6.841/18.405J Advanced Complexity Theory **March 12, 2003**

Lecture 10: Toda's Theorem

Instructor: Prof. Madhu Sudan *Scribe: Max Goldman*

1 $\mathbf{P^{\#P}}$

1.1 Definition and Properties

#P is the class of functions expressible as the number of accepting paths of a nondeterministic Turing machine. So

$$f_M(x) = |\{y \mid M(x,y) \text{ accepts}\}|,\ M \text{ poly-time two-input TM} \Rightarrow f_M(x) \in \#\text{P}$$

Thus $\text{P}^{\#\text{P}}$ is the class of languages decidable in polynomial time with oracle access to some #P function. We can note that

$\text{NP} \subseteq \text{P}^{\#\text{P}}$

$\text{coNP} \subseteq \text{P}^{\#\text{P}}$

$\text{BPP} \subseteq \text{P}^{\#\text{P}}$

$\text{PP} \subseteq \text{P}^{\#\text{P}}$, where PP requires only a strict majority to be correct, as opposed to $\frac{2}{3}$ for BPP

$\text{P}^{\#\text{P}} \subseteq \text{PSPACE}$, but $\text{P}^{\#\text{P}} \stackrel{?}{=} \text{PSPACE}$ is an open question

Complete functions for #P include:

- #SAT: $\phi \rightarrow$ number of satisfying assignments of ϕ
- #HAMCYCLE: $G \rightarrow$ number of Hamiltonian cycles in G
- In fact, #CYCLE is #P-complete by reduction from #HAMCYCLE
- Computing the permanent of a matrix, roughly equivalent to the number of perfect matchings in a bipartite graph; it is worth noting that the permanent computation is very similar to that of the determinant, and the transformation can be done in the case of planar matrices, but this result shows that if it could be done in general, many of our strong beliefs would be violated

Finally, we note that similar to $\text{BPP} \subseteq \Sigma_2^{\text{P}}$, we can show that approx-$\#\text{P} \subseteq \Sigma_3^{\text{P}} \cap \Pi_3^{\text{P}}$.

2 $\mathbf{PH \subseteq P^{\#P}}$

2.1 Operators on Complexity Classes

An operator $\mathcal{O}$ transforms one complexity class into a new one, where $\mathcal{O} \in \{\text{BP}, \exists, \forall, \text{co}, \oplus\}$. We define these operators for a language $L = \{x, y\}$ as

$$\oplus(L) = \{x \mid \text{number of } y\text{'s for which } (x,y) \in L \text{ is odd}\},\ \exists,\ \forall \text{ similar}$$

$$\text{BP}(L) = \begin{cases} x \in \Pi_{\text{YES}} \Rightarrow \Pr_y[(x,y) \in L] \geq 1 - 2^{-|x|} \\ x \in \Pi_{\text{NO}} \Rightarrow \Pr_y[(x,y) \in L] \leq 2^{-|x|} \end{cases}$$

On a class $\mathcal{C}$, we have $\mathcal{O} \cdot \mathcal{C} = \{\mathcal{O}(L) \mid L \in \mathcal{C}\}$. Defining the BP operator in this strong manner (exponentially small error as opposed to a constant fraction) will simply the results to follow.

2.2 Toda's Theorem: Part 1 of n

Toda's Theorem. *$PH \subseteq P^{\#P}$*

We prove this theorem by means of two essentially separate lemmas, from which the desired result follows directly.

Lemma 1. *$PH \subseteq BP \cdot \oplus \cdot P$*

Lemma 2. *$BP \cdot \oplus \cdot P \subseteq P^{\#P}$*

In order to prove these two lemmas, we first note the following:

Claim 0. *Operators BP and $\oplus$ preserve closure under complementation*

Argument for Claim 0. The statement is that co$\cdot$BP$\cdot\mathcal{C} \subseteq$ BP$\cdot\mathcal{C}$, for $\mathcal{C}$ closed under complementation.
In the case of BP, BP$(L) \in$ BP $\cdot\, \mathcal{C}$, and BP$(\overline{L}) = \overline{\text{BP}(L)}$.
For $\oplus$, we see that, for example, if $(\phi, a) \in L$,

$$\oplus(L) = \{\phi \mid \phi \text{ has an odd number of satisfying assignments}\}$$

We can find the complement with the mapping $\phi(x_1, ..., x_n) \to \phi'(x_0, x_1, ..., x_n)$, where ϕ' is satisfied if either $x_0 = 1$ and $x_1 = ... = x_n = 0$ or $x_0 = 0$ and $\phi(x_1, ..., x_n) = 1$, which adds one to the number of satisfying assignments.

Proof of Lemma 1. This proof is done in two parts:

Part 1.1. *$\exists \cdot \mathcal{C} \subseteq BP \cdot \oplus \cdot \mathcal{C}$, where $\mathcal{C} = BP \cdot \oplus \cdot P$*

Part 1.2. *$BP \cdot \oplus \cdot BP \cdot \oplus \cdot P \subseteq BP \cdot BP \cdot \oplus \cdot \oplus \cdot P \subseteq BP \cdot \oplus \cdot P$*

Part 1.1 also implies that $\forall \cdot \mathcal{C} \subseteq$ BP $\cdot \oplus \cdot \mathcal{C}$, so any level of the polynomial hierarchy can be expressed as BP $\cdot \oplus \cdot$ BP $\cdot \oplus \cdots$ BP $\cdot \oplus \cdot$ P. Applying Part 1.2 a finite number of times at each level yields PH $\subseteq$ BP $\cdot \oplus \cdot$ P to prove Lemma 1.

Proof of Part 1.1. We begin by showing NP $\subseteq$ BP$\cdot\oplus\cdot$P, by amplifying the $\frac{1}{\text{poly}}$ result used previously to show USAT is NP-hard under randomized reductions. Recall the Valiant-Vazirani reduction:

$$\phi \overset{?}{\in} \text{SAT} \to \psi \overset{?}{\in} \text{USAT} : \begin{cases} \phi \in \text{SAT} \to \psi \text{ has one satisfying assignment w.p. } \frac{1}{n} \\ \phi \notin \text{SAT} \to \psi \text{ has zero satisfying assignments} \end{cases}$$

Applying this random reduction many times yields several formulas ψ_i:

$$\left.\begin{array}{rcl} \phi \in \text{SAT} & \to & \left.\begin{array}{c} \psi_1 \\ \vdots \\ \psi_m \end{array}\right\} \text{ at least one has an odd number of satisfying assignments w.h.p.} \\ \phi \notin \text{SAT} & \to & \left.\begin{array}{c} \psi_1 \\ \vdots \\ \psi_m \end{array}\right\} \text{ all have an even number} \end{array}\right.$$

We then need to combine $\psi_1 \ldots \psi_m$ into a single formula $\widehat{\psi}$ such that if all ψ_i are even, $\widehat{\psi}$ is even, and if some ψ_i are odd, $\widehat{\psi}$ is odd w.h.p. Applying a parity flip makes this easier; we now need to find a $\widehat{\psi}$ that is odd if all ψ_i are odd and is even otherwise. If we have each $\psi_i(x_i)$ operating on the set of variables x_i, then we simply use

$$\widehat{\psi}(x_1 \ldots x_m) = \bigwedge_{i=1}^{m} \psi_i(x_i)$$

This gives NP $\subseteq$ BP $\cdot \oplus \cdot$ P. The same method works to show $\exists \cdot$ BP $\cdot \oplus \cdot$ P $\subseteq$ BP $\cdot \oplus$BP $\cdot \oplus \cdot$ P, which proves Part 1.1 of the lemma. □

2.3 Operators and Circuits

It can be useful to think of complexity class operators in terms of constructing a circuit, where each operator is a gate with fan-in equal to the number of assignments it quantifies over. Operators $\forall$, $\exists$, and co correspond to the usual AND, OR, and NOT gates, respectively, and $\oplus$ is of course a parity gate. BP corresponds to an "approximate majority" gate, which passes on the majority of its inputs with some error. The result of Part 1.1 above shows that any OR gate in the circuit can be replaced by an approximate majority gate followed by parity gates. We use this understanding of operators to begin the proof of Part 1.2 in the next section.

2.4 Toda's Theorem: Part 2 of n

Proof of Part 1.2. Recall the claim of 1.2, that $\mathrm{BP} \cdot \oplus \cdot \mathrm{BP} \cdot \oplus \cdot \mathrm{P} \subseteq \mathrm{BP} \cdot \mathrm{BP} \cdot \oplus \cdot \oplus \cdot \mathrm{P} \subseteq \mathrm{BP} \cdot \oplus \cdot \mathrm{P}$. This requires that the innermost operators be flipped, and then adjacent operators eliminated. In circuit terms, the flip consists of taking a parity gate whose inputs are BP "approximate majority" gates, and switching them to yield a single BP gate fed by several parity gates. This is done without making any changes to the surrounding circuit.

If the original circuit has a parity gate quantifying over variable y with fan-in 2^a and BP gates over z each with fan-in 2^b, the new circuit will have a BP gate and 2^b parity gates each with fan-in 2^a. These arrangements are shown in Figures 1 and 2. We let 2^{-c} be the error of the original BP gate.

Picking a random z, we see that the z^{th} parity gate in the new circuit computes the incorrect value if there exists a y such that the original BP gate for that y is incorrect. By simply counting the error of each original BP gate, we see that

$$\begin{aligned}\Pr[z^{\text{th}} \ \oplus \ \text{in new circuit incorrect}] &\leq \Pr[\exists y \text{ s.t. BP in original circuit incorrect}] \\ &\leq 2^b \cdot 2^{-c}\end{aligned}$$

Thus, by ensuring that c is larger than b, we can push the error rate arbitrarily low. This shows that the quantifier switch is correct; the rest remains to be shown... in another lecture.

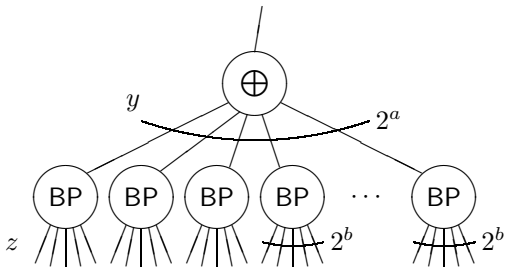

Figure 1: Original circuit

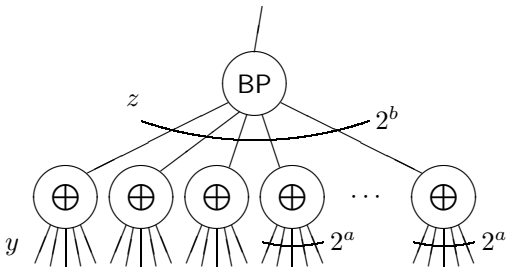

Figure 2: Circuit after transformation