an \( n \)-node graph with colors on its edges, is there a triangle with all three edges having the same color? Note the number of colors here is unbounded: it could be as large as \( n^2 \).
(a) First, let’s suppose all edges are the same color. Show how to determine if an \( n \)-node graph has a triangle in \( O(n^\omega) \) time (here, \( \omega < 2.373 \) is the matrix multiplication exponent).

(b) Again, let’s suppose all edges have the same color, and the graph is in fact sparse: it has at most \( m \) edges. Show how to determine if the graph has a triangle in \( O(m^{2\omega/(\omega+1)}) \) time.

The footnote has a big hint. Please think about it first, before reading the footnote!

(c) Show that if the total number of distinct edge colors in the graph is at most \( K \), then Edge-Monochromatic-Triangle is solvable in \( O(K \cdot n^\omega) \) time. (Use part a.)

(d) Use parts b and c together to show that Edge-Monochromatic-Triangle is in \( O(n^{3-\varepsilon}) \) time for some \( \varepsilon > 0 \).
That is, Edge-Monochromatic-Triangle is in truly subcubic time!
How large can \( \varepsilon > 0 \) be? Can Edge-Monochromatic-Triangle be in \( O(n^\omega) \) time?

(e) Let Edge-Colorful-Triangle be the problem: Given an \( n \)-node graph with colors on its edges, is there a triangle with all three edges having different colors?

Given an edge-colored graph, pick a uniform random mapping from the set of edge colors to the set \( \{R, G, B\} \), and replace the edge colors in the graph with R, G, B appropriately. Show that if the original graph had an edge-colorful triangle, then the new graph still has an edge-colorful triangle with some constant probability greater than zero (compute this constant!).
Use these observations to solve Edge-Colorful-Triangle in \( O(n^\omega) \) randomized time.

4 Fine-Grained Search to Decision

Define the \( k \)-Subgraph-Finding problem to be:
Given a graph \( G \) on \( n \) nodes and graph \( H \) on \( k \) nodes, output a \( k \)-subgraph of \( G \) which is isomorphic to \( H \), if one exists.

Define the \( k \)-Subgraph-Detection problem to be:
Given a graph \( G \) on \( n \) nodes and a graph \( H \) on \( k \) nodes, output “yes” or “no” depending on whether there is a \( k \)-subgraph of \( G \) which is isomorphic to \( H \).

(Note: the \( k \)-subgraph can be any subset of the edges of \( G \), but the edges must be adjacent to exactly \( k \) nodes.)

Prove that: if \( k \)-Subgraph-Detection is in \( f(k) \cdot n^c \) time (for some \( f \) and some \( c \geq 2 \)), then \( k \)-Subgraph-Finding can be solved in \( g(k) \cdot n^c \) time for some \( g \).

That is, we can preserve the exponent \( c \), and turn any detection algorithm into a finding algorithm (at some slight cost to the dependence on \( k \)).
Extra Credit: Try to make your \( g(k) \) as small as possible (as a function of \( f(k) \)). How small can it be?

---

\(^1\)Big Hint. Let \( d \) be a parameter. Divide the problem into two cases, and set \( d \) appropriately: (i) There is a triangle with at least one node of degree at most \( d \). Show this case can be solved in \( O(m \cdot d) \) time. (ii) There is a triangle with all nodes of degree at least \( d \). Show this case can be solved in \( O((m/d)^\omega) \) time, using part a and simple counting.